

Lecture Notes in Electrical Engineering 735

Sabu M. Thampi · Erol Gelenbe ·  
Mohammed Atiquzzaman ·  
Vipin Chaudhary ·  
Kuan-Ching Li *Editors*

# Advances in Computing and Network Communications

Proceedings of CoCoNet 2020, Volume 1

 Springer

# Lecture Notes in Electrical Engineering

## Volume 735

### Series Editors

Leopoldo Angrisani, Department of Electrical and Information Technologies Engineering, University of Napoli Federico II, Naples, Italy

Marco Arteaga, Departament de Control y Robótica, Universidad Nacional Autónoma de México, Coyoacán, Mexico

Bijaya Ketan Panigrahi, Electrical Engineering, Indian Institute of Technology Delhi, New Delhi, Delhi, India

Samarjit Chakraborty, Fakultät für Elektrotechnik und Informationstechnik, TU München, Munich, Germany

Jiming Chen, Zhejiang University, Hangzhou, Zhejiang, China

Shanben Chen, Materials Science and Engineering, Shanghai Jiao Tong University, Shanghai, China

Tan Kay Chen, Department of Electrical and Computer Engineering, National University of Singapore, Singapore, Singapore

Rüdiger Dillmann, Humanoids and Intelligent Systems Laboratory, Karlsruhe Institute for Technology, Karlsruhe, Germany

Haibin Duan, Beijing University of Aeronautics and Astronautics, Beijing, China

Gianluigi Ferrari, Università di Parma, Parma, Italy

Manuel Ferre, Centre for Automation and Robotics CAR (UPM-CSIC), Universidad Politécnica de Madrid, Madrid, Spain

Sandra Hirche, Department of Electrical Engineering and Information Science, Technische Universität München, Munich, Germany

Faryar Jabbari, Department of Mechanical and Aerospace Engineering, University of California, Irvine, CA, USA

Limin Jia, State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing, China

Janusz Kacprzyk, Systems Research Institute, Polish Academy of Sciences, Warsaw, Poland

Alaa Khamis, German University in Egypt El Tagamoa El Khames, New Cairo City, Egypt

Torsten Kroeger, Stanford University, Stanford, CA, USA

Qilian Liang, Department of Electrical Engineering, University of Texas at Arlington, Arlington, TX, USA

Ferran Martín, Departament d'Enginyeria Electrònica, Universitat Autònoma de Barcelona, Bellaterra, Barcelona, Spain

Tan Cher Ming, College of Engineering, Nanyang Technological University, Singapore, Singapore

Wolfgang Minker, Institute of Information Technology, University of Ulm, Ulm, Germany

Pradeep Misra, Department of Electrical Engineering, Wright State University, Dayton, OH, USA

Sebastian Möller, Quality and Usability Laboratory, TU Berlin, Berlin, Germany

Subhas Mukhopadhyay, School of Engineering & Advanced Technology, Massey University,

Palmerston North, Manawatu-Wanganui, New Zealand

Cun-Zheng Ning, Electrical Engineering, Arizona State University, Tempe, AZ, USA

Toyoaki Nishida, Graduate School of Informatics, Kyoto University, Kyoto, Japan

Federica Pascucci, Dipartimento di Ingegneria, Università degli Studi "Roma Tre", Rome, Italy

Yong Qin, State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing, China

Gan Woon Seng, School of Electrical & Electronic Engineering, Nanyang Technological University, Singapore, Singapore

Joachim Speidel, Institute of Telecommunications, Universität Stuttgart, Stuttgart, Germany

Germano Veiga, Campus da FEUP, INESC Porto, Porto, Portugal

Haitao Wu, Academy of Opto-electronics, Chinese Academy of Sciences, Beijing, China

Junjie James Zhang, Charlotte, NC, USA

The book series *Lecture Notes in Electrical Engineering* (LNEE) publishes the latest developments in Electrical Engineering - quickly, informally and in high quality. While original research reported in proceedings and monographs has traditionally formed the core of LNEE, we also encourage authors to submit books devoted to supporting student education and professional training in the various fields and applications areas of electrical engineering. The series cover classical and emerging topics concerning:

- Communication Engineering, Information Theory and Networks
- Electronics Engineering and Microelectronics
- Signal, Image and Speech Processing
- Wireless and Mobile Communication
- Circuits and Systems
- Energy Systems, Power Electronics and Electrical Machines
- Electro-optical Engineering
- Instrumentation Engineering
- Avionics Engineering
- Control Systems
- Internet-of-Things and Cybersecurity
- Biomedical Devices, MEMS and NEMS

For general information about this book series, comments or suggestions, please contact [leontina.dicecco@springer.com](mailto:leontina.dicecco@springer.com).

To submit a proposal or request further information, please contact the Publishing Editor in your country:

#### **China**

Jasmine Dou, Editor ([jasmine.dou@springer.com](mailto:jasmine.dou@springer.com))

#### **India, Japan, Rest of Asia**

Swati Meherishi, Editorial Director ([Swati.Meherishi@springer.com](mailto:Swati.Meherishi@springer.com))

#### **Southeast Asia, Australia, New Zealand**

Ramesh Nath Premnath, Editor ([ramesh.premnath@springernature.com](mailto:ramesh.premnath@springernature.com))

#### **USA, Canada:**

Michael Luby, Senior Editor ([michael.luby@springer.com](mailto:michael.luby@springer.com))

#### **All other Countries:**

Leontina Di Cecco, Senior Editor ([leontina.dicecco@springer.com](mailto:leontina.dicecco@springer.com))

**\*\* This series is indexed by EI Compendex and Scopus databases. \*\***

More information about this series at <http://www.springer.com/series/7818>

Sabu M. Thampi · Erol Gelenbe ·  
Mohammed Atiquzzaman · Vipin Chaudhary ·  
Kuan-Ching Li  
Editors

# Advances in Computing and Network Communications

Proceedings of CoCoNet 2020, Volume 1

 Springer

*Editors*

Sabu M. Thampi  
School of Computer Science & Engineering  
Indian Institute of Information Technology  
and Management-Kerala (IIITM-K)  
Trivandrum, Kerala, India

Mohammed Atiquzzaman  
School of Computer Science  
University of Oklahoma  
Norman, OK, USA

Kuan-Ching Li  
Department of Computer Science  
and Information Engineering  
Providence University  
Taichung, Taiwan

Erol Gelenbe  
Institute of Theoretical and Applied  
Informatics  
Polish Academy of Sciences  
Gliwice, Poland

Vipin Chaudhary  
Department of Computer Science  
University at Buffalo  
State University, Buffalo, NY, USA

Case Western Reserve University  
Cleveland, OH, USA

ISSN 1876-1100

ISSN 1876-1119 (electronic)

Lecture Notes in Electrical Engineering

ISBN 978-981-33-6976-4

ISBN 978-981-33-6977-1 (eBook)

<https://doi.org/10.1007/978-981-33-6977-1>

© The Editor(s) (if applicable) and The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2021

This work is subject to copyright. All rights are solely and exclusively licensed by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

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

# **Organized by**

Vellore Institute of Technology (VIT), Chennai, India

## **Conference Organization**

### **Chief Patron**

Dr. G. Viswanathan, Chancellor, VIT

### **Patrons**

Sankar Viswanathan, Vice-President, Vellore Institute of Technology

Sekar Viswanathan, Vice-President, Vellore Institute of Technology

G. V. Selvam, Vice-President, Vellore Institute of Technology

Sandhya Pentareddy, Executive Director, Vellore Institute of Technology

Kadhambari S. Viswanathan, Assistant Vice-President, Vellore Institute of Technology

Rambabu Kodali, Vice Chancellor, Vellore Institute of Technology

S. Narayanan, Pro-VC, Vellore Institute of Technology, Vellore

V. S. Kanchana Bhaaskaran, Pro-VC, Vellore Institute of Technology, Chennai

P. K. Manoharan, Additional Registrar, VIT Chennai

### **Honorary General Chair**

Raj Jain, Barbara J. and Jerome R. Cox, Jr., Professor of Computer Science and Engineering, Washington University in St. Louis, USA

## **General Chairs**

Erol Gelenbe, Institute of Theoretical and Applied Informatics, Polish Academy of Sciences, Poland

Jayanta Mukhopadhyay, Indian Institute of Technology Kharagpur, India

Jagadeesh Kannan R., VIT Chennai

## **General Executive Chair**

Sabu M. Thampi, School of Computer Science & Engineering, Indian Institute of Information Technology and Management-Kerala (IIITM-K), Trivandrum, Kerala, India

## **Program Chairs**

Mohammed Atiquzzaman, University of Oklahoma, USA

Vipin Chaudhary, University at Buffalo (UB), SUNY, USA

Kuan-Ching Li, Providence University, Taiwan

Peter Mueller, IBM Zurich Research Laboratory, Switzerland

## **Workshop and Symposium Chairs**

Al-Sakib Khan Pathan, Independent University, Bangladesh

Pradeep K. Atrey, State University of New York (SUNY), Albany, USA

## **Industry Track Chairs**

Dilip Krishnaswamy, Reliance Industries Ltd., India

Arpan Pal, TCS Innovation Lab, Kolkata, India

Sougata Mukherjee, IBM India Research Lab, New Delhi, India

Anindita Banerjee, QuNu Labs, Bengaluru, India

## **Organizing Chair**

Geetha S., VIT Chennai

## **Organizing Secretaries**

Sweetlin Hemalatha C., VIT Chennai

Suganya G., VIT Chennai

Kumar R., VIT Chennai

## **Organizing Co-chairs**

Asha S., VIT Chennai

Pattabiraman R., VIT Chennai

Viswanathan V., VIT Chennai

## **Tutorial Chairs**

Domenico Ciuonzo, University of Naples Federico II, Italy

Maheshkumar H. Kolekar, Indian Institute of Technology Patna, India

Pascal Lorenz, University of Haute Alsace, France

## **Demo/Poster Chair**

Sergey Mosin, Kazan Federal University, Russia

## **Advisory Committee**

Mukesh Mohania, Indraprastha Institute of Information Technology, Delhi (IIIT D),  
India

Oge Marques, Florida Atlantic University (FAU) (Boca Raton, Florida), USA

Ajith Abraham, Machine Intelligence Research Labs (MIR Labs), USA

Axel Sikora, University of Applied Sciences Offenburg, Germany



Madhukar Pitke, Professor (Retired) at Tata Institute of Fundamental Research, Mumbai

Selwyn Piramuthu, University of Florida, USA

Juan Manuel Corchado Rodríguez, University of Salamanca, Spain

Mauro Conti, University of Padua, Italy

Nallanathan Arumugam, King's College London, UK

Raj Kumar Buyya, University of Melbourne, Australia

Sameep Mehta, IBM Research—India

Bharat Bhargava, Purdue University, USA

Debabrata Das, International Institute of Information Technology Bangalore

Schahram Dustdar, The TU Wien, Austria

Sherali Zeadally, University of Kentucky, USA

V. N. Venkatakrisnan, University of Illinois at Chicago, USA

Jorge Sá Silva, Universidade de Coimbra, Portugal

Jiankun Hu, University of New South Wales, Australia

Bharat Jayaraman, University at Buffalo, State University of New York, USA

Rajendra Boppana, The University of Texas at San Antonio (UTSA), USA

Jalel Ben-Othman, University of Paris 13, France

Stefan Fischer, University of Luebeck, Germany

Shekar Babu, Amrita Vishwa Vidyapeetham, Bangalore

## **TPC Members**

<http://coconet-conference.org/2020/?q=committee>

# Preface

The Fourth International Conference on Computing and Network Communications (CoCoNet'20) provided a forum for sharing original research outcomes and practical development experiences among experts in the emerging areas of computing and communications. The conference was organized by Vellore Institute of Technology (VIT), Chennai, India. Due to the recent pandemic situation, the conference was conducted as a virtual event during October 14–17, 2020.

The material was presented in a program that consisted of keynote talks, technical sessions, lightning talks, tutorials, symposiums, parallel sessions, workshops and hot off the press. CoCoNet'20 convened a well-tailored and handpicked collection of eminent speakers from renowned universities and industries in and outside India. The array of speakers included Dr. Ian T. Foster, Dr. Arumugam Nallanathan, Dr. Mohammed Atiquzzaman, Dr. Vipin Chaudhary, Dr. Dilip Krishnaswamy and Dr. Ljiljana Trajkovic. The conference received 181 submissions this year, out of which 98 papers (58 regular papers and 40 short papers) had been accepted. The papers were subjected to a rigorous review process that examined the significance, novelty and technical quality of the submission. The papers were presented in different sessions, namely the best paper sessions, regular paper sessions and short paper sessions.

The proceedings of the conference is organized into two volumes. This volume is comprised of 51 papers, and the topical sections include Communications, Control and Signal Processing, Data Analytics and Networked Systems and Security.

The success of the conference depends ultimately on the numerous people who have worked with us to plan and organize both the technical program and local arrangements. In particular, the wise advice and brilliant ideas of the program chairs, workshop and symposium chairs and industry track chairs in the organization of the technical program are gratefully appreciated. We would like to extend our heartfelt gratitude to the organizing committee and advisory committee members.

The accomplishment of the motives of the conference is powered by the tireless efforts of many individuals. We would like to express our sincere gratitude to the TPC chairs, TPC members and additional reviewers who shared their technical expertise and assisted us in reviewing all the submitted papers. We would like to thank the general chairs, organizing committee members, steering committee, keynote speakers, session chairs and the conference attendees. We are thankful to

all the authors for choosing this conference as a venue for presenting their research works. We express our wholehearted appreciation to the contributions of all those who apportioned their valuable time for the success of CoCoNet'20.

We are grateful to Vellore Institute of Technology (VIT), Chennai, for organizing the conference. Recognition should go to the local organizing committee members who all have worked extremely hard for the details of important aspects of the conference programs. We appreciate the contributions of all the faculty and staff of VIT and the student volunteers who have selflessly contributed their time to make this virtual conference successful. We would like to express our gratitude to Senior Editor of Springer Nature, Aninda Bose, for his help and cooperation.

We sincerely hope that CoCoNet'20 turned out to be a forum for excellent discussions that enabled new ideas to come about, promoting collaborative research. We are confident that the proceedings will serve as a momentous source of research references and knowledge, which will lead not only to the scientific and engineering findings but also to the development of new products and technologies.

Trivandrum, India  
Gliwice, Poland  
Norman, USA  
Buffalo\Cleveland, USA  
Taichung, Taiwan  
October 2020

Sabu M. Thampi  
Erol Gelenbe  
Mohammed Atiquzzaman  
Vipin Chaudhary  
Kuan-Ching Li

# Contents

## Communications, Control and Signal Processing

<b>Cost-Effective Device for Autonomous Monitoring of the Vitals for COVID-19 Asymptomatic Patients in Home Isolation Treatment . . . .</b>	<b>3</b>
V. Ashwin, Athul Menon, A. M. Devagopal, P. A. Nived, Athira Gopinath, G. Gayathri, and N. B. Sai Shibu	
<b>Predictive Modeling and Control of Clamp Load Loss in Bolted Joints Based on Fractional Calculus . . . . .</b>	<b>15</b>
Pritesh Shah and Ravi Sekhar	
<b>Resource Allocation for 5G RAN—A Survey . . . . .</b>	<b>33</b>
G. Shanmugavel and M. S. Vasanthi	
<b>Wearable PIFA for Off-Body Communication: Miniaturization Design and Human Exposure Assessment . . . . .</b>	<b>43</b>
Sandra Costanzo, Adil Masoud Qureshi, and Vincenzo Cioffi	
<b>Generalized Symbolic Dynamics Approach for Characterization of Time Series . . . . .</b>	<b>53</b>
S. Suriyaprabhaa, Greeshma Gopinath, R. Sangeerthana, S. Alfiya, P. Asha, and K. Satheesh Kumar	
<b>Smart Mirror-Based Personal Healthcare System . . . . .</b>	<b>63</b>
V. B. Aanandhi, Anshida Das, Melissa Grace Melchizedek, Nived Priyadarsan, and A. Binu Jose	
<b>On-off Thinning in Linear Antenna Arrays Using Binary Dragonfly Algorithm . . . . .</b>	<b>75</b>
Ashish Patwari, Medha Mani, Sneha Singh, and Gokul Srinivasan	
<b>Reduction in Average Distance Cost by Optimizing Position of ONUs in FiWi Access Network using Grey Wolf Optimization Algorithm . . . . .</b>	<b>91</b>
Nitin Chouhan, Uma Rathore Bhatt, and Raksha Upadhyay	

<b>Performance Analysis of Individual Partial Relay Selection Protocol Using Decode and Forward Method for Underlay EH—CRN</b> .....	105
G. Kalaimagal and M. S. Vasanthi	
<b>Building a Cloud-Integrated WOBAN with Optimal Coverage and Deployment Cost</b> .....	119
Mausmi Verma, Uma Rathore Bhatt, and Raksha Upadhyay	
<b>VR Classroom for Interactive and Immersive Learning with Assessment of Students Comprehension</b> .....	133
J. S. Jaya Sudha, Nandagopal Nandakumar, Sarath Raveendran, and Sidharth Sandeep	
<b>Localization of Self-driving Car Using Particle Filter</b> .....	147
Nalini C. Iyer, Akash Kulkarni, Raghavendra Shet, and U. Keerthan	
<b>Convex Combination of Maximum Versoria Criterion-Based Adaptive Filtering Algorithm for Impulsive Environment</b> .....	157
S. Radhika, A. Chandrasekar, and K. Ishwarya Rajalakshmi	
<b>Verifying Mixed Signal ASIC Using SVM</b> .....	167
H. R. Aishwaraya, Saroja V. Siddamal, Aishwaraya Shetty, and Prateeksha Raikar	
<b>Design of High-Speed Turbo Product Code Decoder</b> .....	175
Gautham Shivanna, B. Yamuna, Karthi Balasubramanian, and Deepak Mishra	
<b>Data Analytics</b>	
<b>Extraction and Analysis of Facebook Public Data and Images</b> .....	189
Bala Gangadhara Gutam, D. Subhash Chandra Mouli, and Sudhakar Majjari	
<b>Subspace Clustering Using Matrix Factorization</b> .....	203
Sandhya Harikumar and Shilpa Joseph	
<b>A Data-Driven Approach for Peer Recommendation to Reduce Dropouts in MOOC</b> .....	217
Manika Garg and Anita Goel	
<b>Bag of Science: A Query Structuring and Processing Model for Recommendation Systems</b> .....	231
Prakash Hegade, Vibha Hegde, Sourabh Jain, Rajaram M. Joshi, and K. L. Vijeth	
<b>A Novel Design Approach Exploiting Data Parallelism in Serverless Infrastructure</b> .....	247
Urmil Bharti, Deepali Bajaj, Anita Goel, and S. C. Gupta	

**A LoRa-Based Data Acquisition System for Wildfire Early Detection** ..... 261  
 Stefan Rizanov, Anna Stoynova, and Dimitar Todorov

**Announcer Model for Inter-Organizational Systems** ..... 277  
 Prakash Hegade, Nikhil Lingadhal, Usman Khan, Tejaswini Kale, and Srushti Basavaraddi

**Evaluation of Attributed Network Embedding Algorithms for Patent Analytics** ..... 293  
 Jinesh Jose and S. Mary Saira Bhanu

**A Comparative Analysis of Garbage Collectors and Their Suitability for Big Data Workloads** ..... 305  
 Advithi Nair, Aiswarya Sriram, Alka Simon, Subramaniam Kalambur, and Dinkar Sitaram

**Networked Systems and Security**

**An Innovative and Inventive IoT-Based Navigation Device—An Attempt to Avoid Accidents and Avert Confusion** ..... 319  
 Chennuru Vineeth, Shriram K. Vasudevan, J. Anudeep, G. Kowshik, and Prashant R. Nair

**Deploy—Web Hosting Using Docker Container** ..... 335  
 Minto Sunny, Sen Shaji, Sheen Sabu, Udith Uthaman, and Gemini George

**Enhancement of VerticalThings DSL with Learnable Features** ..... 347  
 Sandesh Ghanta, P. V. Surya Chaitanya, Sai Sarath Chandra Ganti, M. P. V. Roshan Patnaik, and G. Gopakumar

**Demand-Based Dynamic Slot Allocation for Effective Superframe Utilization in Wireless Body Area Network** ..... 361  
 A. Justin Gopinath and B. Nithya

**A Survey on Congestion Control Algorithms of Wireless Body Area Network** ..... 373  
 Vamsikiran Mekathoti and B. Nithya

**Applications of RSSI Preprocessing in Multi-Domain Wireless Networks: A Survey** ..... 389  
 Tapesh Sarsodia, Uma Rathore Bhatt, and Raksha Upadhyay

**Exploring IoT-Enabled Multi-Hazard Warning System for Disaster-Prone Areas** ..... 405  
 Vishal Menon, R. Arjun Rathya, Abhiram Prasad, Athira Gopinath, N. B. Sai Shibu, and G. Gayathri

**IOT Based Smart and Secure Surveillance System Using Video Summarization** ..... 423  
M. Surya Priya, D. Diana Josephine, and P. Abinaya

**An Efficient and Innovative IoT-Based Intelligent Real-Time Staff Assessment Wearable** ..... 437  
J. Anudeep, Shiram K. Vasudevan, G. Kowshik, Chennuru Vineeth, and Prashant R. Nair

**Performance Evaluation of WebRTC for Peer-to-Peer Communication** ..... 455  
Kiran Jadhav, D. G. Narayan, and Mohammed Moin Mulla

**Scalable Blockchain Framework for a Food Supply Chain** ..... 467  
Manjula K. Pawar, Prakashgoud Patil, P. S. Hiremath, Vaibhav S. Hegde, Shyamsundar Agarwal, and P. B. Naveenkumar

**Maximizing Lifetime of Mobile Ad-Hoc Networks with Optimal Cooperative Routing** ..... 479  
K. C. Kullayappa Naik, Ch. Balaswamy, and Patil Ramana Reddy

**On-Demand Multi-mobile Charging Scheduling Scheme for Wireless Rechargeable Sensor Networks** ..... 491  
Charan Ramtej Kodi, Debjit Das, and Shashi Shekar

**CRAWL: Cloud-Based Real-Time Interconnections of Agricultural Water Sources Using LoRa** ..... 509  
P. Sree Harshitha, Raja VaraPrasad, and Hrishikesh Venkataraman

**Link Prediction Analysis on Directed Complex Network** ..... 525  
Salam Jayachitra Devi and Buddha Singh

**Energy-Efficient VM Management in OpenStack-Based Private Cloud** ..... 541  
P. K. Prameela, Priyanka Gadagi, Revathi Gudi, Somashekar Patil, and D. G. Narayan

**Intelligent Transportation System: The Applicability of Reinforcement Learning Algorithms and Models** ..... 557  
S. P. Krishnendhu and Prabu Mohandas

**Speaker Identification Approach for the Post-pandemic Era of Internet of Things** ..... 573  
A. Saleema and Sabu M. Thampi

**Random Permutation-Based Linear Discriminant Analysis for Cancelable Biometric Recognition** ..... 593  
P. Punithavathi and S. Geetha

**A Deep Learning-Based Framework for Distributed Denial-of-Service Attacks Detection in Cloud Environment** ..... 605  
 Amit V. Kachavimath and D. G. Narayan

**Automation for Furnace in Thermal Power Station Using Public Key Cryptography** ..... 619  
 M. Prathyusha, Padmanabha Nikitha, S. Rajashree, and B. Prasad Honnavalli

**Active Dictionary Attack on WPA3-SAE** ..... 633  
 Manthan Patel, P.P Amritha, and R. Sam jasper

**Multiple Hashing Using SHA-256 and MD5** ..... 643  
 Gautham P. Reddy, Anoop Narayana, P. Karan Keerthan, B. Vineetha, and Prasad Honnavalli

**Design and Analysis of a Secure Coded Communication System Using Chaotic Encryption and Turbo Product Code Decoder** ..... 657  
 S. Khavya, Karthi Balasubramanian, B. Yamuna, and Deepak Mishra

**Digital Image Transmission Using Combination of DWT-DCT Watermarking and AES Technique** ..... 667  
 Sudhanshu S. Gonge

**An HTTP DDoS Detection Model Using Machine Learning Techniques for the Cloud Environment** ..... 685  
 N. Muraleedharan and B. Janet

**IoT Device Authentication and Access Control Through Hyperledger Fabric** ..... 699  
 Bibin Kurian and Narayanan Subramanian

**Author Index** ..... 715



## About the Editors

**Sabu M. Thampi** is Professor at the Indian Institute of Information Technology and Management-Kerala (IIITM-K), Technopark Campus, Trivandrum, India. His current research interests include cognitive computing, Internet of Things (IoT), authorship analysis, trust management, biometrics, social networks, nature-inspired computing and video surveillance. He has published papers in book chapters, journals and conference proceedings. He has authored and edited a few books. Sabu has served as Guest Editor for special issues in few journals and a program committee member for many international conferences and workshops. He has co-chaired several international workshops and conferences. He has initiated and is also involved in the organization of several annual conferences/symposiums. Sabu is currently serving as Editor for Elsevier *Journal of Network and Computer Applications* (JNCA), *Connection Science*, Taylor Francis, Associate Editor for *IEEE Access* and *International Journal of Embedded Systems*, Inderscience, UK, and Reviewer for several reputed international journals. Sabu is a senior member of IEEE and ACM. He is Founding Chair of ACM Trivandrum Professional Chapter.

**Erol Gelenbe** is a Turkish-French computer scientist, electronic engineer and applied mathematician who is professor in Computer-Communications at Imperial College. Known for pioneering the field of modelling and performance evaluation of computer systems and networks throughout Europe, he invented the random neural network and the eponymous G-networks. His many awards include the ACM SIGMETRICS Life-Time Achievement Award, and the in Memoriam Dennis Gabor Award of the Hungarian Academy of Sciences. Working as a foreigner everywhere, Gelenbe was born in Istanbul in 1945, to Yusuf Ali Gelenbe, a descendant of the 18th-century Ottoman mathematician Gelenbevi Ismail Efendi, and to Maria Sacchet Gelenbe from Cesiomaggiore, Belluno, Italy. After a childhood spent in Istanbul and Alexandria (Egypt), He graduated from Ankara Koleji in 1962 and the Middle East Technical University in 1966, winning the K. K. Clarke Research Award for work on “partial flux switching magnetic memory systems”. Awarded a Fulbright Fellowship, he continued his studies at Polytechnic University, where he completed a master’s degree and a Ph.D. thesis on “Stochastic automata with structural restrictions”, under the supervision of Edward J. Smith. After graduation

he joined the University of Michigan as an assistant professor. In 1972, and then on leave from Michigan, he founded the Modeling and Performance Evaluation of Computer Systems research group at INRIA (France), and was a visiting lecturer at the University of Paris 13 University. In 1971 he was elected to the second chair in Computer Science at the University of Liège, where he joined Prof. Danny Ribbens in 1973, while remaining a research director at INRIA. In 1973, he was awarded a Doctorat d'État ès Sciences Mathématiques from the Paris VI University with a thesis on “Modélisation des systèmes informatiques”, under Jacques-Louis Lions. He remained a close friend of Prof. Ribbens and of the University of Liège, and in 1979, he moved to the Paris-Sud 11 University, where he co-founded the Laboratoire de Recherche en Informatique and its Ph.D. Program, before joining Paris Descartes University in 1986 to found the Ecole des Hautes Etudes en Informatique. Gelenbe became New Jersey State Endowed Professor at the New Jersey Institute of Technology from 1991 to 1993, and from 1993 and 1998 he was chaired professor and head of Electrical and Computer Engineering at Duke University. From 1998 to 2003 at the University of Central Florida, he founded the Department (School) of Electrical Engineering and Computer Science and developed the Harris Corporation Engineering Centre. In 2003, Gelenbe joined Imperial College London as Dennis Gabor Professor in Computer and Communication Networks and Head of Intelligent Systems and Networks. In 2016 he joined Institute of Theoretical and Applied Informatics, Polish Academy of Sciences, Gliwice Poland.

Gelenbe has contributed pioneering research concerning the performance of multi-programming computer systems, virtual memory management, data base reliability optimisation, distributed systems and network protocols. He formed, led, and trained the team that designed the commercial QNAP Computer and Network Performance Modeling Tool. He introduced the Flexsim Object Oriented approach for the simulation in manufacturing systems. He carried out some of the first work on adaptive control of computer systems, and published seminal papers on the performance optimisation of computer network protocols and on the use of diffusion approximations for network performance. He developed new product form queueing networks with negative customers and triggers known as G-networks. He also introduced a new spiked stochastic neural network model known as the random neural network, developed its mathematical solution and learning algorithms, and applied it to both engineering and biological problems. His inventions include the design of the first random access fibre-optics local area network, a patented admission control technique for ATM networks, a neural network based anomaly detector for brain magnetic resonance scans, and the cognitive packet network routing protocol to offer quality of service to users. From 1984 to 1986 he served as the Science and Technology Advisor to the French Secretary of State for Universities. He founded the ISCIS (International Symposium on Computer and Information Sciences) series of conferences that since 1986 are held annually in Turkey, the USA and Europe to bring together Turkish computer scientists with their international counterparts. According to the Mathematics Genealogy project, Gelenbe has graduated over 72 Ph.D. students, placing him in the Top50 worldwide—all time—Ph.D. supervisors in the mathematical sciences.

**Mohammed Atiquzzaman** (senior member, IEEE) obtained his M.S. and Ph.D. in Electrical Engineering and Electronics from the University of Manchester (UK) in 1984 and 1987, respectively. He currently holds the Edith J. Kinney Gaylord Presidential professorship in the School of Computer Science at the University of Oklahoma. Dr. Atiquzzaman is Editor-in-Chief of *Journal of Networks and Computer Applications* and Founding Editor-in-Chief of *Vehicular Communications* and serves/served on the editorial boards of many journals including *IEEE Communications Magazine*, *Real Time Imaging Journal*, *International Journal of Communication Networks and Distributed Systems* and *Journal of Sensor Networks and International Journal of Communication Systems*. He co-chaired the IEEE High Performance Switching and Routing Symposium (2003, 2011), IEEE GLOBECOM and ICC (2014, 2012, 2010, 2009, 2007, 2006), IEEE VTC (2013) and the SPIE Quality of Service over Next-Generation Data Networks Conferences (2001, 2002, 2003). He was Panel Co-Chair of INFOCOM'05, is/has been in the program committee of many conferences such as INFOCOM, GLOBECOM, ICCCN, ICCIT and Local Computer Networks and serves on the review panels at the National Science Foundation. He is Current Chair of IEEE Communication Society Technical Committee on Communications Switching and Routing. Dr. Atiquzzaman received IEEE Communication Society's Fred W. Ellersick Prize and NASA Group Achievement Award for "outstanding work to further NASA Glenn Research Center's effort in the area of Advanced Communications/Air Traffic Management's Fiber Optic Signal Distribution for Aeronautical Communications" project. He is Co-author of the book *Performance of TCP/IP over ATM Networks* and has over 270 refereed publications. His current research interests are in areas of transport protocols, wireless and mobile networks, ad hoc networks, satellite networks, power-aware networking and optical communications. His research has been funded by National Science Foundation (NSF), National Aeronautics and Space Administration (NASA), and US Air Force, Cisco and Honeywell.

**Vipin Chaudhary** is the Kevin J. Kranzusch Professor and Chair of the Department of Computer and Data Sciences at Case Western Reserve University. Most recently, he was a Program Director at the National Science Foundation where he was involved in many national initiatives and the Empire Innovation Professor of Computer Science and Engineering at SUNY Buffalo. He cofounded Scalable Informatics, a leading provider of pragmatic, high performance software-defined storage and compute solutions to a wide range of markets, from financial and scientific computing to research and big data analytics. From 2010 to 2013, Dr. Chaudhary was the Chief Executive Officer of Computational Research Laboratories (CRL), a wholly owned Tata Sons company, where he grew the company globally to be an HPC cloud and solutions leader before selling it to Tata Consulting Services. Prior to this, as Senior Director of Advanced Development at Cradle Technologies, Inc., he was responsible for advanced programming tools for multi-processor chips. He was also the Chief Architect at Corio Inc., which had a successful IPO in July, 2000. Dr. Chaudhary was awarded the prestigious President of India Gold Medal in 1986 at the Indian Institute of Technology (IIT) Kharagpur where he received the B.Tech.

(Hons.) degree in Computer Science and Engineering and a Ph.D. degree from The University of Texas at Austin.

**Kuan-Ching Li** is Professor in the Department of Computer Science and Information Engineering at Providence University, Taiwan. Dr. Li is the recipient of awards from Nvidia, AWS, Intel, Ministry of Education (MOE)/Taiwan and Ministry of Science and Technology (MOST)/Taiwan, as also distinguished chair professorships from universities in China and other countries. He has been involved actively in conferences and workshops as a program/general/steering conference chairman positions and numerous conferences and workshops as a program committee member and has organized numerous conferences related to high-performance computing and computational science and engineering. Dr. Li is Editor-in-Chief of technical publications *Connection Science* (Taylor and Francis), *International Journal of Computational Science and Engineering* (Inderscience) and *International Journal of Embedded Systems* (Inderscience), also serving a number of journal's editorial boards and guest editorships. In addition, he has been acting as Co-author and Co-editor of several technical professional books, published by CRC Press, Springer, McGraw-Hill and IGI Global. His topics of interest include GPU/cloud/edge computing, parallel software design, performance evaluation and benchmarking. Dr. Li is a member of AAAS, a senior member of the IEEE and Fellow of the IET.

# **Communications, Control and Signal Processing**

# Cost-Effective Device for Autonomous Monitoring of the Vitals for COVID-19 Asymptomatic Patients in Home Isolation Treatment



V. Ashwin, Athul Menon, A. M. Devagopal, P. A. Nived, Athira Gopinath, G. Gayathri, and N. B. Sai Shibu

**Abstract** As the number of COVID-19 cases keeps growing exponentially in the world, the use of the combination of wearable technology and IoT technologies opens up a wide variety of possibilities. An IoT-enabled healthcare device is useful for proper monitoring of COVID-19 patients to increase safety and reduce spreading. The healthcare device is connected to a large cloud network to obtain desirable solutions for predicting diseases at an early stage. This paper presents the design of a healthcare system that makes use of these technologies in a cost-effective and intuitive way which highlights the application of these technologies in the battle against the pandemic. The wearable can give real-time analysis reports of body vitals so that necessary precautions can be taken in case of infection. The wearable is designed in such a way that it can be used as a precautionary measure for people who are not infected with the virus and as a monitoring device for affected patients during the course of their treatment. This low-cost design can not only be used to prevent the community spread of the virus but also for the early prediction of the disease.

**Keywords** CoVID-19 · IoT · Photoplethysmography · Sensors · Cloud

## 1 Introduction

Nations across the world are putting in all possible resources to deploy cutting-edge technologies [1] to mitigate the effects of the ongoing COVID-19 pandemic by identifying citizens at risk to prevent further spreading. In this scenario, wearable

---

V. Ashwin · A. Menon · A. M. Devagopal · P. A. Nived  
Department of Computer Science & Engineering (CSE), Amrita Vishwa Vidyapeetham,  
Amritapuri, India

A. Gopinath · N. B. Sai Shibu (✉)  
Center for Wireless Networks & Applications (WNA), Amrita Vishwa Vidyapeetham,  
Amritapuri, India  
e-mail: [saishibunb@am.amrita.edu](mailto:saishibunb@am.amrita.edu)

G. Gayathri  
Department of Mechanical Engineering (ME), Amrita Vishwa Vidyapeetham, Amritapuri, India

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2021  
S. M. Thampi et al. (eds.), *Advances in Computing and Network Communications*,  
Lecture Notes in Electrical Engineering 735,  
[https://doi.org/10.1007/978-981-33-6977-1\\_1](https://doi.org/10.1007/978-981-33-6977-1_1)

technology and IoT can play a vital role as they can possibly help track the spread of the virus from person to person, identify high-risk zones, predict clusters and mortality rate and control the spreading in real-time as it enables devices to be connected over a network in the hospital and strategic location to enhance the fight against this pandemic.

The interrelationship between factors like heartbeat and illness has always been high in the medical field. Other factors like body temperature can also be strongly correlated with illness. But there are certain factors which are very hard to express as a value so that it can be used for computational purposes in the medical field. There are many challenges in using the heartbeat values to predict a particular outcome as there are many other factors that can also result in an undesirable heart rate value. So, a major challenge we face is to moderate the result we predict from the recorded data considering the other external and internal factors that can result in abnormalities in the output. But even in the presence of all these problems using these methods can help in the fight against the pandemic in a great amount. This combination of various technologies also reduces the need for physical human interaction, thus enabling the quicker recovery of infected patients and prevents the further spread of the disease. When coupled together with other concepts such as machine learning, this can also help predict and gain further insight in this battle. The existing systems of remote monitoring of COVID-19 patients do not include the cloud computing or big data machine learning features. The large amount of data collected from the people can be used to train highly accurate machine learning algorithms to predict the body status of an infected person for the upcoming week. Therefore, more focus is to be given to interpret the data received from the patients and to train relevant models in the future.

In this paper, we propose a system to alert patients and caretakers in the case of risk through a mobile application which receives data from microprocessor connected through WIFI. The microprocessor which resides in the wearable acquires this data through a variety of sensors such as temperature, GPS and heartbeat sensors.

The paper is organised as follows. Section 2 briefly describes the state of the art and other similar works performed at various institutions. Section 3 explains the process in data collection and processing and about implementation, and the results of each model are discussed. The paper is concluded in Sect. 4.

## 2 Related Work

Castaneda et al. [2] discussed the photo plethysmography method for heart monitoring purposes. The paper discussed about various types of PPG-based devices like wristband type, forehead type and ear type devices. The paper focused on the advantage of these devices over the traditional ECG-based devices which includes low cost, high portability and its general convenience. On the other hand, disadvantage of PPG includes the difficulty of obtaining high quality sensor signals due to factors like body movements. The paper highlights many researches that tackle these prob-

lems using accelerometer data. Overall the paper represented PPG as an efficient technology in the healthcare field.

Takashi and Akira [3, 4] discussed the importance of monitoring core temperature that can detect risk of heart stroke. Traditionally, core temperatures are measured as rectal or tympanic temperature, which is not practical for constant monitoring. The authors proposed a method to estimate core temperature using wearable sensors. Using a number of sensors skin temperature, ambient temperatures and metabolic heat production are measured. After obtaining these values using a set of equations, the change of core and skin temperature is simulated. In the paper, the method is implemented to monitor the core temperatures for different people during 60 min exercise session. Different parameters that vary from one person to another were adjusted to get the best results. Overall using the optimal parameters, the core temperature was calculated with an average error of  $0.07^\circ$  in one hour.

Sim et al. [5] proposed a wrist type wearable device that can monitor the skin temperatures in three different parts of the wrist, i.e. radial artery, ulnar artery and upper wrist. The device was implemented to monitor the thermal comfort of the users that can improve the efficiency of users in carrying out daily activities. The device uses a tympanic temperature sensor, skin temperature sensor and an ECG sensor. During the experiment, over 100,000 skin temperatures were measured. Multiple linear regression analysis was used to calculate thermal sensation. The wrist-based device implemented was found to be more effective in measuring than the fingertip-based devices. The paper highlights the importance of such a device that can measure thermal sensation for keeping ourselves comfortable all day, which can boost the quality of life.

A similar research performed by Patrik and Braid [6] proposed a model for better estimation of core body temperature. The study conducted was aimed to define the validity of a multi-parameter model for predicting the rectal temperatures during different environmental conditions and to compare different models for measuring heat flux and insulated skin temperature at different body positions. In the paper, a model was implemented that only took two inputs which included heart rate and insulated skin temperature, whereas previous studies used models that took more input. The proposed model that takes two inputs provided similar results compared to the previously developed model that takes 18 inputs. The core body temperature was estimated with minimum error. The research showed that the minimum input model was not a good option for the estimation of the heat flux and insulated skin temperatures. The paper concluded that using values of insulated skin temperature and heart rate sensor core body temperatures can be calculated with comparable accuracy.

The relevance of IoT and its applications in times of a pandemic are being explored in the paper [1] done by Chamola and Hassi. The paper evaluated the applications of various technologies to help reduce the impact of COVID-19 on the society and improve the recovery process. The paper discusses in detail about the Coronavirus and its impact on different sectors of society. The paper discusses technologies which include IoT devices that collect, analyse and transmit health data efficiently; smart thermometers that monitor the user's data and transfer them to help generate daily



maps showing regions witnessing high fevers, currently a million such thermometers are being used in the US; telemedicine, that allows doctors to diagnose and treat patients without any physical interactions, currently such technology has been utilised by the Andhra Pradesh and Assam government in India; drones are also used in different countries around the world for various purposes like crowd surveillance, public announcement, spraying disinfectants; wearable devices such as Apple watches and Fitbits are used to monitor people's personal health, and newer technologies are also being developed for monitoring COVID-19 patients; various applications are being developed that uses blockchain technology to report, track and monitor COVID-19 patients at times of lockdown, to reduce the risk of physical contact with others; there are also numerous applications that uses AI technology for disease surveillance, risk prediction, medical diagnosis and screening, etc.; 5G technology is used in countries like China for medical imaging, thermal imaging, etc. In conclusion, paper highlights the importance and efficiency of these newer technologies that are currently being used around the world to fight the Coronavirus and help people in need.

The authors, Eknath and Rahul [7], had built a system that provides a digital healthcare assistance to patients. They have considered parameters like blood pressure, glucose level in blood and ECG. The data is processed and is sent from the user's smartphone to doctor's smartphone from where they can suggest solutions. This entire system is built on Consensus Abnormality Motif (CAM). This CAM is actually a measure in the deviation of the patient's value for the parameter with the normal value of the parameters. In their architecture, they had made use of many sensors that is attached to a patient, and accordingly, value from each sensor is compared with the reference to the normal value that is already set in the sensor. This they have visualised as a matrix in which columns of the matrix correspond to the time dependent value from the sensor and row represents value from the same sensors during the same time period. The matrix so obtained is passed to a multiplexer to generate a multi-sensor matrix (MSM) in which values are arranged in a sequence, while taking care of the order of the sensors. The output from the MUX is sent to a Physician Assist Filter (PAF) that will help in CAM discovery and analysis. This engine has three functions that are the preprocessing stage, discovery and alerting engine. In the preprocessing part, each column of the MSM is checked to find out the weighted frequency and is used to get the severity profile matrix (SPM). In the discovery stage, the PAF-CAM engine calculates the severity level of the data from sensor over a period of time. The CAM reduces the deviation in the sensor signals. This makes it a more reliable summarisation technique as far as accuracy is concerned. The last module is the alerting engine that calculates the sensor severity values (SSV). It basically represents the severity range of a certain sensor data over a span of time. They had also made use of an alert measure index (AMI), here an average score is measured from all the sensors, and using that value, severity of the patient is calculated. Based on the value obtained using the AMI, the alert engine will alert the doctor if it is greater than the threshold value.

Manisha et al. [8] implemented a system to continuously monitor health conditions of the patient without affecting their daily lives. The system uses several sensors

attached to the body for obtaining data such as blood pressure and glucose levels. The data is then transferred to a processing, analytics and storage unit. If any issue with the measured data is detected, its severity is determined. Accordingly, the system either notifies the patient about the problem or inform nearby health service provider if patient is in critical condition. The severity of the condition is determined by setting a threshold by taking in account the patient's medical history, doctor's input and some machine learning models. If the sensor data crosses this threshold value, the patient might require immediate medical attention.

Rangan, Ekanath and Pathinarupothi, Rahul conducted a research [9] that discussed the implementation of cost-effective system that monitors the health conditions of patients who are critically ill. The system uses multiple body sensor to measure blood pressure, oxygen intake, ECG, etc. The Physician Assist Filter enables the analysis of these measured data. The PAFs discovers the most abnormal pattern sequence also known as Cofonsensus Abnormal Motifs. The main aim of the implemented algorithm is to detect CAM accurately. The CAM, in readable format, can give all the details about the patient's health. Such a data can be very helpful diagnostic for the physician treating the patient. Such a system has achieved acceptance by the doctors in the field and has been deployed on a large scale for improving the health conditions of people in need.

The research conducted by Char and Magnus [10] discusses the implementation of remote stress detector. Machine learning model is used to determine the persons medical condition from the measured data, and IoT is used for communication. Medical studies have studied the correlation between stress and diseases such as cancer, heart diseases and other terminally illness. The proposed model determines the stress by monitoring the variation in heart rate. Each device is calibrated for each user for getting reliable values. Once the sensor values are obtained, different algorithms like logistic regression, support vector machine models were used for classification. The results showed that the SVM model was more accurate than the other models in predicting the stress. The paper concludes by highlighting the importance of such a model in the healthcare field.

### **3 Proposed System Architecture for the Vital Monitoring Device**

As explained in [11–14], the heartbeat sensor considered in this paper also works on the principle of photo plethysmography. This method monitors the volumetric change of blood in microvascular tissues. It uses an LDR as the light detector and a LED emitted. The resistor operates in such a way that the value of resistance varies when light falls on it. And it is to be noted that this relation between light and resistance is inversely proportional, that is if the light source of higher intensity falls on the resistor, the resistor decreases its value. As a result of the decrease in resistance, the voltage drop associated with the resistor also decreases as a side effect of Ohm's law.

It then compares the output voltage from the light detector with the threshold voltage value with the help of a comparator. The threshold voltage measures the potential drop across the light detector when light with fixed intensity falls directly on it. The non-inverting terminal of the comparator is joined to the light detector, while the other terminal that is the inverting terminal is joined to the potential divider which is tuned to the threshold voltage. So, whenever human tissues are exposed to a light source, the intensity usually decreases. When this light which is of a lower intensity falls on the light detector, resistance associated with it increases as a consequence of the previously said principle. This increase in resistance affects the voltage drop by increasing it. When the potential drop across the detector exceeds the inverting input, a logically high signal is generated, and on the other case, a logically low signal is generated at the output of the comparator. Like this, the signal generated as output will be a series of pulses which can be fed to a microcontroller for processing information to get the heartbeat rate value and can be displayed on the display screen associated with that microcontroller. We use an NTC thermistor to calculate the body temperature, and the advantage of using this technique is due to its cost effectiveness. Since this thermistor has a negative coefficient, it gives low resistance when the body temperature is high and high value for resistance when there is low body temperature.

The proposed design of the body vitals monitoring system provided in Fig. 1 consists of a heartbeat sensor and a thermistor that measure the real-time heartbeat rate of the user in terms of beats per minute (BPM) and a thermistor to measure the real-time temperature of a person. A GPS module is also included in the device to estimate the location of the user for further analysis based on the location. The output from the GPS module will be in the form of its native language and can provide the time, latitude and longitudinal coordinates of the user.

A rechargeable battery is being used to provide power to the device. The OLED module displays the real-time BPM value and temperature of the person using it. The data received from the sensors will be sent to the AWS cloud via the WIFI module connected to the device. In case a WIFI network is unavailable, the data will be sent

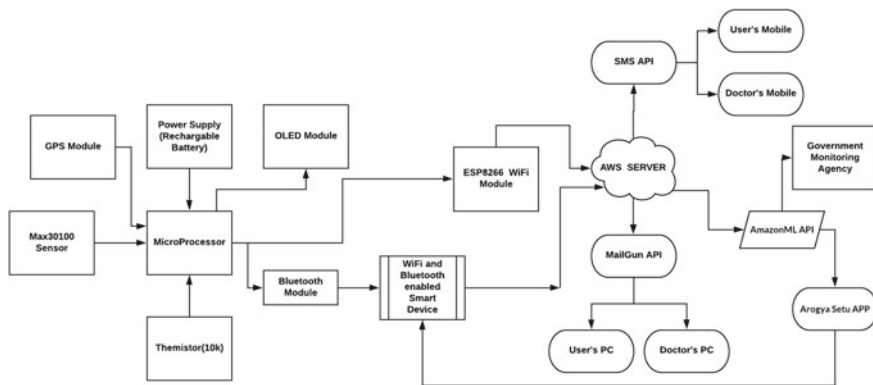


Fig. 1 COVID-19 vitals monitoring device block diagram

via the Bluetooth module to the paired smartphone of the user. The data collected by the smartphone from the device will then be sent to the cloud. An SMS alert will be sent to the user and the doctor if BPM value or the body temperature increases than a particular threshold which would be the body temperature of a healthy person in our case. Detailed automated emails are also sent to the user and the doctor from time to time using mailing APIs. The data from all the users is stored in the AWS EC2 virtual cloud server instance. The server will be scaled automatically on receiving new data objects from new patients. The health of the instances will be managed by a load balancer. The data received from the users will be sent to the model endpoints via an API Gateway. The AWS Gateway acts as a connection between the model endpoint and the Internet. The AWS Lambda service will be used as a reference point. This deployment architecture provides an output for the real-time data in sub second latency. The cloud services and the resources will be secured using a virtual private cloud (VPC). The dataset of previously affected COVID-19 patients will be used to train a model that can predict whether a person is infected with the virus based on their temperature, BPM and GPS data. The GPS data will be used in the batch processing algorithms and will also be sent to a government monitoring agency to design route maps and monitoring users on the core COVID affected regions. All the data will be transferred through the healthcare application in the smart device to communicate with the servers.

### 3.1 System Description

This following section gives a short description about the specifications of the components used.

- **Max 30100 Oximeter:** The Max 30100 pulse oximeter [15] has a low power operation mode and uses LED current to minimise the energy consumption rate, and it also has ultra low shut down current and has high sample rate capability and fast data output capability. The oxygen saturation or SPO2 level is one of the vital parameters monitored in a COVID-19 patient. SPO2 values that are less than 94% are threatening.
- **U-blox NEO-6M GPS Module:** U-blox NEO-6M GPS Module [16] operates at 40 TO 85 °C temperature range, and it uses a supply voltage of 3.3 V. UART protocol is used to collect information of latitude and longitude.
- **Negative Temperature Coefficient thermistor:** The input voltage for this negative temperature coefficient (NTC) thermistor [17] is in the range 3.3–5 V and operates at a temperature range of –25 to 80 °C with 0.1 °C precision. It has both analog and digital signals as outputs.
- **OLED Module:** The OLED display module [18] suggested in this architecture has a resolution of 128p × 64p with visual angle greater than 160°. The input voltage is 3.3–6 V. SPI protocol is used to display the text on the screen.

- **Microprocessor:** The microprocessor we are using is the Arduino Pro Mini [19] which operates at a voltage 5 V and has a clock frequency of 16 MHz. There are 14 digital I/O pins and eight analog input port. Arduino Pro Mini also supports UART, I2C and SPI communication protocols.
- **Bluetooth Module:** HC-05 Bluetooth module [20] is used in the proposed system. This module is connected to the microprocessor through UART. This module supports Bluetooth P2P and A2DP protocols. Other Bluetooth devices such as mobile phones can be paired to it and can communicate wireless.
- **WIFI Module:** We use the ESP-12F ESP8266 WIFI module [21] which has the communication interface voltage as 3.3 V and a max working current of 240 mA. The serial port baud rate is 115200 by default but can be modified to other values through AT commands.
- **Power Management:** A 3 V, 150 mAh lithium polymer (lipo) battery with battery management system was used to power the entire circuit. The battery can be recharged, and it provided up to 12 h of battery life for the proposed system. The microprocessor enters a deep sleep state and makes the sensors and communication modules idle. At this stage, it consumes 10 A, and during measurement state, around 10 mA was consumed.

## 4 Smart Device Application Implementation

A smartphone application is designed for the user to get an analysis his personal health data. The App will work on the architecture provided in Fig. 2. The application has a user interface for the initial set-up of the heart monitoring device. The device can also provide a heat map of the COVID-19 affected regions using the user's GPS location. For location tracking, permission has to be granted by the user for the application to collect the user's GPS data. An account can be created for a first-time user of the App by linking the app with their Arogya Setu or other relevant credentials. A connection will be established between the App and the wearable via Bluetooth. The weekly heart report and temperature analysis can be viewed in the app. All the health data received from the user will be sent to the cloud via this application. Continuous monitoring of the user's body vitals will be performed, and in case, the vitals data indicates that the user is at risk and several precautionary measures can be made. The users near to a certain radius of the high-risk user will also be notified to be extra cautious, hence decreasing the chance of communal spread of the virus. The integration with government COVID tracing applications helps in the seamless flow of data. Customer services can also be availed from the app.

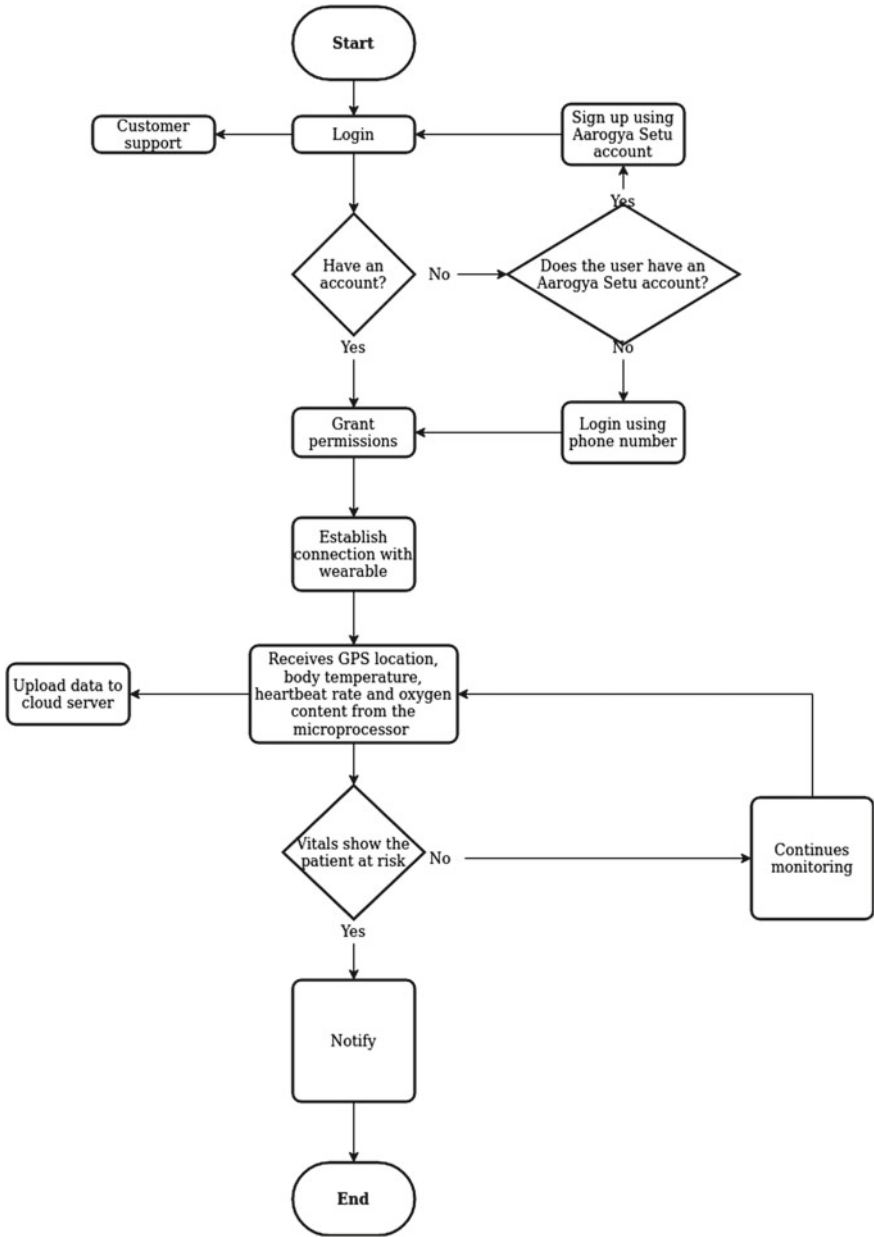


Fig. 2 Smart device application control flow

## 5 Conclusion and Future Work

In this paper, we proposed a design that captures the heartbeat rate as well as body temperature of a person using a wearable so that if the temperature and heartbeat count exceed a certain threshold, concerned government authorities can be informed, and from there, further analysis can be made to check whether the person is a victim of COVID-19.

But the temperature calculated from the skin does not provide accurate results. For accurate results, the temperature must be the rectal temperature or temperatures found from axilla, auricle canal which are considered to be the alternative areas from which core temperature can be found out. But to implement this method, it requires the usage of non-invasive IR sensors which cannot be compacted with our design.

So, our future work involves the addition of a non-invasive IR temperature sensor to our existing model so that much more accurate results can be obtained. Heartbeat sensor used in our research can measure the oxygen saturation value of the blood. But the extent of change of the oxygen saturation in the body in COVID-19 affected patients is not known so our future work will also include finding a way to incorporate the oxygen saturation level value on the prediction models.

**Acknowledgements** We express our deep gratitude to our beloved Chancellor and world-renowned humanitarian leader Shri. (Dr) Mata Amritanandamayi Devi (AMMA), for the inspiration and motivation. We would like to thank the staff and faculty members of the department for providing immense support and suggestions to improve this paper.

## References

1. V. Chamola, V. Hassija, V. Gupta, M. Guizani, A comprehensive review of the Covid-19 pandemic and the role of IoT, Drones, AI, Blockchain, and 5G in managing its impact. *IEEE Access* **8**, 90225–90265 (2020)
2. D. Castaneda, A. Esparza, M. Ghamari, C. Soltanpur, H. Nazeran, A review on wearable photoplethysmography sensors and their potential future applications in health care. *Int. J. Biosens. Bioelectron.* **4**(4), 195 (2018)
3. T.H. Takashi Hamatani, A. Uchiyama, Estimating core body temperature based on human thermal model using wearable sensors, in *SAC '15: Proceedings of the 30th Annual ACM Symposium on Applied Computing*, vol. (4), pp. 521–526 (2015)
4. M.B.B.B. Tan Suryani Sollu, Alamsyah, Monitoring system heartbeat and body temperature using Raspberry Pi, in *The 3rd International Conference on Energy, Environmental and Information System (ICENIS 2018)*, vol. 73 (2018)
5. S.Y. Sim, M.J. Koh, K.M. Joo, S. Noh, S. Park, Y.H. Kim, K.S. Park, Estimation of thermal sensation based on wrist skin temperatures. *MDPI J.* **16**(4)
6. P. Eggenberger, B.A. MacRae, S. Annaheim, Prediction of core body temperature based on skin temperature, heat flux, and heart rate under different exercise and clothing conditions in the heat in young adult males. *PMC* **9**
7. E. Rangan, R. Pathinarupothi, Rapid healthcare alerts using multiple sensors (2016)
8. M.V. Ramesh, R.K. Pathinarupothi, E.S. Rangan, P. Durga, P. Venkat Rangan, Systems, methods, and devices for remote health monitoring and management using internet of things sensors. U.S. Patent application publication US 2019/0046039 A1, Feb 14-2019

9. R.K. Pathinarupothi, P. Durga, E. S. Rangan, Data to diagnosis in global health: a 3P approach. *BMC Med. Infor. Dec. Making* **18**(1), 78:1–78:13 (2018). <https://doi.org/10.1186/s12911-018-0658-y>. <https://dblp.org/rec/journals/midm/PathinarupothiD18.bib>
10. M.D. Char, N.H. Shah, Implementing machine learning in health care—addressing ethical challenges (2018). <https://doi.org/10.1056/NEJMp1714229>
11. E.S. Rangan, R.K. Pathinarupothi, Rapid healthcare alerts using multiple sensors, in *38th IEEE Annual International Conference of Engineering in Medicine and Biology*, At Orlando, FL, USA
12. M.V. Ramesh, R.K. Pathinarupothi, Systems and methods for remote health monitoring and management
13. E.S. Rangan, R.K. Pathinarupothi, Multi-sensor architecture and algorithms for digital health at every doorstep, in *IEEE International Conference on Electrical, Computer and Communication Technologies*, Coimbatore, India (2017)
14. R.K. Pathinarupothi, E.S. Rangan, Effective prognosis using wireless multi-sensors for remote healthcare service, in *Healthwear 2016: International Conference on Wearables in Healthcare in Budapest*, Hungary (2016)
15. A. Onubeze, Developing a Wireless Heart-Rate Monitor with MAX30100 and nRF51822
16. M. Giammarini, D. Isidori, E. Conettoni, C. Cristalli, M. Fioravanti, M. Pieralisi, Design of wireless sensor network for real-time structural health monitoring, pp. 107–110 (2015)
17. S. Yun, M. Lee, K.G. Lee, J. Yi, S.J. Shin, M. Yang, N. Bae, T.J. Lee, J. Ko, S.J. Lee, An integrated and wearable healthcare-on-a-patch for wireless monitoring system, pp. 1–4 (2015)
18. I. Taryudi, A.W. Prasetyo, R.S. Nugraha, Ammar, Health care monitoring system based-on internet of things. *J. Phys.: Conf. Ser.* **1413**, 012008 (2019)
19. C.G. Butca, G. Suciuc, A. Ochian, O. Fratu, S. Halunga, Wearable sensors and cloud platform for monitoring environmental parameters in e-health applications, pp. 1–4 (2014)
20. M.A. Al-Taei, N.A. Jaradat, D.M.A. Ali, Mobile phone-based health data acquisition system using bluetooth technology, pp. 1–6 (2011)
21. Y. Yu, F. Han, Y. Bao, J. Ou, A study on data loss compensation of WiFi-based wireless sensor networks for structural health monitoring. *IEEE Sens. J.* **16**(10), 3811–3818 (2016)



# Predictive Modeling and Control of Clamp Load Loss in Bolted Joints Based on Fractional Calculus



Pritesh Shah and Ravi Sekhar

**Abstract** Safety of bolted joints in industrial machinery is of paramount importance. In this paper, fractional calculus-based predictive modeling has been investigated to control clamping force losses in bolted joints under service loads. Clamp load loss occurs in bolted joints due to application and subsequent removal of an externally applied separating service load on a fastener preloaded beyond its elastic limit. In this work, five different model structures were tried for system identification-based predictive modeling of joint clamp load loss. These structures were the first-order integer, second-order integer, first generation CRONE, fractional integral and fractional-order models. These models were validated by statistical parameters such as FIT,  $R^2$ , mean squared error, mean absolute error, and maximum absolute error. The fractional-order model with three parameters provided most accurate estimate of the system performance. It also took minimum iterations to reach the optimum controller parameter settings. This model was controlled using PID and fractional PID controllers. Fractional PID controller was designed to minimize integral of squared error (ISE) and toward the convergence of gain/order parameters. The PID controller response exhibited better time domain characteristics as compared to the fractional PID, but suffered from a maximum overshoot as well. In a physical bolted joint, clamp load loss and external service load overshoots may lead to joint failures. Maximum overshoot was totally eliminated by fractional PID controller, proving its safe applicability to the bolted joint system. By choosing a realistic set point for clamp load loss, the maximum permissible external service loading conditions were predicted successfully.

**Keywords** Predictive modeling · System identification · Fractional-order model · Fractional PID controller · Bolted joints · Clamp load loss

---

P. Shah (✉) · R. Sekhar

Symbiosis Institute of Technology (SIT), Symbiosis International (Deemed University) (SIU),  
Pune, India

e-mail: [pritesh.shah@sitpune.edu.in](mailto:pritesh.shah@sitpune.edu.in)

R. Sekhar

e-mail: [ravi.sekhar@sitpune.edu.in](mailto:ravi.sekhar@sitpune.edu.in)

## 1 Introduction

### 1.1 *Clamp Load Loss in Bolted Joints*

Bolted joints are used mainly for fastening together mechanical parts. In spite of significant advancements in joining technology, threaded fasteners remain preferred medium of clamping parts together for assembly. Bolted joints find clamping applications across industries for all kinds of assembly requirements because they can be disassembled for maintenance. Loss in clamping force or clamp load loss may lead to joint failures, potentially leading to large-scale losses depending on the nature of industrial/structural joint application. Many researchers have investigated the nature, magnitude, and mechanics of clamp load loss in bolted joints.

Nasser and Matin [1] investigated clamp load losses for fasteners tightened beyond elastic limit of bolt material, under separating service loads. Lambert [2] studied the effects of friction coefficient variations on clamping force of bolted connections. Fazekas [3] developed a linear model of bolted joint subjected to alternating tension with a modified Goodman diagram. Groper [4] developed a methodology to measure preload in fastener using Preload Direct Measuring Device. Monaghan [5] tightened high strength fasteners to yield for maximizing joint clamp loads under lubrication conditions. Duffey [6] determined optimal prestress to minimize peak bolt stress for closure bolting systems. Pai and Hess [7] studied loosening of threaded fasteners under dynamic shear. Investigators have come up with a number of linear/nonlinear mathematical models describing the phenomenon of clamp load under different conditions. However, it seems that fractional calculus has not been applied for system identification of bolted joint systems for clamp load losses. This kind of modeling is necessary in order to control and limit clamp load loss generation in a bolted joint.

### 1.2 *Fractional Modeling*

Fractional calculus precedes classical calculus by more than three centuries. However, it requires more exploration by researchers [8]. Recently, there has been a steady rise in fractional calculus-based research investigations in engineering, science, and non-engineering fields (fractional PID controller, speech signal processing, modeling of physical systems, cancer dynamics, finance, and more) [9–15]. The fractional calculus models have been implemented for system identification of semiconductor diodes, capturing better dynamic characteristics often overlooked by classical models [16]. Fractional model has proved to accurately estimate nonlinear high-level voltage dynamics for ultracapacitance impedance modeling [17]. The fractional-order tracking differentiator (FOTD) model has exhibited satisfactory performance in many instances [18]. Nonlinear fractional model has been applied in modeling of thermal systems for large temperature variations [19]. Fractional model has been applied to control the robotic systems as well [20]. In the current work, fractional calculus has

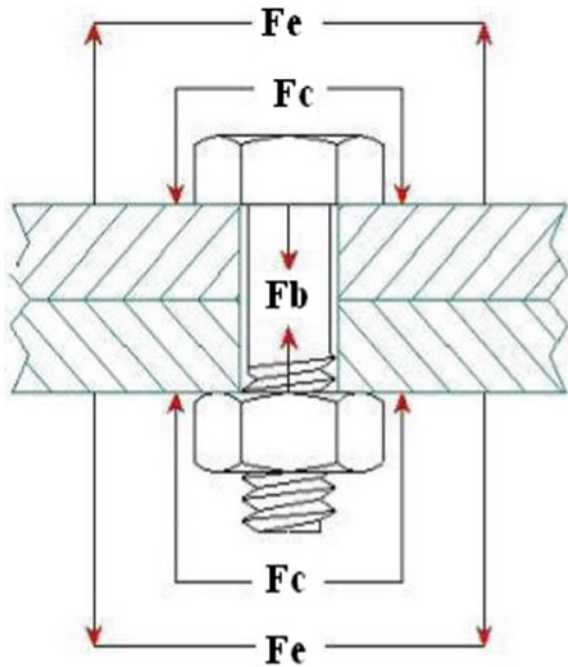
been employed to identify the clamp load loss system in a bolted joint based on analytical formulations from literature and experimental data.

## 2 Clamp Load Loss Determination

In a bolted joint system, initial tightening  $F_i$  produces bolt tension  $F_b$  and clamp load  $F_c$  (Fig. 1) opposite in nature but equal in magnitude.  $F_e$  is an externally applied service load that attempts to loosen the joint.

Clamp load loss  $\Delta F_c$  can be approximated by the following formulation [1].

**Fig. 1** Schematic of a bolted joint [21]



**$F_e$  = Separating Force**

**$F_b$  = Bolt Tension**

**$F_c$  = Clamping Force**

$$\Delta F_c = \frac{\left( \frac{K_b + K_c}{K_c + \frac{NF_e^{1-1/n}}{L \left( \frac{1}{A_0 K} \right)}} - 1 \right)}{\left( 1 + \frac{K_b}{K_c} \right)} F_e \quad (1)$$

where  $K$  is strain hardening coefficient,  $n$  is the strain hardening exponent,  $L$  is effective fastener grip length,  $A_0$  is fastener tensile stress area,  $K_b$  is bolt stiffness, and  $K_c$  is joint stiffness.

## 2.1 Experimentation Details

A tensile test was conducted for a Class 8.8 M108 \* 150 bolt, and the material strain hardening exponent  $n$  and strain hardening coefficient  $K$  were determined. Modulus of elasticity of the material was also determined from the same, which was used to compute bolt material stiffness,  $K_b$ . As a thumb rule, joint stiffness is designed to be much higher than bolt stiffness (around five times) for joint safety [22]. So,  $K_c$  was assumed to be equal to  $5 K_b$ . Bolt was preloaded (tightened) to 63,600 N ( $F_e$ ) to ensure it reaches plastic elongation range (i.e., beyond the elastic limit determined experimentally, at 62,140 N).

Externally applied separating force,  $F_e$ , was varied from zero 7000 N to estimate values of clamp load loss,  $\Delta F_c$  based on the formulation from literature (Eq. 1).

## 3 System Identification Using Fractional and Integer Models

System identification is used for control design, predicting system behavior, fault diagnosis, etc. At present, many industries have automated their processes. Due to this, their process data is easily available. Model development based on system inputs and outputs is called system identification. System identification can be performed for open-loop and closed-loop experimental data based on applications. It involves data collection, model structure selection, model parameter estimation, and model validation. Of these, model structure selection is the most vital step. The probable model structures can range from transfer functions and ordinary differential equations (ODE) to fractional differential equations (FDE), state space models, time series models and others.

### 3.1 Model Structures

The present work considered fractional order, fractional integer, and CRONE (first generation) model structures for identifying the clamp load loss system under consideration. Integer models (first and second order) were also explored to compare with the performance of the fractional structures. In the first-order model, two parameters were minimized namely gain ( $K$ ) and time constant parameters ( $\tau$ ), whereas in second-order model, three parameters were minimized namely  $K$ , damping ration  $\zeta$  and natural frequency of oscillation ( $\omega_n$ ).

### 3.2 Estimation of Model Parameters

After model structure selection, parameters of the selected model structure were estimated by minimizing the sum of square error (SSE) function (Fig. 2). In the present work, model parameters were tuned by genetic algorithm through minimization of SSE function.

The output error method [23] can be utilized for model parameter estimation. It is given as follows:

$$e_n = y_n^* - \hat{y}_n(u_n, \hat{\theta}) \tag{2}$$

where  $\theta$  is the exact model parameter under consideration and  $\hat{\theta}$  is its required estimation considering the dataset  $\{u_n, y_n^*\}$  having  $n$  points for identification.  $y_n^*$  is the noise parameter corresponding to the exact system output  $y_n$ .  $\hat{\theta}$  can be optimized by minimizing mean square predication error given as follows:

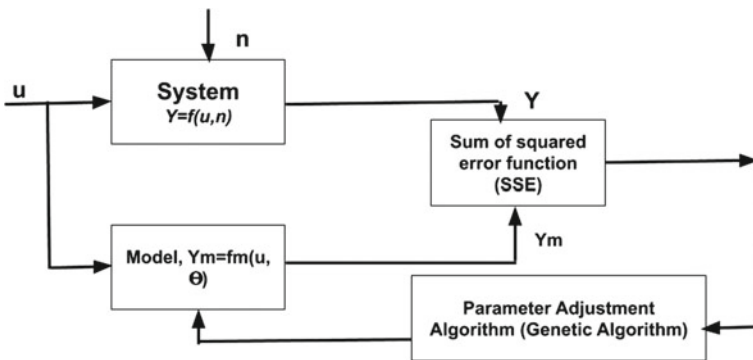


Fig. 2 Parameter estimation for system identification

$$J = \sum_{n=1}^N e_n^2 \quad (3)$$

### 3.3 Model Validation

Model validation must include multiple indices to ensure effectiveness of the model selection procedure. First among such indices is the FIT factor, calculated as follows

$$\text{FIT} = 100 * \left( 100 - \frac{\|y - \bar{y}\|_2}{\|y - \text{mean}(y)\|_2} \right) \quad (4)$$

where  $y$  is the actual output and  $\bar{y}$  is model output. For a perfect fit, this value will reach 100 [24]. Other indices included sum of square error, R-squared, mean squared error (MSE), mean absolute error (MAE), and maximum absolute error (MaxAE), defined as follows:

$$\text{SSE} = (y - \bar{y})^2 \quad (5)$$

$$R^2 = 1 - \frac{\sum_{n=1}^N (y_n - \text{mean}(y))^2}{\sum_{n=1}^N (y_n - \bar{y})^2} \quad (6)$$

$$\text{MAE} = \text{mean}(\text{abs}(y - \bar{y})) \quad (7)$$

$$\text{MaxAE} = \max(\text{abs}(y - \bar{y})) \quad (8)$$

$$\text{MSE} = \text{mean}((y - \bar{y})^2) \quad (9)$$

All of the above-mentioned validation indices were considered together for validating system models in the current work.

## 4 Control Using PID and Fractional PID Controllers

PID controller is widely used in industry [25] owing to its simplicity. A PID controller is represented mathematically as:

$$C(s) = P + I \frac{1}{s} + D \frac{N}{1 + N \frac{1}{s}} \quad (10)$$

where  $P$  is the proportional gain constant,  $I$  is the integration gain constant,  $D$  is the derivative gain constant, and  $N$  is the filter coefficient.

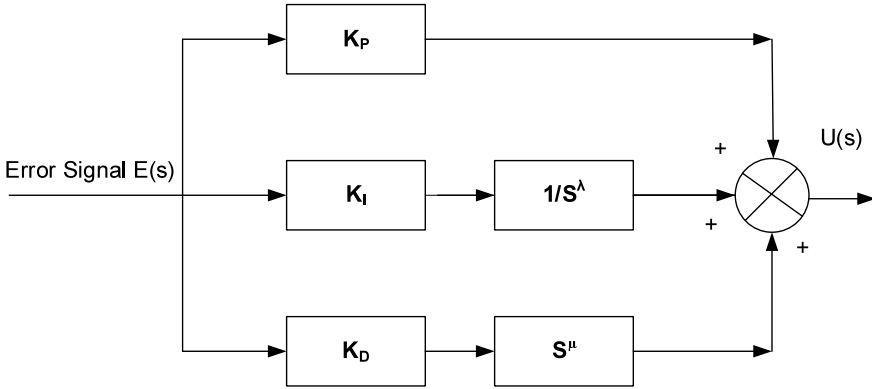


Fig. 3 Block diagram of fractional PID controller

In 1994, Podlubny introduced the fractional-order controller for fractional-order systems [26–28]. This controller is more robust against variations in system variables and controller tuning parameters [29, 30] as compared to classical controllers due to its iso-damping characteristics [26]. There are five parameters available in this controller for tuning (Eq. 11). A fractional PID controller can be represented schematically as shown in Fig. 3. The fractional PID controller is represented mathematically as [31, 32]:

$$C(s) = \frac{U(s)}{E(s)} = K_P + \frac{K_I}{s^\lambda} + K_D s^\mu, (\lambda, \mu \geq 0) \tag{11}$$

where,  $C(s)$  is controller transfer function,  $U(s)$  and  $E(s)$  are the control and error signals.  $K_P$ ,  $K_I$  and  $K_D$  are the proportional, integral and derivative gain constants, respectively.  $\mu$  and  $\lambda$  are differentiation and integration orders.

For most applications, the order of the fractional-order PID controller is kept in the range 0 to 2 [33–35]. Further literature [36–38] may be consulted for detailed information on characteristics of fractional PID controllers.

## 5 Results and Discussions

### 5.1 System Identification of Clamp Load Loss

Table 1 shows model structures, sum of squared errors, R-squared, number of iterations, MAE, MSE, etc., for each of the developed models. With respect to the  $R^2$  parameter, all models perform well except for the first generation CRONE model. In fact, CRONE model appears to disappoint on all statistical parameters for the current bolt joint system. Best FIT is offered by the fractional-order model, while

**Table 1** Comparison of models trained for bolt System

S. no.	Model	Model structure	$R^2$	FIT	SSE	MAE	MSE	MaxAE	Iterations
1	First generation CRONE	$23.4824s^{0.8361}$	0.402	22.6653	$4.77E+07$	229.2125	$6.80E+04$	426.7315	200
2	Fractional integral model	$\frac{0.1213}{s^{0.0651}}$	0.9987	96.3946	103556.09	10.9259	147.8585	22.3949	500
3	Fractional-order model	$\frac{15.1527}{s^{1.8039}+90.9163}$	1	99.9997	$3.39E-04$	$6.97E-04$	$1.04E-06$	0.0181	108
4	First-order integer model	$\frac{0.2016}{1+74.4211s}$	0.9957	93.4612	$3.41E+05$	18.7392	486.3469	38.6698	200
5	Second-order integer model	$\frac{80.34}{s^2+2.941s+482.1}$	1	99.9767	4.3338	0.0656	0.0062	0.1563	300



the second-order integer model came a close second. The fractional-order model scores significantly better over all others in error estimates of SSE, MAE, MSE, and MaxAE.

CRONE model is composed of two parameters and takes 200 iterations to reach optimal parameter solutions.

Fractional integer model has two parameters and takes 500 iterations to reach optimal parameter solutions. Fractional-order model has three parameters and reaches optima at 108 iterations, the least among all models. Fractional models give better responses as compared to CRONE model because of the presence of integration structure in fractional models.

Predictions of all identified models were compared with actual output (Figs. 4, 5, 6, 7 and 8). The fractional-order model (Fig. 6) estimated the actual clamp load loss data very well.

The response of first and second integer-order models is shown in Figs. 7 and 8.

The fractional-order model has been taken as a reference to compare other models because of its superior estimation characteristics listed in Table 1.

### 5.2 Control of Bolt System

PID and fractional PID model controllers were applied on the fractional-order model (Model 3) to check how effectively clamp load loss could be controlled at a given set point. The control signal outputs were also plotted for corresponding external loading conditions.

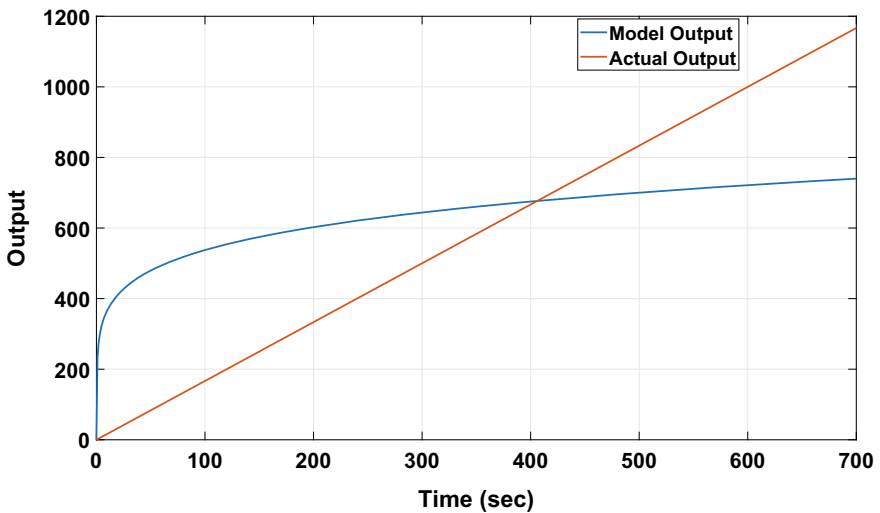


Fig. 4 Prediction using fractional CRONE model (Model 1)

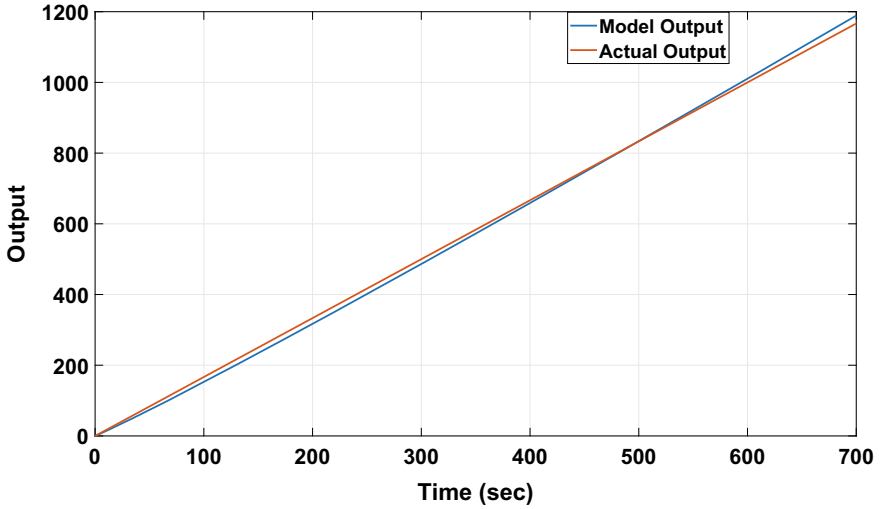


Fig. 5 Prediction using fractional integration model (Model 2)

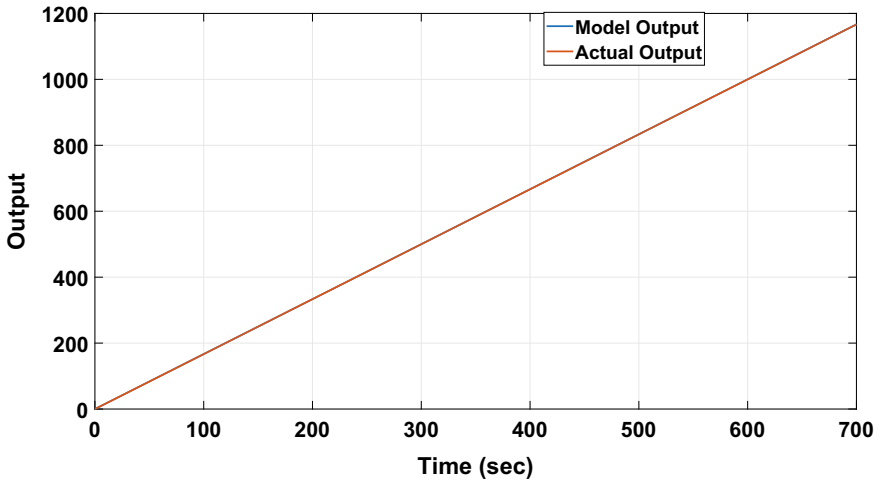


Fig. 6 Prediction using fractional-order model (Model 3)

Figures 9 and 10 are PID controller response and control signal plots, respectively. In this case,  $\Delta F_C$  (clamp load loss) setpoint was kept at 1 N. The PID controller response contained an initial overshoot followed by a tiny undershoot before settling at the designated set point. PID control signal exhibited high under and overshoots before settling in 6 N value for  $F_e$  (external load). Table 2 shows the PID controller parameters for the same.

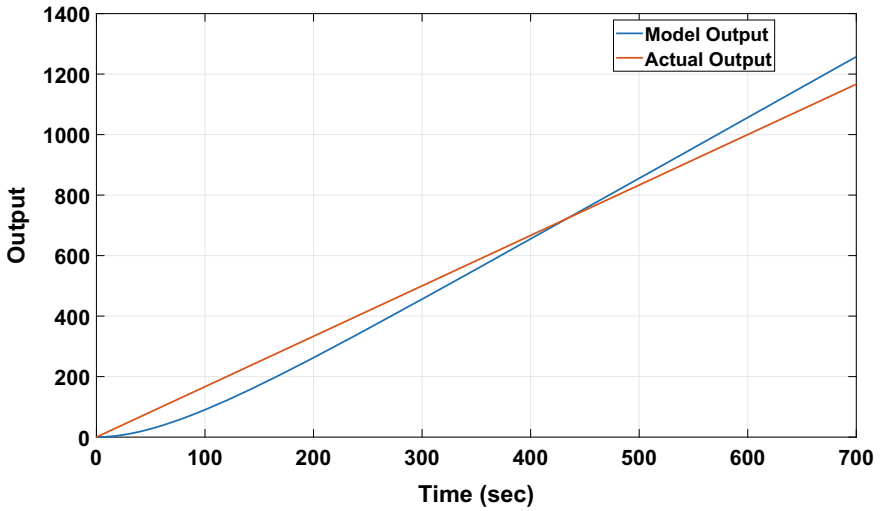


Fig. 7 Prediction using first-order integer model (Model 4)

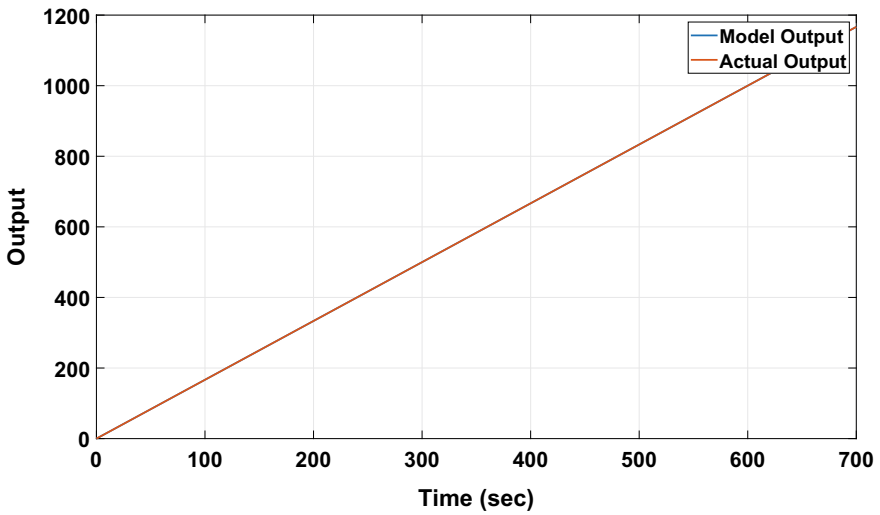


Fig. 8 Prediction using second-order integer model (Model 5)

Table 2 PID controller parameters

S. no.	Parameter	Value
1	$K_p$	2707.18
2	$K_i$	91992.41
3	$K_d$	16.67
4	$N$	2808.02

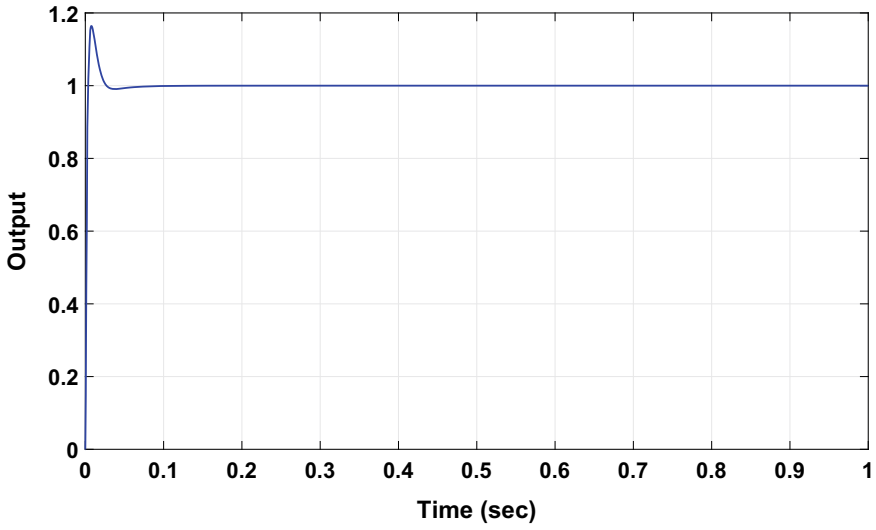


Fig. 9 Fractional-order model (three parameters): PID controller response ( $\Delta F_c$  set point 1 N)

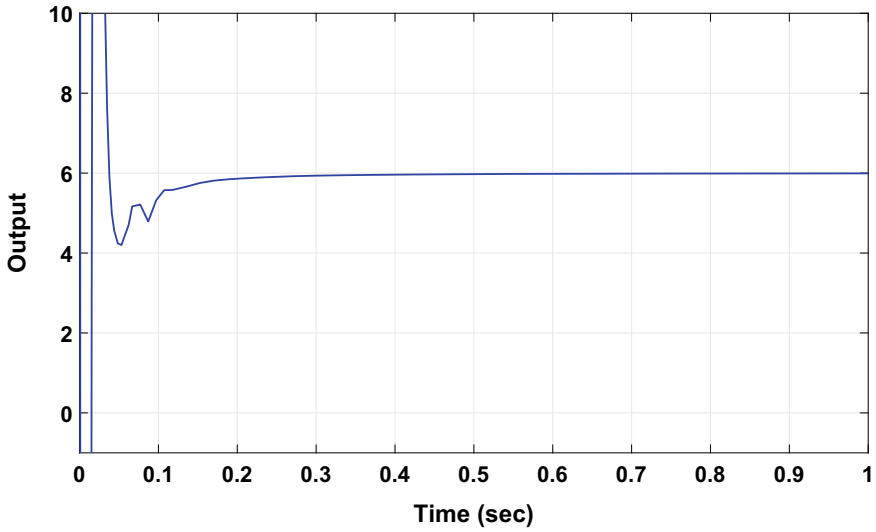


Fig. 10 Fractional-order model (three parameters): PID control signal ( $F_e$  at  $\Delta F_c$  set point 1 N)

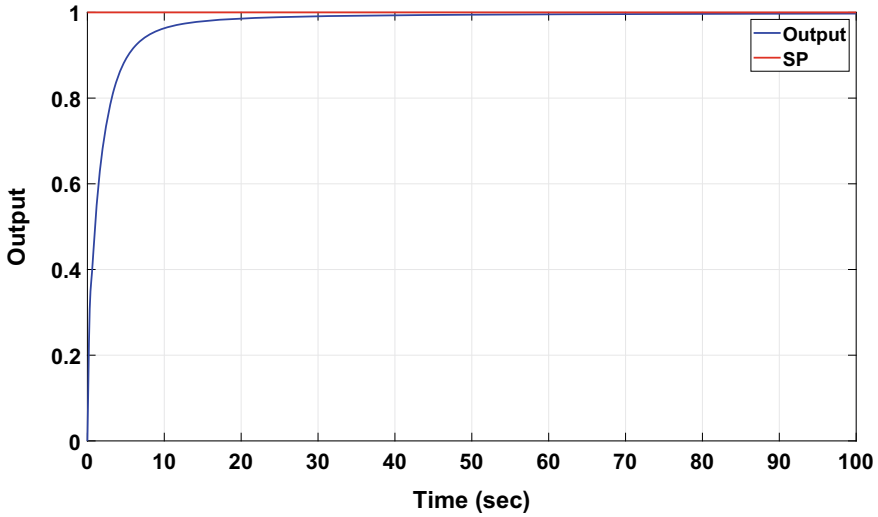


Fig. 11 Fractional-order model (three parameters): fractional PID controller response ( $\Delta F_c$  set point 1 N)

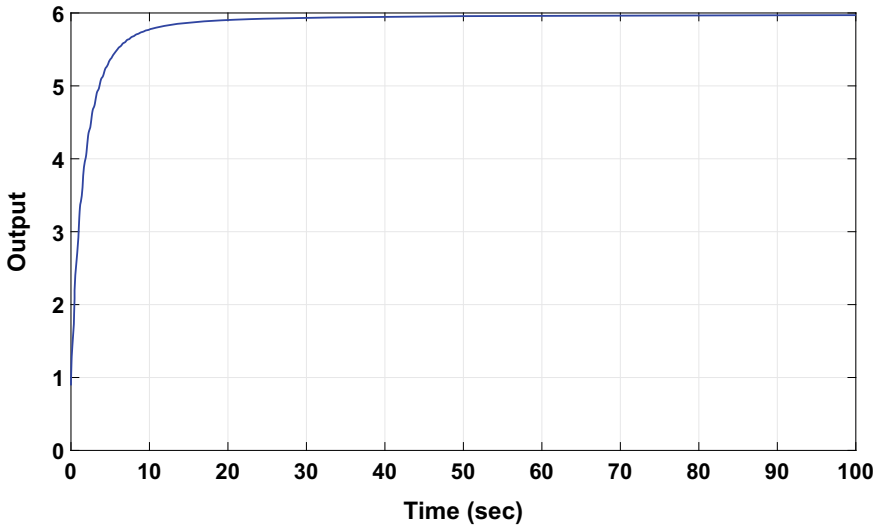


Fig. 12 Fractional-order model (three parameters): fractional PID control signal ( $F_c$  at  $\Delta F_c$  set point 1 N)

Figures 11 and 12 are FPID controller response and control signal plots for the fractional-order model (Model 3). For comparisons with PID responses,  $\Delta F_c$  (clamp load loss) setpoint was kept at 1 N. It may be observed how FPID controller response removes the initial overshoot seen in PID controller response before settling at the

**Table 3** Fractional PID controller parameters

S. No.	Parameter	Value
1	$K_p$	0.89156
2	$K_i$	3.9378
3	$\lambda$	0.89776
4	$K_d$	0.0007068
5	$\mu$	0.11502

designated set point. Similarly, FPID control signal is able to smoothen over and undershoots in PID responses before settling in 6N value for  $F_e$  (external load). Table 3 shows the FPID controller parameters for the same.

After successful control of the bolted joint plant by FPID controller, the response set point was changed to 10,000 N as a realistic setting to limit clamp load loss ( $\Delta F_c$ ) and avoid joint failure. This set point was also successfully attained as shown in Fig. 13. As may be observed from Fig. 14, the control signal output for the same manipulated variable: external load ( $F_e$ ) is 60,000 N.

This result implies that the external loading conditions are to be limited to 60,000 N in this bolted joint's service cycle in order to limit the joint clamp load loss to 10,000 N post withdrawal of the external load. In other words, a residual clamp load of only 53,600 N (63,600 N initial preload minus 10,000 clamp load loss) may be expected due to the application and subsequent removal of an externally separating tensile load of 60,000 N on the joint. The ratio of the joint preload to the external load limit determined by FPID control signal, i.e.,  $F_i/F_e$  ratio is 1.06, which is a good estimate of actual service loading conditions [1].

Controller time domain specifications are shown in Table 4. Fractional PID controller achieves 0% maximum overshoot whereas PID controller exhibits 18% overshoot. However, the PID controller has lower rise, peak and settling times over fractional PID controller. Considering physical bolted joint systems, overshoot may damage the physical system architecture. In the current case of bolted joints as well, overshoot in  $\Delta F_c$  can lead to failure of the actual physical joint. Similarly, extreme over and undershoots of the external load  $F_e$  observed in the PID control signal output can lead to incorrect service load design specifications. Hence, the fractional PID controller response is preferable as per these vital safety aspects as compared to the PID controller.

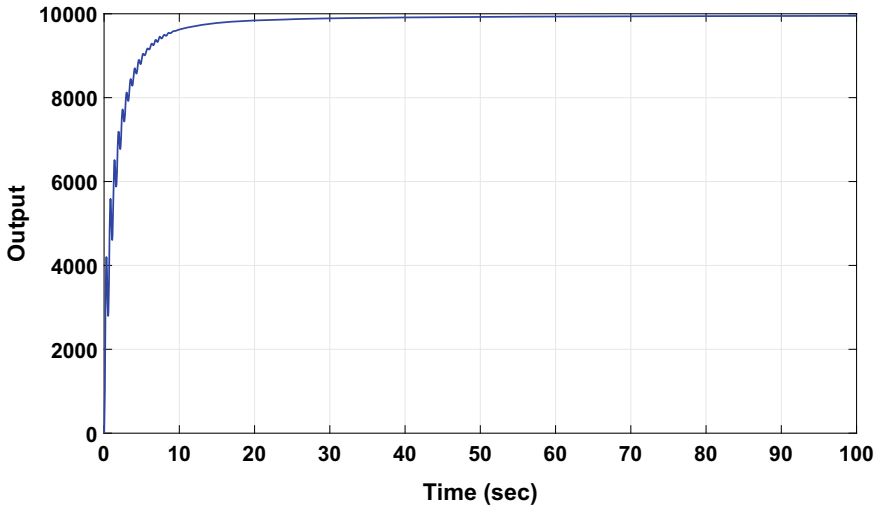


Fig. 13 Fractional-order model (three parameters): fractional PID controller response ( $\Delta F_c$  set point 10,000 N)

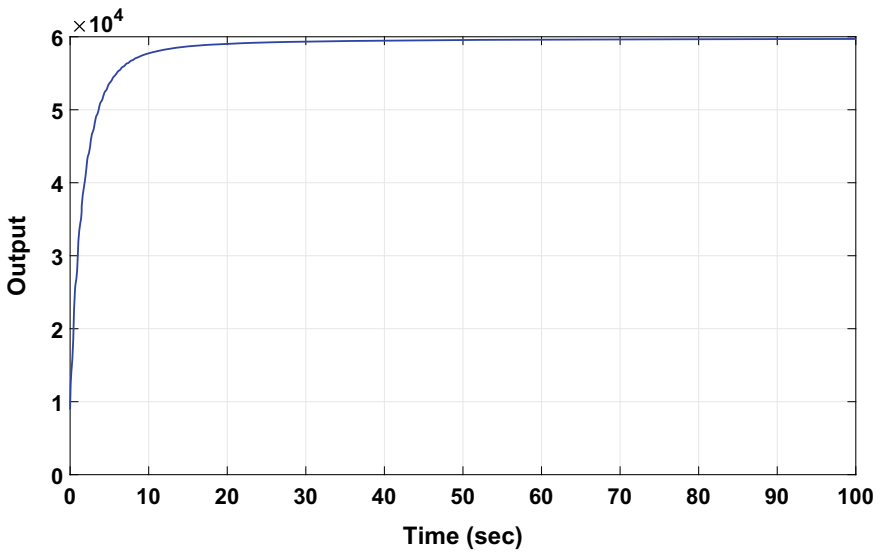


Fig. 14 Fractional-order model (three parameters): fractional PID control signal ( $F_c$  at  $\Delta F_c$  set point 10,000 N)

**Table 4** Time domain specifications table

S. no.	Open loop	PID controller	Fractional PID controller
Rise time (sec)	NA	0.01	7
Peak time (sec)	NA	0.02	NA
Maximum overshoot (%)	67	18	0
Settling time (sec)	NA	0.05	22

## 6 Conclusions

Accurate estimation and control of clamp load loss are of critical importance in bolted connections. In the present work, different models were implemented to identify clamp load loss system in a bolted joint plant. These models included first- and second-order integer models; fractional integral and fractional-order models; and a first generation CRONE model as well. To compare their relative statistical performances, parameters like  $R^2$ , FIT, maximum absolute error, etc., were evaluated. Model predictions were plotted against actual system responses. The fractional-order model with three parameters emerged as the best predictive model. This model was controlled by PID and fractional PID controllers at different set points of the output parameter (clamp load loss,  $\Delta F_c$ ). Fractional PID controller consumed more rise, peak, and settling times as compared to the PID controller responses. However, maximum overshoot observed in PID controller response (clamp load loss,  $\Delta F_c$ ) was completely eliminated by the fractional PID controller. Similarly, control signal (external load  $F_e$ ) over and undershoots were also effectively avoided by fractional PID controller. In an actual bolted joint, such overshoots in clamp load loss and/or external service load may lead to joint failures. Thus, fractional PID controller was proved to be safer for predictive modeling of real physical systems such as the bolted joint considered in the present study. Based on the fractional PID control signal response at realistic physical clamp load loss limit set point, the appropriate external service loading conditions were successfully predicted.

## References

1. S.A. Nassar, P.H. Matin, Nonlinear strain hardening model for predicting clamp load loss in bolted joints. *J. Mech. Des.* **128**(6), 1328–1336 (2006)
2. T. Lambert, Effects of variations in the screw thread coefficient of friction on the clamping force of bolted connections. *J. Mech. Eng. Sci.* **4**(4), 401–406 (1962)
3. G. Fazekas, On optimal bolt preload. *J. Eng. Ind.* **98**(3), 779–782 (1976)
4. M. Groper, Measuring preload in fasteners. *Exp. Tech.* **9**(1), 28–29 (1985)
5. J.M. Monaghan, The influence of lubrication on the design of yield tightened joints. *J. Strain Anal. Eng. Des.* **26**(2), 123–132 (1991)



6. T. Duffey, Optimal bolt preload for dynamic loading. *Int. J. Mech. Sci.* **35**(3–4), 257–265 (1993)
7. N.G. Pai, D.P. Hess, Dynamic loosening of threaded fasteners. *Noise Vib. Worldwide* **35**(2), 13–19 (2004)
8. Y. Chen, B.M. Vinagre, *Fractional-Order Systems and Controls: Fundamentals and Applications* (Springer, Berlin, 2010)
9. S. Das, *Functional Fractional Calculus* (Springer Science & Business Media, Berlin, 2011)
10. R. Sekhar, T. Singh, P. Shah, ARX/ARMAX modeling and fractional order control of surface roughness in turning nano-composites, in *2019 International Conference on Mechatronics, Robotics and Systems Engineering (MoRSE)* (IEEE, New York, 2019), pp. 97–102
11. P. Shah, R. Sekhar, Closed loop system identification of a DC motor using fractional order model, in *2019 International Conference on Mechatronics, Robotics and Systems Engineering (MoRSE)* (IEEE, New York, 2019), pp. 69–74
12. P. Shah, R. Sekhar, S. Agashe, Application of fractional PID controller to single and multi-variable non-minimum phase systems. *Int. J. Recent Technol. Eng.* **8**(2), 2801–2811 (2019)
13. E. Balç, I. Ozturk, S. Kartal, Dynamical behaviour of fractional order tumor model with caputo and conformable fractional derivative. *Chaos, Solitons & Fractals* **123**, 43 – 51 (2019)
14. K. Fatmawati et al., A fractional model for the dynamics of competition between commercial and rural banks in Indonesia. *Chaos, Solitons & Fractals* **122**, 32–46 (2019)
15. W. Ma, M. Jin, Y. Liu, X. Xu, Empirical analysis of fractional differential equations model for relationship between enterprise management and financial performance. *Chaos, Solitons & Fractals* **125**, 17–23 (2019)
16. J.T. Machado, A.M. Lopes, Fractional-order modeling of a diode. *Commun. Nonlinear Sci. Numer. Simul.* **70**, 343–353 (2019)
17. J.-D. Gabano, T. Poinot, H. Kanoun, LPV continuous fractional modeling applied to ultracapacitor impedance identification. *Control Eng. Pract.* **45**, 86–97 (2015)
18. Y. Wei, Q. Gao, Y. Chen, Y. Wang, Design and implementation of fractional differentiators, part I: system based methods. *Control Eng. Pract.* **84**, 297–304 (2019)
19. A. Maachou, R. Malti, P. Melchior, J.-L. Battaglia, A. Oustaloup, B. Hay, Nonlinear thermal system identification using fractional volterra series. *Control Eng. Pract.* **29**, 50–60 (2014)
20. A.P. Singh, D. Deb, H. Agarwal, On selection of improved fractional model and control of different systems with experimental validation. *Commun. Nonlinear Sci. Numer. Simul.* **79**, 104902 (2019)
21. R. Sekhar, V. Jadhav, Effect of strain hardening rate on the clamp load loss due to an externally applied separating force in bolted joints. *Indian J. Appl. Res.* **1**(10), 61–63 (2011)
22. J.H. Bickford, *An Introduction to the Design and Behavior of Bolted Joints* (Dekker, 1995)
23. T. Poinot, J.-C. Trigeassou, Identification of fractional systems using an output-error technique. *Nonlinear Dyn.* **38**(1–4), 133–154 (2004)
24. L. Chen, B. Basu, D. McCabe, Fractional order models for system identification of thermal dynamics of buildings. *Energy Build.* **133**, 381–388 (2016)
25. K.J. Åström, T. Hägglund, P.I.D. Controllers, *Theory, Design, and Tuning*, vol. 2 (Instrument society of America Research, Triangle Park, NC, 1995)
26. I. Podlubny, *Fractional-order systems and fractional-order controllers*, in *UEF-03-94* (Institute of Experimental Physics of the Slovak Academy Science, Kosice, 1994), pp. 1–24
27. I. Podlubny, *Fractional Differential Equations an Introduction to Fractional Derivatives, Fractional Differential Equations, to Methods of their Solution and Some of Their Applications* (Academic Press, San Diego, 1999)
28. I. Podlubny, L. Dorcak, I. Kostial, On fractional derivatives, fractional-order dynamic systems and  $PI^\lambda D^\mu$ -controllers, in *Proceedings of the 36th IEEE Conference on Decision and Control, 1997*, vol. 5, pp. 4985–4990 (1997)
29. Y. Luo, Y.Q. Chen, C.Y. Wang, Y.G. Pi, Tuning fractional order proportional integral controllers for fractional order systems. *J. Process Control* **20**, 823–831 (2010)
30. H. Malek, Y. Luo, Y. Chen, Identification and tuning fractional order proportional integral controllers for time delayed systems with a fractional pole. *Mechatronics* **23**(7), 746–754 (2013)

31. P. Shah, S. Agashe, and A. Singh, Design of fractional order controller for undamped control system, in *2013 Nirma University International Conference on Engineering (NUiCONE)*, pp. 1–5 (2013)
32. R.S. Barbosa, J. Tenreiro Machado, A.M. Galhano, Performance of fractional PID algorithms controlling nonlinear systems with saturation and backlash phenomena. *J. Vib. Control* **13** (9-10), 1407–1418 (2007)
33. C.I. Muresan, S. Folea, G. Mois, E.H. Dulf, Development and implementation of an FPGA based fractional order controller for a DC motor. *Mechatronics* **23**(7), 798–804 (2013)
34. B.B. Alagoz, A. Ates, C. Yeroglu, Auto-tuning of PID controller according to fractional-order reference model approximation for DC rotor control. *Mechatronics* **23**(7), 789–797 (2013)
35. P. Shah, S. Agashe, Experimental analysis of fractional PID controller parameters on time domain specifications. *Progress Fract. Different. Appl.* **3**, 141–154 (2017)
36. P. Shah, S. Agashe, Review of fractional PID controller. *Mechatronics* **38**, 29–41 (2016)
37. P. Shah, S. Agashe, A.J. Kulkarni, Design of a fractional  $PI^\lambda D^\mu$  controller using the cohort intelligence method. *Front. Inform. Technol. Electron. Eng.* **19**, 437–445 (2018)
38. P. Shah, A.J. Kulkarni, Application of variations of cohort intelligence in designing fractional PID controller for various systems, in *Socio-Cultural Inspired Metaheuristics* (Springer, Berlin, 2019), pp. 175–192

# Resource Allocation for 5G RAN—A Survey



G. Shanmugavel and M. S. Vasanthi

**Abstract** Resource allocation (RA) is a fundamental task in the design and management of wireless signal processing and communication networks. In a wireless communication, we must wisely allocate some available radio resources like time slots, transmission power, frequency band, and transmission waveforms or codes across multiple interfering links as to accomplish a better framework execution while guaranteeing user fairness and quality of service (QoS). In fifth generation (5G) of wireless communication system provides a better mobile service with improved QoS everywhere. Considering the dense deployment and more number of network nodes, RA and interference management are the important research issues in heterogeneous mobile networks. In this, we need to utilize the available radio resources efficiently, for that the RA is of much importance in future wireless communication systems (5G/6G). In this survey, we consider various resource allocation methods for different radio access network (RAN) architecture; several authors have implemented some techniques and algorithms to achieve better resource allocation with the help of existing literature survey, we explore ways to allocate the radio resources for next generation wireless communication.

**Keywords** 5G · Radio access network · Resource allocation · Reinforcement learning · NOMA · Markov decision process · Fog RAN · Cloud RAN

---

G. Shanmugavel (✉)

Department of Electronics and Communication Engineering, SRM Institute of Science and Technology, Kattankulathur, Chennai 603203, Tamilnadu, India

e-mail: [shanmugg@srmist.edu.in](mailto:shanmugg@srmist.edu.in)

M. S. Vasanthi

Department of Telecommunication and Engineering, SRM Institute of Science and Technology, Kattankulathur, Chennai 603203, Tamilnadu, India

e-mail: [vasanthm2@srmist.edu.in](mailto:vasanthm2@srmist.edu.in)

## 1 Introduction

In fifth-generation (5G) mobile communication systems, in addition to provide a wide coverage, improved spectral efficiency, ultra-reliable low-latency communication, reduction in energy usage, more number of connected devices, improved accessibility, and also providing a very huge data rates in terms of giga-bit-per-second (Gbps) all the cell coverage area [1]. The cell coverage range and total sum capacity solutions are proposed for next generation mobile communication, beam-forming, carrier aggregation, higher level modulation, and dense deployment of small cells [2]. For high bandwidth capacity in 5G, the millimeter wave communication is used [3]. Using multiple input and multiple output (MIMO) in 5G, it provides high spectral bandwidth efficiency and better energy saving. [4]. To increase more IoT nodes or equipments in a cell, it is providing more traffic congestion to satisfy all user requirements; the cloud radio access network (C-RAN) is introduced to 5G mobile communication; the best powerful cloud controller (CC) in C-RAN network architecture has the remote radio units (RRU) and baseband units (BBU) [5, 6].

5G wireless communication requires more QoS and also wants to overcome the drawbacks of the previous generation mobile communication systems. Some of the research challenges like massive type communication (mMTC), enhanced mobile broadband (EMBB), and ultra-reliable low-latency communication (uRLLC) are required. These types of services are lead to find new RAN. Compared to C-RAN, the F-RAN provides the uRLLC, this type of service required in future generation wireless communication [7].

Delivery of huge value of data from UE to fog access point it need a large amount of bandwidth, and very high spectral bandwidth is demand in radio resources. For very high spectral efficiency, very low-latency requirement, and multiple access facility, NOMA is one of the best method [8]. In NOMA, SIC can used in F-AP to separate various user's signals. A general fact is that the F-AP usually do not have enough storage capacity and computing resources and may not meet large-scale users' services; therefore, the implementation scheme of NOMA will have great impact on F-RAN. There are various types of researches about NOMA and F-RAN [9]. Implementation of F-RAN based of NOMA RA technique it increase the QoS and reduce the latency [10]. In the transmitter and receiver section of NOMA system, the successive interference cancelation is implemented in receiver side. In [11], multiuser detection and decoding is implemented to optimize receiver ability. In [12], influence of error vector magnitude to various SIC in DL NOMA is very accuracy. In [13], the research challenges in multi-tier heterogeneous networks are resource allocation, dense deployment, huge network nodes, and interference management. The radio resource allocation issues in multi-tier OFDMA based in 5G LTE-A. Specifically, author introduced three novel methodologies for distributed RA in such networking ideas of message passing, distributed auction, and stable matching. In [14], the cloud node characteristics and 5G services are taken by the joint optimal virtual network functions (VNF).

To improve the system throughput of the used level resource scheduling algorithm, a slice level scheduling algorithm is proposed based on the requirements of wireless network slicing technology and the definition of functional scenarios. VNF provides better radio resource utilization and huge gain in network slicing for 5G communications. The slicing characteristics of every user and the scheduling method priority are calculated by PF algorithm [15].

## 2 Heterogeneous IoT and F-RAN

In C-RAN, the latency issue becomes very high, so it is not suitable for IoT services. For 5G need a very low-latency communication service, for that we move C-RAN to F-RAN (Fig. 1).

The fog node provides heterogeneous low-latency requirements of the IoT application devices, and it is directly linked to the cloud network through fronthaul connections. Red lines mentioned a local service by fog node to fulfill low-latency needs.

A hybrid cloud supporting C-RAN system (see Fig. 2). A different set of requirements are provided by RRHs like massive machine type communications (mMTC), ultra-reliable low-latency communications (URLLC) and enhanced mobile broadband (eMBB), (see Fig. 3).

## 3 Resource Allocation Methodologies

The different resource allocation methods are considered from various existing possible techniques or algorithm like reinforcement learning (RL), cooperative edge computing, Markov decision process (MDP), soft resource reservation mechanism, fixed power allocation (FPA), fractional transmit power allocation (FTPA), improved fractional transmit power allocation (I-FTPA), message passing (MP), stable matching (SM), distributed auction method, inter linear programming (ILP), and PF algorithms are considered to achieve better resource allocation. Here, we will see the merits, demerits, and improvement needed for the future research in next generation mobile communication systems.

### 3.1 RA in Fog RAN Using RL

In [16], in fog RAN, the reinforcement learning (RL)-based resource allocation provides the requirements of low latency. In fog RAN, the URLLC cannot consent more delay. In FN, the IoT application required low latency. The Markov decision

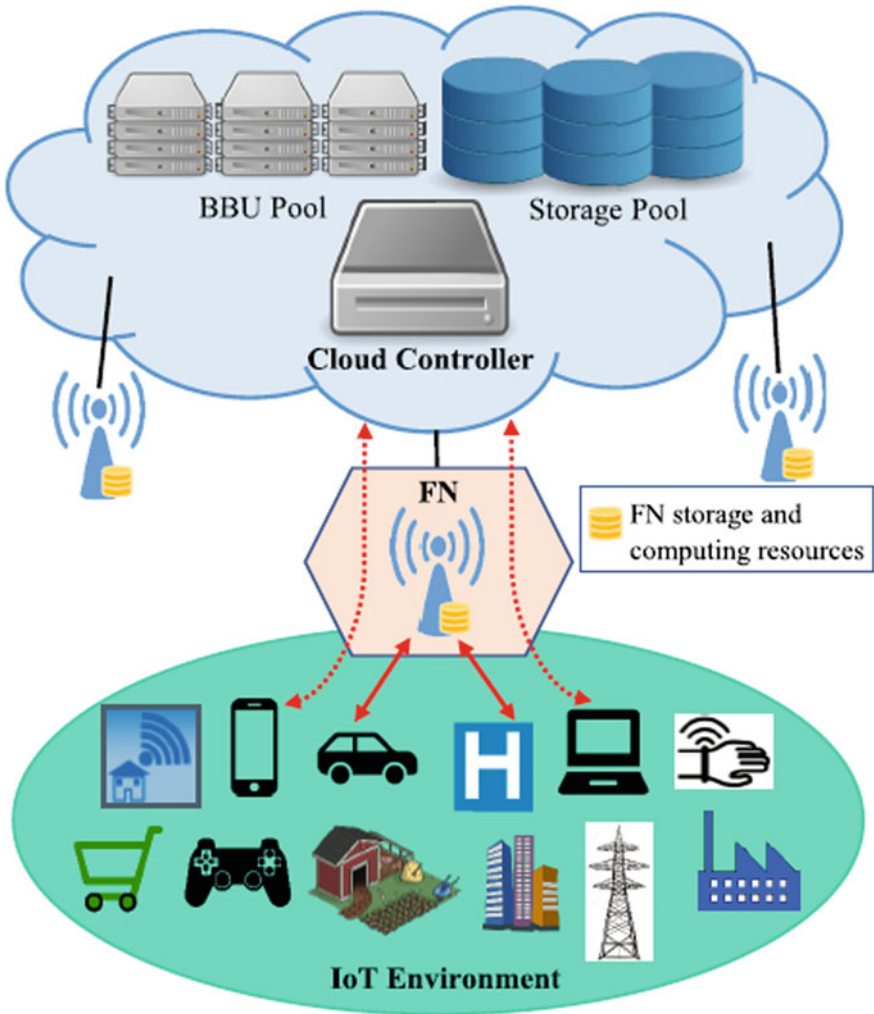


Fig. 1 Fog-RAN architecture

process (MDP) provides a low-latency communication in fog RAN. Like Q-learning, SARASA, expected SARASA, and Monte Carlo method are to solve MDP problem.

The objective of IoT applications in a FN the MDP provide the service to N number of resource blocks with perfect timing. The problem in MDP the FN getting services from the IoT side, after that the service will be continue or drop. The optimum policy utilizing Monte Carlo algorithm in the IoT environment if done initially, then the next algorithm for MDP by SARSA and QL.

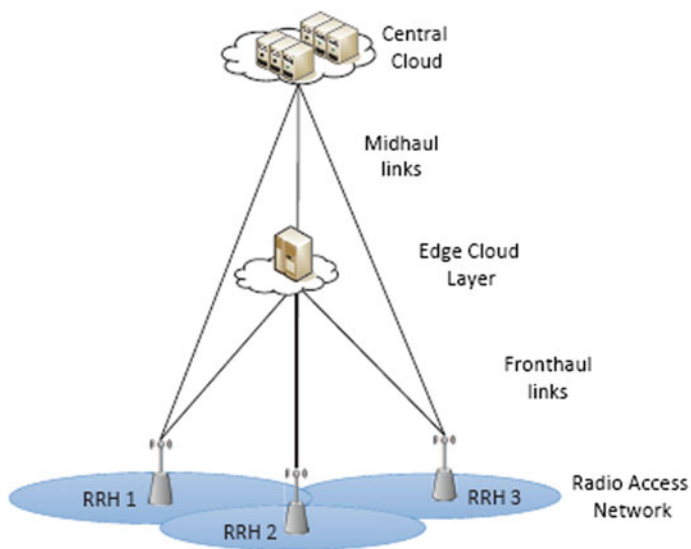


Fig. 2 C-RAN architecture with hybrid cloud

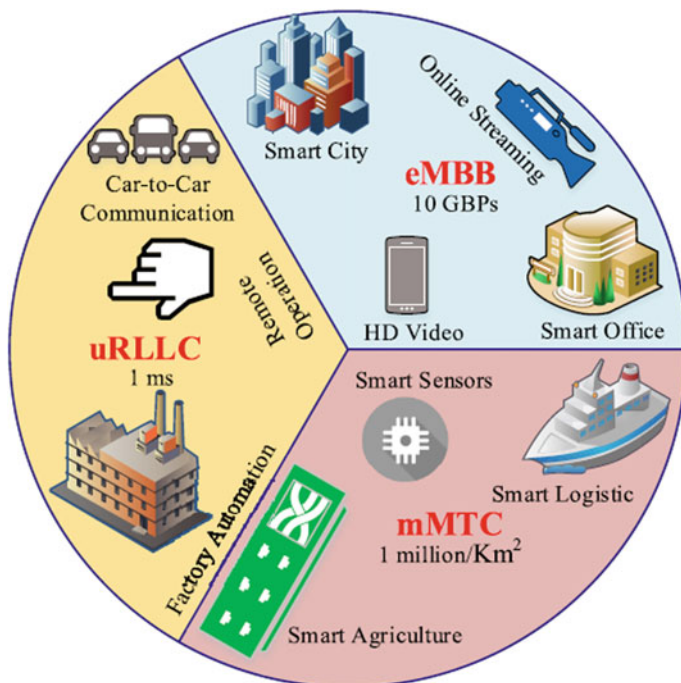


Fig. 3 5G mobile communication service categories

### 3.2 NOMA-Based Fog RAN Channel Power Allocation

In different RAN architectures in fog RAN [17] to increase the QoS and reduce the delay service is getting by edge computing. According to various power factor assignment of each and every NOMA user information is multiplexed through the corresponding data, and the receiver side the SIC is used to receive the signal. At the transmitting side, a three user multiplexing system is implemented.

In Fig. 4, user equipment (UE) first user is a near the user, this user is near the base station, second user is a mid-user this user some distance away from the base station, and third user is more away from the base station or an edge user. The sender signal or data of first user ( $i = 1,2,3$ ) is  $X_i$  and signal power assign to each user is  $P_i$ . As indicated by power allocation (PA) of NOMA [19], since the third user getting a very low signal strength so the more transmitter power is assign, then the second user getting moderate signal strength so the required power is allocated, and in the first user getting high signal power so the assignment of transmitter power is very small as required.

The user power allocation is important thing. The various methods of PA algorithms are: (i) fixed power allocation (FTP), (ii) fractional transmit power allocation (FTPA), and in [17], the method of improved fractional transmit power allocation (I-FTPA) was implemented.

The FPA depends upon the channel gain of the each and every individual user the different fixed power is allocated; in the poor channel gain user, the more power is allocated; in high channel gain user, minimum power is assigned. FPA does not have any specific power allocation strategy for every individual user which depends upon their channel gain. The main drawback for FPA is that it does not perform to allocate the power according to randomly varying nature of channel gain.

In I-FTPA planned for taking care of the above issue in FPA, the fractional transmit power allocation (FTPA) is implemented based on signal decay factor [18]. Author proposed an Improved fractional transmit power allocation (I-FTPA) calculation.

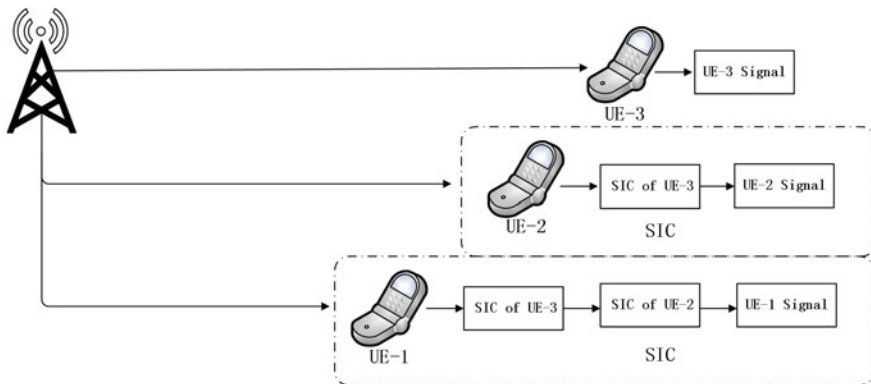


Fig. 4 Simple NOMA system



Contrasted with FTPA, this has just a signal degradation factor for power allocation, and I-FTPA has distinctive fluctuating signal degradation factor to modify every client's channel gain, which gives advantageous to change transmitted power as per channel changes consequently. The estimation varying the decay factor is somewhere in the range of 0 and 1.

Successive Interference Cancellation is applied at the receiver end in order to decode the user signal. For better receiving in the receiver end, the symbol level SIC (SLIC) and code-word level SIC (CWIC) are used. Important variation between these two methods is: CWIC demodulation and decoding will be done; it does not reduce the error propagation at the cost of complexity.

### ***3.3 Optimal Resource Allocation and Virtual Network Function Using in a Hybrid CRAN***

In [14], a joint optimal arrangement of virtual network functions (VNFs) and the RA in a hybrid cloud environment gives the requirement of next generation system of the cloud access point. The relationship between computational requirements, latency constraints, and design an integer linear programming (ILP) these all are completed by fixed assignment system.

An analysis of least computational rate is needed to fulfill every VNF necessity, by constant assigned edge or central access point. At that point, utilizing this information, author can arrange an integer ILP that optimizing the using of cloud VNFs.

VNF provides the least computational rate to fulfill its low-latency and computational needs to depend on this or that it is in the common cloud of its neighboring VNFs.

### ***3.4 Distributed Resource Allocation***

The large number of network nodes, [13] resource allocation, and interference management are the important research challenges in multi-tier heterogeneous networks. In OFDMA-based 5G cellular network, the three novel resource allocation methods are implemented, namely: (i) stable matching, (ii) message passing, and (iii) distributed auction method. Here, the matching theory permits a low-complexity algorithmic control to give a decentralized self-sorting solution to the resource allocation problems. In matching-based resource allocation, each of the agent's radio resources and transmitter nodes ranks the other using a preference relation. The solution of the matching concept is fix an available sources to the sender based on the selection.

**Table 1** Characteristics of RAN and resource allocation algorithms

Characteristics	C-RAN	Fog-RAN
Storage [13, 16]	Centralized	Centralized and distributed
Communication [13, 16]	Centralized	Centralized and distributed
Backhaul interference [17]	CN	CN and BBU pool
Fronthaul complexity [17]	Low	Medium
Data processing [13]	Cloud data center	Near to device
Transmission delay [16]	Long	Low
Latency [16]	High	Low
Reliability [13]	Medium	High
Energy consumption [13]	Medium	Low
Resource allocation algorithms and techniques used	<ul style="list-style-type: none"> <li>• Reinforcement learning [16]</li> <li>• Markov decision process [16]</li> <li>• Improved fractional transmit power allocation [17]</li> <li>• Message passing, stable matching and auction-based RA [13]</li> <li>• Integer linear programming [14]</li> </ul>	

In resource allocation using a message passing is produces a polynomial time complexity solution by assigning the calculated load among the access point of the system. In the RF source distributing issue, the decided authority system to the radio sources and the sender form a virtual graphical system. Every access point can exchanges some information with nearby access points to get the solution of RA.

Auction-based resource allocation algorithms provide polynomial complexity solutions, which are shown to output near nodes. The method of tender is the one of the best system to assign the service of the service provider. The unused spectrum is bids for all service provider agents. Who is coat a high value that agent or service provider can get the spectrum.

Through this survey, we tabulated the characteristics of different types of RAN architecture and what are all the algorithms and techniques were used by different authors are listed in Table 1.

## 4 Future Research Direction

In [16], several RL, methods are considered for the better solution of optimal system. A better action over a filtering method based of network slabbing. In future research to expanding the present RA framework is limited to fixed FN system. In [17], implementation of NOMA to F-RAN and the three user multiplexed transmitter and CWIC receiver is implemented. By varying the signal decay factor depending upon the channel gain, the better power allocation is assigned which depends upon the user requirement. In this the multiplexed transmitted and receiver, they considered

the single transmit and single receive antenna method. In future research, the design of multi-transmit antenna where use the diversity gain which will be increase in user equipment's. It may improve the SINR to provide a better QoS by using the I-FTPA. In [14], the issue of the optimal RA method and network slab usage is in a hybrid cloud environment which is considered and systemized by inter linear programming. Considering a hybrid cloud with respect C-RAN system, composed only in a central access point, in the service of low latency needed. Some of the future research work progress is based on usage of VNF. In [13] stable matching, message passing and auction-based resource allocation methods are implemented. Future research work to develop and implement an advanced game model resource allocation problems and analyses the response of 5G system with the data rate and spectrum and energy efficiency.

## 5 Conclusion

With the dramatically increasing end user amount, we want to improve the resource allocation for the future 5G/6G wireless communication. Due to spectrum on demand like, we want to utilize the available radio resources, and we did a survey with different resource allocation problems in existing wireless communication systems. In future research work, we want to overcome a problems faced in existing resource allocation algorithms and techniques discussed in this paper; these identified problem are given to invent a novel RA algorithm. An idea to implement the artificial intelligence for RA in 5G RAN will give the optimal solutions for better resource allocation the next generation wireless communication systems.

## References

1. A.I. Sulyman, A.T. Nassar, M.K. Samimi, G.R. MacCartney Jr., T.S. Rappaport, A. Alsanie, Radio propagation path loss models for 5G cellular networks in the 28 GHz and 38 GHz millimeter-wave bands. *IEEE Commun. Mag.* **52**(9), 78–86 (2014)
2. B. Yang, Z. Yu, J. Lan, R. Zhang, J. Zhou, W. Hong, Digital beamforming-based massive MIMO transceiver for 5G millimeter-wave communications. *IEEE Trans. Microw. Theory Techn.* **66**(7), 3403–3418 (2018)
3. S. Rangan, T.S. Rappaport, E. Erkip, Millimeter-wave cellular wireless networks: potentials and challenges. *Proc. IEEE* **102**(3), 366–385 (2014)
4. J. Zhang, Z. Zheng, Y. Zhang, J. Xi, X. Zhao, G. Gui, 3D MIMO for 5G NR: Several observations from 32 to massive 256 antennas based on channel measurement. *IEEE Commun. Mag.* **56**(3), 62–70 (2018)
5. S.-H. Park, O. Simeone, S. Shamai (Shitz), Joint optimization of cloud and edge processing for fog radio access networks. in *Proceedings IEEE International Symposium Information Theory (ISIT)*, July (2016), pp. 315–319
6. M. Peng, Y. Sun, X. Li, Z. Mao, C. Wang, Recent advances in cloud radio access networks: system architectures, key techniques, and open issues. *IEEE Commun. Surveys Tuts.* **18**(3), 2282–2308 (2016)

7. H. Zhang, Y. Qiu, X. Chu, K. Long, V.C. M. Leung, 'Fog radio access networks: mobility management, interference mitigation, and resource optimization. *IEEE Wireless Commun.* **24**(6), 120–127 (2017)
8. Y. Wang, B. Ren, S. Sun, S. Kang, X. Yue, Analysis of non-orthogonal multiple access for 5G. *China Commun.* **13**(2), 52–66 (2016)
9. H. Zhang, Y. Qiu, K. Long, G.K. Karagiannidis, X. Wang, A. Nallanathan, Resource allocation in NOMA-based fog radio access networks. *IEEE Wireless Commun.* **25**(3), 110–115 (2018)
10. C. Yan, A. Harada, A. Benjebbour, Y. Lan, A. Li, H. Jiang, Receiver design for downlink non-orthogonal multiple access (NOMA). in *Proceedings IEEE 81st Vehicles Technology Conference (VTC)*, May (2015), pp. 1–6
11. M. Al-Imari, P. Xiao, M.A. Imran, 'Receiver and resource allocation optimization for uplink NOMA in 5G wireless networks. in *Proceedings International Symposium Wireless Communications Systems (ISWCS)*, August (2015), pp. 151–155
12. K. Saito, A. Benjebbour, A. Harada, Y. Kishiyama, T. Nakamura, 'Link-level performance of downlink NOMA with SIC receiver considering error vector magnitude. in *Proceedings IEEE 81st Vehicle Technology Conference (VTC)*, May (2015), pp. 1–5
13. M. Hasan, E. Hossain, Distributed resource allocation in 5G cellular networks. (Wiley, 2017)
14. A. De Domenico, Y.-F. Liu, W. Yu, Optimal computational resource allocation and network slicing deployment in 5G hybrid C-Ran. (IEEE, 2019)
15. M. Liang, X. Wang, Application of 5G-based mobile communication technology in network resource scheduling. (IEEE, 2019)
16. A. Nassar, Yasin, Reinforcement learning for adaptive resource allocation in fog RAN for IoT with heterogeneous latency requirements. *IEEE Access* **7**, 529–551 (2019)
17. W. Bai, T. Yao, H. Zhang, V.C.M. Leung, Research on channel power allocation of fog wireless access network based on NOMA. vol. 7, (IEEE, 2019)
18. A. Benjebbour, A. Li, Y. Saito, Y. Kishiyama, A. Harada, T. Nakamura, 'System-level performance of downlink NOMA for future LTE enhancements. in *Proceedings IEEE Globecom Workshops (GC Wkshps)*, December (2013), pp. 66–70
19. F.-L. Luo, C. Zhang, in *Signal Processing for 5G: Algorithms and Implementations*. (Wiley, Hoboken, NJ, USA, 2016)

# Wearable PIFA for Off-Body Communication: Miniaturization Design and Human Exposure Assessment



Sandra Costanzo, Adil Masoud Qureshi, and Vincenzo Cioffi

**Abstract** A miniaturized Planar Inverted-F Antenna (PIFA) design tailored for wearable devices is presented in this work. The proposed antenna operates in the ISM band (from 2.4 to 2.5 GHz) used by common wireless communication standards. A felt textile substrate is used to allow easy integration into everyday clothing. A side-fed coaxial cable is also adopted to give a low profile. To assess human exposure, SAR analysis is conducted on the designed antenna and simulated results are presented. The SAR level of the antenna is successfully limited to comply with international guidelines, by introducing significant modifications on the antenna parameters.

**Keywords** PIFA · SAR · Wearable application

## 1 Introduction

Wearable antennas for off-body communication are increasingly ubiquitous in the modern world [1]. Consumer devices like smartwatches, ankle tags used by law enforcement, and health sensors to monitor at-risk patients, all of them make use of wearable antennas. Ensuring reliable communication independently of user's posture, while also being as unobtrusive as possible, gives some very challenging requirements for the antenna design [2]. Wearable antennas are required to be small in size, resistant to detuning and able to provide a reasonable gain [3]. Apart from being small and reliable, wearable antennas must also be safe for everyday use. Strict requirements are in place for Specific Absorption Rate (SAR) levels of wearable devices, in order to avoid any harmful effects. One of the most common antenna

---

S. Costanzo (✉) · A. M. Qureshi · V. Cioffi  
University of Calabria, 87036 Rende, CS, Italy  
e-mail: [costanzo@dimes.unical.it](mailto:costanzo@dimes.unical.it)

S. Costanzo  
CNR–Institute for Electromagnetic Sensing of the Environment (IREA), 80124 Naples, Italy  
ICEmB, Inter-University National Research Center On Interactions Between Electromagnetic Fields and Biosystems, 16145 Genova, Italy

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2021  
S. M. Thampi et al. (eds.), *Advances in Computing and Network Communications*,  
Lecture Notes in Electrical Engineering 735,  
[https://doi.org/10.1007/978-981-33-6977-1\\_4](https://doi.org/10.1007/978-981-33-6977-1_4)

designs is given by the Planar Inverted-F Antenna, commonly known as PIFA [4]. Every modern smartphone includes more than one example of the PIFA, operating in various frequency bands. The PIFA is an electrically small antenna by design, and thanks to its ground plane, it radiates mostly in only one hemisphere. The directional radiation ensures stable (not easily detuned) as well as safe (low SAR) operation [5]. These features make it a good choice for portable as well as wearable communication devices. Nevertheless, the trend towards smaller and more compact devices has led to research methods for further miniaturization of the PIFA. The oldest and most basic method is based on the adoption of higher permittivity substrates, in order to increase the electrical size of the antenna [6]. However, high permittivity substrates increase losses, thus leading to lower efficiency and lower gains. Loading the PIFA with reactive or resistive components can also lead to miniaturization, but causing similar losses in the efficiency [7]. Metamaterials have been shown to reduce the resonant frequency of a PIFA, although the complexity of such designs increases manufacturing costs [8]. Another method for achieving PIFA miniaturization is given by the adoption of a fractal geometry for the radiating element. Earlier authors' work has shown that the Minkowski fractal shape can be used to reduce the operating frequency of a reflect array without increasing the physical size of the cells [9]. The same technique has also been successfully applied to a PIFA operating into free space [10]. Further work by the authors demonstrated a simplified slotted geometry for a miniaturized PIFA [11].

In the present work, a compact PIFA design is presented, which is specifically tailored for wearable devices. The proposed design includes a new lateral feed arrangement that allows for easy integration into clothing. Simulated results for the new design, including SAR performance, are reported. A technique for reducing the SAR level of the antenna based on earlier work by the authors is also described, and the comparison of SAR levels before and after optimization is illustrated.

## 2 Antenna Design

The design process for the proposed wearable PIFA starts from the miniaturized PIFA working in the ISM band, presented in [11]. The earlier design has a coaxial probe-type input that connects through the back of the antenna. This kind of input is not ideal for a wearable design, since the back or the ground plane of the antenna is expected to sit flush with the fabric or body of the user. In order to face this problem, a lateral coaxial feed is proposed, as shown in Fig. 1. The coaxial cable enters from the left side of the antenna, and it places between the ground plane and the radiating patch. The outer conductor is bonded to the ground plane, while the inner conductor is turned 90° to connect with the patch.

The feed cable is normally oriented to the polarization axis of the antenna, in order to minimize its effects on the electric field. As shown in Fig. 2, the radiation pattern of the antenna remains almost unchanged due to the introduction of the lateral feed.

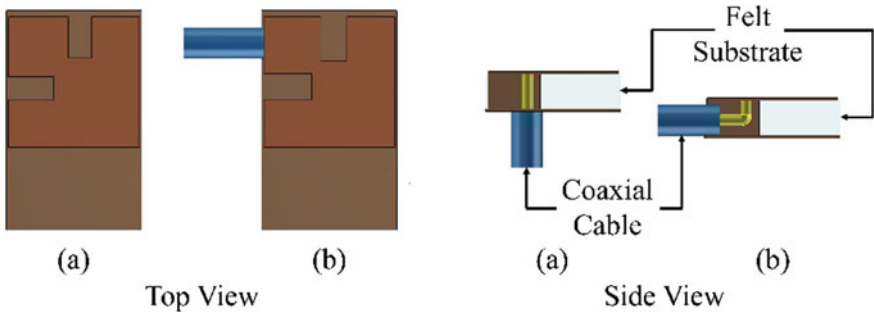


Fig. 1 a Probe feed used in the earlier design [11] versus b lateral feed of the wearable PIFA

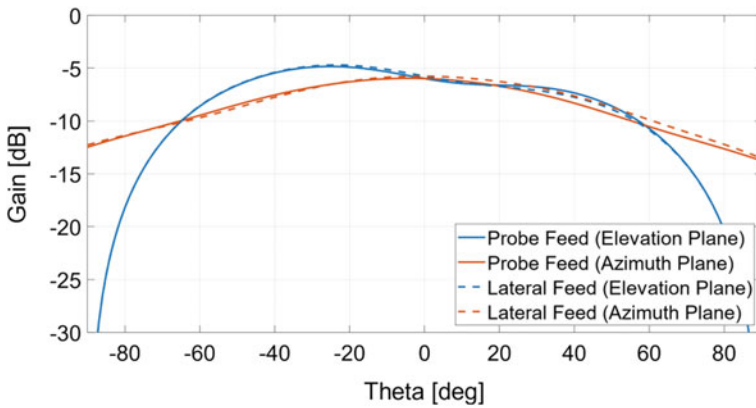
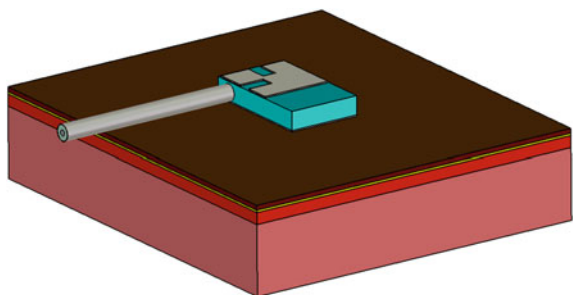


Fig. 2 Radiation pattern of the PIFA antenna with the two different feed arrangements

The wearable antenna is modelled with felt fabric as a substrate, since it is expected to be incorporated into clothing. To simulate the human body effect, a simplified multilayer phantom is adopted (Fig. 3).

Fig. 3 Wearable antenna model in CST software with multilayer body phantom

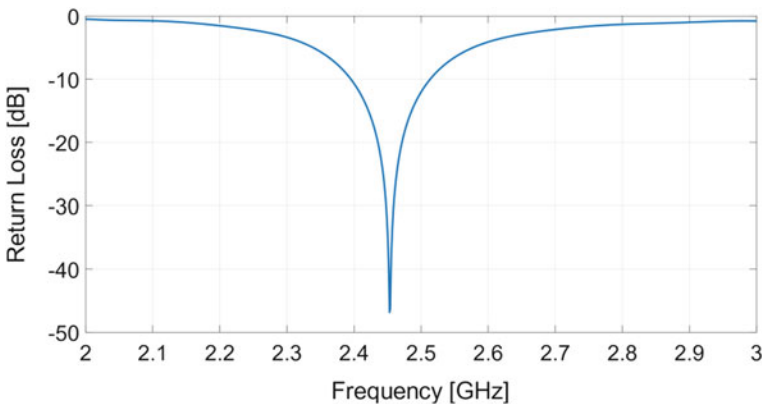


**Table 1** Dielectric characteristics of the felt substrate and the multilayer body model

Medium	Thickness [mm]	$\epsilon_r$	$\tan\delta$	$\sigma$ [S/m]	Density [kg/m <sup>3</sup> ]
Felt (substrate)	4	1.45	0.02	–	–
Dry skin	0.015	38.06	0.2835	1.5	1000
Wet skin	0.985	42.92	0.2727	1.62	1000
Fat	0.5	10.84	0.1808	0.27	850
Blood	2.5	58.53	0.1743	1.41	1060
Muscle	3	52.34	0.1893	1.37	1050

The dielectric characteristics of each layer of the assumed body model, as well as the felt substrate, are listed in Table 1. The combined effect of the higher permittivity felt substrate and the body proximity reduces the resonant frequency of the antenna. The dimensions of the antenna are optimized for operation in the ISM band from 2.4 to 2.5 GHz (Fig. 4). The final antenna design is equal to  $15.5 \times 25.25$  mm in size, with a substrate height of 4 mm.

The direct contact with the lossy body media makes it impossible to use a defected ground plane for bandwidth enhancement [12]. As a result, the bandwidth of the present design is much smaller than the free space variant. On the other hand, the proximity to the skin allows for a smaller ground plane to be used, thus enhancing the miniaturization. The wearable antenna design is  $\sim 37\%$  shorter than the free space variant, while operating in the same frequency band.

**Fig. 4** Simulated return loss of the proposed wearable PIFA



### 3 Preliminary SAR Analysis

Human body exposure to the EM radiation is monitored through several parameters, such as the temperature increase, the exposure to time-varying electric and magnetic fields, or the commonly known Specific Absorption Rate (SAR).

SAR is a parameter which measures the speed at which energy is absorbed by the human body when it is exposed to electromagnetic fields with a carrier frequency between 100 kHz and 10 GHz. It is used in different fields; in mobile telephony, the SAR establishes the amount of energy absorbed by a particular mass of human tissue within a certain period of time. SAR is calculated in units of power per mass (W/kg) and is given by (1):

$$\text{SAR} = \frac{\sigma |E|^2}{\rho} \left[ \frac{\text{W}}{\text{Kg}} \right] \quad (1)$$

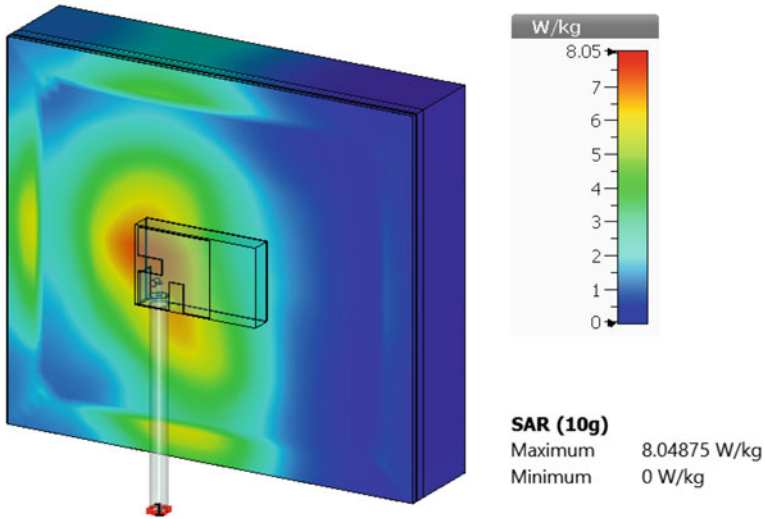
where  $\sigma$  is the conductivity of the tissue (S/m),  $\rho$  is the mass density of tissue (Kg/m<sup>3</sup>), and  $E$  is the root mean square of electric field strength (V/m).

Numerical simulations are conducted to evaluate the SAR levels of the designed PIFA. These simulations are carried out using CST microwave studio, with the reference standard safe levels reported below:

- **FCC [13]:**
  - Body, trunk, head: 1 g-SAR with limit 1.6 W/Kg;
  - Limbs: 10 g-SAR with limit 4 W/Kg.
- **CE [14]:**
  - Body, trunk, head: 10 g-SAR with limit 2 W/Kg;
  - Limbs: 10 g-SAR with limit 4 W/Kg.

The proposed sensor should work on the human arm; thus, a safe limit equal to 4 W/Kg is assumed for any 10 g of tissue. The result of the first SAR analysis performed on the presented PIFA in the flat condition is reported in Fig. 5. The same multilayer body phantom already used for the antenna design is again adopted for the SAR simulations.

As it can be observed from Fig. 5, the SAR levels of the initial design exceed the safe limit of 4 W/Kg, as the maximum SAR value is equal to 8.04 W/Kg. The input power assumed for simulations is equal to 0.5 W. Although quite high, this value is chosen as the worst possible condition.



**Fig. 5** SAR analysis for the initial PIFA configuration

## 4 Reduction of SAR Level

The original PIFA configuration presents a very high SAR level, if considering the standard safe limits. In order to reduce the SAR value, some modifications are introduced on the original antenna geometry.

Some approaches exist in the literature to control the SAR levels, for example, by using metamaterial to restrict the propagation of surface waves within a specific frequency band and therefore reducing the level of unwanted radiations towards the human body [15]; or by adopting a Perfect Electric Conductor (PEC) reflector [16] as a shielding layer between the antenna and human body. An alternative method is based on a proper choice of the dielectric substrate parameters.

In this work, the antenna is modified in terms of ground plane and substrate thickness. Indeed, the ground plane is enlarged for better protection towards the human body, while the substrate thickness is sufficiently increased [17]. A parametric analysis was conducted to better understand the effect of the ground plane size and the substrate thickness on the SAR. The results of the parametric analysis are show in Tables 2 and 3.

**Table 2** SAR values at different substrate thickness

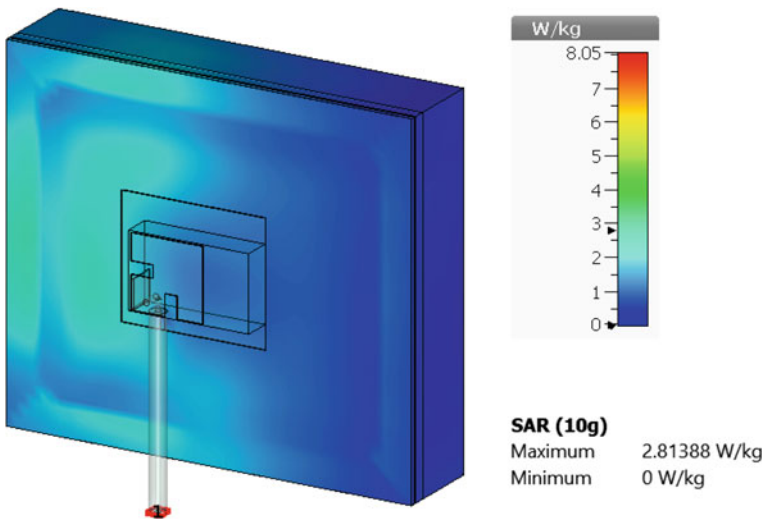
Thickness [mm]	SAR [W/Kg]
4	8.05
6	2.81
7	2.75
8	2.66

**Table 3** SAR values at different ground plane dimensions (felt = 6 mm)

Ground plane [mm]	SAR [W/Kg]
20.25 × 25.5	4.18
21.25 × 25.5	3.45
22.25 × 25.5	2.88
30.25 × 25.5	2.81

The final ground plane dimensions are equal to 25.5 × 30.25 mm, while the substrate thickness is equal to 6 mm. Using this new antenna geometry, the SAR simulation is repeated, and the enhanced result is reported in Fig. 6.

In this case, the SAR value exhibits a sharp decrease, with a new value equal to 2.81 W/Kg, which is perfectly within the safe limits for humans. It is evident that the strategy to modify the ground plane and the substrate thickness gives excellent results, in terms of safe human exposure. The final design with reduced SAR performs better than the original design as the power being absorbed by the body is reduced. The efficiency enhancement can be observed from the improved gain and radiation patterns of the final design (Fig. 7).



**Fig. 6** SAR analysis using the new PIFA geometry

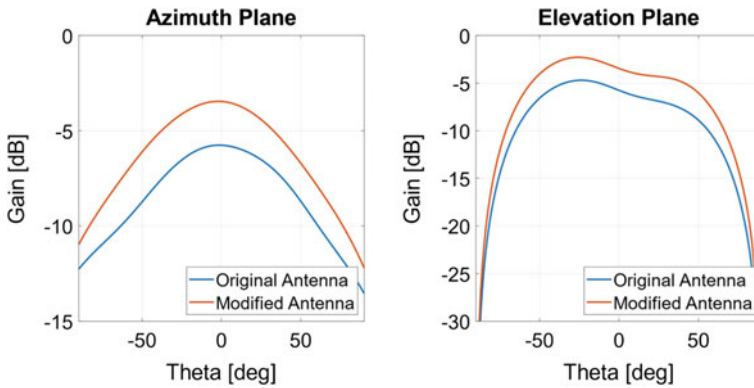


Fig. 7 Radiation patterns of the original antenna and final design with improved SAR

## 5 Conclusion

A new miniaturized PIFA design for wearable communication devices has been presented in this work. The proposed antenna is slightly over one inch in length, and it covers the entire ISM band (2.4–2.5 GHz). A sideways coaxial feeding mechanism has been designed in order to make it easier to integrate the antenna into textiles. SAR levels of the proposed design have been successfully reduced by a proper modification of the adopted geometry. The final design is well within the established safety limits of SAR for wearable applications.

## References

1. N.H.M. Rais, P.J. Soh, F. Malek, S. Ahmad, N.B.M. Hashim, P.S. Hall, A review of wearable antenna. in *2009 Loughborough Antennas and Propagation Conference*. (IEEE, Loughborough, 2009), pp. 225–228
2. C. Roblin, J. Laheurte, R. D’Errico et al., Antenna design and channel modeling in the BAN context—part I: antennas. *Ann. Telecommun.* **66**, 139–155 (2011)
3. P.R. Young, C.K. Aanandan, T. Mathew, D.D. Krishna, Wearable antennas and systems. *Int. J. Antennas Propag.* 1–2 (2012)
4. K. Fujimoto (ed.), *Mobile Antenna Systems Handbook*, 3rd edn. (Artech House, Boston, 2008)
5. G. Gao, B. Hu, S. Wang, C. Yang, Wearable planar inverted-F antenna with stable characteristic and low specific absorption rate. *Microwave Opt. Technol. Lett.* **60**(4), 876–882 (2018)
6. T.K. Lo, Y. Hwang, Bandwidth enhancement of PIFA loaded with very high permittivity material using FDTD. in *IEEE Antennas and Propagation Society International Symposium 1998 Digest*. Antennas: Gateways to the Global Network. Held in conjunction with: USNC/URSI National Radio Science Meeting (Cat. No.98CH36) vol. 2 (1998)
7. R.B. Waterhouse (ed.), in *Printed antennas for wireless communications*. (Wiley, Chichester, England; Hoboken, NJ 2007)
8. G. Gao, C. Yang, B. Hu, R. Zhang, S. Wang, A wearable PIFA with an all-textile metasurface for 5 GHz WBAN applications. *IEEE Antennas Wirel. Propag. Lett.* **18**(2), 288–292 (2019)

9. S. Costanzo, F. Venneri, Miniaturized fractal reflectarray element using fixed-size patch. *IEEE Antennas Wirel. Propag. Lett.* **13**, 1437–1440 (2014)
10. S. Costanzo, A.M. Qureshi, Miniaturized wearable minkowski planar inverted-F antenna. in *Information Technology and Systems*, ed. by Á. Rocha, C. Ferrás, C. Montenegro Marin, V. Medina García. *ICITS 2020 Advances in Intelligent Systems and Computing*, vol. 1137. (Springer, Cham, 2020)
11. S. Costanzo, A.M. Qureshi, Compact slotted planar inverted-f antenna: design principle and preliminary results. in *Trends and Innovations in Information Systems and Technologies*, ed. by Á. Rocha, H. Adeli, L. Reis, S. Costanzo, I. Orovic, F. Moreira. *WorldCIST 2020. Advances in Intelligent Systems and Computing*, vol 1161. (Springer, Cham, 2020)
12. F. Wang, Z. Du, Q. Wang, K. Gong, Enhanced-bandwidth PIFA with T-shaped ground plane. *Electron. Lett.* **40**, 1504–1505 (2004). <https://doi.org/10.1049/el:20046055>
13. <https://www.fcc.gov/search/#q=specific%20absorption%20rate&t=web>
14. Official Journal of the European Communities, L 199/59, 2 July (1999)
15. U. Ali et al., Design and SAR analysis of wearable antenna on various parts of human body using conventional and artificial ground planes. *J. Electri. Eng. Technol.* **12**(1), 317–328 (2017)
16. H.K. Chan, M.K. Chan, L.C. Fung, S.W. Leung, Effects of using conductive materials for SAR reduction in mobile phones. *Microwave Opt. Technol. Lett.* **44**(2), 140–144 (2005)
17. S. Costanzo, V. Cioffi, Preliminary SAR analysis of textile antenna sensor for non-invasive blood-glucose monitoring. in: *ICITS'20* (Bogotá, Colombia, 2020)

# Generalized Symbolic Dynamics Approach for Characterization of Time Series



S. Suriyaprabhaa, Greeshma Gopinath, R. Sangeerthana, S. Alfiya, P. Asha, and K. Satheesh Kumar

**Abstract** Various nonlinear methods have been developed to analyze the underlying dynamics of a nonlinear time series. Dynamic characterization using symbolic dynamics approach has been found to be a good alternative for the analysis of chaotic time series. As per this method, the given time series is first transformed into a single bit binary series. The single bit encoding limits its ability to capture the dynamics faithfully. This paper aims to provide a generalization of the symbolic dynamics method for better capturing the dynamical characteristics such as Lyapunov exponents of a time series. The effectiveness of the generalized method is demonstrated by employing a logistic map. The results of the analysis indicate that higher-order encoding can capture the bifurcation diagram more effectively compared to the original single bit encoding used in symbolic dynamics.

**Keywords** Chaos · Dynamic characterization · Symbolic dynamics · Bifurcation · Logistic map

## 1 Introduction

The complexity and irregularity observed in nature is becoming an interesting topic for scientists from various disciplines. There emerges a new way of looking at this complexity, which is termed as chaos theory. Chaos is the complex behavior exhibited by the nonlinear system due to their sensitive dependence on initial conditions [1–3]. For a nonlinear chaotic system, the small uncertainty in initial value grows exponentially and gives rise to unpredictable values in the future. Thus, it is difficult to predict the long-term behavior of a nonlinear system [1, 4, 5]. Hence, analyzing such kind of system with their time series is a good alternative [6]. Time series is the set of all measurements carried out over time which can be used to forecast the future by the proper knowledge of the past [7]. Nonlinear time series are generally used to

---

S. Suriyaprabhaa · G. Gopinath · R. Sangeerthana · S. Alfiya · P. Asha · K. Satheesh Kumar (✉)  
Department of Futures studies, University of Kerala, Kariavattom, Kerala 695581, India  
e-mail: [kskumar@keralauniversity.ac.in](mailto:kskumar@keralauniversity.ac.in)

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2021  
S. M. Thampi et al. (eds.), *Advances in Computing and Network Communications*,  
Lecture Notes in Electrical Engineering 735,  
[https://doi.org/10.1007/978-981-33-6977-1\\_5](https://doi.org/10.1007/978-981-33-6977-1_5)

get information about the dynamical systems. Recently, different alternative methods have been developed to analyze time series. The role of symbolic description for dynamics was first recognized by Morse, Harold Marston [8], and phase space coarse graining algorithm that transforms a time series into a directed and weighted complex network presented by Wang and Tian [9]. The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis by Huang et al. [10] and the Fourier decomposition method for nonlinear and non-stationary time series analysis by Singh et al. [11] are notable advancements in this field. The applied aspect of symbolic dynamics—DCSD approach which is the main reference for our paper was proposed by Freitas et al. [12]. Dynamical characterization using symbolic dynamics (DCSD) approaches can be used to analyze dynamical system instead of analyzing with Lyapunov exponents and bifurcation process.

Symbolic dynamics is becoming an influential tool in the study of periodic and chaotic motion in nonlinear systems [13]. As it provides a natural link between the chaotic dynamics and information theory, it is used to describe time evolution of a nonlinear chaotic system [14]. This method usually replaces continuous time series data with discrete symbols. Even though a lot of information may be lost by doing so, the inherent properties of dynamics like periodicity, chaotic property kept invariant [15, 16]. Thus, we got a reliable and powerful technique whose application does not involve many computational resources [17]. With the DCSD approach, each data point in time series is converted into its equivalent binary values depending on its value of the data points [12, 18]. Each binary value is merged with its neighboring binary values and converted to its equivalent decimal values. Each decimal value represents node  $N$  of the network. However, the symbolic dynamics approach currently employs single bit encoding for transforming a time series into binary series. This single bit encoding limits its ability to capture the underlying dynamics of a given time series as demonstrate by Freitas et al. [12]. In order to overcome this limitation, we propose a generalized symbolic dynamics method and demonstrate that the generalized approach can more effectively capture the underlying dynamical characteristics.

## 2 Generalized Method of DCSD Approach

DCSD method [12, 17, 18] introduced by Freitas et al. [12] is found to be a good alternative for the analysis of dynamical systems. The aforementioned paper explained the underlying concepts of DCSD approaches. According to this approach, the median of the time series data points is calculated first. Those values lie in between the minimum values and median are treated as “0” and others as “1” to obtain symbolic binary series. After that, they cluster the first 10 bits (i.e., bin size  $B_s = 10$ ) and converted them into corresponding decimal values. The process continues by shifting a bit to 1 unit right till the series reaches its end. Then, the obtained decimal values are used as nodes of a network, and 2 decimal values adjacent in the symbolic binary series are connected by an edge. It is observed that the approach cannot faithfully

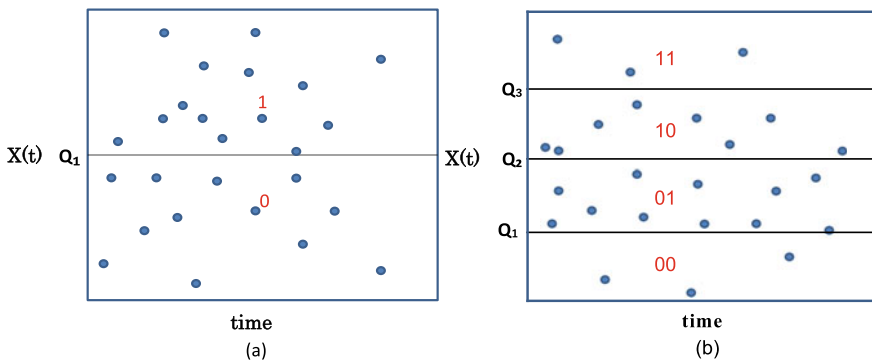
capture the bifurcation diagram. In this paper, we present a more generalized method of encoding. Instead of encoding into single bits of 0's and 1's, it is extended up to 6-bit encoding, and we tested the approach using logistic map [17].

Time series are generated with a logistic map, the simple, one-dimensional, discrete equation that produces chaos at certain growth rates [19] given by

$$X_{n+1} = rX_n(1 - X_n) \tag{1}$$

where  $X_n$  is the dimensionless measure of population in  $n^{th}$  generation, and  $r$  is the growth rate. The positive value of  $r$  represents exponential growth [4]. Here,  $X_n$  lies in is between 0 and 1, and the parameter  $r$  lies inside  $[0, 4]$  [1]. It is a model based on the common  $s$ -curve logistic function that shows how a population grows gently, then suddenly, before drop off as it reaches its environment's carrying capacity [20, 21].

For single bit encoding, each component of time series obtained from the logistic map is converted into symbols of 0s and 1s as per the method proposed by Freitas et al. [12], and it is illustrated in Fig. 1(a). The black filled circles represent the time series data points and each data point above  $Q_1$  replaced with 1 and rest are with 0. Then, we considered the double bit encoding. As shown in Fig. 1b, for a double bit encoding ( $b = 2$ ), we have partitioned the values of time series into four layers, where the first layer corresponds to values less than or equal to the first quartile  $Q_1$ , second layer with values above  $Q_1$  and values up to  $Q_2$ , and the third layer of values above  $Q_2$  and up to third quartile  $Q_3$ , and the last bin with values above  $Q_3$ , respectively. All the time series data point in first layer is replaced by 00, second layer by 01, third layer by 10, and last layer by 11. The length of the resulting symbolic binary series will be the twice the length of the original time series. To convert it into decimal numbers, we have to append the symbolic binary numbers to its adjacent binary numbers. For a single bit and double bit encoding approaches, the bin size is varied from 2 to 20, and analysis is done. After that, for convenience, we fix the bin size as 10. Bin is the series of fixed number of bits we have used for analysis, and bin



**Fig. 1** Diagrammatic representation of **a** single bit encoding, **b** double bit encoding



size ( $B_s$ ) is the number of bits we appended each time for the generation of decimal series. The first segment of bin size  $B_s$  bits is converted into a decimal number. After skipping 2 bits, the second segment is similarly converted into next decimal number and so on, i.e., symbolic binary series converted into a decimal series.

In general, for  $b$ -bit encoding, we will have  $2^b$  layers. Data points obtained from the time series are distributed in these layers. Each data point will be assigned by a binary value based on the layer in which it is distributed. For a  $b$ -bit encoding, every time series data points are replaced by  $b$ -bit binary number to form the symbolic binary series. For generating decimal series, binary segments are formed by skipping  $b$ -bits until it reaches the end.

The analysis is done as follows:

1. In order to access the effect of bin size,  $B_s$  it is varied from 2 to 20 for single and double bit encoding.
2. We use a bin size of 10 for analyzing 2 to 6 bit encoding.
3. In order to analyze the effect of shift length  $s$  of symbolic series, we varied  $s$  from 1, 2,  $\dots$ , 10.

Each sequence is converted into a decimal number which gives rise to decimal series.

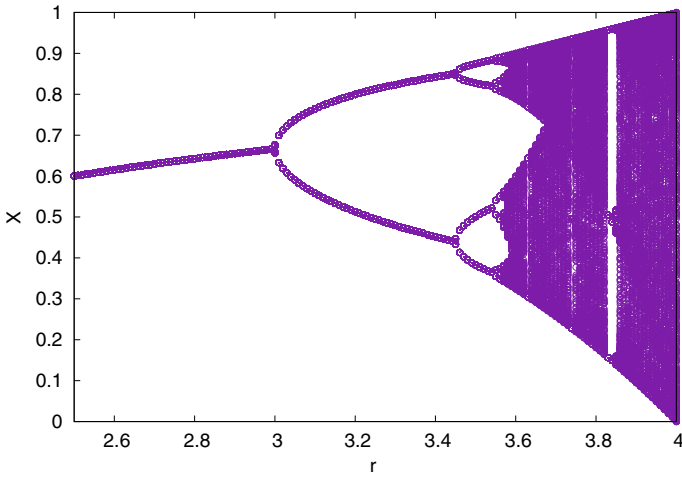
### 3 Result and Analysis

Using Eq. 1 of logistic map, time series of length 1000 is generated with  $r$  ranging from 2.5 to 4 at an interval of 0.001, and the initial condition is  $X_0 = 0.1$ . The first 500 values were neglected as transients. The remaining values were analyzed for bifurcation analysis using DCSD approach. The plot of bifurcation diagram of logistic map  $X(n)$  is shown in Fig. 2.

The time series for each value of  $r$  is then converted into binary values and then converted into decimal numbers according to DCSD procedure described in the previous section.

#### 3.1 The Effect of Bin Size

As a first step, the analysis is carried out by varying the bin size from 2 to 20 for single bit encoding and double bit encoding. It is observed that the point where the system starts exhibiting the chaotic behavior remains unchanged irrespective of the length of the bin size. So, we have taken fixed bin size of 10; hereafter, for analysis, as varying the bin size does not affect the bifurcation. As explained by Lacerda et al. [17] and V.L.S. Freitas et al. [12], the single bit encoding of DCSD approach



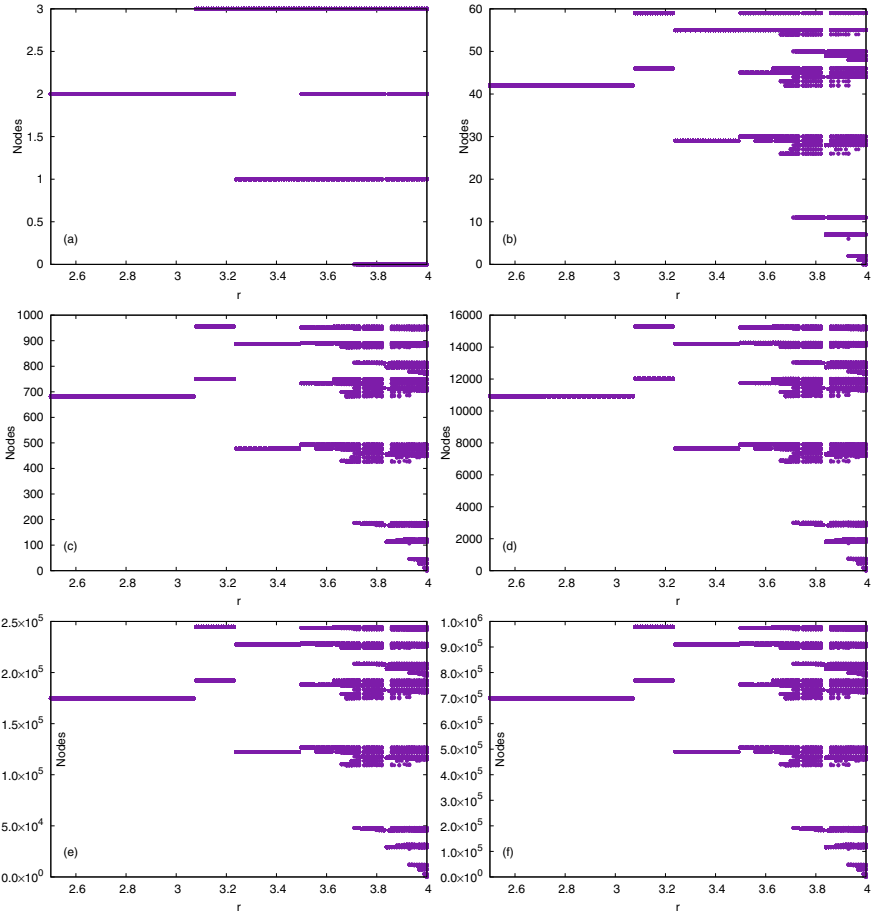
**Fig. 2** Bifurcation diagram of logistic map obtained directly from time series

**Table 1** Range of encoded decimal series for various bin size of double bit encoding

Bin size	The possible values of nodes
2	0-3
4	0-15
6	0-63
8	0-255
10	0-1023
12	0-4095
14	0-16383
16	0-65535
18	0-262143
20	0-1048575

fails to faithfully capture the bifurcation diagram of logistic map. The value of the encoded decimal series represented hereafter as node is given in Fig. 3. The range of the corresponding encoded decimal series for various bin size of double bit encoding is given in Table 1.

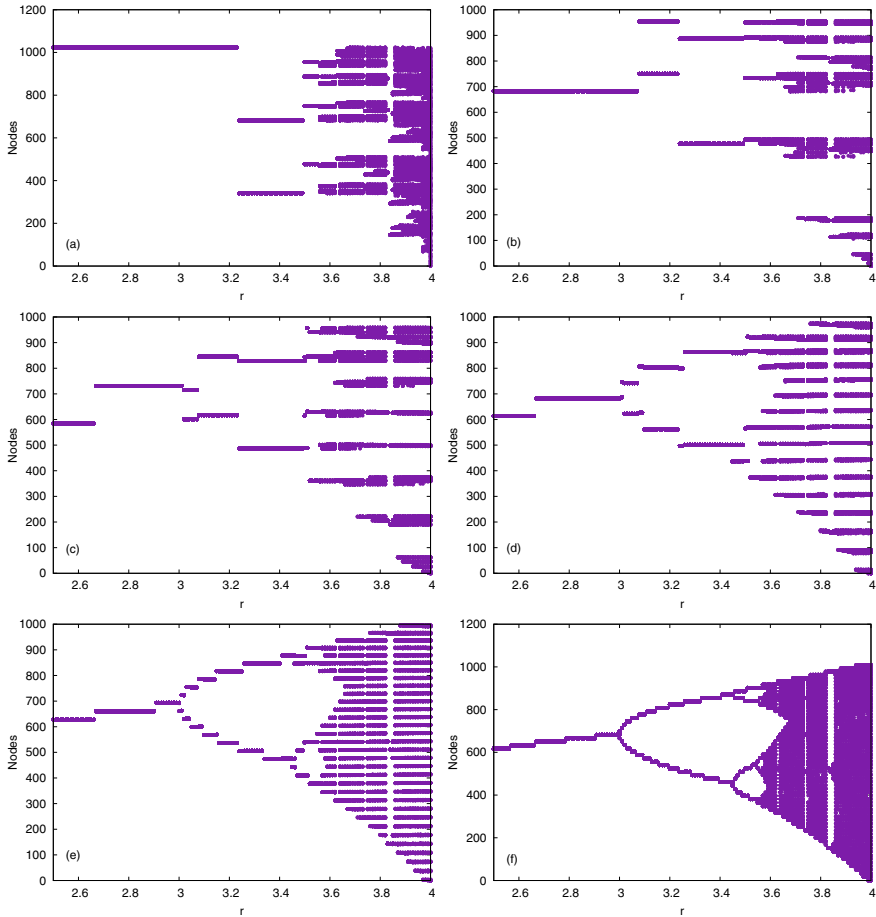
As can be seen from Fig. 3, our analysis for single and double bit encoding shows that the bifurcation diagram is not effectively captured for bin size varying from 2 to 20.



**Fig. 3** Plots of nodes versus  $r$  for varying bin size **a**  $B_s = 2$ , **b**  $B_s = 6$ , **c**  $B_s = 10$ , **d**  $B_s = 14$ , **e**  $B_s = 18$ , **f**  $B_s = 20$

### 3.2 The Effect of Encoding

To analyze the effect of encoding, as a next step, we varied  $b$ -bit encoding for  $2^b$  layers for  $b = 1, 2, \dots, 6$ . In all these analysis, a fixed bin size  $B_s$  of 10 is used. The plots of nodes of  $b$ -bit encoding against  $r$  for  $b = 1$  to 6 are given in Fig. 4. From the figure, it is clear that the pattern approaches original bifurcation diagram with the increase in  $b$ . It may be noted that by 6-bit encoding, the plots of the nodes almost faithfully captured bifurcation diagram, whereas single bit encoding introduced by Lacerda et al. [17] and Freitas et al. [12] fail to capture it effectively.

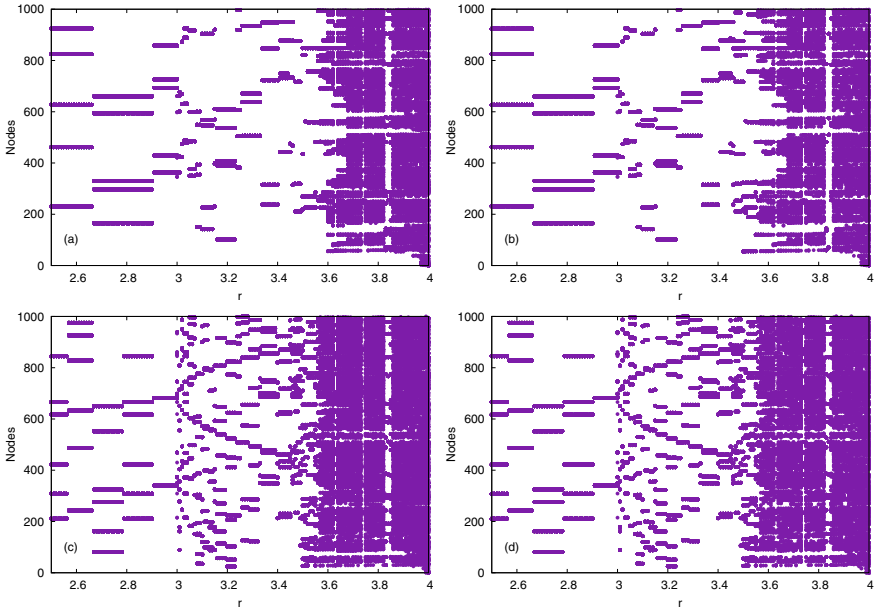


**Fig. 4** Plot of nodes versus  $r$  for various encoding with bin size 10 and  $b$  bit shifting for  $b$  bit encoding. **b** 2 bit encoding, **c** 3 bit encoding, **d** 4 bit encoding, **e** 5 bit encoding, **f** 6 bit encoding

### 3.3 The Effect of Shifting of Bin

As we have discussed earlier, the symbolic binary series is converted into decimal series by segmenting the binary series. At first, we consider a segment of binary number with bin size  $B_s$  and converted it into a decimal number. The next segment is of same length  $B_s$  after skipping the first  $s$  bits and so on. In this section, we present the results of analysis of length of shifting size  $s$ . The typical plots of  $b$ -bit encoding with  $s \neq b$  are shown in Fig. 5. The bin size is fixed to be 10.

It can be seen that distortion occurs in all these figures. Our analysis for various encoding shows that optimum performance is achieved when shift length  $s$  is equal to the encoding size  $b$  as evident from Fig. 4.



**Fig. 5** Plot of nodes against  $r$  for encoding length differs shift width. **a** 5 bit encoding with 4 bit shifting, **b** 5 bit encoding with 6 bit shifting, **c** 6 bit encoding with 5 bit shifting, and **d** 6 bit encoding with 7 bit shifting

## 4 Conclusion

In this paper, we introduced a generalized symbolic dynamics method and demonstrated the procedure that can better capture the dynamical characteristics of a given time series. According to the method, a given time series is transformed first into binary values with  $b$ -bit encoding of DCSD approaches by splitting the time series into  $m = 2^b$  layers. The resulting binary series is then converted to a decimal series. We demonstrated that the variation of the bin size in single and double bit encoding could not capture the bifurcation of the original series. Also, at each time when the bin is shifted less than or greater than  $b$ , the encoded series is observed with distortion. The results of the analysis show that as the order of encoding increases, the corresponding diagram approaches to original bifurcation diagram. The network constituted by the nodes of the decimal series will be analyzed in the future to investigate how the structural properties correspond to the time series parameter.

## References

1. K.-S. Chan, H. Tong, A note on noisy chaos. *J. R. Stat. Soc. Ser. B (Methodol.)* **56**(2), 301–311 (1994)
2. S. Iqbal, et al., Study of nonlinear dynamics using logistic map, in *LUMS 2nd International Conference on Mathematics and its Applications in Information Technology (LICM08)* (2008)
3. M.A. Savi, *Nonlinear dynamics and chaos*, in *Dynamics of Smart Systems and Structures* (Springer, Cham, 2016), pp. 93–117
4. R.M. May, Simple mathematical models with very complicated dynamics. *Nature* **261**(5560), 459–467 (1976)
5. K. Mischaikow et al., Construction of symbolic dynamics from experimental time series. *Phys. Rev. Lett.* **82**(6), 1144 (1999)
6. Z. Liu, Chaotic time series analysis, in *Mathematical Problems in Engineering 2010* (2010)
7. T. Raicharoen, C. Lursinsap, P. Sanguanbhokai, Application of critical support vector machine to time series prediction, in *Proceedings of the 2003 International Symposium on Circuits and Systems (ISCAS'03)*, vol. 5 (IEEE, 2003)
8. H.M. Morse, Recurrent geodesics on a surface of negative curvature. *Trans. Am. Math. Soci.* **22**(1), 84–100 (1921)
9. M. Wang, L. Tian, From timeseries to complex networks: the phase space coarse graining. *Physica A* **461**, 456–468 (2016)
10. N.E. Huang, et al., The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis. *Proc. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci.* **454**(1971), 903–995 (1998)
11. P. Singh, et al. The Fourier decomposition method for nonlinear and non-stationary time series analysis. *Proc. R. Soc. A Math. Phys. Eng. Sci.* **473**(2199), 20160871 (2017)
12. V.L.S. Freitas, J.C. Lacerda, E.E.N. Macau, Complex networks approach for dynamical characterization of nonlinear systems. *Int. J. Bifurcation Chaos* **29**(13), 1950188 (2019)
13. Z. Wei-Mou, B.L. Hao, Applied symbolic dynamics, in *Experimental Study and Characterization of Chaos: A Collection of Reviews and Lecture Notes* (1990), pp. 363–459
14. E.M. Bollt et al., Validity of threshold-crossing analysis of symbolic dynamics from chaotic time series. *Phys. Rev. Lett.* **85**(16), 3524 (2000)
15. B.L. Hao, Symbolic dynamics and characterization of complexity. *Physica D Nonlinear Phenomena* **51**(1–3), 161–176 (1991)
16. B.L. Hao, Applied symbolic dynamics. arXiv preprint chao-dyn/9806025 (1998)
17. J. Lacerda, E. Macau, Metodo baseado em redes complexas para a caracterizacao da dinamica caotica, in *Proceeding Series of the Brazilian Society of Computational and Applied Mathematics 5.1* (2017)
18. J. Lacerda, V. Freitas, E. Macau, Dynamical characterization of nonlinear systems through complex networks, in *Proceeding of the Series of the International Conference on Nonlinear Science and Complexity* (2016)
19. G. Boeing, Visual analysis of nonlinear dynamical systems: chaos, fractals, self-similarity and the limits of prediction. *Systems* **4**(4), 37 (2016)
20. C.M. Danforth, Chaos in an atmosphere hanging on a wall. *Math. Planet Earth* **17** (2013)
21. W. Li, K. Wang, H. Su, Optimal harvesting policy for stochastic logistic population model. *Appl. Math. Comput.* **218**(1), 157–162 (2011)

# Smart Mirror-Based Personal Healthcare System



V. B. Aanandhi, Anshida Das, Melissa Grace Melchizedek,  
Nived Priyadarsan, and A. Binu Jose

**Abstract** A smart mirror is a device that is an extended and enhanced version of a conventional mirror. It is a two-way mirror with an inbuilt display behind the glass. It allows the user to access and interact with various features that can be found in smart devices, like smart phones and tablets, etc. A smart mirror can display images, videos, current time, weather forecast, news feed, upcoming appointments, and all kinds of data supported by a smart device. The existing smart mirror frameworks were developed just to show time, date, and weather. After some updates, it contained schedules, alerts, and notices, and later, it got updated with music player and voice acknowledgment. The smart mirror will be redefined by remembering each and every aspect and disadvantages of the existing systems. The smart mirror offers unique features to improve the user experience and system security through biometric authentication, multimedia capabilities, and customized user profiles. This device can replace a wide range of household utilities like clocks, calendars, and external virtual assistants like Amazon Echo, Google Home, etc. The mirror will provide personalized healthcare services for each of its users. This includes analyzing varying health patterns of the user, like sleep patterns and body parameters and providing suggestions to improve their lifestyle. It also displays the medicine timetables of each user. Thus, it will be beneficial for elderly people and anybody with a busy routine.

---

V. B. Aanandhi (✉) · A. Das · M. G. Melchizedek · N. Priyadarsan · A. Binu Jose  
Department of Computer Science, Mar Baselios College of Engineering and Technology,  
Thiruvananthapuram, Kerala, India  
e-mail: [aanandhi.vb@gmail.com](mailto:aanandhi.vb@gmail.com)

A. Das  
e-mail: [anshidadas17@gmail.com](mailto:anshidadas17@gmail.com)

M. G. Melchizedek  
e-mail: [melissagrace97@gmail.com](mailto:melissagrace97@gmail.com)

N. Priyadarsan  
e-mail: [nivedofficial98@gmail.com](mailto:nivedofficial98@gmail.com)

A. Binu Jose  
e-mail: [binu.jose@mbcet.ac.in](mailto:binu.jose@mbcet.ac.in)

## 1 Introduction

The world is constantly evolving, and everything around us is developing. As technology and science progress, we are heading into a more digital lifestyle. We have got smart cities, smart houses, electric cars, and more. This fast way of life needs a further advancement of projects, which makes our life much simpler. A mirror is something that is essential in our day-to-day life to find out how we look. What if you could have a mirror telling you that it is cold outside, and recommend wearing a sweater? What if a mirror could monitor your day-to-day life? What if a mirror could suggest you a better lifestyle? When a mirror is made to do a task in addition to its normal functionality, with interconnected smart devices and other technologies with embedded intelligence that offers additional functionality, is what that makes a mirror smart. There are a few smart mirror projects which have been already developed. Some of them are Philips HomeLab System, an interactive mirror by Sam Ewen and Alpay Kasal, HUD Mirror, etc.

The Smart Mirror-Based Personal Healthcare System helps in developing a smart home and gives a unique environment to the users. It will also provide a set of features that helps to improve the user's experience. The system provides security through biometric authentication and uses IoT that makes devices to work according to the user's preferences, thus providing a whole new experience to the user. In addition to all these, the mirror will also analyze the health status of the user with the help of the Health Status Prediction feature and a user application and suggest a healthy lifestyle to the user. Thus, it will be beneficial for the elderly and anybody with a busy routine. By incorporating these features into a mirror, all the relevant information can be viewed in such a way that it fits seamlessly with the everyday routine of the user.

## 2 Literature Review

In the paper, "Smart Mirror—A Home Automation System Implemented Using Ambient Artificial Intelligence" [1] by Dhamangi et al. discusses about a smart mirror which is both interactive and futuristic with artificial intelligence for home automation as well as it is also helpful in commercial uses and public environments. Most of the work done by the smart mirror is controlled by the Raspberry Pi. The basic requirements of the smart mirror are—microphone, speakers, an LCD monitor covered with a sheet of two-way acrylic mirror, and it is connected by a Raspberry Pi. The mirror also provides the basic functionalities performed by a smart device such as displaying of weather, latest updates of news and headlines, local time corresponding to a particular location, etc. Using voice commands, the user is able to interact with the mirror. In addition to all these, to make it more user-friendly, Remote Configuration Tool (RCT) is also created to help the user with the working of the mirror.



In “Raspbian Magic Mirror—A Smart Mirror to Monitor Children by Using Raspberry Pi Technology” [2] by Siripala et al. contains information related to smart mirrors with reference to problems faced by parents/guardians nowadays in monitoring their children while they are away at work. In that case, a demand arises for a system which can be easily handled, and at the same time, it should be smart in accordance with rapid advancements in technology. This system is based on Internet of Things (IoT) by using Raspberry Pi technology. It is a smart mirror which will have the ability to display all the advanced details and connect with the user with the help of an Android application. Even though many smart mirror-related projects have been already developed, Raspbian Mirror which is demonstrated in this paper is much more interactive and advanced; moreover, it primarily targets working parents. The Raspbian Magic Mirror, in addition to displaying all the basic details, it will help the parents to monitor their children and assist them in their studies. This mirror can also be used as an ordinary mirror that will make day-to-day life easier, which is also an integral part of home automation.

The paper “Smart Mirror using Raspberry Pi” [3] by Pathak et al. contains details about the design and development of a smart mirror using Raspberry Pi with additional features such as face recognition for security and smart unlocking process. Here, they aim to create a system where the face is detected using OpenCV. The mirror will identify user’s face, and it will be processed using Raspberry Pi and then displays that user’s details. User’s image will be stored in a database. The mirror will also display basic details like weather, time, date, etc. The concept of Internet of Things (IoT) is another domain of this mirror.

By referring the above research papers, keeping in mind all the merits and demerits of the existing systems, we have designed and developed a smart mirror that provides easy access for a person to receive all the information that could affect how they prepare for the day. Through the use of LED displays and a two-way mirror, weather, time, date, news, and other useful information is available at a glance. The smart mirror offers unique features to improve the user experience and system security through biometric authentication (Facial Recognition) and customized user profiles. The mirror will also provide personalized healthcare services for each of its users by reminding them of their medicine schedules and suggesting them a healthy lifestyle based on his/her sleep patterns and body parameters. Thus, it will be beneficial for the elderly and anybody with a busy routine. By building these features into a mirror, it is possible to present the relevant information in such a way that it will seamlessly blend together with the user’s daily routine.

### 3 Facial Recognition

A facial recognition system is generally a system that is capable of verifying or identifying a user from a digital image. It is usually used to increase the security of any standard system that requires biometric authentication. It can also be compared with other biometrics, for example, fingerprint and iris recognition systems. The accuracy

of the iris recognition and fingerprint systems is much more as compared to facial recognition system, but still it is widely adopted in many systems due to its contactless and non-invasive process. Recently, it was known for commercial identification and as a marketing tool. There are many algorithms used to implement facial recognition. The main role of these algorithms is to pick specific details in a person's face such as shape of face, chin, etc. convert them into mathematical representations and compare it with data in the database. The data collected from a particular face is called face template, and it is often different from normal photographs because it contains details that can be used to distinguish one face from another.

In our system, facial recognition is used mainly for identification of the user and for creating customized user profiles. Firstly, the owner of the system is given an admin panel, through which he/she can add as many users as they want, into the system. The owner can register a user into the smart mirror system and give the user a unique ID. When a user stands in front of the mirror and if the user is using the mirror for the very first time, then the camera captures the image of the user, and it will be stored in the database which will be referred in future. The ID given to the user will be used for two purposes: one is for registering in the Android application, and secondly, it will be used to generate a user profile, which contains all details related to that user, for the newly registered user. Once the user registers in the Android application, certain inputs will be taken from the user's side. There are a total of seven inputs taken, in which four of them are user inputs and remaining three are user sleep parameters, which are explained further later on. These are just one time processes; the next time when the same user wants to use the mirror, the camera captures the user's image and identifies the user. Finally, after the identification, the user's profile is activated, and the rest of the functionality will be done according to the user's preference.

If the registered user has to log into the system, then he/she has to be authenticated first. To achieve this, we use:

- Face Detection
- Face Recognition

### ***3.1 Face Detection***

Each person registered into the system has an initial training set of ten images. The size of the training set has to be increased for improving accuracy, and for this, Image Data Augmentation technique is used. The Image Data Augmentation method creates different forms of the images in the training dataset, which results in the expansion of the training dataset. The different versions of the image created by this technique will belong to the same class as the original image. The training dataset is expanded so that the ability and the performance of the model to generalize can be improved. The same process was applied on the test set as well.

The Image Data Augmentation techniques used on the distinct images were:

- Horizontal Shift
- Vertical Shift
- Random Brightness
- Random Zoom

Face detection is used to detect the face in the given image. From the video stream, we extracted a frame, and this frame is preprocessed to obtain an image blob. This image blob is given to our extractor, which is a pre-trained Caffe deep learning model. This model localizes the faces in the image blob and returns the face coordinates. Then, these face coordinates are given to an embedder which is again another pre-trained model called OpenFace deep learning model. This embedder generates 128 D face vectors via triplet loss function, and these vectors act like the summary of the face.

### 3.2 Face Recognition

It is a process of verifying or identifying a person from a video source. We have used a SVM model for facial recognition. The SVM model is trained using the 128 D face vectors of the images in the training set. The 128 D face vectors of a person generated by embedder are given to the SVM model. The SVM model based on Euclidean similarity measure will classify the face as 'unknown' if it is an unregistered user and classifies a registered user under his/her name (Figs. 1 and 2).

## 4 Personalized Healthcare Services

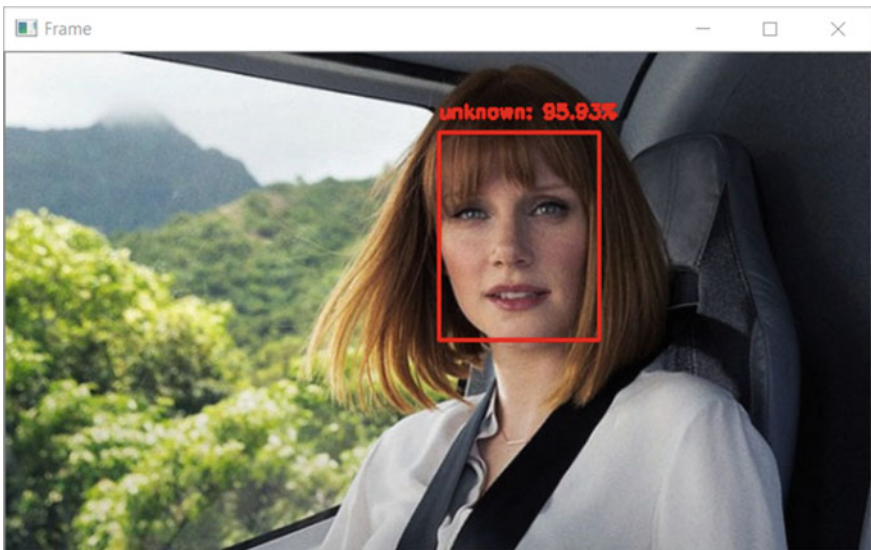
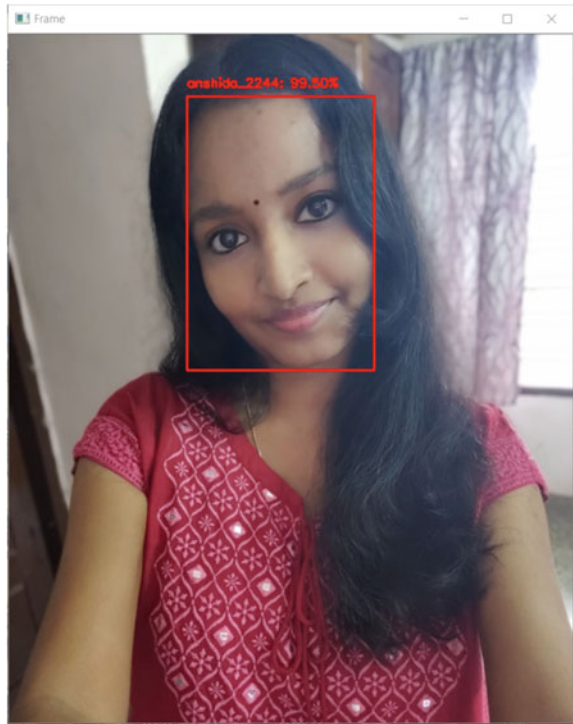
In this fast-growing world of technological innovation and knowledge, one of the most common symptoms felt in the third millennium is anxiety and immense pressure due to a busy lifestyle. Some of the most common effects of busy lifestyles are unhealthy and chaotic nutrition, restless or insufficient sleep, and lack of sport and physical activities. The tendency of ignoring health or giving less attention towards one's personal health will always resort to major health problems. So we designed a system that will suggest the user a better lifestyle based on their health parameters.

The main feature of Smart Mirror-Based Personal Healthcare System is User Health Monitoring, and its main aim is to accept certain parameters from the user, and based on these inputs taken, the system will tell the user whether the user's health is good or degrading. If the user has degrading health, then the system will suggest them a better lifestyle.

To implement this feature, we have two modules, they are:

- (1) Health Status Prediction
  - Sleep Pattern Recognition Model
  - Body Parameter Prediction Model

**Fig. 1** Face Recognition feature successfully recognizing the face of a registered user



**Fig. 2** Face Recognition feature successfully rejecting the face of an unregistered user

## (2) Android Application

The models for Health Status Prediction resides in the server. The user inputs and user sleep parameters generated from the Android application are send to these models via post requests. The Sleep Pattern Recognition model predicts the user's sleep status, whether user is having good, bad, or moderate sleep, based on the user sleep parameters. The Body Parameter Prediction model predicts the user's body parameters, like BMI and BMR, based on the user inputs. The predicted values of BMI and BMR are compared against their standard ranges to ultimately conclude whether the user is physically healthy or not. The results from both these models are analyzed together to obtain the final health report or improvised lifestyle suggestions. The final health report of the user is send back to the Android application by the server as the response.

### ***4.1 Health Status Prediction Using Machine Learning***

In Smart Mirror-Based Personal Healthcare System, the aim of using the health status prediction feature is to know the health status of the user. When seven inputs that is generated by the Android application is received by the health status prediction models residing in the server, they will be analyzed, and whatever be the health status, it will be displayed on the mirror as well as in the app. In order to analyze the inputs, both classification and regression are performed.

In order to train our models for both classification and regression, respectively, we used a dataset, which was customized from the ISRUC-Sleep Dataset. The data in this dataset was collected from human adults, including healthy subjects with sleep problems and subjects with insomnia-medication effects. Each recording has been selected from the PSG recordings collected by the Coimbra University Hospital (CHUC) sleep medicine center. The dataset consists of three classes of results:

- Data for 100 subjects, with one recording session per subject.
- Data obtained from 8 subjects, two recording sessions were held per subject.
- Data obtained from a single recording session for ten healthy subjects.

#### **Sleep Pattern Recognition Model (SVM)**

In machine learning, the Support Vector Machine (SVM) is a supervised learning model which is associated with learning algorithms that analyzes the data used for classification (differencing between groups), regression (creating a mathematical model to predict certain things), and even outlier detection. Linear SVM works by drawing a line between two groups. The aim of SVM is to mention a hyperplane which maximizes the margin between different classes.

In our project, SVM is used to perform multiclass classification in order to predict whether the user's sleep is good, bad, or moderate based on the user's sleep parameters. In the beginning, we tested the dataset with six algorithms that is Naïve Bayes,

Logistic Regression, KNN, Random Forest and Decision Tree and SVM gave the highest accuracy (0.89655) among the six. So, to perform classification, we opted the SVM algorithm.

### Body Parameter Prediction Model (RF Regressor)

Random Forest is a combination technique that can be performed for both regression and classification tasks using multiple decision trees. In our project, regression is used to predict the user's body parameters, that is, Body Mass Index (BMI) and Body Metabolic Rate (BMR) based on the user inputs (age, weight, height, gender). For performing regression, we tested our dataset with two algorithms, that is, Linear Regression and Random Forest Regressor algorithm. Ultimately, Random Forest Regressor algorithm was chosen as it gave the lowest root mean squared error.

Body Metabolic Rate determines the amount of calories that get burned when a person remains at rest. To calculate BMR, we used a formula which is different for both males and females. The equation we have used here is Harris–Benedict equation, it is given as follows:

For females:

$$\begin{aligned} \text{BMR} = & 665.1 + (9.6 \times \text{weight in kg}) \\ & + (1.8 \times \text{height in cm}) - (4.68 \times \text{age in years}) \end{aligned} \quad (1)$$

For males:

$$\text{BMR} = 66.47 + (13.7 \times \text{weight in kg}) + (5 \times \text{height in cm}) - (6.78 \times \text{age in years}) \quad (2)$$

The BMR range of both males and females is as follows:

- For females: 1400–1550 cal
- For males: around 1800 cal

If the calculated BMR is high, then it means that the user's metabolic rate is also high, indicating that more calories will be burned, and hence, the weight of the user decreases. When BMR is too high, then more calories will be burned, and the user will eventually become too thin, and if BMR is too low, then user will become obese. For a healthy lifestyle, it is advisable to maintain a moderate BMR.

Body Mass Index (BMI) is the person's weight by their height squared. The equation is as follows:

$$\text{BMI} = \frac{\text{weight}}{\text{height}^2} \quad (3)$$

It is used to check if the person's weight is appropriate to his/her height. If the BMI is less than 18.5, then the person is said to be under weight. If the BMI is between 18.5–24.9, then the person has normal weight. If the BMI range is from 25.0 to 29.0,

then the person is said to be overweight, and range above 30 means that the person is obese.

## 4.2 *Android Application*

In Smart Mirror-Based Personal Healthcare System, the main function of the Android application is to generate inputs. Once the user registers in the Android application using their unique ID, certain inputs will be taken from the user. There are a total of seven inputs taken in which four of them are user inputs, and remaining three are user sleep parameters. The user inputs are: gender, age, height, and weight. The user sleep parameters are Total Time in Bed, Total Awake Time, and Total Sleep Time. The sleep parameters are explained as follows:

- Total Awake Time: Total time the subject was awake from the beginning and till the end of the whole session.
- Total Sleep Time: Total time the subject was asleep from the beginning and till the end of the whole session.
- Total Time in Bed: Total time in bed of a subject is the sum of how many minutes was he/she awake (Total Awake Time) and how many minutes was he/she asleep (Total Sleep Time) during the time he/she got in bed and got out of bed.

Whenever the user uses this application, two main questions will be asked which are: “When do you go to sleep?” and “When do you wake up?” The user needs to set the time and press the “Track Sleep” button for sleep tracking to start. Even if the user forgets to press the button, the sleep tracking will automatically start at the “When do you sleep?” time set by the user. It is to be noted that before going to sleep, the user should place the phone on their mattress in such a way that it does not fall off easily. When the user gets up the very next day, he/she will get the sleep results of the previous night and his/her health report in the app. For example, if the user gave 11 pm as the “When do you sleep?” time and “When do you wake up?” time as 6 am, then the duration from 11 pm to 6 am is 7 hrs and that will be taken as our first parameter, i.e., Total Time in Bed. The next parameter is Total Awake Time, and to calculate this, we should know how long the user has used their mobile phone in this duration. To calculate this, we have introduced two factors: accelerometer sensor and screen activity. The accelerometer sensor is used to detect the device motion, and the screen activity is to check whether the screen is on or off.

There are users who suffer from sleep disorders, where they are extremely disturbed in their sleep. They turn about in their sleep, wakeup abruptly, etc. When user’s phone screen is on in the 7 hrs duration, we assume that user is using the phone. Suppose user’s phone screen was off, then we assume that the user is not using the phone. But in order to detect disturbances in their sleep and even when their phone is off, accelerometer sensor is used. While the user sleeps, the accelerometer sensor detects slightest disturbances made by the user. If the disturbances go on for more than an hour, it means that the user is highly disturbed in their sleep and could be

suffering from sleep disorders like sleep apnea, restless legs syndrome, insomnia, REM sleep behavior disorder, etc. When phone screen is off, accelerometer sensor is activated else screen activity is activated. The time intervals for which accelerometer sensor and screen activity were in action, respectively, are added together to obtain the Total Awake Time.

Thus, the second parameter is calculated. So, as of now, we have two parameters, and from this, the third parameter is calculated, i.e., Total Sleep Time. The Total Sleep Time is calculated as the difference between the Total Time in Bed and Total Awake Time. We have now obtained all the seven inputs, and using a post request, these inputs will be send to the health status prediction models (Sleep Pattern Recognition Model and Body Parameter Prediction Model) which are residing inside the server.

### ***4.3 User Prescription Alerts***

This feature of our system mainly allows a user to add or remove reminders about his/her medicine intake through the Android application. According to this information, the mirror will display the medicine prescription whenever user's face is recognized and their profile gets activated, thereby alerting the user about their health (Fig. 3).

## **5 Home Automation Using IoT**

Another main feature of our project is home automation using IoT. This smart mirror is meant to simulate all of the normal mirror interfaces. The regular mirror and mirror functionality are demonstrated using a two-way acrylic mirror. A flat LED monitor that is powered by Raspberry Pi is used for display. The Raspberry Pi runs Python scripts that send requests using different protocols. It is through these requests, the NodeMCU take input on whether to turn on/off different relays that control each appliance in the user's home. NodeMCU is a microcontroller programmed using Arduino IDE. Each pin can be set as input or output. The microcontroller is connected to the Internet with a built-in Wi-Fi module. The requests from the Raspberry Pi are received by the NodeMCU which would be already preprogrammed on how to respond to each call through its IDE. The relays are the most basic part of the IoT system. The connections would be in a "Normally Open" sequence which basically means that the relay will close and complete the circuit if it gets turned on by the microcontroller and vice versa. Thus, it works as a controllable switch. When user's profile gets activated after user identification, his/her home appliances, like lights, A/C, etc., will start to work according to their preferences.



**Fig. 3** Mirror displaying user Anshida’s sleep status and medicine prescriptions



## 6 Conclusion

The smart mirror is a smart home device that helps the users to store their day-to-day events and reminders to keep them updated. It also helps the user to replace clock, calendar, and smart home assistants with a single device. In the world of connected devices today, safety cannot be compromised; hence, the mirror makes use of facial recognition for identifying the user. The smart mirror integrates several impressive features and is user and developer friendly. The home automation feature provides a highly personalized experience for the user, where users can set their preferred appliance settings, and these settings will be activated once the user’s profile in the mirror is activated. It can be improvised further depending on the strength of the household equipment and access to personalized information services. The smart mirror allows its users to experiment with it and encourages the users to customize the device based on their personal needs. It successfully monitors the user’s body parameters and sleep patterns through the personalized healthcare feature and suggests them a better lifestyle.

## 7 Future Scope

Sleep plays a crucial role in our daily lives, and it is an essential factor that people overlook while considering their overall health. Studies have shown that lack of sleep or disrupted sleep cycles can lead to sleep disorders and other critical medical conditions, including heart disease, diabetes, and obesity. Keeping this in mind, we have come up with the innovative design of a highly cost-effective smart mirror that provides personalized healthcare services based on sleep. Our smart mirror analyzes real-time sleep patterns of the user and keeps them updated about the quality of their sleep and the possible health risks that they are bound to face if they have a low-quality sleep cycle. The mirror also provides different solutions with which the user can overcome their insomniac state.

The mirror does not limit itself to the user's sleep, but along with it, it also considers important body parameters, like BMI and BMR, and uses them to generate a health report which gives the user a complete overview of their health. It also reminds the users to take their medications on time. The home automation feature provides a highly personalized experience for the user, where users can set their preferred appliance settings (dim the lights, start light music, lock doors, switch off other appliances), which helps them to get proper sleep, and these settings will be activated once user's profile in the mirror is activated. In future, with the aid of evolving smart mirror technology, further progress can be made by converting it to touch screen mode. On advanced integration with the medical field, the smart mirror can help doctors to get first-hand data about their patient's sleep patterns, as well as other parameters necessary to make a proper diagnosis. This improves the overall effectiveness of the smart mirror as it involves the user into the whole experience. This system can also be made to help people set daily fitness goals and can be linked to wearables which will help in getting more first-hand activity of a person.

In addition to the existing ones, more widgets can be added to the smart mirror, and to increase the security, thumb impressions can also be used along with the facial recognition technology. The home automation feature of the mirror can be improvised further depending on the strength of the household equipment and access to personalized information services. With better advancement in technology, smart mirrors can also be used in other fields such as business, marketing, fashion, and so on.

## References

1. J.R. Dhamanigi, et al., Smart Mirror—A Home Automation System Implemented Using Ambient Artificial Intelligence (2017)
2. R.M.B.N. Siripala, M. Nirosha, P.A.D.A. Jayaweera, N.D.A.S. Dananjaya, S.G.S. Fernando, Raspbian Magic Mirror—A Smart Mirror to Monitor Children by Using Raspberry Pi Technology. *Int. J. Sci. Res. Public.* 7(12), 281–295 (2017)
3. A. Pathak, A. Mishra, R. Sarate, S. Bhavsar, N. Patel, Smart Mirror using Raspberry Pi. *Int. J. Recent Trends Eng. Res.* 4(3), 353–358 (2018)

# On-off Thinning in Linear Antenna Arrays Using Binary Dragonfly Algorithm



Ashish Patwari, Medha Mani, Sneha Singh, and Gokul Srinivasan

**Abstract** The aim of this work is to study the suitability of two newly introduced bio-inspired algorithms, namely the dragonfly algorithm (DA) and the salp swarm algorithm (SSA) for thinning a linear antenna array. In array thinning, a fully populated array is chosen as a starting point, and a thinned array is obtained through careful deactivation of select sensors such that the residual active sensors enable the array to achieve a desired side-lobe performance. In this paper, we apply the binary versions of DA and SSA, namely the binary dragonfly algorithm (BDA), and the binary salp swarm algorithm (BSSA) to thin a symmetric linear array with uniform inter-element spacing of half wavelength. Extensive simulations were performed in MATLAB by considering arrays of different sizes. The results obtained from BDA and BSSA were compared against those obtained from the binary versions of two benchmark algorithms, namely the genetic algorithm (GA) and the gray wolf optimizer (GWO). Relative side-lobe level (RSL) and filling percentage were used as performance comparison metrics. It has been observed that both BDA and BSSA offer promising results in line with BGA and BGWO. More specifically, BDA was found to be faster than BSSA.

**Keywords** Array Synthesis · Binary Dragonfly Algorithm (BDA) · Binary Salp Swarm Algorithm (BSSA) · Linear antenna arrays · Relative side-lobe level (RSL) · Thinned arrays

## 1 Introduction

Sensor arrays have long been used in radar, sonar, wireless communications, medical imaging, astronomy, and many other fields of study; as they offer better directional properties than a single sensor. A sensor array is formed by arranging at least two sensors according to a specific geometric layout. A linear array is the most basic and

---

A. Patwari (✉) · M. Mani · S. Singh · G. Srinivasan  
School of Electronics Engineering, Vellore Institute of Technology, Vellore 632014, Tamil Nadu, India  
e-mail: [ashish.p@vit.ac.in](mailto:ashish.p@vit.ac.in)

straightforward method to obtain a sensor array. Active or adaptive arrays have the ability to adjust their radiation patterns according to the changes in the signal environment and could, therefore, overcome noise, intentional jamming and interference using techniques such as beam-steering and null-steering [1].

The radiation pattern of an array represents the spatial directions along which it radiates/receives energy and is characterized by parameters such as the main lobe width and side-lobe levels (SLL). The height of the side-lobes relative to the main lobe height is known as relative side-lobe level (RSLL) of the array. A uniform linear array (ULA) with uniform amplitude weighting across all its elements has an RSLL of approximately  $-13$  dB. High values of RSLL are not desired as the presence of active interferers or intentional jammers in the side-lobe region can compel the array to receive high amounts of unnecessary energy from undesired directions [2]. Hence, in many practical applications, it is desired to have low RSLLs [3].

Density tapering or array thinning is a combinatorial technique used to determine the optimal ON-OFF pattern (thinning pattern) for individual array elements such that the thinned array achieves a desired side-lobe performance. After thinning, a few elements over the array's span remain turned-OFF, thereby causing variations in the density of the array. This process breaks the inherent periodicity among array elements and therefore disrupts the nature and extent of sidelobes.

Though thinning remained an active research topic for the past few decades, it has assumed even more importance in the modern era. With the advent of Internet of Things (IoT) and fifth-generation (5G) communication in recent years, a plethora of devices are densely packed in a given area (say, a living room or a hotel lobby). SLL minimization plays a major role in reducing the interference between such closely spaced devices [4–7]. The choice of millimeter wave frequencies (above 10 GHz) for 5G has paved a way to accommodate large antenna arrays (with hundreds of elements) within a small space, thanks to the tiny wavelengths [8]. Antenna arrays and associated signal processing techniques are geared up to play a big role in the last mile of 5G/6G networks [9, 10]. The properties of linear antenna arrays in relation to the millimeter wave frequencies have been studied in recent years [11–14].

Evolutionary and swarm-based algorithms have been applied extensively for array pattern optimization. Techniques such as genetic algorithm, particle swarm optimization, invasive weed optimization, fruit fly optimizer, gray wolf optimizer, flower pollination algorithm, grasshopper optimization, ant lion optimization, and many others have been applied for array synthesis [3, 15–25]. A comprehensive review about the use of various nature inspired algorithms for linear array synthesis was reported in recent years [26]. It is not uncommon to see the use of hybrid algorithms (obtained by combining the best features of two different algorithms) for array synthesis [27–29]. However, as per our current knowledge, salp swarm optimization [30] and dragonfly algorithms [31] have not been studied for linear array thinning. As widely known from the No-Free-Lunch (NFL) theorem in optimization theory [32], there is no single algorithm that can solve all optimization problems. In other words, not all algorithms are equally effective against a given objective function. Therefore, there is always a scope to test the suitability of newer algorithms in optimizing a particular objective function.

The rest of this paper is organized as follows: Section 2 presents the array model. Section 3 presents a brief review of various bio-inspired algorithms considered in this paper. Section 4 outlines the methodology followed for simulations. Section 5 presents the numerical simulation results and Sect. 6 concludes the paper with a few future directions.

## 2 The Array Model

In this section, we present the array model considered for this study. A symmetric linear antenna array with  $M = 2N$  elements is considered along the  $x$ -axis as shown in Fig. 1. There are  $N$  antenna elements on either side of the origin. The array is assumed to be steered toward the broadside (azimuth steering  $90^\circ$  and elevation steering  $0^\circ$ ). An inter-element spacing of half wavelength is assumed.

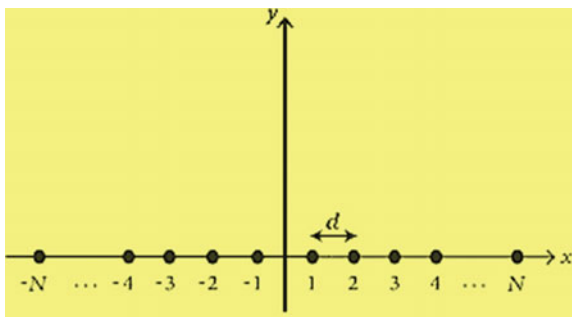
The array factor of such a symmetric linear array with a uniform inter-element spacing of  $d$  is given as follows

$$AF(u) = 2 \sum_{n=1}^N w_n \cos((n - 0.5)kdu), \quad (1)$$

where  $w_n$  denotes the binary (on-off) element weights,  $k = 2\pi/\lambda$  denotes the wave number, and  $u = \cos\phi$  denotes the  $u$ -space in the azimuth range of  $0^\circ \leq \phi \leq 180^\circ$ . The term  $(n - 0.5)d$  denotes the position of the  $n$ th sensor along the array axis, and by default,  $d = \lambda/2$  is assumed. The cost function for SLL minimization in the above array is given by

$$C = \min(\max(20 \log_{10}(AF(u))); \frac{\lambda}{Nd} \leq |u| \leq 1), \quad (2)$$

**Fig. 1** Symmetric linear antenna array with  $2N$  elements



where  $\frac{\lambda}{Nd} \leq |u| \leq 1$  denotes the side-lobe region in the  $u$ -space that starts soon after the first null. The aim of this work is to minimize this cost function using the bio-inspired algorithms in question.

### 3 Brief Review of Algorithms Used

Bio-inspired algorithms are inspired from nature or biology. In particular, swarm optimization algorithms are based on the social or collective behavior of a group of animals. Here, we briefly describe the main features of the four bio-inspired or meta-heuristic algorithms that we considered in this paper.

#### 3.1 *The Genetic Algorithm (GA)*

The genetic algorithm (GA) is one of the oldest evolutionary algorithms and has been largely accepted for optimization problems. It is based on Darwin's evolution theory. The GA and its parts, i.e., selection, crossover, mutation, fitness, and elitism in relation to antenna array optimization have been widely studied in literature [2, 3, 16]. This algorithm has been widely used for array thinning, phase-only nulling, beam-forming, and many other array synthesis applications.

#### 3.2 *The Gray Wolf Optimizer (GWO)*

Gray wolf optimization (GWO) is a meta-heuristic algorithm that mimics the social hierarchy and hunting mechanism of gray wolves [21, 33]. The hierarchy of gray wolves is as follows: the leaders are male and female, called alpha ( $\alpha$ ) wolves. The second level within the hierarchy comprises the beta ( $\beta$ ) wolves. These are followed by delta ( $\delta$ ) wolves. Finally, the least-ranked gray wolves are the omega ( $\omega$ ). The position of the wolves leading the pack has to be followed by the remaining wolves. The main steps followed by gray wolves while hunting are as follows: (i) tracking, chasing, and approaching the prey, (ii) cornering and immobilizing the prey, and (iii) attacking the prey.

#### 3.3 *The Dragonfly Algorithm (DA)*

Dragonfly algorithm (DA) is yet another meta-heuristic algorithm developed in the recent times [31]. It is inspired by the food searching behavior of dragonflies. In

this algorithm, dragonflies reach the food source by using five important processes, namely separation, alignment, cohesion, food attraction, and enemy repulsion.

Dragonflies move in swarms mainly for two purposes: hunting and migration. Hunting swarm is called the static (feeding) swarm and represents the exploration phase. The migratory swarm is called a dynamic swarm and indicates the exploitation phase. In static swarm behavior, the dragonflies fly back and forth in small groups to hunt for other flying insects over a smaller area. It is characterized by local movements and abrupt changes in the flying path. In the dynamic swarm, a huge number of dragonflies come together to migrate in one direction over long distances. The mathematical equations and calculations related to the algorithm can be found in [31]. However, a few important definitions are repeated here for easy understanding.

Separation, alignment, and cohesion are the basic traits of any swarm. Apart from these, attraction to food and avoidance of enemies are the additional features in dragonfly algorithm. Separation defines the distance between individual dragonflies and helps in avoiding collisions. Alignment is the ability of an individual to match the speed of the swarm. Cohesion is the binding force that glues the entire swarm closer to the center of mass. The separation, alignment and cohesion of the  $p$ th individual with the rest of the swarm are defined as follows:

$$\begin{aligned} S_p &= - \sum_{q=1}^K Y - Y_q \\ A_p &= \sum_{q=1}^K U_q / K \\ C_p &= \left( \sum_{q=1}^K Y_q / K \right) - Y, \end{aligned} \quad (3)$$

where  $Y$  denotes the position of the individual dragonfly in consideration.  $Y_q$  denotes the position of the  $q$ th neighboring dragonfly.  $K$  denotes the number of neighboring dragonflies.  $U_q$  is the velocity of the  $q$ th neighbor. The other two parameters, i.e., attraction to food and enemy repulsion of the  $p$ th individual are given by

$$\begin{aligned} F_p &= Y^+ - Y \\ E_p &= Y^- + Y, \end{aligned} \quad (4)$$

where  $Y^+$  is the position of the food source, and  $Y^-$  is the enemy position. The above five parameters control the behavior of the swarm of dragonflies. The movement of individual dragonflies within the search space is governed by the step vector ( $\Delta Y$ ) and the position vector ( $Y$ ). The step vector is similar to the velocity vector of the particle swarm optimizer (PSO) and defines the direction along which the dragonfly is moving. The step vector of a particular dragonfly is given by

$$\Delta Y_{t+1} = (sS_p + aA_p + cC_p + fF_p + eE_p) + w\Delta Y_t, \quad (5)$$

where  $t$  is the current iteration number. A total of six weights are used to update the step vector. The inertial weight  $w$  is multiplied to the current step vector in order to get the updated step vector. The position vector is then obtained using the step vector and is given as

$$Y_{t+1} = Y_t + \Delta Y_{t+1}. \quad (6)$$

Unlike the GWO, it can be seen that there is no hierarchy among dragonflies in the DA algorithm.

### 3.4 The Salp Swarm Algorithm (SSA)

The salp swarm algorithm (SSA) is one of the recent meta-heuristic algorithms that imitates the social behavior of salp swarms while navigating and foraging in oceans [30]. A new binary version of the SSA, namely the BSSA, has been proposed recently to solve discrete or binary optimization problems [34]. Salps have see-through barrel-shaped bodies and belong to the Salpidae family. They resemble jelly fishes in their movement and tissue structure. Salps are known to form the largest swarm among all creatures on the planet. A swarm of salps is called as a salp chain. The salp swarm algorithm (SSA) mathematically models the salp chains. Firstly, the swarm is divided into two groups: leader and followers. The leader salp is at the front of the swarm. The remaining salps in the rest of the swarm are referred to as followers. The leader guides the salp chain toward the food source  $F$  within the search space.

The mathematical formulation and details about the algorithm can be found in [30]. However, we repeat the definitions of a few important parameters here for easy reference. The position of the leader salp is updated using the following equation

$$\begin{aligned} x_i^1 &= F_i + k_1((ub_i - lb_i)k_2 + lb_i); k_3 \geq 0.5 \\ x_i^1 &= F_i - k_1((ub_i - lb_i)k_2 + lb_i); k_3 < 0.5, \end{aligned} \quad (7)$$

where the subscript  $i$  indicates the  $i$  th dimension.  $x_i^1$  denotes the position of the leader,  $F_i$  denotes the position of the food source,  $ub_i$  is the upper bound, and  $lb_i$  is the lower bound.  $k_1$ ,  $k_2$ , and  $k_3$  are random numbers. Among them,  $k_1$  is the most important parameter as it balances the exploration and exploitation phases of the SSA. It is given by the following expression

$$k_1 = 2e^{-\left(\frac{l}{L}\right)^2}, \quad (8)$$

where  $L$  is the maximum number of iterations, and  $l$  denotes the present iteration. As the iterations of the SSA progress, the value of  $k_1$  is adaptively decreased so that the algorithm first explores the search space and then exploits it. Parameters  $k_2$  and



$k_3$  follow a uniform distribution between 0 and 1. They help in deciding the step size and also whether the next position proceeds toward positive infinity or negative infinity. The position of the  $m$  th follower in the  $i$  th dimension is updated using the following equation.

$$x_i^m = \frac{x_i^m + x_i^{m-1}}{2}, \quad (9)$$

where  $m > 1$ . It is known that  $m = 1$  corresponds to the leader salp. Equation (7) shows that the position of the leader is updated only on the basis of the food source. The salp chain can be simulated using Eqs. (7) and (9). To achieve global optimization, the SSA updates the food source position  $F$  with the location of the leader salp in each iteration so that the remaining salp chain can chase the food source. It can be observed that the SSA algorithm is simple and easy to implement; as it has only one major parameter to control, namely, the  $k_1$  value as given in Eq. (8).

## 4 Proposed Method and Methodology

As mentioned earlier, due to array symmetry, it is sufficient to optimize the right half of the array in order to obtain the thinning pattern of the whole array. In addition, it is assumed that the first sensor and the  $N$  th sensor on the either side of the array are always ON. These sensors do not participate in the thinning process so as to preserve the array aperture. Therefore, the array thinning problem essentially boils down to optimizing the right half of the array (barring the first and the last sensor). Therefore, only  $N - 2$  array elements or binary variables remain to be optimized.

The cost function for SLL minimization defined in Eq. (2) serves as an objective function to the optimization algorithms. Various on-off combinations of the array elements are generated randomly to act as the initial population for the algorithms. The fact that the first and the last elements are always ON acts as a constraint on the objective function. This constraint ensures that the array aperture remains unaltered during the thinning process. Hence, the angular resolution and beam-width of the array remain intact. This constraint is generally referred to as the fixed first null beam-width (FNBW) constraint.

The four algorithms considered here (i.e., the genetic algorithm, the gray wolf optimizer, the dragonfly algorithm and the salp swarm algorithm) are applicable only for solving continuous problems in their original forms. However, array thinning is a discrete problem and needs binary algorithms. Therefore, a V-shaped transfer function has been used to determine the position vectors as needed for binary algorithms.

Symmetric arrays with 20, 60, and 100 elements were considered for thinning. Simulations were carried out in MATLAB 2016a. No additional toolboxes were needed. The filling percentage and RSSL were considered as metrics for comparing the performance of different algorithms. The filling factor ( $f$ ) is defined by

$$f = \frac{e_{on}}{e_{tot}}, \tag{10}$$

where  $e_{on}$  denotes the number of ones (active antennas) in the thinning pattern, and  $e_{tot}$  denotes the total number of active antennas before thinning. In principle,  $e_{tot} = N$ , if the thinning pattern of the right side alone is considered. The filling percentage can easily be evaluated using the above filling factor.

Figure 2 gives an overview of the methodology followed for MATLAB coding and simulations. The population size for each algorithm was taken as  $P = 50$ . Completion of hundred iterations or reaching a cost of  $-50$  dBi was set as the stopping criteria

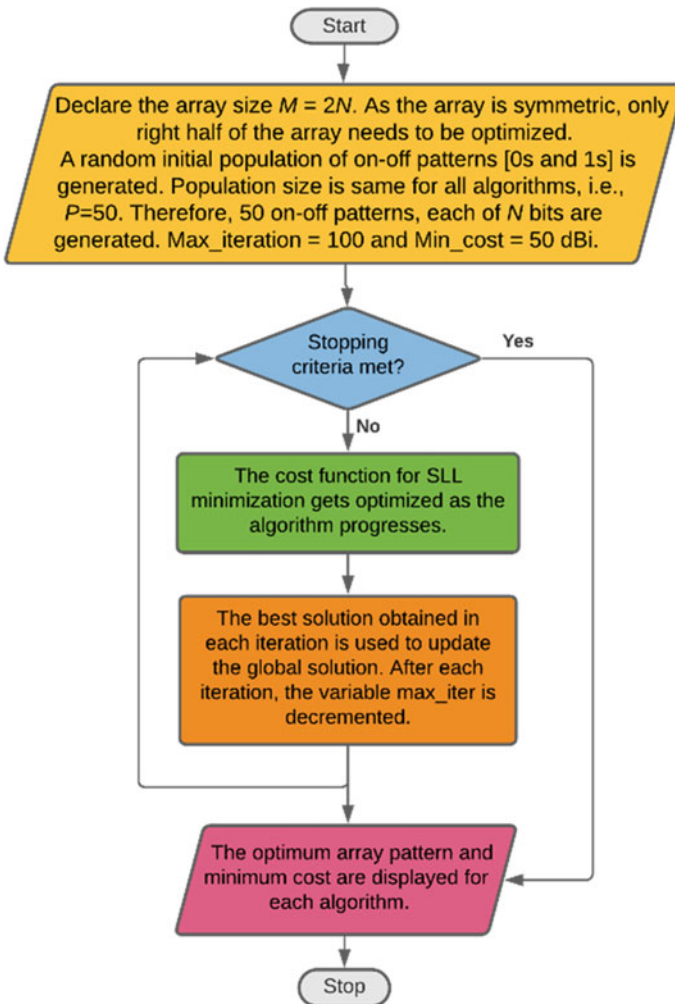


Fig. 2 Simulation methodology

for each algorithm. The MATLAB command `[ones (P,1) round (rand (P, N-2)) ones (P,1)]` was used to generate the initial random population needed for various algorithms. This command creates  $P$  binary vectors, each made of  $N$  binary variables, representing the initial thinning patterns.

## 5 Results and Discussion

This section describes the numerical simulation results obtained by following the methodology mentioned above.

### 5.1 Broadside Arrays

As a starting point, the BDA algorithm was used to thin a symmetric linear array of 20 elements ( $M = 20$ ). Figure 3 shows the results of thinning. The optimal thinning pattern of the right half of the array is obtained as 1,111,111,101. Only half of the array is represented here as the exact same behavior is observed on the other half due to symmetry. Also seen in Fig. 3 are the array radiation patterns before and

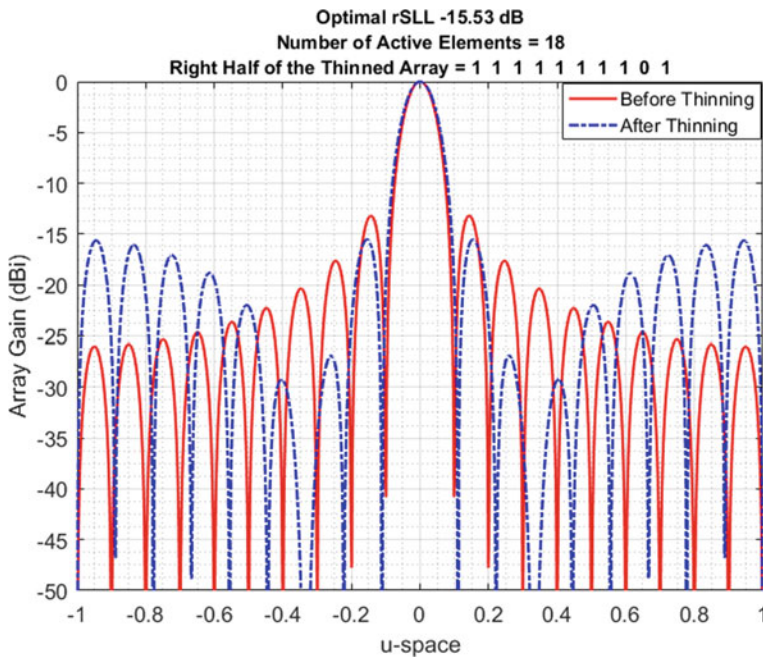
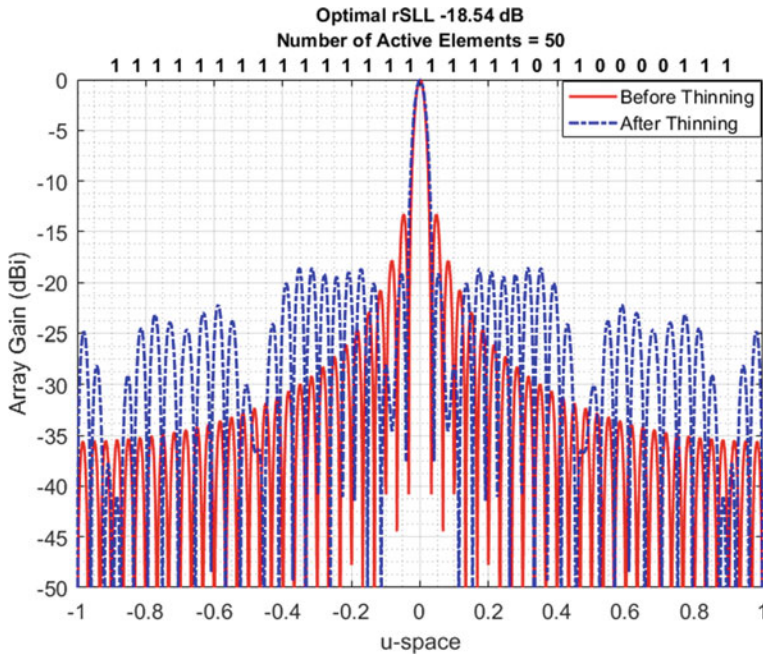


Fig. 3 Array pattern of the 20-element linear array before and after thinning



**Fig. 4** Array pattern of 60-element symmetric linear array before and after thinning

after thinning. The optimum RSLL obtained is  $-15.53$  dB, which accounts to a 2 dB improvement over the uniform array. After getting a basic idea of on-off array thinning from the above simulation, the methodology was extended to larger antenna arrays.

In a similar manner, Fig. 4 shows the thinning results for a 60-element symmetric linear array. It can be seen that this array provides further improvement in the SLL performance compared to the 20-element array. The RSLL is  $-18.54$  dB which is nearly 6 dB better than the uniform array of same size. The filling factor is  $f = 0.83$  indicating that 83% of the elements in the thinned array are active. Figure 5 shows the thinning pattern for a 100-element array. The 100-element array provides a sidelobe of almost  $-20$  dB.

Table 1 summarizes the array thinning results obtained from the four algorithms, namely the BGA, the BGWO, the BDA, and the BSSA, by considering arrays of size 20, 60 and 100 elements, respectively. In Table 1,  $a^r$  indicates  $r$  consecutive occurrences of the symbol  $a$ . This notation is only for concise representation.

It can be observed from Table 1 that the two new algorithms in question, i.e., the BDA and the BSSA offer comparable results as the benchmark algorithms, i.e., the BGA and the BGWO. It can be seen that the SLL performance is more or less similar among all the algorithms and is in agreement with the results reported in existing literature [35].



We have also compared the time taken by each algorithm to complete 100 iterations by executing each program 20 times on a laptop with 8 GB RAM and i5 processor. It was found that BGA, BGWO, and BDA needed less than 5 s for execution, whereas BSSA needed more than 100 s. Hence, in terms of speed of execution, the BDA outperforms the BSSA.

## 5.2 Beam-Steered Arrays

Many-a-times, it is needed to be able to steer the array's main lobe toward a desired direction in order to maximize the radiation toward a particular source or user. This can be achieved through beam-steering, where the main beam of the array is steered towards a specified angle through electronic phase shifting. The array factor for a beam-steered symmetric linear array is given by

$$AF(u) = 2 \sum_{n=1}^N w_n \cos((n - 0.5)kd(u - u_s)), \quad (11)$$

where  $u_s = \cos\phi_s$  and  $\phi_s$  denotes the azimuth steering angle.

Since BDA was better in terms of convergence and speed, we have tried to thin a beam-steered array using the BDA algorithm and obtained the following results. It has to be noted that the cost function remains same as in Eq. (2), but the limits for  $u$  are given by  $\frac{\lambda}{Nd} \leq |u - u_s| \leq 1$ . Figure 6 shows the thinning pattern of a 60-element array beam-steered toward  $u_s = 0.2$ . It can be seen that the main beam is now centered at  $u = 0.2$ . An SLL of almost  $-20$  dB was obtained, indicating the suitability of the BDA in thinning a beam-steered array.

## 6 Conclusion and Future Scope

The suitability of two recent bio-inspired algorithms, namely the BDA and the BSSA for array thinning application, has been presented in this paper. Four bio-inspired meta-heuristic algorithms were compared in terms of the RSL values and filling percentages, and it was found that the BDA and BSSA are no inferior to established optimization algorithms (such as binary genetic algorithm and binary GWO), when it comes to on-off array thinning. More specifically, the BDA was found to be faster than the BSSA for thinning application and was also found to be suitable for thinning beam-steered arrays. As a future extension, BDA and BSSA could be used to thin circular and planar arrays. They can also be used to detect sensor failures in large arrays and for subsequent pattern correction using the residual healthy elements in the array.

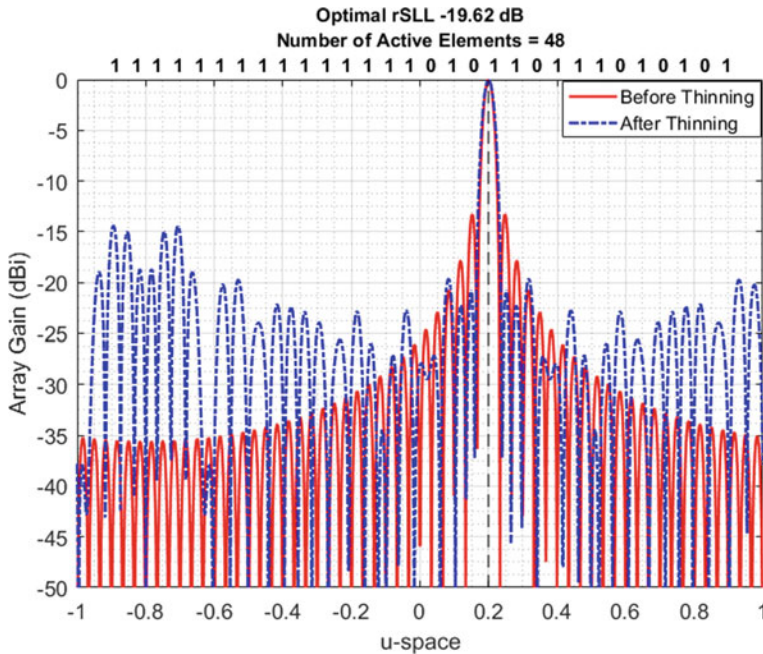


Fig. 6 Thinned pattern for a 60-element beam-steered array

## References

1. R.A. Monzingo, R. Haupt, T. Miller, in *Introduction to Adaptive Arrays*, 2nd edn. (Institution of Engineering and Technology, 2011)
2. F. Gross, in *Smart Antennas with MATLAB* 2nd edn. (McGraw-Hill Education, 2015)
3. R.L. Haupt, Thinned arrays using genetic algorithms. *IEEE Trans. Antennas Propag.* **42**(7), 993–999 (1994). <https://doi.org/10.1109/8.299602>
4. G.G. Lema, D.H. Hailu, T.B. Wuneh, SLL attenuation-based thinned antenna design for next-generation communications. *EURASIP J. Wireless Commun. Netw.* **2019**(1), 225 (2019). <https://doi.org/10.1186/s13638-019-1547-5>
5. G. Buttazzoni, F. Babich, F. Vatta, M. Comisso, Geometrical synthesis of sparse antenna arrays using compressive sensing for 5G IoT applications. *Sensors* **20**(2), 2 (2020). <https://doi.org/10.3390/s20020350>
6. G. Sun, Y. Liu, J. Li, Y. Zhang, A. Wang, Sidelobe reduction of large-scale antenna array for 5G beamforming via hierarchical cuckoo search. *Electron. Lett.* **53**(16), 1158–1160 (2017). <https://doi.org/10.1049/el.2016.4768>
7. M.Z. Hasan, H. Al-Rizzo, Beamforming optimization in internet of things applications using robust swarm algorithm in conjunction with connectable and collaborative sensors. *Sensors* **20**(7), 7 (2020). <https://doi.org/10.3390/s20072048>
8. T. Bai, A. Alkhateeb, R. Heath, Coverage and capacity of millimeter-wave cellular networks. *IEEE Commun. Mag.* **52**(9), 70–77 (2014). <https://doi.org/10.1109/MCOM.2014.6894455>
9. S. Kuttu, D. Sen, Beamforming for millimeter wave communications: an inclusive survey. *IEEE Commun. Surveys Tutorials* **18**(2), 949–973 (2016). <https://doi.org/10.1109/COMST.2015.2504600>

10. M. Wang, F. Gao, S. Jin, H. Lin, An overview of enhanced massive MIMO with array signal processing techniques. *IEEE J. Selected Topics Signal Process.* **13**(5), 886–901 (2019). <https://doi.org/10.1109/JSTSP.2019.2934931>
11. A. Patwari, G.R. Reddy, A conceptual framework for the use of minimum redundancy linear arrays and flexible arrays in future smartphones. *Int. J. Antennas Propag.* **2018**(9629837), 12 (2018). <https://doi.org/10.1155/2018/9629837>
12. A. Patwari, R.R. Gudheti, Novel MRA-based sparse MIMO and SIMO antenna arrays for automotive radar applications. *Progress Electromagnet. Res.* **86**, 103–119 (2020). <https://doi.org/10.2528/PIERB19121602>
13. A. Patwari, G.R. Reddy, DOA estimation and adaptive nulling in 5G smart antenna arrays for coherent arrivals using spatial smoothing. *IJMETS* **9**(11), 614–628 (2018)
14. A. Patwari, G.R. Reddy, H. Gupta, V. Nigam, Suitability of conventional 1D noise subspace algorithms for DOA estimation using large arrays at millimeter wave band. *Int. J. Appl. Eng. Res.* **12**(8), 1591–1597 (2017). <https://doi.org/10.37622/IJAER/12.8.2017.1591-1597>
15. H.M. Elkamchouchi, M.M. Hassan, Array pattern synthesis approach using a genetic algorithm. *IET Microwaves Antennas Propag.* **8**(14), 1236–1240 (2014). <https://doi.org/10.1049/iet-map.2013.0718>
16. J.R. Mohammed, Thinning a subset of selected elements for null steering using binary genetic algorithm. *Progress Electromagnet. Res.* **67**, 147–155 (2018). <https://doi.org/10.2528/PIERM18021604>
17. T.B. Chen, Y.B. Chen, Y.C. Jiao, E.S. Zhang, Synthesis of antenna array using particle swarm optimization. in *2005 Asia-Pacific Microwave Conference Proceedings*, December 2005, vol. 3, (2005) pp. 4. <https://doi.org/10.1109/APMC.2005.1606685>
18. M.M. Khodier, C.G. Christodoulou, Linear array geometry synthesis with minimum sidelobe level and null control using particle swarm optimization. *IEEE Trans. Antennas Propag.* **53**(8), 2674–2679 (2005). <https://doi.org/10.1109/TAP.2005.851762>
19. S. Pal, A. Basak, S. Das, A. Abraham, Linear antenna array synthesis with invasive weed optimization algorithm. in *2009 International Conference of Soft Computing and Pattern Recognition*, December (2009), pp. 161–166. <https://doi.org/10.1109/SoCPar.2009.42>
20. P. Saxena, A. Kothari, Ant lion optimization algorithm to control side lobe level and null depths in linear antenna arrays. *AEU—Int. J. Electron. Commun.* **70**(9), 1339–1349 (2016). <https://doi.org/10.1016/j.aeue.2016.07.008>
21. P. Saxena, A. Kothari, Optimal pattern synthesis of linear antenna array using grey wolf optimization algorithm. *Int. J. Antennas Propag.* (2016). <https://www.hindawi.com/journals/ijap/2016/1205970/> (Accessed May 13, 2020).
22. N. Mhudgetong, C. Phongcharoenpanich, S. Kawdungta, Modified fruit fly optimization algorithm for analysis of large antenna array. *Int. J. Antennas Propag.* (2015). <https://www.hindawi.com/journals/ijap/2015/124675/> (Accessed May 13, 2020)
23. L. Polo-López, J. Córcoles, J.A. Ruiz-Cruz, Antenna design by means of the fruit fly optimization algorithm. *Electronics* **7**(1), 1 (2018). <https://doi.org/10.3390/electronics7010003>
24. U. Singh, R. Salgotra, Pattern synthesis of linear antenna arrays using enhanced flower pollination algorithm. *Int. J. Antennas Propag.* (2017). <https://www.hindawi.com/journals/ijap/2017/7158752/> (Accessed May 13, 2020)
25. H. Wang, C. Liu, H. Wu, B. Li, X. Xie, Optimal pattern synthesis of linear array and broadband design of whip antenna using grasshopper optimization algorithm. *Int. J. Antennas Propag.* (2020). <https://www.hindawi.com/journals/ijap/2020/5904018/> (Accessed May 13, 2020)
26. A.K. Yerrola, P. Spandana, Optimization of linear antennas—a survey. *Int. J. Comput. Appl.* **171**(3), 17–20 (2017)
27. D. Prabhakar, M. Satyanarayana, Side lobe pattern synthesis using hybrid SSWOA algorithm for conformal antenna array. *Eng. Sci. Technol. Int. J.* **22**(6), 1169–1174 (2019). <https://doi.org/10.1016/j.jestch.2019.06.009>
28. A.A. Amaireh, A.S. Al-Zoubi, N.I. Dib, Sidelobe-level suppression for circular antenna array via new hybrid optimization algorithm based on antlion and grasshopper optimization algorithms. *Progress Electromagnet. Res.* **93**, 49–63 (2019). <https://doi.org/10.2528/PIERC19040909>



29. Z. Liang, J. Ouyang, F. Yang, A hybrid GA-PSO optimization algorithm for conformal antenna array pattern synthesis. *J. Electromagnet. Waves Appl.* **32**(13), 1601–1615 (2018). <https://doi.org/10.1080/09205071.2018.1462257>
30. S. Mirjalili, A.H. Gandomi, S.Z. Mirjalili, S. Saremi, H. Faris, S.M. Mirjalili, Salp swarm algorithm: a bio-inspired optimizer for engineering design problems. *Adv. Eng. Softw.* **114**, 163–191 (2017). <https://doi.org/10.1016/j.advengsoft.2017.07.002>
31. S. Mirjalili, Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems. *Neural Comput Appl.* **27**(4), 1053–1073 (2016). <https://doi.org/10.1007/s00521-015-1920-1>
32. D.H. Wolpert, W.G. Macready, No free lunch theorems for optimization. *IEEE Trans. Evol. Comput.* **1**(1), 67–82 (1997). <https://doi.org/10.1109/4235.585893>
33. S. Mirjalili, S.M. Mirjalili, A. Lewis, Grey wolf optimizer. *Adv. Eng. Softw.* **69**, 46–61 (2014). <https://doi.org/10.1016/j.advengsoft.2013.12.007>
34. R.M. Rizk-Allah, A.E. Hassanien, M. Elhoseny, M. Gunasekaran, A new binary salp swarm algorithm: development and application for optimization tasks. *Neural Comput. Appl.* **31**(5), 1641–1663 (2019). <https://doi.org/10.1007/s00521-018-3613-z>
35. H. Rezagholizadeh, D. Gharavian, A thinning method of linear and planar array antennas to reduce SLL of radiation pattern by GWO and ICA algorithms. *AUT J. Electri. Eng.* **50**(2), 135–140 (2018). <https://doi.org/10.22060/ej.2018.13697.5182>

# Reduction in Average Distance Cost by Optimizing Position of ONUs in FiWi Access Network using Grey Wolf Optimization Algorithm



Nitin Chouhan, Uma Rathore Bhatt, and Raksha Upadhyay

**Abstract** Fiber-Wireless is the promising next generation broadband access network. FiWi integrates the technical merits of the optical access network and wireless access network. ONU placement is the most important issue in FiWi as it affects the network cost and network performance. The present research work considers the ONU placement issue and proposes a novel algorithm for finding an optimum position of ONUs. For this, a nature-inspired grey wolf optimization (GWO) algorithm is applied in the FiWi network. To the best of our knowledge, this algorithm has not been used for the ONU placement problem in the FiWi network. GWO provides the optimum position of every ONU, where the average distance cost (ADC) is minimum. ADC is the average of the distance of ONU and its associated wireless routers. To check the effectiveness of the proposed work, simulation is done for varying numbers of wireless routers. The proposed work is compared with well-known algorithm, namely teaching learning-based optimization (TLBO) algorithm. The result shows the reduction in ADC after applying the GWO algorithm than the initial placement and TLBO algorithm for all the cases considered for simulation. Hence, to deploy a cost-efficient FiWi network, proposed work may be one of the best solutions.

**Keywords** Fiber-wireless (FiWi) · ONU placement · Grey wolf optimization algorithm (GWO) · Average distance cost (ADC) · TLBO

---

N. Chouhan · U. R. Bhatt (✉) · R. Upadhyay  
Institute of Engineering and Technology, Devi Ahilya University, Indore, India  
e-mail: [uvrathore@gmail.com](mailto:uvrathore@gmail.com)

N. Chouhan  
e-mail: [nitin\\_chouhan27@yahoo.com](mailto:nitin_chouhan27@yahoo.com)

R. Upadhyay  
e-mail: [raksha\\_upadhyay@yahoo.co.in](mailto:raksha_upadhyay@yahoo.co.in)

# 1 Introduction

Recently, the fast development of Internet technology creates more challenges for researchers to design a network that provides a variety of services at a faster rate and lower cost to users as per requirement. Traditionally, optical access network and wireless access network are the two technologies which provide services to users. Optical access network provides longer distance communication, larger bandwidth, and better stability to the users. However, the huge cost is required to deploy fiber and optical devices. Alternately, a wireless access network provides services at a lower cost with better flexibility and easy deployment. However, it is not suitable for longer distance communication and its bandwidth is limited. Considering the advantageous features of both technologies, fiber-wireless (FiWi) [1–4] has been proposed which is the combination of the optical access network at the back-end and wireless access network at the front-end. It provides services to users at higher bandwidth, lower price, better stability, and better quality of services (QoS).

Figure 1 shows the architecture of the FiWi access network. It comprises the optical network at the back-end and wireless network at the front-end. At the back-end, multiple ONUs are connected to OLT via fibers and splitters. Front-end consists of wireless routers and end-users. Each ONU is connected with a wireless gateway through which communication is possible between both ends. FiWi supports upstream and downstream modes of communication. In upstream mode, the data is

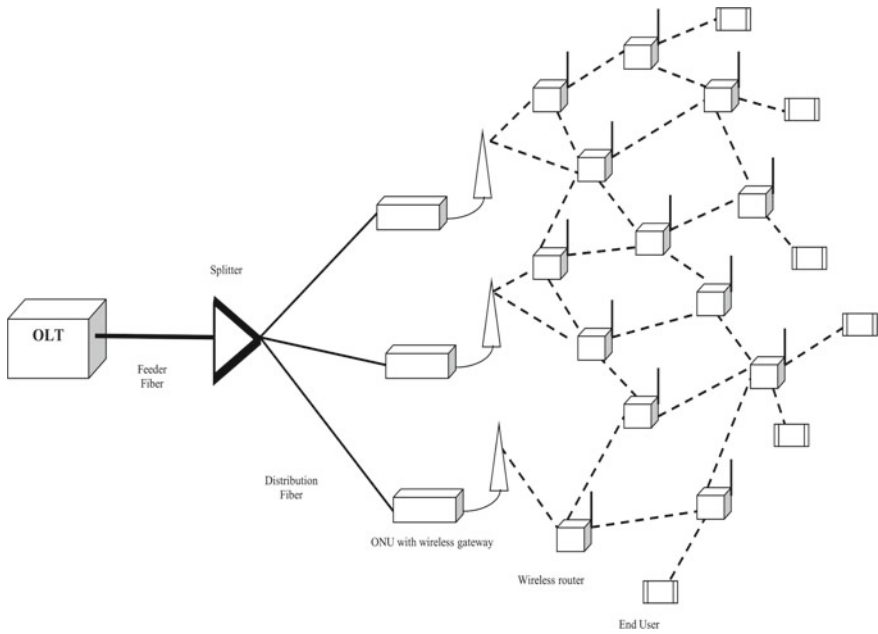


Fig. 1 FiWi architecture [5]

delivered from the end-user to OLT. Firstly, the user sends their data to nearby wireless router. The router sends this data to its primary ONU through multiple routers and wireless gateways. Lastly, ONU injects this data to OLT which acts as a service provider in the back-end. In downstream mode, the data is delivered from OLT to end-user, and the process is repeated in a reverse fashion.

Routing, energy consumption, survivability, and ONU placement are some of the research issues which can be tackled in the planning of the FiWi network [6–8]. Among all, ONU placement is one of the critical issues since it affects the cost and performance of the network. ONU placement means to optimally place minimum ONUs in the network such that overall cost is minimized, and all the users get connected in the network. On taking consideration of this issue, we propose an efficient nature-inspired grey wolf optimization (GWO) algorithm for finding an optimum position of ONUs in the FiWi network. GWO algorithm is based on the hunting behavior of grey wolves. It gives the optimum position of each ONU, where average distance cost (ADC) among ONU, and its associated routers is lowest. We compare the performance of the proposed algorithm with the initial placement of the FiWi network and the well-known TLBO algorithm. Result shows that the GWO algorithm always returns the lowest value of ADC for all cases compared to TLBO algorithm.

The rest of the paper is arranged as follows: Sect. 2 overviews the literature review on ONU placement in FiWi network. The proposed work explains in Sect. 3. Section 4 covers the simulation results and their analysis. Section 5 concludes the work.

## 2 Literature Review

FiWi access network has been broadly researched by the researchers. A wide range of algorithms have been proposed for different issues of FiWi network such as ONU placement, survivability, routing, energy-saving, and network performance. ONU placement is important in terms of cost and network performance. In [9], author proposed a greedy and simulated annealing theorem for the placement of ONUs in the network. Both the algorithm optimizes the position of ONUs such that the cost function is minimized in the network. Cost function is the distance between ONUs and end users. The author in [10] proposed a mixed integer linear programming (MILP)-based primal model for optimally placed ONUs under various constraint. For doing this, Lagrangean relaxation method is used. In [11], author optimizes the position of ONUs such that the overall network throughput is increased. Tabu search heuristic method is used for finding the optimum position of ONUs. Load balanced ONU placement (LBOP) algorithm is proposed in [12] which works in two stages. In the first stage, the minimum number of ONUs are placed such that all the users can communicate with hop constraint. In the second stage, load balancing takes place among ONUs to satisfy load balancing constraints.

In [13], the author implemented a hybrid algorithm to make a cost-efficient FiWi network. They proposed the genetic algorithm for optimizing the position of ONUs

followed by the reduction of ONUs. The hybrid algorithm is outperformed than the LBOP algorithm in terms of minimum ONUs in the network. The authors in [14–16] implemented different optimization algorithms, i.e., particle swarm optimization (PSO), teaching learning -based optimization (TLBO), and ant colony optimization (ACO) for optimizing the position of ONUs. The outcomes of these algorithms are the minimum ONUs required than the existing algorithms. In [17], author proposed an encircling mechanism of whale optimization algorithm (WOA) for optimizing the position of ONUs. The same author extends this work in [18] in which both mechanisms of WOA, i.e., encircling prey and spiral update mechanism is proposed. In both the papers, the ONUs positions are optimized in such a way that with minimum number of ONUs, FiWi network can be deployed. The author of [5] implemented various algorithms, i.e., Genetic, TLBO, and WOA for optimizing the position of ONUs. They analyze the effect of ONU placement on energy and survivability issues of the FiWi network. In [19], the author optimized the position of ONU using a whale optimization algorithm to minimize the distance between ONU and its associated wireless routers.

In the literature reported, so far none of the authors implemented the grey wolf algorithm for optimizing the position of ONUs. Therefore, in this paper, we implemented the grey wolf algorithm and analyzed its effect on optimizing the position of ONUs.

### 3 Proposed Work

The ONU placement is an eminent problem in the FiWi network. The deployment cost of the FiWi network should be minimized by optimally placing the ONUs. In the proposed work, we apply a nature-inspired grey wolf optimization (GWO) algorithm for finding the optimum position of ONUs. The optimum position is such that where the average distance cost of each ONU and its associated routers is minimized.

In this section, first, we mathematically formulate a system model, and then, we discuss the GWO algorithm for finding the optimum position of ONUs.

#### 3.1 System Model

The following notations are taken to design system model.

- $L \times L$ : Network Area.
- $G_s \times G_s$ : Grid size of the network.
- $N_{ONU}$ : Number of ONUs in the network.
- $ONU_i$ : ONU indexed as  $i$ .
- $(ONU_{X_i}, ONU_{Y_i})$ :  $i$ th ONU X/Y-coordinates.
- $NWR$ : Number of wireless routers in the network.

- WR<sub>*i*</sub>: Wireless router indexed as *i*.
- (WR<sub>*X<sub>i</sub>*</sub>, WR<sub>*Y<sub>i</sub>*</sub>): *i*th Wireless router *X/Y*-coordinates.
- $D_{WR_j}^{ONU_i}$ : Distance between *i*th ONU and *j*th wireless routers.
- $S_{WR}^{ONU}$ : Set of wireless routers for each ONU.
- WH<sub>ONU-WR</sub>: Wireless hop number between ONU and wireless routers.
- ADC<sub>ONU</sub>: Average distance cost of ONU.

The FiWi network is modeled in the  $L \times L$  network area. We assumed GPON and WLAN at the back-end and front-end, respectively. OLT and ONUs and ONU and WRs communicate with each other in TDMA manner. ONUs and wireless routers communicate if they are in the transmission range of each other. Each router has only one primary ONU. Therefore, each ONU form set of wireless routers which is given by

$$S_{WR}^{ONU} = WR_i | WH_{ONU-WR} \leq \text{predefined hops} \quad (1)$$

The distance cost between ONU and its associated routers is given as

$$DC_{ONU_i} = \sum_{j=1}^{OWR_i} \sqrt{((ONU_{X_i} - WR_{X_j})^2 + (ONU_{Y_i} - WR_{Y_j})^2)} \quad (2)$$

where  $OWR_i$  is the number of routers connected to *i*th ONU.

The average distance cost of ONU is given as

$$ADC_{ONU} = DC_{ONU_i} / OWR_i \quad (3)$$

The objective of the paper is to find the optimum position of ONU such that the average distance cost between ONU and its associated routers is minimized.

Optimize  $(ONU_{X_i}, ONU_{Y_i})$  where  $i = 1 : NONU$

Subject to

Minimize ADC<sub>ONU</sub>

### 3.2 Proposed Algorithm

In this subsection, we explain, in brief, the proposed algorithm, i.e., grey wolf optimization (GWO) [20] algorithm applied for optimizing the position of ONUs in the FiWi network.

GWO algorithm is inspired by the social hierarchy and hunting behavior of grey wolves belongs to the Canidae family. In this algorithm, the fittest solution is considered as the alpha ( $\alpha$ ). Consequently, the second and third best solutions are considered

as beta ( $\beta$ ) and delta ( $\delta$ ), respectively. The rest of the candidate solutions are assumed to be omega ( $\omega$ ).

Firstly, grey wolves encircling the prey by knowing its location. The following equations are used to encircle the prey.

$$D = C \times X_{\text{PREY}}(t) - X(t) \quad (4)$$

$$X(t + 1) = X_{\text{PREY}}(t) - A \times D \quad (5)$$

where  $X$  is the position of individual wolves,  $X_{\text{PREY}}$  is the position of prey,  $A$  and  $C$  are the coefficient vectors of the algorithm, and  $t$  is the current iteration. The  $A$  and  $C$  are calculated as follows:

$$A = 2 \times a \times r_1 - a$$

$$C = 2 \times r_2$$

$a$  is decreased from 2 to 0 over the iteration, and  $r_1$  and  $r_2$  are random vector between 0 to 1.

After encircling the prey, wolves are started attacking the prey. In the search space, we have no idea of the location of the prey. But we suppose that the search agents (best solutions), i.e., alpha, beta, and delta have best knowledge about the location of prey. Therefore, omegas are changing their position as per guided by alpha, beta, and delta using the following equations:

$$D_\alpha = C_1 \times X_\alpha - X_{\text{Oldposition}} \quad (6)$$

$$X_{NP1} = X_\alpha - A_1 \times D_\alpha \quad (7)$$

$$D_\beta = C_2 \times X_\beta - X_{\text{Oldposition}} \quad (8)$$

$$X_{NP2} = X_\beta - A_2 \times D_\beta \quad (9)$$

$$D_\delta = C_3 \times X_\delta - X_{\text{Oldposition}} \quad (10)$$

$$X_{NP3} = X_\delta - A_3 \times D_\delta \quad (11)$$

$$X_{\text{Newfinalposition}} = (X_{NP1} + X_{NP2} + X_{NP3}) \div 3 \quad (12)$$

where  $X_{NP1}$ ,  $X_{NP2}$ , and  $X_{NP3}$  are the position of omega according to alpha, beta, and delta, respectively.  $X_{Newfinalposition}$  is the final position of wolves. Finally, grey wolves finish the hunt by attacking the prey when it stops moving.

### 3.3 Implementation of GWO for Optimizing the Position of ONUs

For finding the optimum position of ONUs, we apply the GWO algorithm in the FiWi network. The steps are as follows:

1. Firstly, wireless routers are randomly placed in the FiWi network area.
2. ONUs are placed in the network such that with minimum ONUs all the routers are connected.
3. We form the set of wireless routers for each ONU according to predefined hops according to Eq. 1.
4. We find the premium routers for each ONU.
5. Now, we find the average distance cost for each ONU according to Eqs. 2 and 3.
6. Alpha is that ONU that has the least distance cost among all the ONUs.
7. Similarly, beta and delta ONUs are found for the GWO algorithm.
8. Rest ONUs are the omegas of the network.
9. While  $t < \text{Maximum iteration}$

for  $i = 1$ : NONU.

Update the position of the ONUs according to Eqs. 6–12.

end for.

Update  $a$ ,  $A$  and  $C$ .

Repeat steps 3–8.

$t = t + 1$ .

end while.

Find the average distance cost of all the ONUs according to Eqs. 2 and 3.

At the end, GWO gives the optimum position, where average distance cost is minimized for every ONU.

## 4 Simulation Settings and Result Analysis

We performed simulation experiments in the MATLAB environment. In the simulation scenario, the FiWi network is modeled in a  $1000 \times 1000$  m square area.  $N_{WR}$  routers are randomly placed, where  $N_{WR}$  takes value  $\{30, 40, 50, 60, \text{ and } 70\}$  for different cases. Six ONUs are placed such that it communicates with all the routers present in the network.



We illustrate the proposed work with an example. In the example scenario, 50 wireless routers are randomly placed in the network. Six ONUs are placed in the network as shown in Fig. 2.

Now, we form the set of each ONUs according to predefined hops and find the premium routers for each ONU. Then, we find the average distance cost of each ONU as shown in Table 1.

Now, we apply the existing well-known TLBO algorithm for optimizing the position of ONUs. Initial position and final position (Optimum position) of ONUs achieve by TLBO are shown in Fig. 3, and their respective average distance cost is shown in Table 2.

In order to apply the proposed GWO algorithm, first, we find the alpha, beta, and delta in the network. According to Table 1, ONU<sub>6</sub> has minimum distance cost so it becomes the alpha in the GWO algorithm. Similarly, ONU<sub>3</sub> and ONU<sub>2</sub>

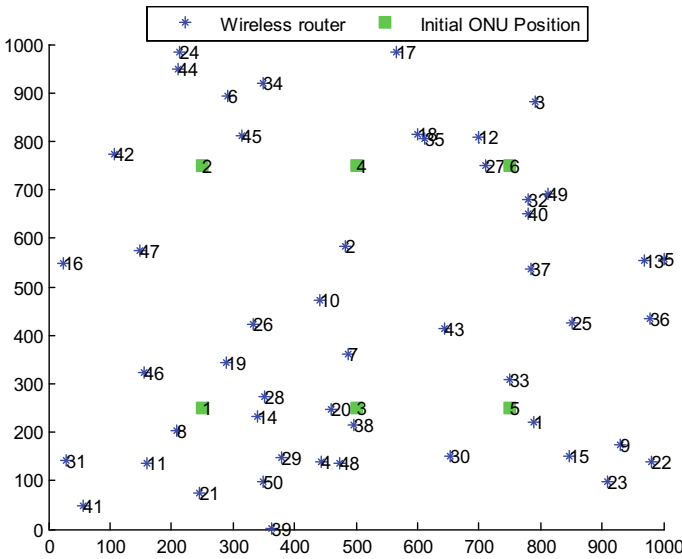


Fig. 2 Initial placement of ONUs in FiWi network

Table 1 Average distance cost of ONUs in initial placement

S. no.	ONU <sub>i</sub>	Average distance cost (in meter)
1	ONU <sub>1</sub>	214.1049
2	ONU <sub>2</sub>	201.2751
3	ONU <sub>3</sub>	191.8462
4	ONU <sub>4</sub>	232.5517
5	ONU <sub>5</sub>	210.6707
6	ONU <sub>6</sub>	174.6699

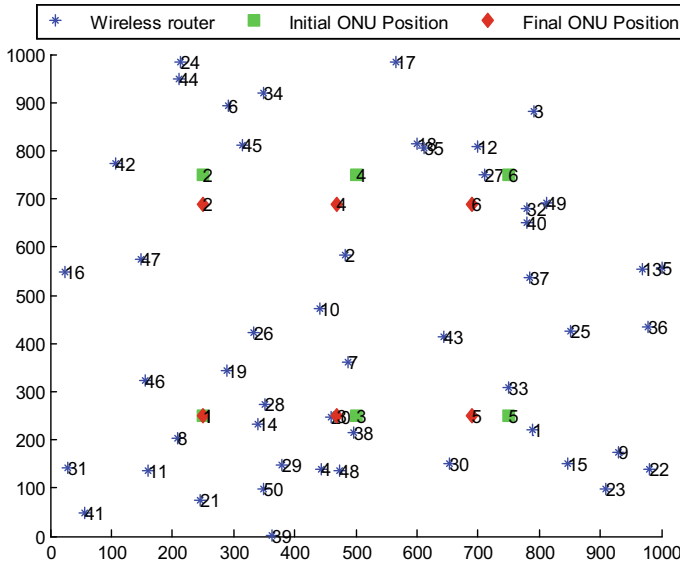


Fig. 3 Optimum position of ONUs using TLBO algorithm in the FiWi network

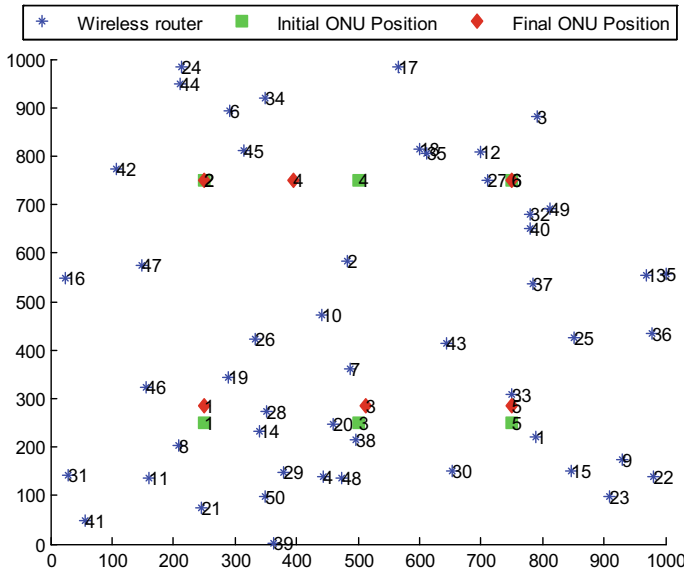
Table 2 Average distance cost of ONUs after applying TLBO algorithm

S. no.	ONU <sub><i>i</i></sub>	Average distance cost (in meter)
1	ONU <sub>1</sub>	198.7568
2	ONU <sub>2</sub>	193.3352
3	ONU <sub>3</sub>	186.3843
4	ONU <sub>4</sub>	229.5292
5	ONU <sub>5</sub>	207.8597
6	ONU <sub>6</sub>	164.8416

become beta and delta in the algorithm. Then, we displace the position of ONUs according to the GWO algorithm. The optimum positions (final position) of each ONUs corresponding to initial placement are shown in Fig. 4.

After finding an optimum position, again we form the set of routers and find premium routers for each ONU. The average distance cost of each ONU after applying the GWO algorithm is shown in Table 3. The cost saving in GWO algorithms for ONU<sub>S1-6</sub> as compared to initial placement and TLBO algorithm are also shown in the table. The overall cost improvement (average value) for this scenario is 5.6 and 1.99% compared to initial placed and TLBO algorithm, respectively.

Simulation results for a varying number of wireless routers, namely for 30, 40, 60, and 70 are shown in Figs. 5, 6, 7 and 8, respectively. From all the figures, we can see that the average distance cost in initial placement of ONU is always greater for all the ONUs. For further reduction of ADC in the network, we optimize the



**Fig. 4.** Optimum position of ONUs using GWO algorithm in the FiWi network

**Table 3** Average distance cost of ONUs after applying GWO algorithm

S. no.	ONU <sub><i>i</i></sub>	Average distance cost (in meter)	Cost saving as compared to initial placement in %	Cost saving as compared to initial placement in %
1	ONU <sub>1</sub>	190.7883	10.89	4.0
2	ONU <sub>2</sub>	188.6281	6.283	2.43
3	ONU <sub>3</sub>	186.4593	1.24	0
4	ONU <sub>4</sub>	227.9779	1.96	0.67
5	ONU <sub>5</sub>	203.5539	3.37	2.07
6	ONU <sub>6</sub>	159.0938	8.19	3.1

position of ONUs using the proposed algorithm. The result of proposed algorithm is compared with the TLBO algorithm. It is observed from all the figures that the proposed GWO algorithm outperforms than TLBO algorithm in terms of reduced ADC in the network. The reason is that the GWO algorithm efficiently optimizes the position of ONUs as compared to TLBO network. With the decreasing distance cost, the transmission energy may also decrease since it depends on distance among ONUs and associated wireless routers. Hence, the proposed work may also provide better energy efficiency than TLBO algorithm.

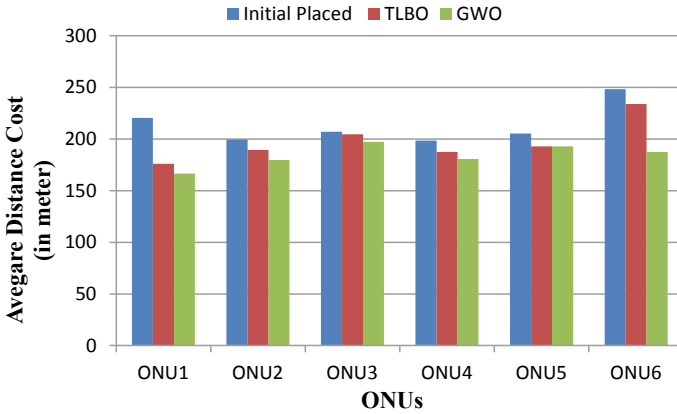


Fig. 5. Value of average distance cost for 30 wireless routers

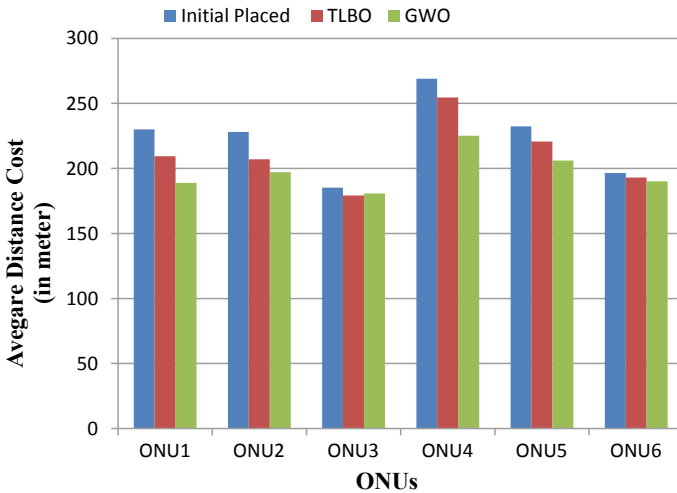


Fig. 6. Value of average distance cost for 40 wireless routers

## 5 Conclusion

This paper considered the optimum placement of ONUs in the FiWi network. Regarding this, an efficient grey wolf optimization algorithm is applied on the FiWi network. GWO optimizes the position of ONUs such that the average distance cost among ONUs, and its associated wireless routers is minimized in the network. An extensive simulation is performed to analyze the performance of the proposed algorithm. The result shows the reduction in average distance cost in the GWO algorithm than the initial placement and TLBO algorithm. The overall cost improvement of

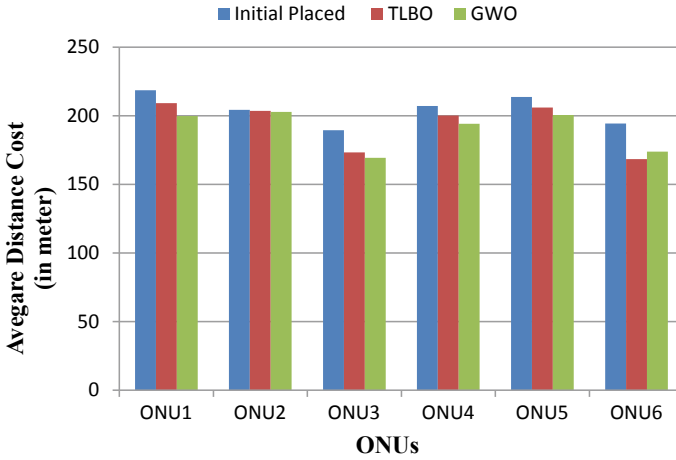


Fig. 7. Value of average distance cost for 60 wireless routers

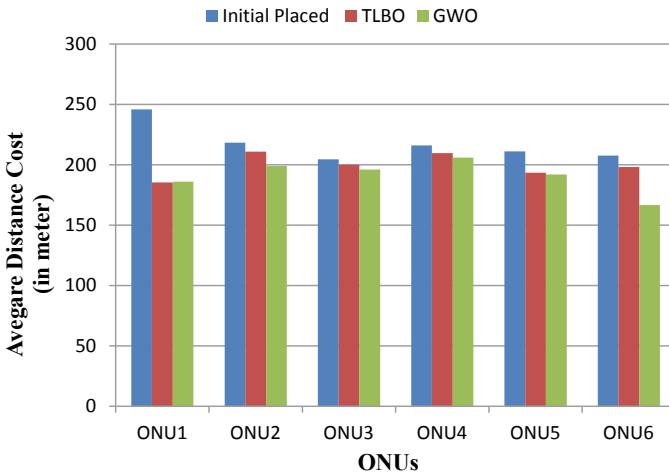


Fig. 8. Value of average distance cost for 70 wireless routers

GWO for 30–70 wireless routers are 6.7, 6, 1.99, 1.75, and 4.32%, respectively, compared to TLBO algorithm. In future, we can explore the influence of this algorithm on other network parameters. Besides, we can compare the performance of this algorithm with other optimization algorithms also.

**Acknowledgements** We would like to acknowledge IET, DAVV, Research Center, Indore, India. This paper can be used as a part of Ph.D. thesis in future for the first author.

## References

1. Y. Liu, L. Guo, B. Gong et al., Green survivability in fiber-wireless (FiWi) broadband access network. *Opt. Fiber Technol.* **18**, 68–80 (2012). <https://doi.org/10.1016/j.yofte.2011.12.002>
2. R. Shaddad, A. Mohammad, M. Elmagzoub et al., A survey on access technologies for broadband optical and wireless networks. *J. Netw. Comput. Appl.* **41**, 459–472 (2014). <https://doi.org/10.1016/j.jnca.2014.01.004>
3. N. Ghazisaidi, M. Maier, C. Assi, Fiber-wireless (FiWi) access networks: a survey. *IEEE Commun. Mag.* **47**, 160–167 (2009). <https://doi.org/10.1109/MCOM.2009.4785396>
4. L. Guan, H. Yang, M. Cheriet, Throughput-oriented power allocation scheme based on convex optimization for cache-enabled FiWi access network in 5G IoT scenario. in *Proceeding of International Wireless Communications and Mobile Computing (IWCMC), IEEE Conference*, (Limassol, Cyprus, 2020), pp. 1043–1046. <https://doi.org/10.1109/IWCMC48107.2020.9148344>
5. U.R. Bhatt, N. Chouhan, R. Upadhyay, An optimization framework for FiWi access network: comprehensive solution for green and survivable deployment. *Opt. Fiber Technol.* **53**, 1–16 (2019) <https://doi.org/10.1016/j.yofte.2019.102002>
6. N. Ghazisaidi, M. Maier, Fiber-wireless (FiWi) access networks: challenges and opportunities. *IEEE Netw.* **25**, 36–42 (2011). <https://doi.org/10.1109/MNET.2011.5687951>
7. U.R. Bhatt, N. Chouhan, ONU placement in fiber wireless (FiWi) networks. in *Proceeding Of Nirma University International Conference on Engineering (NUICONE), IEEE Conference*, (Ahmedabad, India, 2013). <https://doi.org/10.1109/NUICONE.2013.6780115>
8. U.R. Bhatt, N. Chouhan, R. Upadhyay, Cost efficient algorithm for ONU placement in fiber-wireless (FiWi) access networks. *Procedia Comput. Sci.* **46**, 1303–1310 (2015). <https://doi.org/10.1016/j.procs.2015.01.055>
9. S. Sarkar, H. Yen, S. Dixit et al., Hybrid wireless-optical broadband access network (WOBAN): network planning and setup. *IEEE J. Sel. Areas Commun.* **26**, 12–21 (2008). <https://doi.org/10.1109/JSACOCN.2008.032207>
10. S. Sarkar, H. Yen, S. Dixit, B. Mukherjee, A mixed integer programming model for optimum placement of base stations and optical network units in a hybrid wireless-optical broadband access network (WOBAN). in *Proceedings of Wireless Communications Networks Conference WCNC, IEEE Conference*, (2007), pp. 3907–3911. <https://doi.org/10.1109/WCNC.2007.714>
11. Z. Zheng, J. Wang, X. Wang, ONU placement in fiber-wireless (FiWi) networks considering peer-to-peer communications. in *Proceedings of GLOBECOM, IEEE Conference* (2009), pp. 1–7. <https://doi.org/10.1109/GLOCOM.2009.5425913>
12. L. Yejun, Q. Song, B. Li et al., Load balanced optical network unit (ONU) placement in cost-efficient fiber-wireless (FiWi) access network. *Optik* **124**, 4594–4601 (2013). <https://doi.org/10.1016/j.ijleo.2013.01.063>
13. U.R. Bhatt, N. Chouhan, R. Upadhyay, Hybrid algorithm: a cost efficient solution for ONU placement in fiber-wireless (FiWi) network. *Opt. Fiber Technol.* **22**, 76–83 (2015). <https://doi.org/10.1016/j.yofte.2015.01.010>
14. U.R. Bhatt, R. Upadhyay, D. Kothari, N. Chouhan, Cost efficient low convergence ONU placement algorithm for deployment of FiWi network. *Trends Opto-Electro Opt. Commun.* **6**, 1–17 (2016). <https://doi.org/10.37591/toeoc.v6i1.1757>
15. U.R. Bhatt, N. Chouhan, R. Upadhyay, C. Agrawal, ONU Placement in FiWi access network using teacher phase of TLBO algorithm. in *Proceeding of 3rd International Conference on Computational Intelligence and Communication Technology (CICT)*, (Ghaziabad, India, 2017). <https://doi.org/10.1109/CIACT.2017.7977312>
16. U.R. Bhatt, N. Chouhan, R. Upadhyay et al., Efficient placement of ONUs via ant colony optimization algorithm in FiWi access networks. in *Proceeding of Progress in Advanced Computing and Intelligent Engineering*, Advances in Intelligent Systems and Computing AISC series of Springer (2016). [https://doi.org/10.1007/978-981-10-6872-0\\_49](https://doi.org/10.1007/978-981-10-6872-0_49)

17. U.R. Bhatt, N. Chouhan, R. Upadhyay, Potential sites searching for ONUs in FiWi network. in *Proceedings of Recent Advances in Interdisciplinary Trends in Engineering and Applications* (RAITEA), (SSRN Elsevier, Indore, India< 2019). <https://doi.org/10.2139/ssrn.3364212>
18. U.R. Bhatt, N. Chouhan, R. Upadhyay, Performance evaluation of fiber wireless (FiWi) access network using position optimization of ONUs. Accepted article in *Int. J. Sens. Wireless Commun. Control Bentham Sci.* (2020). <https://doi.org/10.2174/2210327910666200304131411>
19. U.R. Bhatt, N. Chouhan, R. Upadhyay et al., Fiber wireless (FiWi) access network: ONU placement and reduction in average communication distance using whale optimization algorithm. (*Heliyon Elsevier*, 2019) <https://doi.org/10.1016/j.heliyon.2019.e01311>
20. S. Mirjalili, S.M. Mirjalili, A. Lewis, Grey wolf optimizer. *Adv. Eng. Softw.* **69**, 46–61 (2014). <https://doi.org/10.1016/j.advengsoft.2013.12.007>

# Performance Analysis of Individual Partial Relay Selection Protocol Using Decode and Forward Method for Underlay EH—CRN



G. Kalaimagal and M. S. Vasanthi

**Abstract** The paper investigates the performance of the underlay cognitive radio network. We propose a relay selection protocol to enhance the throughput and to obtain reduced outage probability in the ad hoc network. The proposed work is based on relay selection aided with energy harvesting to serve communication between the secondary nodes. We have formulated a closed-form expression for the proposed work and differentiated the outage probability and throughput with other partial relay selection techniques. Further, decode and forward (DF) relaying with Rayleigh fading channel is considered in this work to improve the end-to-end channel gain. The performance evaluation indicates that the proposed cooperative relay selection scheme has marginally enhanced by increasing the number of relay node.

**Keywords** Cooperative relaying · Partial relay selection · Outage probability · Underlay CRN

## 1 Introduction

Cognitive radio technology is a smart technology that enables radio frequency (RF) communication to function effectively at a restricted bandwidth. The cognitive radio network is categorized as underlay, overlay, and interweave according to the information of the primary users. Cognitive radio operations depend on the primary user's location and spectrum. The benefit of the underlay cognitive radio network (CRN) over other groups is that the secondary user (unlicensed user) does not impact primary user service and interferes with the primary user at some stages.

---

G. Kalaimagal (✉) · M. S. Vasanthi  
SRM Institute of Science and Technology, Kattankulathur, India  
e-mail: [kalaimag@srmist.edu.in](mailto:kalaimag@srmist.edu.in)

M. S. Vasanthi  
e-mail: [tcevasanthi@gmail.com](mailto:tcevasanthi@gmail.com)



The fundamental concept of this ad hoc wireless network is the usage and reliability of data transmission. However, data transmission and reliability are significantly impacted by shadows, degradation channels, and route loss in the cellular network [1]. Cooperative communication will overcome these challenges in the CRN. This communication is commonly used in mobile ad hoc network (MANET), vehicular ad hoc network (VANET), and other wireless networks without any intervention of servers and routers [2]. The relaying methods can be operated either in distributed or centralized, focused on the network's application [3]. The key concept of the relay system can be defined as follows.

In cooperative communication, the information will be exchanged by two most relay systems, such as a decode and forward (DF) and amplify and forward (AF), between the source and intended nodes [4]. The former system has more advantages compared to the latter, but the drawback of decode and forward technique is complex, unlike amplify and forward. Therefore, two separate relays (i.e., hybrid relay-AF & DF) in the underlay CRN can be implemented to increase the gain in diversity [5]. To maintain communication between the source and destination nodes without any intervention, the nodes can be replaced by wireless battery storage.

A potential approach or a process for energy storage and processing from the external source is known as energy harvesting (EH) [2]. The harvesting system can either be designed online or offline. In the offline system, the amount of energy obtained from the source, the channel conditions and the amount of input data for transmission are known. The system should have causal details on the channel conditions and collected energy and the volume of data to be sent to the online system. Various types of research have been performed using online joint power management algorithms to balance the energy levels of the battery and the fading condition [6].

The proposed research focuses on the relay of the energy harvested cognitive system. Our analysis is based on the CRN network of secondary users (source and destination nodes), primary users, and base stations. The secondary node transmits the information to primary users or other secondary nodes. Some of the secondary nodes have been considered as relays that also play the role of collecting energy [7]. The secondary user serves as an EH source and as a relay during data transmission. In the proposed work, the selection of optimal relay is based on the EH channel link [8], i.e., which relay or secondary node can harvest sufficient amount of energy from the beacon signal radiated from base station [9]. The output performance and expressions are developed using the cumulative density and probability function in this proposed model. The mathematical expression of three separate relay systems in the presence of slowly fading Rayleigh channel is theoretically simulated.

The important contribution of our work is summarized as follows:

- We propose three different relay selection protocols for DF cooperative relaying; in the proposed hybrid relay partial scheme (proposed HRPS) scheme, the relay is chosen based on maximum energy harvesting from the base station as compared to the existing work, and selection of optimal relay is based on mode of communication.

- On the other hand, best opportunistic relay selection (Best-ORS) and conventional opportunistic relay selection scheme (C-ORS) select the best relay based on end-to-end communication.
- We determine the relay performance of the system proposed with Best-ORS and C-ORS by threshold, an increasing number of relays, interference links and various locations of relay nodes.
- Closed-form expression is used for the verification of the simulation results.

The paper is summarized as follows. The relevant work is outlined in Sect. 1.2, and further in Sect. 1.3 which explains the system model of the proposed work. The problem formulation for the proposed work for targeting instantaneous SNR and EH relaying has been analyzed in Sect. 2. The strategy for a potential choice of relays is discussed in Sect. 2.3.

The outputs of the proposed work are elaborated in Sect. 5. The conclusion of this paper is discussed in Sect. 6.

## ***1.1 Related Work***

Some of the related research is carried out here by the scholars based on partial relay selection, and EH is detailed; in this reference paper [10], both the secondary node and relay nodes initially gather energy from the power beacon and act as data transmission systems. Moreover, the authors have highlighted that these nodes can be fitted with multiple antennas and also illustrated the effectiveness of the secondary node and relay node. Tourki et al. [11] presented their work on the opportunistic relaying scheme and analyzed outage probability by deriving closed form of expression using probability density function. Their output result correlates with the analytical data over different network architectures.

Moreover, Xu et al. [12] have submitted their research based on underlay CRN with multi-hop relays in which the power beacon is used as the operator for secondary user's data transmission and power generation. Further, the author has achieved the outage performance using joint optimization technique, i.e., optimizing both transmit power and energy harvesting. The power allocation strategy is studied in [13], and the researchers have achieved better performance by increasing the number of relays. The researchers [14, 15] analyzed their work on partial relaying using DF protocol and proved their better outage performance at low SNR over Rayleigh fading channel. There were significantly more focused works in underlay CR network aided EH with a signal to interference plus noise ratio constraints and given suggestions for the channel quality [16].

### 1.2 System Model

In the proposed work, the underlay CRN network identified in Fig. 1 operated using a dual-hop DF relaying scheme. The system consists of a source ( $S$ ) that communicates to the destination ( $D$ ) via a relay ( $R$ ). The network includes ‘ $r$ ’ number of relays, and one of them has been chosen to support data communication, while other nodes perform EH. The signal received between the two nodes can be represented in the following Eq. (1):

$$y_{x,y} = \sqrt{P}h_{x,y}x + n_{x,y} \tag{1}$$

The maximum channel gain is selected based on the relaying scheme. Assume that the source and relay power for the communication depends on the energy harvester [17] which is deployed in different locations near the base station ( $B$ ) and primary user ‘ $P$ ’. The primary user considered in the proposed work as  $P_1, P_2, P_3 \dots P_n$ . The term  $n_{x,y}$  represented as additive white Gaussian noise (AWGN) with zero mean and with variance. The source node harvests energy from the beacon signal which has  $K$ -antenna installed on the base station considered during the first phase, while in the second phase, relay harvests energy from the beacon signal with the same number of the antenna. The EH channel link was taken when the secondary node acts a relay.

The expression  $h_{S,R_r}$  and  $h_{R_r,D}$  represents as channel gains between  $S$  to  $R_r$  and  $R_r$  to  $D$  respect, where  $r = 1, 2, 3 \dots R$ . The interference and data links for the relay and the primary user are chosen as  $I_I$  and  $I_D$ . In this network model, the channel is assumed as slow fading Rayleigh distribution channel. The source and relays are considered to have a single antenna that can harvest energy from a  $K$ -antenna

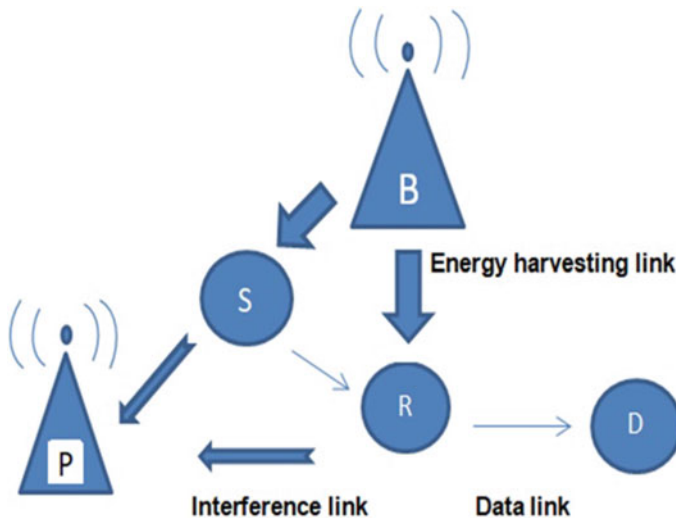


Fig. 1 Proposed model of underlay CR with relay selection scheme

mounted in the base station ( $B$ ). To analyze the shadowing effect, the transmission of data is performed via best relay instead of using direct communication between  $S$  and  $D$ . Let the period for each data transmission taken as  $T_{S,R_r} = T_{R_r,D} = T$ .

## 2 Problem Formulation

### 2.1 Instantaneous SNR

Let us denote  $h_{a,b}$  as channel gain between two ends  $a$  &  $b$ , where  $(a, b) \in \{S, D, R_r, B_K, P_n\}$  with  $r = 1, 2, \dots, R$ ,  $K = 1 \& 2$  and  $n = 1, 2, 3, \dots, N$ .

Consider the general Eq. (2) for the instantaneous SNR ( $y_{a,b}$ ) for end-to-end communication with power written as

$$y_{a,b} = \frac{P_x h_{a,b}}{P_x h_{a,b} + \sigma_0^2} \quad (2)$$

The formulation of instantaneous signal-to-noise ratios for  $S \rightarrow R_r$ ,  $R_r \rightarrow D$  in underlay CRN as follow as,

During the first hop, the instantaneous signal-to-noise ratio between source and relay can be identified as,

$$y_{1r} = \tilde{y}_{S,R_r} = \frac{P_S h_{B_K,S}}{P_S h_{B_K,S} I_D + \sigma_0^2} \quad (3)$$

On the other hand, at the second hop, the instantaneous signal-to-noise ratio is given in Eq. (4)

$$y_{2r} = \tilde{y}_{R_r,D} = \frac{P_R h_{B_K,R_r}}{P_R h_{B_K,R_r} I_D + \sigma_0^2} \quad (4)$$

The transmit power of relay and source as  $P_R$  and  $P_S$  are taken as equal power harvested from the base station. The term  $h_{B_K,S}$  and  $h_{B_K,R_r}$  represents the channel gains between the source and the base station and relay to base station. Also,  $\sigma_0^2$  represents AWGN variance whose value for each channel can be considered to be one.

## 2.2 EH Relaying

In the proposed model, the energy required for the source ( $E_s$ ) and the relay.

( $E_{R_r}$ ) uses all the antennas installed at the base station. The energy harvested at the  $R_r$  and  $S$  is given by the expression as follows,

$$E_{R_r} = \frac{\alpha T \eta P_B \sum_{k=1}^K h_{B_k, R_r}}{(1 - \alpha)T/2} \quad (5)$$

$$E_S = \frac{\alpha T \eta P_B \sum_{k=1}^K h_{B_k, S}}{(1 - \alpha)T/2} \quad (6)$$

where  $\eta$  is the energy efficiency at source and relay,  $\frac{2\eta\alpha}{1-\alpha} = \mu$ , and  $\sum_{k=1}^K h_{B_k, S} = \Omega_S$  are denoted as EH links. The time interval for RF-EH process by the source is  $\alpha T$ , while the remaining time interval for transmission and reception of relay is  $(1 - \alpha)T/2$  for ( $0 < \alpha \leq 1$ ).

The source and relay nodes must modify their transmitting power in underlay CR to fix interference. Therefore, the equation can be formulated by,

$$P_S = \min(E_S, I_S) = P_{B_k} \min(\mu\Omega_S, \frac{\lambda}{\tilde{E}_S}) \quad (7)$$

$$P_{R_r} = \min(E_{R_r}, I_{R_r}) = P_{B_k} \min(\mu\Omega_S, \frac{\lambda}{\tilde{E}_{R_r}}) \quad (8)$$

where  $\tilde{E}_S = \max_{k=1,2,\dots,K} (h_{B_k, S})$  and  $\tilde{E}_{R_r} = \max_{k=1,2,\dots,K} (h_{B_k, R_r})$ . In addition,  $I_S = \left( \frac{I_{\text{threshold}}}{I_I \tilde{E}_S} \right)$  and  $I_{R_r} = \left( \frac{I_{\text{threshold}}}{I_I \tilde{E}_{R_r}} \right)$  are considered as the minimum thresholds [18] for the primary user. Furthermore, the constraint threshold for interference with respec-

tive of power beacon to prevent the fading of channel can be represented as  $\lambda = \frac{\kappa}{I_I}$ ,

where  $\kappa = \frac{I_{\text{threshold}}}{P_{B_k}}$ .

## 2.3 Relay Selection Techniques

### 2.3.1 Proposed Hybrid Partial Relay Selection (Proposed HPRS)

In the cooperative strategy with underlay CR network, the best relay can be chosen by using various techniques of relay selection, i.e., based on high channel gain to forward the data to the destination. Similar to the conventional method, in the proposed hybrid partial relay selection, the best relay selection is done based on the maximum channel gain between  $B_k, S$  and  $R_r$  can be represented as  $(h_{B_k, S}, h_{B_k, R_r})$ .

The maximum and minimum method for DF relaying scheme can be expressed as

$$R_x : \min(\tilde{y}_{S, R_x}, \tilde{y}_{R_x, D}) = \max_{r=1, 2, \dots, R} \min(y_{1r}, y_{2r}) \quad (9)$$

where  $R_x$  is the chosen relay which is near to the base station with  $x \in \{1, 2, 3, \dots, R\}$ .

## 3 Performance Analysis

### 3.1 Outage Probability and Throughput

The performance of channel quality can be measured by using the outage probability and throughput. The end-to-end channel capacity for DF relaying [19] can be calculated using the instantaneous SNR equation from (3) and (4) given as,

$$\Gamma_m = \frac{(1 - \alpha)}{2} \log_2(1 + \min(y_{1r}, y_{2r})) \quad (10)$$

The probability of an  $\epsilon$  outage is defined as the probability of the end-to-end capacity ( $\Gamma_m$ ) is lesser than the positive threshold ( $\Gamma_{th}$ ) expressed as,

$$O = \Pr(\Gamma_m < \Gamma_{th}) \quad (11)$$

Using the Eq. (11), then the throughput for an end-to-end communication can be formulated using  $\epsilon$  outage probability as,

$$T_p = (1 - \alpha)T\Gamma_{th}(1 - O) \quad (12)$$

where  $(1 - \alpha)T$ , the total time for the data transmission between  $S$ ,  $R$  and  $D$ .

From the relay selection technique, the expression for the outage probability and throughput for the proposed HPRS, Best-ORS and C-ORS [19] relaying scheme can be framed as follows,

Consider in the first hop, the chosen relay with the high channel gain for data transmission to the other node defined as

$$R_{x_1} : \tilde{y}_{S, R_{x_1}} = \max_{r=1,2,\dots,R} (y_{1r}) \quad (13)$$

where  $R_{x_1}$  is the selected relay with  $x_1 \in \{1, 2, \dots, R\}$  and  $R$  is the number of the relay in the network. The relay with the highest channel gain during the second hop is chosen as the best equation represented by,

$$R_{x_2} : \tilde{y}_{S, R_{x_2}} = \max_{r=1,2,\dots,R} (y_{2r}) \quad (14)$$

where  $R_{x_2}$  is the selected relay with  $x_2 \in \{1, 2, \dots, R\}$ . Combining the Eqs. (10) and (11) and also Eqs. (13) and (14), the end-to-end outage probability of the partial relay selection using instantaneous SNR [20] between source to relay ( $O_{\text{PRS}_1}$ ) and relay to destination ( $O_{\text{PRS}_2}$ ) calculated as follows,

$$O_{\text{PRS}_1} = \Pr(\Gamma_{x_1} < \Gamma_{\text{th}}) = \Pr\left(\frac{(1-\alpha)T}{2} \log_2(1 + \min(y_{1x_1}, y_{2x_1})) < \Gamma_{\text{th}}\right) \quad (15)$$

$$O_{\text{PRS}_2} = \Pr(\Gamma_{x_2} < \Gamma_{\text{th}}) = \Pr\left(\frac{(1-\alpha)T}{2} \log_2(1 + \min(y_{1x_2}, y_{2x_2})) < \Gamma_{\text{th}}\right) \quad (16)$$

Using the Eqs. (15) and (16), the outage probability used for the proposed scheme can be expressed as,

$$O_{\text{proposedHPRS}} = \min(O_{\text{PRS}_1}, O_{\text{PRS}_2}) \quad (17)$$

If  $O_{\text{PRS}_1} \leq O_{\text{PRS}_2}$ , the transmission takes place using the relay  $R_{x_1}$  and on the other hand if  $O_{\text{PRS}_2} > O_{\text{PRS}_1}$ , then the relay  $R_{x_2}$  will be selected for the transmission in our proposed scheme. Similarly, the throughput of the proposed hybrid partial relay selection scheme can be calculated using outage probability formulated as,

$$T_{\text{proposedHPRS}} = (1-\alpha)T\Gamma_{th}(1 - O_{\text{proposedHPRS}}) \quad (18)$$

The end-to-end outage probability of different relay selection techniques  $Z$ , where  $Z \in \{\text{Proposed HPRS, Best-ORS, C-ORS}\}$ , can be written as,

$$O_Z = \Pr(\min(\Theta_1, \Theta_2)) < \psi \quad (19)$$

For different protocols, the first hop and second hop can be regarded as  $\Theta_1$  and  $\Theta_2$  with known CSI. Equation (19) can be expressed in terms of the first and second stages of transmission [21],

$$O_Z = 1 - \Pr(\Theta_1 \geq \psi, \Theta_2 \geq \psi) \quad (20)$$

Further, the data relaying based on the threshold (Cth) [22] is given by the equation

$$\psi = 2^{\frac{2C_{th}}{(1-\alpha)T}} - 1 \quad (21)$$

Also, for other relaying protocol such as Best-ORS [23] and C-ORS [11], their outage probability and throughput expression during the first hop and the second hop can be defined in terms of the maximum end-to-end SNR. Similar to the proposed scheme, Eqs. (22) and (23) represent outage probability, and throughput for the other two relaying protocol can be stated as,

$$O_{BC} = O_{\text{BestORS}} = O_{\text{C-ORS}} = \Pr\left(\frac{(1-\alpha)T}{2} \log_2(1 + \min(y_{1r}, y_{2r})) < \Gamma_{th}\right) \quad (22)$$

$$T_{p(\text{BestORS})} = T_{p(\text{C-ORS})} = (1-\alpha)T\Gamma_{th}(1 - O_{BC}) \quad (23)$$

Moreover, the outage performance for different DF relaying protocols [24] in terms of power transmission and interference from the Eqs. (7) and (8) with respect to threshold can be compiled as,

$$O_Z = 1 - \Pr((1 - I_D\psi) \min(\mu\Omega_S, \frac{\lambda}{E_S})h_{S,R_r} \geq \psi), (1 - I_D\psi) \min(\mu\Omega_R, \frac{\lambda}{E_{R_r}})h_{R_r,D} \geq \psi) \quad (24)$$

**Proof of Lemma**

$$\begin{aligned} O_{PRS_1} = 1 - & \left[ \sum_{x=0}^{K-1} \sum_{r=0}^{R-1} (-1)^r \frac{2C_{R-1}^r}{x!} (r+1)^{\frac{(x-1)}{2}} \left(\frac{h_{B,S}h_{S,R}\lambda}{\mu}\right)^{\frac{(x+1)}{2}} K_{1-x}\left(\frac{(r+1)h_{B,S}h_{S,R}\lambda}{\mu}\right) \right. \\ & \left. - \sum_{x=0}^{K-1} \sum_{n=1}^N \sum_{r=0}^{R-1} (-1)^{n+r+1} C_{R-1}^r C_N^n \frac{2Rh_{S,R}}{x!} \left(\frac{\lambda}{nh_{S,Pk}+(r+1)h_{S,R}\lambda}\right)^{\frac{(1-x)}{2}} \left(\frac{h_{B,S}\lambda}{\mu}\right)^{\frac{(x+1)}{2}} \right] \\ & \times K_{1-x} \left( 2\sqrt{\frac{h_{B,S}(nh_{S,Pk} + (r+1)h_{S,R}\lambda)}{\mu}} \right) \times \sum_{x=0}^{K-1} \frac{2}{x!} \left(\frac{h_{B,R}h_{R,D}\lambda}{\mu}\right)^{\frac{(x+1)}{2}} \\ & K_{1-x} \left(\frac{2\sqrt{h_{B,R}h_{R,D}\lambda}}{\mu}\right) - \sum_{x=0}^{K-1} \sum_{n=1}^N (-1)^{n+1} C_N^n \frac{2h_{R,D}}{x!} \\ & \times \left(\frac{h_{B,R}\lambda}{\mu}\right)^{\frac{(x+1)}{2}} \times K_{1-x} \left(\frac{2\sqrt{h_{B,R}(nh_{R,Pk} + h_{R,D}\lambda)}}{\mu}\right) \end{aligned} \quad (25)$$



$$\begin{aligned}
O_{PRS_2} = & 1 - \left[ \sum_{x=0}^{K-1} \sum_{r=0}^{R-1} (-1)^r \frac{2C_{R-1}^r}{x!} (r+1)^{\frac{(x-1)}{2}} \left( \frac{h_{B,R} h_{R,D} \lambda}{\mu} \right)^{\frac{(x+1)}{2}} K_{1-x} \left( \frac{(r+1) h_{B,R} h_{R,D} \lambda}{\mu} \right) \right. \\
& \left. - \sum_{x=0}^{K-1} \sum_{n=1}^N \sum_{r=0}^{R-1} (-1)^{n+r+1} C_{R-1}^r C_N^n \frac{2^{Rn} h_{R,D}}{x!} \left( \frac{\lambda}{nh_{R,P} \kappa + (r+1) h_{R,D} \lambda} \right)^{\frac{(1-x)}{2}} \left( \frac{h_{B,R} \lambda}{\mu} \right)^{\frac{(x+1)}{2}} \right] \\
& K_{1-x} \left( 2 \sqrt{\frac{h_{B,S} (nh_{S,P} \kappa + (r+1) h_{S,R} \lambda)}{\mu}} \right) \times \sum_{x=0}^{K-1} \frac{2}{x!} \left( \frac{h_{B,S} h_{S,R} \lambda}{\mu} \right)^{\frac{(x+1)}{2}} \\
& \times K_{1-x} \left( \frac{2 \sqrt{h_{B,S} h_{S,R} \lambda}}{\mu} \right) - \sum_{x=0}^{K-1} \sum_{n=1}^N (-1)^{n+1} C_N^n \frac{2 h_{S,R}}{x!} \left( \frac{\lambda}{nh_{S,P} \kappa + h_{S,R} \lambda} \right)^{\frac{(1-x)}{2}} \\
& \left( \frac{h_{B,S} \lambda}{\mu} \right)^{\frac{(x+1)}{2}} \times K_{1-x} \left( \frac{2 \sqrt{h_{B,S} (nh_{S,P} \kappa + h_{S,R} \lambda)}}{\mu} \right) \quad (26)
\end{aligned}$$

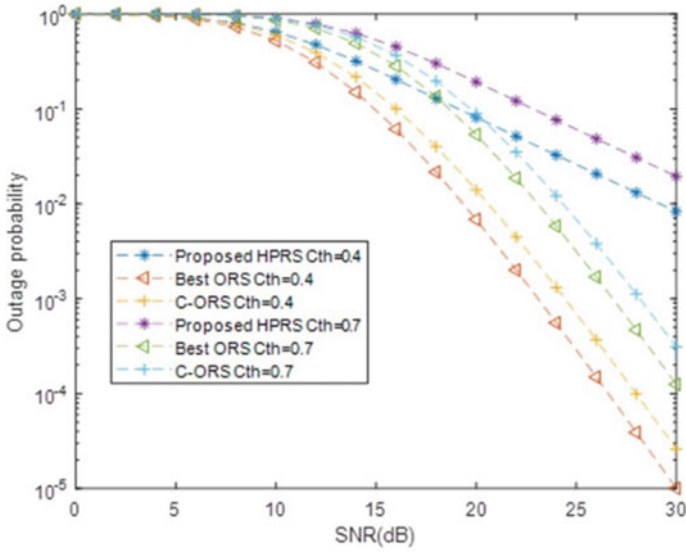
## 4 Simulation Results

The simulation parameters are considered within a section, and results are calculated for the outage probability and throughputs of the proposed method with other partial relaying protocol such as opportunistic partial relay selection (C-ORS), best opportunistic relay selection (Best-ORS). Table 1 displays the parameters and assumptions of the simulation. Initial energy is taken as 0.3 J for the source node, and initial energy for relays is given as (0.1–0.5 J). The Rayleigh fading channel of 1 MHz is taken into account in this approach. The coordinates at  $h_{S,R}$ ,  $h_{R,D}$ ,  $h_{B,R}$ ,  $h_{S,P}$ ,  $h_{B,S}$  are between range of (0.1–0.5).

Different values of SNR ( $P_B$ ) versus outage probability ( $O_{\text{proposedHPRS}}$ ) are plotted using Eqs. (25) and (26), as shown in Fig. 2. In addition, the location of relays also determines the outage performance. The outage of different relay selections is investigated by using two conditions, if  $(1 - I_D \psi) > 0$  and  $(1 - I_D \psi) < 0$ . With the

**Table 1** Simulation parameters

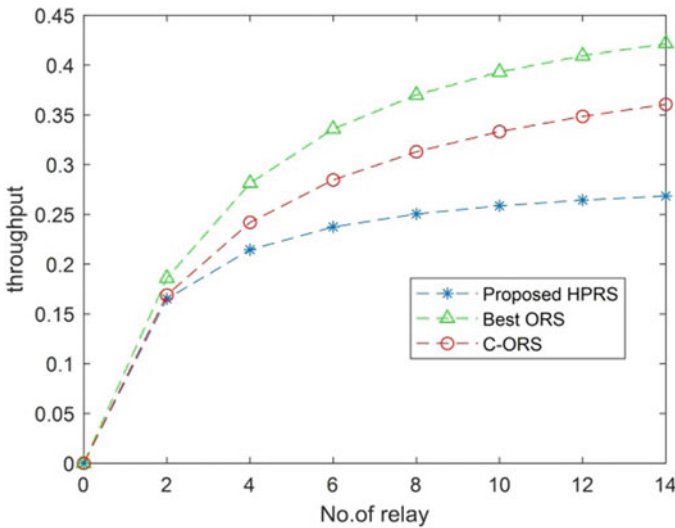
Parameters	Assumptions
Number of relay node( $R$ )	5–8
Number of primary users ( $P$ )	2
Number of power beacon( $B$ )	2
$C$ th	0.4–0.7
$I_I$	0–0.05
$T$	1 ms
$I_D$	0–0.6
$\eta$	1
Data rate	100–500 kb/s



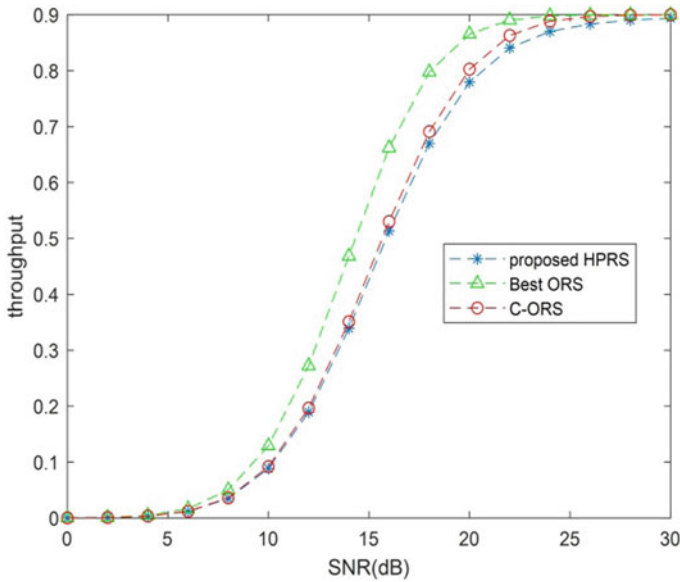
**Fig. 2** OP versus SNR (dB) when  $N = 3, R = 4, k = 2, \alpha = 0.3, \kappa = 0.05$  and path loss exponent = 3

assumption of limited interference ( $\lambda$ ), the proposed scheme outperforms better at higher SNR than the other relaying protocol.

In Fig. 3, the throughput has been seen as a function of the allocated time fraction



**Fig. 3** Throughput as a function of number of relay when  $SNR = 15 \text{ dB}, k = 2, \lambda = 0.5$  and  $\kappa = 0.01$



**Fig. 4** Throughput as a function of SNR (dB) when  $c_{th} = 0.5$ ,  $\alpha < 1$ ,  $R = 4$

( $\alpha$ ) for the EH method. The interference link between the nodes is considered to be negligible. In this simulation, with two primary sources and the value of EH processing time,  $\alpha < 1$  is considered.

In Fig. 4, results prove that throughput increases at different SNR values. If the value of  $\alpha$  is taken as high, then the throughput decreases. Other protocols perform better than the proposed HPRS because the optimum relay is based on high energy harvesting.

## 5 Conclusions

This paper aims to enhance selection of optimal relay-assisted underlay CR performance under constraints on interference. We suggested three relays, in which the multi-antenna power beacon is used for the dual-hop DF relay operation. The probability and efficiency of the outage were extracted in the proposed protocols with multiple power beacon and Rayleigh fading channel. The simulation results prove to have a better probability of outage and can achieve throughput from the maximum energy harvested by the relay. Therefore, from the outage probability analysis, it proves that at higher SNR, the performance of the secondary user is enhanced in underlay model. Finally, it is possible to boost the device efficiency of the proposed protocols through the establishment of a half-duplex or full-duplex in optimal power allocation scheme, and setting an estimated energy harvesting ratio for various relays.

## References

1. L. Wang, F. Hu, Z. Ling, B. Wang, Wireless information and power transfer to maximize information throughput in WBAN. *IEEE Internet of Things J.* **4**(5), 1663–1670 (2017)
2. A.M. Zungeru, L.-M. Ang, S. Prabaharan, K.P. Seng, Radio frequency EH and management for wireless sensor networks. in *Green Mobile Devices and Networks: Energy Optimization and Scavenging Techniques*, vol. 13, (CRC Press, New York, NY, USA, 2012) pp. 341–368
3. M.E. Bayrakdar, Cooperative communication based access technique for sensor networks. *Int. J. Electron.* **107**(2), 212–225 (2020)
4. I. Krikidis, J. Thompson, S. McLaughlin, N. Goertz, Amplify-and-forward with partial relay selection. *IEEE Commun. Lett.* **12**(4), 235–237 (2008)
5. A. Ghasempour, Impact of a time-varying rician fading channel on the performance of alamouti transmit diversity technique. in *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, (Athens, 2007), pp. 1–4
6. M. Dong, W. Li, F. Amirnavaei, Online joint power control for two-hop wireless relay networks with EH. *IEEE Trans. Signal Process.* **66**(2), 463–478 (2018)
7. A.A. Nasir, Zhou, S. Durrani, R.A. Kennedy, Relaying protocols for wireless energy harvesting and information processing. *IEEE Trans. Wireless Commun.* **12**, 3622–3636 (2013)
8. X. Lu, P. Wang, D. Niyato, D.I. Kim, Z. Han, Z., Wireless networks with RF EH: A contemporary survey. *IEEE Commun. Surveys Tutorials* **17**(2), 757–789 (2014)
9. N.P. Le, Throughput analysis of power-beacon-assisted eh wireless systems over non-identical nakagami-m fading channels. *IEEE Commun. Lett.* **22.4** (2017)
10. N.P. Le, Outage probability analysis in power-beacon assisted EH cognitive relay wireless networks *Wireless Commun. Mobile Comput.* (2017)
11. K. Tourki, H.-C. Yang, M.-S. Alouini, Accurate outage analysis of incremental decode-and-forward opportunistic relaying. *IEEE Trans. Wireless Commun.* **10**(4), 1021–1025 (2011)
12. C. Xu, Z.M. Liang, W. Yu, H. Liang, Y.C. , Outage performance of underlay multihop cognitive relay networks with EH. *IEEE Commun. Lett.* **20**, 1148–1151 (2016)
13. P.T. Tin, T.T. Duy, Power allocation strategies for dual-hop relay protocols with best relay selection under constraint of intercept probability. *ICT Express* **5.1**, 52–55 (2018)
14. A.K. Varma, D. Lahiri, Selection of optimum sensors for cooperative sensing in cognitive radio. in *Information and Communication Technology for Intelligent Systems*. (Springer, Singapore, 2019), pp. 587–594
15. Y. Zou, et al., Secrecy outage probability of cooperative relay networks with channel estimation error. in *IEEE Global Communications Conference (GLOBECOM)*. (IEEE, 2018)
16. A. Masadeh, A.E. Kamal, Z. Wang, Cognitive radio networking with cooperative relaying and EH. in *2017 IEEE 86th Vehicular Technology Conference (VTC-Fall)*. (IEEE, 2017)
17. I. Ahmed, M.M. Butt, C. Psomas, A. Mohamed, I. Krikidis, M. Guizani, Survey on EH wireless communications: challenges and opportunities for radio resource allocation. *Comput. Netw.* **88**, 234–248 (2015)
18. H. Chen, S. Tan, F. Zhao, Outage performances of relay-assisted transmissions in cognitive full-duplex relay networks. *EURASIP J. Wireless Commun. Netw.* **2015**(1), 1–11 (2015)
19. C. Xu, M. Zheng, W. Liang, H. Yu, Y.C. Liang, End-to-end throughput maximization for underlay multi-hop cognitive radio networks with RF energy harvesting. *IEEE Trans. Wireless Commun.* **16**, 3561–3572 (2017)
20. T.D. Hieu, T.T. Duy, S.G. Choi, Performance evaluation of relay selection schemes in beacon-assisted dual-hop cognitive radio wireless sensor networks under impact of hardware noises. *Sensors* **18.6**, 1843 (2018)
21. T.D. Hieu, T.T. Duy, S.G. Choi, Performance enhancement for harvest-to-transmit cognitive multi-hop networks with best path selection method under presence of eavesdropper. in *Proceedings IEEE 2018 20th ICACT* (2018), pp. 323–328
22. V.V. Huynh, H.S. Nguyen, L.T.T. Hoc, T.S. Nguyen, M. Voznak, Optimization issues for data rate in EH relay-enabled cognitive sensor networks. *Comput. Netw.* **157**, 29–40 (2019)

23. T.T. Duy, H.Y. Kong, Performance analysis of incremental amplify-and-forward relaying protocols with  $n$ th best partial relay selection under interference constraint. *Wireless Personal. Commun.* **71**, 2741–2757 (2013)
24. Al. Haija, A. Abu, Vu. Mai, Outage analysis for coherent decode-forward relaying over Rayleigh fading channels. *IEEE Trans. Commun.* **63**(4), 1162–1177 (2015)

# Building a Cloud-Integrated WOBAN with Optimal Coverage and Deployment Cost



Mausmi Verma, Uma Rathore Bhatt, and Raksha Upadhyay

**Abstract** Increasing demand of new services and application by the users poses a challenge on the communication network. Cloud computing serves this purpose by providing a shared pool of resources such as storage, servers, services, etc. Such technology uses backbone network for every applications to be served and thus results in high latency. To overcome such problem, cloudlets are used which are deployed in decentralized way. Cloudlets are clusters of computers which are connected to the users either directly or in maximum two wireless hops without affecting the latency of the network. There is another important factor, cost efficiency, which plays a very important role in the deployment of cloudlets in cloud-integrated wireless optical broadband access network (CIW). In this paper, we proposed an algorithm to find the optimum position in the network for the deployment of cloudlets by taking coverage and cost as a trade-off.

**Keywords** Cloudlets · Cloud components · Cloud-integrated WOBAN (CIW) · Optical network units (ONU)

## 1 Introduction

With the enormous growth of users in the telecom sector, the demand of the bandwidth is also increasing. To fulfill these demands, different types of access networks are available such as wireless network, optical network, etc., with their own pair of advantages and disadvantages. Wireless access network can be available everywhere and thus provides flexible and cost effective communication but restricted in

---

M. Verma · U. R. Bhatt (✉) · R. Upadhyay  
Institute of Engineering and Technology, Devi Ahilya University, Indore 452017, India  
e-mail: [uvrathore@gmail.com](mailto:uvrathore@gmail.com)

M. Verma  
e-mail: [mausmi.7nov@gmail.com](mailto:mausmi.7nov@gmail.com)

R. Upadhyay  
e-mail: [raksha\\_upadhyay@yahoo.co.in](mailto:raksha_upadhyay@yahoo.co.in)

bandwidth and highly susceptible to channel impairments. On the other hand, optical fiber access network provides huge amount of bandwidth but is limited in physical availability [1, 2]. Thus, meeting the demands of the users is focusing the research towards the hybrid fiber-wireless access network (FiWi) which is also referred as hybrid wireless optical broadband access network (WOBAN). FiWi access networks serve as a “last mile” for future access networks and provide a perfect solution for ubiquitous, flexible, and cost effective access network. The architecture of FiWi comprises an optical back-end typically a passive optical network (PON) and a wireless front end which is a wireless mesh network (WMN) [3]. PON consists of optical network units (ONUs) which is connected to optical line terminal (OLT). WMN consists of wireless routers and gateways connected in a multi-hop fashion. These gateways connect wireless front-end with optical back-end. The front-end wireless routers send/receive the end user’s traffic to/from the gateway to get served by the optical back-end.

Along with many advantages in the FiWi network, it is subjected to various limitations as well. In a wireless mesh network, the entire service requests generated by the end users and their responses propagate to and fro in the network and are responsible for the channel capacity consumption which is already scarce. And almost all the traffic in the network has to pass through the gateways and thus creates a bottleneck in the links near them and results in reducing the efficiency of WOBAN. Thus, a proficient solution to these problems is the integration of cloud components such as servers, storages, etc., with FiWi network, and this created network is known as cloud-integrated WOBAN (CIW) [4, 5] as shown in Fig. 1. The need for these cloud components in WOBAN is significant as various local cloud service requests (e.g., finding parking area, parking bills, etc.) generated by the users can be served locally by using these cloud components. These cloud components are linked with the wireless router in the wireless front-end. The generated service requests by the users initially forwarded to the wireless router with the associated cloud component. If the requested service is available with the cloud component, it gets served. Otherwise, the request is forwarded to the next wireless node with linked cloud component. If the cloud components are unable to serve the requested service, then the service is finally forwarded to the OLT. This CIW architecture can be implemented by two ways as CIW-I and CIW-S. If the wireless router is available with some extra memory and capacity, then this availability can be utilized by integrating cloud component within the wireless node to host the requested service, thus called CIW-I. On the other hand, when there is no supplementary development space is available in the WMN, then additional cloud components are connected to wireless nodes via Ethernet called CIW-S [4]. The cloud computing architecture follows distributed or centralized structure to provide various services within the access network of the users [6]. These cloud services are available by deploying low-cost cloud components in the access network. The centralized cloud computing benefits users with various types of on-demand interactive applications and computing resources such as storage, servers etc. Using centralized backbone network to serve these applications results in latency. To overcome this latency, wireless nodes are not directly connected to centralized clouds but via small clusters of computers known as cloudlets. These cloudlets are connected

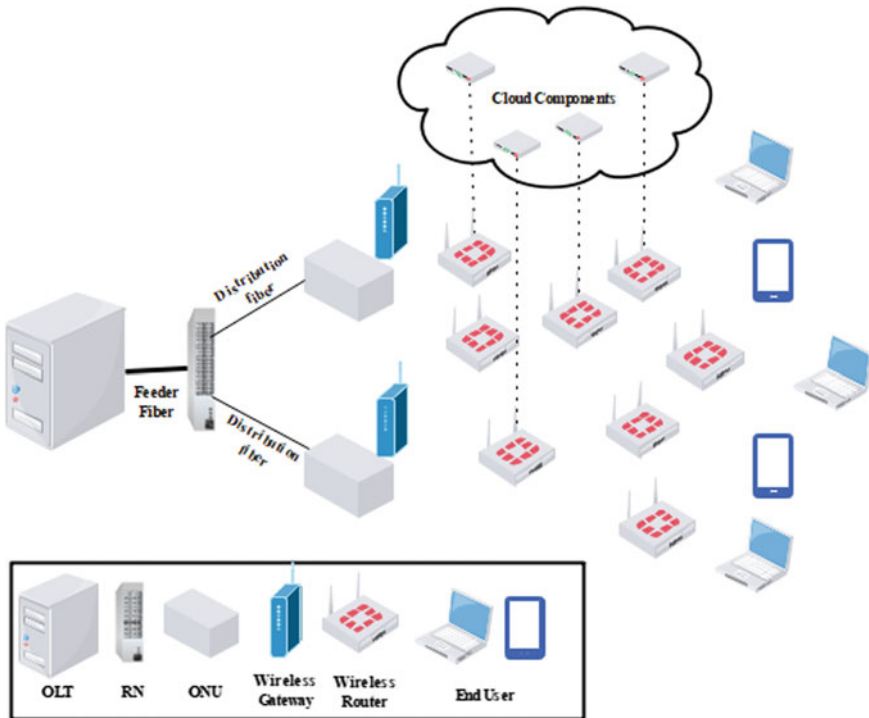


Fig. 1 Cloud-integrated WOBAN architecture [5]

near the edge of wireless nodes to provide the computing resources to the users in one or two wireless hops [7]. While deploying cloudlets various issues were considered such as cloud components placement, computation offloading, energy saving, cost effectiveness, etc. Various works have been proposed covering all the defined parameters related to cloudlets deployment. In this paper, we focus on providing a trade-off between coverage and cost for the deployment of cloudlets in the network.

The rest of the paper is structured in the following way. Section 2 presents the CIW related work such as its various architectures or issues related to cloudlet deployment. Section 3 provides the detailed explanation of the proposed work in the paper. Simulation results are discussed in Sect. 4. Finally, Sect. 5 concludes the paper and presents future scope of the work.

## 2 Related Work

In the study [8], authors investigated the best usage of optical network and proposed architecture to facilitate cloud services through PON. The authors considered ONU as a wimpy node to provision cloud computing services via PON using fast array



of wimpy node (FAWN) architecture. In paper [9], the authors investigated the PON computing and storage capabilities by integrating cloud computing with PON. In [10], authors proposed an architectural framework to find a suitable position for the cloudlet over an optical network for optimum placement considering capacity and latency constraints. The authors in [11] considered energy saving issue related to CIW architecture and proposed a new approach through routing mechanism which saves energy, and that mechanism is called green routing for CIW (GRC). This allows CIW to perform self-management on the activation and deactivation of network components, such as ONUs and CCs, so as to reduce the overall power consumption. Authors in [12] considered the survivability issue and proposed a protection strategy known as protection scheme with maximal coverage ratio (PSMCR). This strategy aims that wireless routers provide maximal coverage against distribution fiber link failure. Considering the situation of multiple fiber link failure, the authors in [13] proposed a scheme known as survivability strategies against multi-fiber failure (SSMF) which not only covers survivability issue but also optimizes ONU placement solution. In [14], the authors proposed an algorithm that aims to find the exact location of the critical routers. These critical routers play an important role while transferring data traffic from the ONU which gets affected in case of distributed fiber link failure to the backup ONU.

In [15], the authors merged two technologies, i.e., centralized cloud computing and mobile edge computing (MEC), and integrated it with FiWi to evaluate the performance gain of the network. They also developed a probabilistic model against optical and MEC against network link failure. In [16], the authors proposed a collaborative computation offloading method using game theory model for IOT over fiber-wireless networks. In [17], the authors considered the situation where some cloudlets are overcrowded and some cloudlets are under loaded in the optical network with edge computing technology. Considering such situation, authors provided a non-cooperative game theorybased mechanism for computation offloading.

To deploy the services or allocate resources, the service providers need deployable locations, and this issue in CIW architecture is considered by authors in [18]. The authors provided an auction mechanism in which there is a price deal between the service providers as buyers and the location providers as sellers. They also proved the truthfulness that by doing such deals, both the buyer and seller get profit. In [19], the authors proposed an algorithm to optimize the position of ONUs and then deployed cloud components using cluster-based approach so as to provide a cost-effective solution. In [20], the authors proposed a probabilistic coverage model which assures the QoS regardless of any unwanted conditions gets imposed by either the sensor nodes or the environment of the network.

### 3 Proposed Work

In the proposed work, we presented an algorithm that finds the optimum position for placing the cloudlets in the network. This algorithm also provides the minimum number of cloudlets required for providing maximum coverage of the network by presenting a trade-off between cost and coverage. Initially, we placed the wireless routers (WR) randomly in the predefined network area. After placing wireless routers, we deployed minimum required ONUs in the network within hop constraint to provide maximum wireless routers coverage. We now determine the exponential coverage [20] of each wireless router using Eq. 1, which depends on the distance (DIS) between the selected wireless routers with other wireless routers.

$$\text{Cov} = e^{-\beta \times \text{DIS}} \quad (1)$$

where DIS is the distance between two WRs, and  $\beta$  is the parameter related to physical characteristics of WR.

We then determine a relationship matrix for wireless routers based on this exponential coverage as shown in Eq. 2.

$$\text{RM} = \begin{cases} \text{Cov} & \text{if DIS} < 350 \text{ m} \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

This relationship matrix contains the information of the connectivity of each and every wireless router to other wireless routers of the network on the basis of distance constraint [7]. Based on this information, the connectivity index of every router is found. Here, the connectivity index is defined by the connectivity of a particular router to other routers (total number of wireless routers connected with particular wireless router). After finding the connectivity index, next process is to find the possible location for the deployment of cloudlets. For that, firstly, we select the wireless router with maximum connectivity index and deploy the first cloudlet on that wireless router location while ensuring all its associated users get served by this cloudlet. Once first cloudlet is deployed, the relationship matrix gets updated by deleting all the routers covered by this cloudlet, and based on this updated relationship matrix, again connectivity index is determined. Also, a list that contains all the WRs gets updated by deleting those wireless routers which gets covered by the deployed cloudlet. By using this updated relationship matrix, next wireless router with maximum connectivity index is taken into account. The selected wireless router is checked for its associated connectivity index as zero (no wireless router covered) or one (single wireless router covered). If no such condition is achieved, then second cloudlet is deployed on the selected wireless router location, and then the process is repeated till to achieve the condition of zero or one connectivity index. At last, the remaining wireless routers in the updated list of WRs are considered as the location for the deployment of cloudlets. In this manner, we find out number of cloudlets, their

positions, and connected wireless routers to each of the cloudlet to provide 100% network coverage.

The notations used in simulation and psuedo-code of the proposed algorithm are as follows:

**Notations**

WR: Wireless router

NWR: Total number of WRs used in the network

RM: Relationship matrix showing connectivity among WRs

TR: List contains all the WRs

$v$ : Variable which gives the maximum value of the connectivity index

$k$ : Variable which gives the linear index of value in “ $v$ ”

CC: Set which tells the position of cloudlet to be deployed

PCC: List contains minimum number of cloudlets required for 100% coverage

DWR ( $i, j$ ): Euclidean distance between  $i$ th cloudlet with  $j$ th WR

**Step 1:** Initially divide the network area in squares and then randomly place WRs in the network.

**Step 2:** Deploy minimum number of ONUs required within hop constraint so as to ensure coverage of all the WRs

**Step 3:** Finding the relationship matrix RM for each WR using equation 2, assuming  $\beta=0.5$

**Step 4:** Find the connectivity index for all the WRs based on the RM

**Step 5:** TR= [1: NWR]

CC= [ ]

for i=1: length (RM)

    if (associated connectivity index of all the WRs is either 0 or 1)

        PCC= [CC TR]

        break

    else

        [v k]=max (connectivity index of RM)     // if more than one routers with same connectivity index then router with lower subscript is selected

        CC(i)=Deploy cloudlet on k<sup>th</sup> location

        delete all the routers covered by k<sup>th</sup> cloudlet from RM and TR

    end if

update RM& TR

find the connectivity index of all the WRs based on the updated RM

end for

**Step 6:** Find the Euclidean distance DWR of each router with all the cloudlets

find minimum value of DWR for each WR from cloudlets

    if ( the value is less than or equal to 350m)

        the wireless router is within the coverage of that associated cloudlet

    else

        the wireless router is cloudlet itself

end if

### 4 Illustrative Example and Simulation Results

To study the worthiness of the proposed work, we performed the simulation in the MATLAB environment. In the simulation setup, we considered a hypothetical cloudlet-integrated WOBAN (CIW-I) of 1000m × 1000 m square-shaped region with a different number of randomly placed wireless routers, i.e.,  $NWR = \{10, 20, 30, 40, 50\}$ . To understand the functioning of proposed algorithm, its stepwise illustration has been carried out which is as follows:

According to step 1 of the proposed algorithm, the considered area is divided into  $5 \times 5$  squares, and 20 wireless routers are randomly placed. Step 2 determines the total number of ONUs (7) required to cover all the wireless routers within two-hop constraint as shown in Fig. 2. After ONUs placement, number of cloudlets required in the network using proposed approach is determined. Follow step 3, to generate a relationship matrix RM of wireless routers using exponential coverage assuming  $\beta = 0.5$ . Step 4 finds the connectivity index of all the WRs taking RM into account and is shown by the initially generated column in Table 1. Finally, step 5 explains the analysis for finding the location of cloudlets to be deployed. Initially consider a list TR for 20 WRs.

$$TR = [1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8 \ 9 \ 10 \ 11 \ 12 \ 13 \ 14 \ 15 \ 16 \ 17 \ 18 \ 19 \ 20]$$

From the initially generated column given in Table 1 select WR with maximum connectivity index (i.e., 13th wireless router with connectivity index 10). This is the position to deploy first cloudlet. After deploying first cloudlet, initially generated

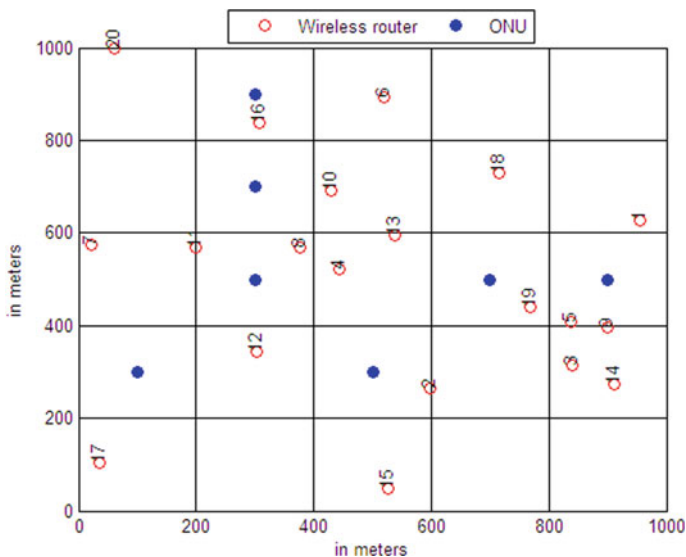


Fig. 2 Configured FiWi architecture

**Table 1** Connectivity index of every WR

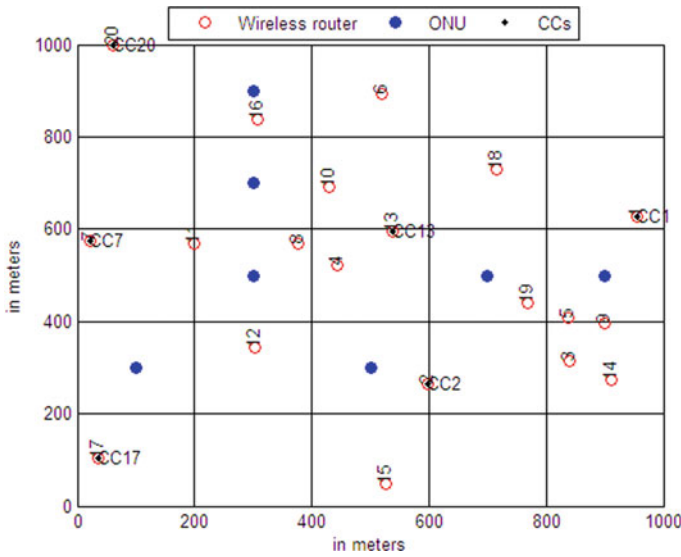
S. No.	WRs numbering	Associated connectivity Index		
		Initially generated	After Iteration-1	After Iteration-2
1	1	5	3	0
2	2	9	6	0
3	3	6	4	1
4	4	9	1	0
5	5	7	4	1
6	6	4	1	0
7	7	1	0	0
8	8	6	1	0
9	9	6	4	1
10	10	7	1	0
11	11	7	2	1
12	12	5	1	0
13	13	10	0	0
14	14	5	3	0
15	15	1	0	0
16	16	7	2	1
17	17	0	0	0
18	18	7	3	1
19	19	9	6	1
20	20	1	0	0

connectivity index from Table 1 and TR list are updated. In the updation process, all the WRs covered by firstly deployed cloudlet are deleted from RM and TR. And based on this new RM, updated connectivity index is determined. This updated connectivity index is shown by iteration-1 column of Table 1. Updated TR list with deleted WRs covered by first deployed cloudlet is [1 3 5 7 9 13 14 15 17 20].

Again from iteration-1 column, WR with maximum connectivity index is selected (i.e., 2<sup>nd</sup> WR with connectivity index 6). The second cloudlet is deployed at 2<sup>nd</sup> WR position. After deploying second cloudlet, once again the connectivity index (Iteration-1 column) and TR list get updated by deleting all the WRs covered by second deployed cloudlet from RM. Updated connectivity index set is shown by iteration-2 column of Table 1. Also the updated TR list is given as [1 7 17 20].

It is clear from iteration-2 column of Table 1 that all the WRs are with either one or zero connectivity index. Thus, all the remained WRs in the TR list are the location for the rest of the cloudlets. Finally, a list PCC is generated with the total deployed cloudlet numbering as: PCC = [1 2 7 13 17 20].

It is obvious from the PCC list that 6 cloudlets are sufficient to provide 100% coverage to the WRs as shown in Fig. 3. To find the coverage of each cloudlet in



**Fig. 3** Identification of position of cloudlets using proposed approach

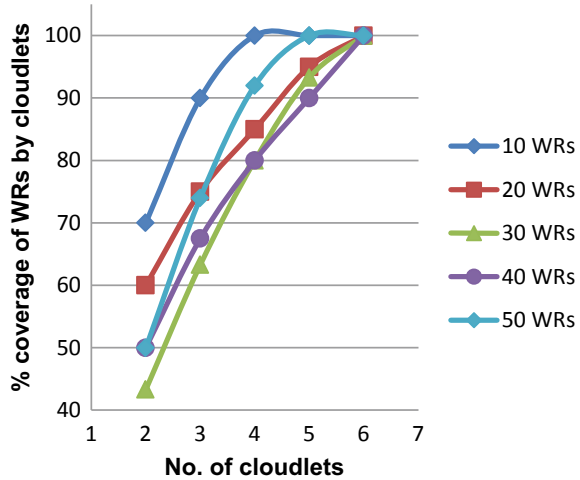
terms of wireless routers is analyzed by step 6. By following this step, initially, the distance of each cloudlet from all the wireless routers is determined. After finding the distance, we considered each and every wireless router one by one and checked for the minimum distance for every cloudlet. For the considered wireless router, if the minimum distance is within the constraint (less than or equal to 350 m), then that wireless router will be covered by its associated cloudlet. Otherwise, the selected wireless router will be cloudlet itself. After performing the analysis for every wireless router, the coverage of each cloudlet is given in Table 2.

It is clear from Table 2 that for 20 wireless routers, total 6 cloudlets provide 100% coverage to all wireless routers. Cloudlet deployed at wireless router number 17 is serving to the users of this router only. If we sacrifice the connectivity to these users by removing this cloudlet, then 17% of the total cost (in terms of cloudlets) will be saved with 95% coverage to the remaining users. Removing one more cloudlet (connected

**Table 2** Coverage of each cloudlet for 20 WRs

Cloudlet No.	Cloudlet deployed at WR position	Covered WRs
1	1	1, 5, 9
2	2	2, 3, 12, 13, 14, 15, 19
3	7	7, 11
4	13	2, 4, 6, 8, 10, 13, 18
5	17	17
6	20	16, 20

**Fig. 4** Percentage coverage of wireless routers by varying cloudlets and wireless routers



with minimum number of routers) will further result in 85% network coverage area and 33% of cost saving. So, in this way, there will always be a trade-off between network cost and coverage area.

An extensive simulation is carried out for a varying number of wireless routers, namely 10, 30, 40, and 50 as shown in Fig. 4. Increase in number of wireless routers will not increase the network latency since we considered 350 m coverage constraint [7] for each case. From the figure, we can find out the total number of required cloudlets for given number of wireless routers to ensure 100% connectivity to all the users. It is clear from Fig. 4 that for the desired level of coverage, we can find out required number of cloudlets; hence, we can calculate the network cost in terms of deployed cloudlets. Reduction in number of cloudlets will decrease the network coverage but will result in cost saving. Hence, proposed algorithm offers a solution to the network service providers in terms of network coverage and cost.

## 5 Conclusion

Cost efficiency and coverage are the two important parameters considered while deploying the communication network components. In cloud-integrated WOBAN, it is an important measure to optimize cloudlet numbers and position in order to provide maximum coverage and minimum deployment cost. The proposed work fulfills this requirement and suggests a trade-off between the number of cloudlets and percentage network coverage without affecting the latency of the network.

In the future work, number of cloudlets can further be optimized, and its utility can be enhanced by shifting wireless routers in the network without affecting network latency.



**Acknowledgements** We would like to acknowledge IET, DAVV, Research Center, Indore, India. This paper can be used as a part of Ph.D. thesis in the future for the first author.

## References

1. M.Z. Chowdhury, M.K. Hasan, M. Shahjalal, E.B. Shin, Y.M. Jang, Opportunities of optical spectrum for future wireless communications, in *International Conference on Artificial Intelligence in Information and Communication (ICAIIIC)* (2019)
2. U.R. Bhatt, P.K. Yadav, N. Chouhan, Fi-Wi architecture and survivability, in *International Conference on Recent Advances in Interdisciplinary Trends in Engineering & Applications* (2019)
3. S. Sarkar, S. Dixit, B. Mukherjee, Hybrid wireless-optical broadband access network (WOBAN): a review of relevant challenges. *IEEE/OSA J. Lightwave Technol.* **25**(11), 3329–3340 (2007)
4. A. Reaz, V. Ramamurthi, M. Tornatore, Cloud-over-WOBAN (CoW): an offloading-enabled access network design, in *IEEE International Conference on Communication*, pp. 1–5 (2011)
5. S. Solanki, R. Upadhyay, U.R. Bhatt, Cloud-integrated wireless-optical broadband access network with survivability. *Int. J. Sens. Wireless Commun. Control* (2020)
6. U.R. Bhatt, A.B. Marmat, R. Upadhyay, Cloud integrated WOBAN: revisited for cloud, cloud services, performance issues and challenges, in *International conference on Recent Advances in Interdisciplinary Trends in Engineering & Applications* (2019)
7. D. Fesehaye, Y. Gao, K. Nahrstedt, G. Wang, Impact of cloudlets on interactive mobile cloud applications, in *IEEE 16th International Enterprise Distributed Object Computing Conference* (2012)
8. M. Taheri, N. Ansari, A feasible solution to provide cloud computing over optical networks. *J. IEEE Network* **27**(6), 31–35 (2013)
9. Y. Luo, F. Effenberger, M. Sui, Cloud computing provisioning over passive optical networks, in *1st IEEE International Conference on Communications in China (ICCC)* (2012)
10. S. Mondal, G. Das, E. Wong, Compassion: a hybrid cloudlet placement framework over passive optical access networks, in *IEEE Conference on Computer Communications* (2018)
11. A. Reaz, V. Ramamurthi, M. Tornatore, B. Mukherjee, Green provisioning of cloud services over wireless-optical broadband access networks, in *Global Telecommunication Conference (IEEE Globecom)*, pp. 1–5 (2011)
12. Y. Yu, Y. Liu, Y. Zhou, P. Han, Planning of survivable cloud-integrated Wireless-optical broadband access network against distribution fiber failure. *J. Opt. Switch. Network.* **14**, 217–225 (2014)
13. Y. Yu, Y. Zhou, Y. Liu, P. Han, Survivable deployment of cloud-integrated fiber-wireless networks against multi-fiber failure. *Photonic Network Commun.* **31**(3), 559–567 (2016)
14. U.R. Bhatt, S. Sadafal, K. Chaurasiya, A. Awasthi, Critical routers identification to handle distribution fiber failure in wireless optical broadband access network (WOBAN), in *Third International Conference on Computing and Network Communications (CoCoNet)*, pp. 2186–2194 (2019)
15. B.P. Rimal, D. Van Pham, M. Maier, Mobile-edge computing versus centralized cloud computing over a converged Fi-Wi access network. *J. IEEE Trans. Network Service Manag.* **14**(3), 498–513 (2017)
16. H. Guo, J. Liu, H. Qin, Collaborative mobile edge computation offloading for IoT over fiber-wireless networks. *J. IEEE Network* **32**(1), 66–71 (2018)
17. S. Mondal, G. Das, E. Wong, Computation offloading in optical access cloudlet networks: a game-theoretic approach. *IEEE Commun. Lett.* **22**(8), 1564–1567 (2018)

18. S. Dai, Y. Li, L. Hai, A truthful auction mechanism for service deployment in cloud-integrated WOBAN, in *16th International Conference on Optical Communications and Networks (ICOON)* (2017)
19. U.R. Bhatt, N. Chouhan, A.B. Marmat, R. Upadhyay, Deployment of cost-efficient cloud integrated WOBAN: a cluster-based approach. *Int. J. Senso. Wireless Commun. Control* (2020)
20. Z. Taghikhaki, N. Meratnia, P.J.M. Havinga, A trust-based probabilistic coverage algorithm for wireless sensor networks. *Proc. Comput. Sci.* **21**, 455–464 (2013)

# VR Classroom for Interactive and Immersive Learning with Assessment of Students Comprehension



J. S. Jaya Sudha, Nandagopal Nandakumar, Sarath Raveendran, and Sidharth Sandeep

**Abstract** The virtual classroom environment is created using virtual reality that enables multiple students to enter as if in a real class but with better learning environment. Conventional learning is currently limited in the current model of textbook teaching. An interactive and visual environment provided for learning enhances the rate at which the student grasps concepts. Even though many modern online teaching methods are available today, it is not possible to check whether a student is paying attention or not. Technology is evolving at a very fast rate, and this research is an apt integration of two modern technologies: machine learning and virtual reality, so as to increase the quality of education for students. A shared VR environment, optimised for learning, will be created. Students can wear a head-mounted display and select an avatar for themselves, which will be seen by other students and teachers. The VR environment is created using Unity3D software. Students will also have to wear an EEG scanner on their heads. The output of this scanner will be fed to the machine learning subpart. Neural networks are used to identify whether the student is paying attention or not. If a student is not paying attention, the teacher will be informed about it, with a message near the student's avatar. It has many advantages over traditional learning techniques, like usage of multiple senses and inclusivity for differently abled students.

**Keywords** Virtual reality · Machine learning · Recurrent neural network

---

J. S. Jaya Sudha · S. Raveendran · S. Sandeep (✉)  
Computer Science and Engineering, Sree Chitra Thirunal College of Engineering,  
Thiruvananthapuram, India  
e-mail: [sidharthsandeep97@gmail.com](mailto:sidharthsandeep97@gmail.com)

J. S. Jaya Sudha  
e-mail: [jayasudhajs@gmail.com](mailto:jayasudhajs@gmail.com)

N. Nandakumar · S. Raveendran · S. Sandeep  
Sree Chitra Thirunal College of Engineering, Thiruvananthapuram, India  
e-mail: [nandg8@gmail.com](mailto:nandg8@gmail.com)

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2021  
S. M. Thampi et al. (eds.), *Advances in Computing and Network Communications*,  
Lecture Notes in Electrical Engineering 735,  
[https://doi.org/10.1007/978-981-33-6977-1\\_11](https://doi.org/10.1007/978-981-33-6977-1_11)

## 1 Introduction

Virtual reality is currently one of the most trending technologies today. It is a simulated experience that mimics a real-world environment using computer-generated imagery. The user will be completely immersed in this generated virtual environment and will interact within its bounds. This provides a higher level of interactivity for students.

Machine learning is another trending topic in technology. It can be defined as the application of artificial intelligence (AI) that provides the ability to automatically learn and improve from experience without being explicitly programmed. This can be used to assess the learning levels of students. Virtual reality and machine learning are not utilised to the fullest, in the field of education. The education system still holds on to its traditional methods. Our aim is to use these two technologies to create a better, interactive and visual environment for teaching and learning that enhances the rate at which the student grasps concepts.

The existing models of teaching are extremely limited in terms of comprehension and interactiveness. Technology has improved at a phenomenal rate but no significant application of it has been implemented to enhance the learning experience. The existing model does not provide an immersive learning environment for students nor does it allow teachers to gain an idea of student's comprehension. Virtual reality environment increases the interest of students with immersive learning and better teaching methods can be adopted by analyzing the concentration of students using EEG signals.

The general aims to be attained by this research are: To create a shared virtual reality environment optimised for learning. To analyse whether the VR environment, accessible from anywhere by students and teachers, can be efficiently used instead of conventional teaching methods. To use machine learning to analyse concentration of students by measuring their EEG signals in real time and analyse whether this can be used to measure the level of attention of students since the teacher and learners are not physically present at the same place.

## 2 Literature Review

Aftab et al. demonstrate that individualised instruction is superior to the traditional one-size-fits-all teaching approach [1]. The use of 3D virtual environments for educational purposes is becoming attractive because of their rich presentation, user-friendly interaction techniques and adaptive capabilities. A fuzzy logic-based approach is used to quantitatively measure the level of learning of students, and it is used as adaptation criterion for changing the contents of 3D-virtual learning environments. The system displays the customized teaching materials for different students, which results in improved learning. The experimental results showed that the proposed approach is effective and can be efficiently used to enhance the learning capabilities of students in 3D-VLEs.

Georgios et al. describe a software tool aimed to improve teaching effectiveness [2]. The tool utilises virtual reality technologies such as OCULUS RIFT, virtual reality helmet and Unity3D game engine in an attempt to simulate a virtual classroom experience. The tool focuses on offering an asynchronous distance learning experience to the students via a 3D application, which allows students to participate in lectures, ask questions to the virtual instructors and receive pre-stored or generated answers. This research focused to simulate the experience of a student in a classroom, including the interactions between the students and the lecturer.

Chung-Yen et al. conducted a research to measure human's brain waves; the evaluation is built based on the energy of watching a video [3]. Brainwave reflects the change in electrical potential resulting from the conjunction between the thousands of brain neurons. A neuron can receive signals from other neurons and starts off cyclic discharge reaction when sufficient energy is accumulated. That is also the reason why people persistently emit brainwaves. However, some people rely on their brain to deal with many things, and it may lead to learning status. The learning status includes good, questionable and bad. This learning status can be classified based on their response. Users click choices in the system. Then, using deep learning to predict their learning status through the experiments. The experimental results of this research indicated that this method is valuable for learning status prediction. This type of learning status classification can be used to assess whether students are confused or attentive in class.

### **3 Overview**

A multiuser environment with a virtual interface provides access to teachers and students. This is used to create and access the virtual environment when required. In this project work, the classroom, the lecturer and the virtual students were simulated using 3D models. The EEG device collects the students data, analyses it and makes a prediction of whether the student is confused or not and displays this to the teacher. The research can be divided into three main subsystems: Virtual reality subsystem, machine learning subsystem and multiuser subsystem. Integration of the various components and the hardware was included. The use case diagram is shown in Fig. 1.

#### ***3.1 Hardware***

A virtual reality headset allows the user to access the virtual environment. The device is mounted on the user's head. A virtual reality headset is mandatory for users to participate in the virtual from any location. The project was completed on Oculus Rift VR headset.environment

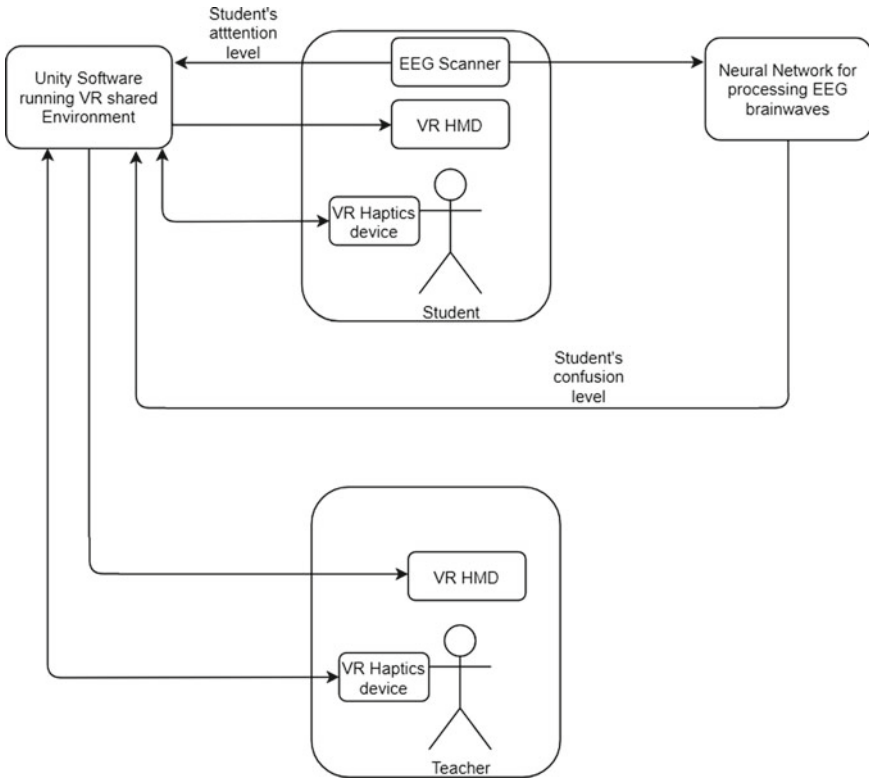


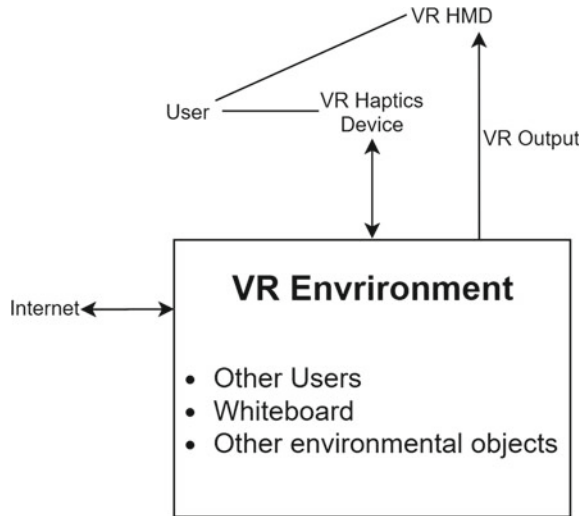
Fig. 1 Use case diagram

An EEG headset was used to measure the brainwaves of the user. The device used was Neurosky Mindwave Mobile 2 which is a portable EEG headset. It uses TGRAM1 module which reads the user’s raw EEG signal, delta, theta, alpha, beta and gamma waves [4]. The data was read by connecting the device using Bluetooth COM port.

### 3.2 Virtual Reality Subsystem

The classroom 3D environment is created as part of this subsystem. 3D models that are created using 3D modelling software are placed in appropriate places, so as to make up the environment. This subsystem also deals with user motion, interaction and animation. The important components developed are: humanoid avatar for students and teachers, whiteboard and virtual interface for interaction within the environment. The overview of the VR subsystem is shown in Fig. 2.

**Fig. 2** Overview of VR subsystem



The humanoid avatar is a 3D model that resembles a human being. This model is assigned to the users within the virtual environment. The hands of this humanoid avatar mimic the movement of the controller which the user moves. The parts of the avatar are animated and moved accordingly using inverse kinematics [5]. Kinematics describes the rotational and translational motion of points and objects without any reference to mass, force or torque. This method of posing a skeleton model is known as forward kinematics. However, it is often useful to look at the task of posing joints from the opposite point of view, given a chosen position in space, and working backwards and orienting the joints so that the end point lands at that position in the environment. This approach is known as inverse kinematics (IK). In this project, only simple locomotion needs to be implemented, so a heuristic approach is sufficient. Heuristic solvers do not take into consideration spatio-temporal corrections between nearby joints, as they treat each joint's constraint independently with no global constraints. In such a way, the humanoid avatar can be implemented. Also, the user's name is displayed on top of the avatar. In case of student, the confusion value predicted by the machine learning model and the attention value given by the EEG hardware are also displayed on top.

The whiteboard is a 3D surface with white texture applied on it, i.e. the colour white is applied on each of the pixel of the whiteboard surface texture. The whiteboard pen is another 3D model, which can be interacted with, i.e. grabbed and moved, by the user. A technique called raycast touch is used to detect when the user touches the whiteboard with the pen. Raycasting is a process by which a 3D object casts a ray that is cast by a 3D object in some direction. In this case, the ray is cast by the tip of the pen, parallel to its direction.

The VR user interface includes the VR main menu, login screen, etc. The user interaction is done with the help of straight-line pointers. When the user holds the

haptic device in a direction and presses a button, a straight line pointer is shown in that direction. If a 3D object comes in the path of the pointer, the pointer stops there. If the object is an interactable object like the login button or a key on the virtual keyboard, it gets highlighted in another colour. The user can then press another button to select that option.

### **3.3 Machine Learning Subsystem**

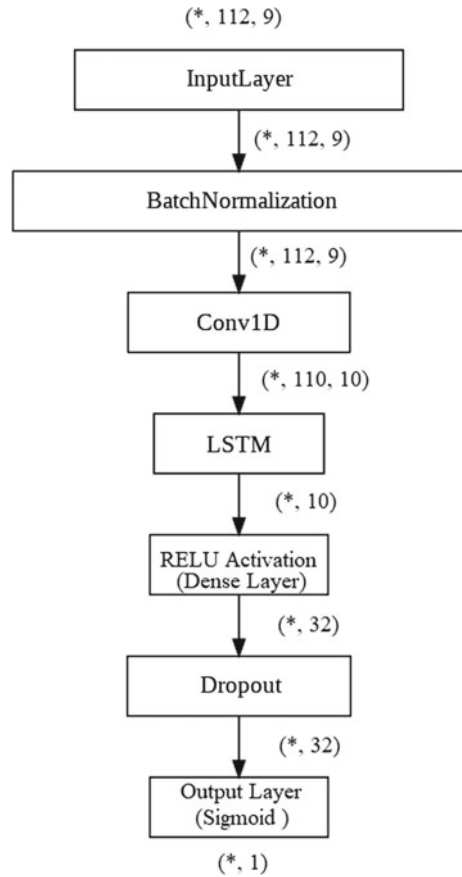
The machine learning subsystem consists of collecting the EEG data, analysing it and producing the output showing the student's confusion level, on a scale of 0–100 (0-least confused, 100-most confused). It also sends the average attention value of the last minute, as calculated by the hardware, as an output. The result is shown on top of the student's avatar, which can be seen by the teacher.

The datasets were available of motor movements, emotions, event related and visual recognition. One dataset found was “EEG dataset of fusion relaxation and concentration moods,” Ahmed Albasri [6]. However, this dataset was not suitable since relaxation and concentration were not an accurate criteria for whether the student is confused or not. The dataset suitable for and selected was: “Confused student EEG brainwave data” by Wang [7]. This data had been collected from ten college students while they each watched ten MOOC video clips. The students wore a single-channel wireless headset that measured activity over the frontal lobe. After each session, the students rated their confusion level on a scale of 1–7, where one corresponded to the least confusing and seven corresponded to the most confusing. These labels if further normalised into binary labels of whether the students are confused or not. The data was sampled at 0.5 Hz. So, for every second, there will be two rows of data. Therefore, for a minute, there are 120 rows of data. That is, each 120 rows corresponds to one data point. In total, there are 120,000 rows, which corresponds to 100 data points.

Some of the videos were not exactly one minute in length. Some were a few seconds short and some longer. But for training purposes, all data points should contain the same number of rows. The shortest number of rows for a data point was found to be 112 (the shortest video was 56 s in length). So, for every datapoint, the rows after 112 were removed. The features attention, mediation and predefined label are removed. The features attention and mediation are proprietary values which are calculated by the device itself, and these are device specific and are not suitable for the project. The features selected for the model are raw, delta, theta, alpha low, alpha high, beta low, beta high, gamma low, gamma high waves [8] and the label. The data is rearranged into a 3D array with dimension 112,100, 9. 112 number of rows for each datapoint, 100 being the number of datapoints and 9 being the number of features. Since a single data point has 112 rows, it takes 56 s to be created, at a sampling rate of 0.5 Hz. So, in real time, the ML model can predict a new confusion value every 56 s.



**Fig. 3** Machine learning model



**Machine Learning Model** Long short-term memory (LSTM) is used for creating the model for classifying the student. LSTM is an artificial recurrent neural network (RNN) architecture [9] (Fig. 3).

First a sequential model, which is a linear stack of layers, is created. Then, batch normalisation is done. A convolution network is used with ten filters and kernel size 3 (length of convolution window) [10]. AN LSTM network is implemented. A layer with dimensionality 32 and activation function “relu” is applied. 0.2 fractions of the input units are dropped to prevent overfitting. The activation function “sigmoid” along with dimensionality 1 is implemented. The dataset contains labels 0 and 1. Since the final layer is a sigmoid activation function, the output of the neural network is a value between 0 and 1, which is the probability of the label to be 1. In order to give output as in a classic binary classification problem, a threshold, say 0.5, can be set, and if the probability is greater than that, the label is 1, else, it is 0. But in this project, we multiply the probability by 100 to obtain an approximate scale of confusion of the student.

### 3.4 *Multi-user Subsystem*

The VR environment has to be shared by multiple users, that is, teacher and students at the same time. The three main functions of this subsystem are: user authentication, join/create rooms and synchronising user actions. The users are given the ability to create accounts and login. A server has to be maintained, and connection-oriented communication is used since reliability is of high priority. NoSQL database using JSON objects are used for storing user data.

Whenever a user registers a new account, with username, password and email parameters, the credentials are validated and a new object is created with the above fields. A unique user ID is assigned to each user. The timestamp of creation of account is also stored. When a user tries to login with a username and password, the objects are checked whether a user with such credentials exists and is logged in based on that. Once the user has logged in, they have to create or join a room. Usually, a teacher creates a room and students join after that using connection-oriented communication. The room data is also stored as JSON objects. When a teacher tries to create a new room, the objects are checked whether a room of the same name already exists. If not, the room is created. When a student tries to join a room, the objects are checked whether such a room exists, and if it exists, a new object entry is made with the username of the student. In this manner, the room continues to be in existence even if the teacher leaves, as long as there is at least one student remaining in the room.

**Synchronise user actions** The various actions done by each user have to be reflected to each user in the room using connectionless communications since speed has higher priority over reliability. The main tasks involved are given below.

**User movement:** Whenever a user changes their position, the new position value has to be sent to every other user and that user's avatar has to be moved to the new position in every user's view.

**User animation:** Similar to movement, every user's animations have to be sent to every other user and applied everywhere.

**User confusion value updated:** Whenever the machine learning subsystem updates the confusion and attention value of user, it also has to be sent to other users and has to be displayed above the particular user's avatar in every user's view.

**Voice communication:** Whenever a user speaks, it has to be recorded, sent to all other users and played there.

### 3.5 *Implementation*

Unity game engine is used to create, both 2D and 3D, experiences or environments [11]. The engine's primary scripting API is done in C sharp. Unity was used to create a virtual classroom which mimics a real-life classroom. It contains models

of 3D models of desks, chairs and a whiteboard. The users on accessing the virtual environment are assigned avatars which are modelled after human beings. These are visible to the users, so they can interact with each other. The teacher can use the blackboard to write and draw. This would be visible to all the users.

A framework called VRTK is used for interfacing the Oculus rift with Unity and for implementing the various VR specific actions in the environment like setting the camera in front of the HMD, setting up VR interactable objects like grabbable pen, VR UI [12]. Another framework named VRIK is used for implementing the inverse kinematics associated with the humanoid avatar.

Machine learning model is implemented in the Python programming language. A library called Keras is used to create the model as mentioned in the design. Keras is an open-source neural network library written in Python. It uses TensorFlow as back end. TensorFlow is a free and open-source software library for dataflow and differentiable programming across a range of tasks. It is also used for machine learning applications such as neural networks. Also, the Pandas library is used while building the model, in order to retrieve the dataset stored as a CSV file. Pandas is a software library written for data manipulation and analysis. The NumPy library is used for manipulations such as preprocessing the dataset into a 3D array. NumPy is a library adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays. The built model is saved as .sav file and is loaded into another script, for prediction.

NeuroPy is a library written in Python to connect, interact and get data from NeuroSky's MindWave EEG headset. This was used to collect the EEG data of the students in real time. The EEG headset is connected using Bluetooth, and a Python script is used to read the EEG signals at a periodic rate and pass it to algorithm for analysis.

PlayFab is a back-end platform, developed by Microsoft for building and operating live environments. It is used for creating and authenticating user accounts which is integrated to the virtual environment. It uses TCP connection-oriented communication [13]. It was integrated with Unity engine. Photon Unity networking (PUN) is a Unity package for multiuser environments. It allows frameworks for getting users into rooms where objects can be synchronised over the network. Photon is used for implementing the managing of rooms and synchronising user actions. TCP connection-oriented communication is used for managing rooms, and UDP connectionless communication is used for synchronising player actions. Photon voice is used for implementing voice communication.

The source code for our proposed system has been made available in GitHub under the open-source license ([www.github.com/nandg81/VRClassroom](https://www.github.com/nandg81/VRClassroom)). A website has been created for viewing the details and teaching/learning in the virtual classroom ([www.vrclassroom.cl.biz](http://www.vrclassroom.cl.biz)).

### **3.6 System Integration**

The ML subsystem and VR subsystem are integrated using the ZeroMQ protocol [14]. ZeroMQ is a high-performance asynchronous messaging library, aimed at use in distributed or concurrent applications. The Python script running the ML model acts as the server, and the Unity C sharp script in the VR environment acts as the client. Sockets are created on both sides, binding is done on the server side and client connects to the server. The Unity client continuously requests the server. Whenever the ML model predicts a new value, the server sends that value and the average attention value over the last 56 s to the client. The Unity script then update the value on top of the user avatar.

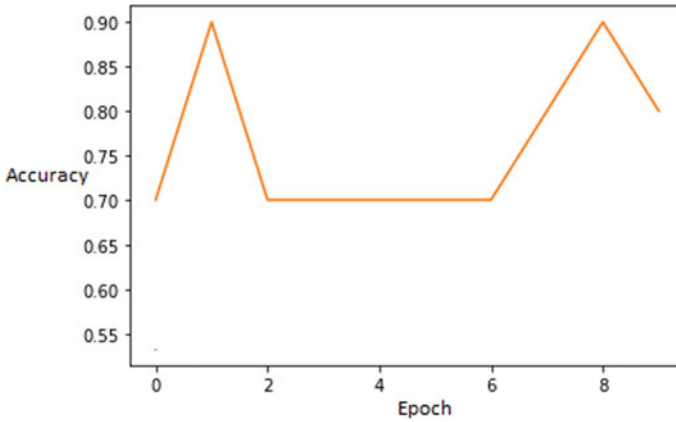
## **4 Results and Discussion**

### **4.1 ML Model Performance**

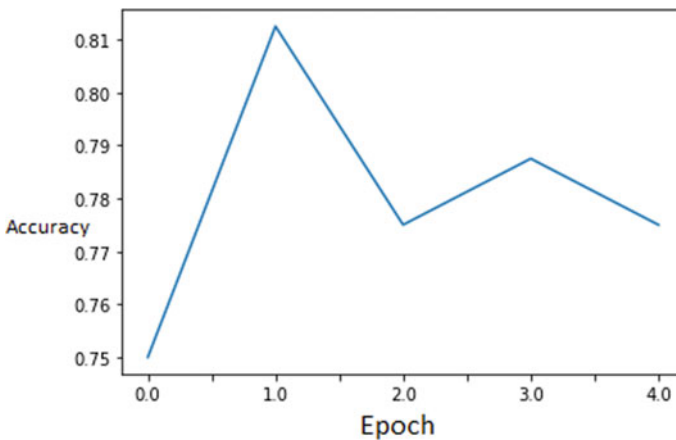
The original dataset had 100 instances. First, it was split as 90–10, 90 for the training set and 10 for the test set. An average accuracy of around 76% was obtained. Since there are dropout layers involved, random connections are dropped every time it is trained or tested. This is done so as to avoid overfitting. This has led to deviating behaviour while training, even with same data, since each time it is trained and tested, different connections are dropped randomly. Still, overall the accuracy ranges from 75–80%. The model loss was also found to decrease, as training progressed. As another test, K-fold cross-validation was applied on the same dataset. The K-fold split was 5. The average accuracy obtained was 78%. The dataset, as mentioned by the creators itself, is an extremely challenging dataset to perform binary classification. Some of the reasons are the presence of a large number of features as well the large number of rows present, just for a single data point. The execution of the model was very fast. Although the TensorFlow initialisation takes 10–20 s, predictions were made within 1–2 s (Figs. 4 and 5).

### **4.2 VR and Multi-user Subsystems Performance**

The VR environment created in Unity was running smoothly on the tested Oculus hardware and a PC with average configuration as sixth gen Intel Core i5, 4 GB RAM and a fairly good GPU. The 3D rendering of the environment was smooth and fast. The initialisation of the environment takes just 1–2 s. The speed and efficiency of the multi-user subsystem were also remarkable. For user authentication, with an average 4G Internet connection, PlayFab did the user authentication and registration with 3–6 s. Photon was able to synchronise data at a very fast rate. Changes made in one



**Fig. 4** 90–10 split model accuracy



**Fig. 5** K-fold cross-validation model accuracy

user's environment was reflected in other users' within 1–2 s. Joining and creating room were done in 3–10 s. The communication between Unity and Python was very fast. Data was transmitted within a second of prediction.

### 4.3 Discussion

A real-world test was conducted for the accuracy of the ML system. On watching a video lecture of a difficult topic, 7 out of 10 predictions classified the student as confused. Similarly, on watching a video lecture of an easy topic, 7 out 10 predictions classified the student as not confused. Thus, the system on average worked as theoritized with an average accuracy of 70%. For the virtual environment, users were

found to be more attentive than in a traditional classroom, based on the brain activity measured using EEG waves. The virtual user interface removed the need for the user to take off the VR headset until the end of the class, this increased the immersiveness. A traditional login selection system using keyboard and mouse was found to break the immersiveness.

The same topics were taken for a set of learners first in conventional form, then using the VR environment. The interest of students were captured for a longer period of time when the VR environment was used. Also the teacher was able to identify which topics were confusing due to the feedback from the neural network. These topics were taught as usual in the conventional form. But in the VR classroom, the teacher was able to identify them and put more emphasis on these topics.

## 5 Conclusion and Future Work

Virtual reality applications are increasingly being used in education now. The aim of this research work is to supplement conventional learning techniques, which are limited in scope, with the help of an interactive VR environment. The VR classroom provides this by helping students to take part in classes irrespective of their physical location. This is very important in the current context, with the spread of COVID-19 leading to many universities resorting to online classes. VR classroom can be used as a direct alternative to online video classes. It gives students and teachers a more realistic feel of a classroom and has a higher chance of increasing the concentration of students. But one of the hindrances is the high cost of VR and EEG hardware. By using Unity and VRTK, the project is compatible with most devices and will be easy to maintain and update. With the usage of EEG headsets to analyse the student's attentiveness in class, the teacher is notified if the student is not grasping concepts, and this provides an additional increment to the rate of learning. The teacher can assess whether the current method of teaching is sufficient. Due to these features, this research work would be useful in any part of the world for providing better education.

In the research work done, students and teachers are required to choose an avatar and that avatar is displayed to the other users. An improvement that can be done is to make avatars actually look like the user. By capturing photos of the user from multiple angles, modern techniques like DeepFakes can be used to display the face of the user itself on the avatar and do lip syncing. This makes the VR classroom even more closer to the real classroom. Another improvement that can be done in future is to use the VR environment for various teaching purposes, like displaying 3D models in the VR environment, demonstrating real-life processes to help the students understand better.

## References

1. A. Alam, S. Ullah, N. Ali, The effect of learning-based adaptivity on students' performance in 3D-virtual learning environments. *IEEE Access* (2017)
2. G. Tsaramirsis, S.M. Buhari, K.O. Al-Shammari, Towards simulation of the classroom learning experience: virtual reality approach. *IEEE Access* (2016)
3. C. Liao, R. Chen, Using EEG brainwaves and deep learning method for learning status classification (2018)
4. R. Robbins, M. Stonehill, Investigating the NeuroSky MindWave EEG headset. Published project report / TRL PPR; 726 (2014)
5. A. Aristidou, J. Lasenby, Y. Chrysanthou, A. Shamir, Inverse kinematics techniques in computer graphics: a survey. *Comput. Graph. Forum* (2017)
6. A. Albasri, EEG dataset of fusion relaxation and concentration moods. *Mendeley Data*, v1 (2019)
7. H. Wang, EEG brain wave for confusion. [www.kaggle.com/wanghaohan/eeg-brain-wave-for-confusion](http://www.kaggle.com/wanghaohan/eeg-brain-wave-for-confusion) (2016)
8. N.-H. Liu, C.-Y. Chiang, H.-C. Chu, Recognizing the degree of human attention using EEG signals from mobile sensors. *Sensors* (2013)
9. Z. Ni, A.C. Yuksel, X. Ni, M.I. Mandel, L. Xie, Confused or not confused? Disentangling brain activity from EEG data using bidirectional LSTM recurrent neural networks, in *ACM-BCB'17*, August 20–23, 2017
10. K. O'Shea, R. Nash, An introduction to convolutional neural networks. *ArXiv* 2015 (November 2015)
11. P.P. Patil, R. Alvares, Cross-platform application development using unity game engine. *Int. J. Adv. Res. Comput. Sci. Manage. Stud.* **3**(4) (2015)
12. L. Chen, Z. Luo, Practice and research of HTC vive controller functions in virtual reality interaction, in *Advances in Social Science. Education and Humanities Research (ASSEHR)*, vol. 93 (2017)
13. PlayFab Technical Whitepaper. <https://docs.microsoft.com/en-us/gaming/playfab/> Microsoft (2016)
14. J. Lauener, W. Sliwinski, CERN, Geneva, How to design and implement a modern communication middleware based on ZeroMQ, in *16th International Conference on Accelerator and Large Experimental Control Systems* (2017)

# Localization of Self-driving Car Using Particle Filter



Nalini C. Iyer, Akash Kulkarni, Raghavendra Shet, and U. Keerthan

**Abstract** Autonomous system or self-driving car needs to localize itself very frequently or sometimes continuously to determine its proper location that is essential to perform its navigation functionality. The probabilistic models are among the best methods for providing a real-time solution to the localization problem. Current techniques still face some issues connected to the type of representation used for the probability densities. In this paper, we attempt to localize the self-driving car using particle filter with low variance resampling. Particle filter is a recursive Bayes filter, non-parametric approach, which models the distribution by samples [1–3]. A specially modified Monte Carlo localization method is used for extracting the local features as the virtual poles [4, 5]. Simulations results demonstrate the robustness of the approach, including kidnapping of the robot’s field of view [6]. It is faster, more accurate, and less memory-intensive than earlier grid-based methods.

## 1 Introduction

Autonomous cars also known as self-driving cars have been studied and researched by many research centers, automotive companies, and other industries around the world since mid-70s. Some of the most important examples of self-driving cars research platforms are the vehicle of the University of the Bundeswehr Munich, Navlab’s mobile platform, UniBw Munich’s, and Daimler-Benz’s vehicles “VaMp” and “VITA-2.”

The recognition system of the self-driving car is responsible for estimating the state of the car and also for creating a representation of the environment around it, using data captured by on-board sensors. The decision-making system is responsible

---

N. C. Iyer · A. Kulkarni (✉) · R. Shet · U. Keerthan  
KLE Technological University, Hubballi 580031, India  
e-mail: [akash.kulkarni@kletech.ac.in](mailto:akash.kulkarni@kletech.ac.in)

N. C. Iyer  
e-mail: [nalinic@bvb.edu](mailto:nalinic@bvb.edu)



for navigating the car from its initial position to the final position as defined by the user.

In order to navigate the car throughout, the decision-making system needs to know where the car is. The localization module is responsible for estimating the car state in relation to static or offline maps of the environment. These offline maps or static maps are calculated before the autonomous operation, by using the sensors of the car. A car might use one or more different maps. We will discuss the methods of generating the landmark maps for localization in the coming sections.

The localizer module receives as input the static maps and sensors data and generates output the location and state of the car. Though GPS may help for the localization process, but it alone may not be enough for localization in urban environments due interference caused by the trees, building, tunnels, etc. The GPS uses trilateration method to locate our position. In these measurements, there may be an error from 1 to 10 m. This error is too important for the car and can potentially be fatal for the autonomous vehicle.

Localization is the step implemented in the majority of robots and vehicles to locate with a small error with respect to ground truth. A brief of the different technologies or methods used for localization as reported by various authors is summarized below. Various localization methods that do not depend on GPS have been proposed in the literature. They can be divided into three classes: LIDAR-based localization, LIDAR plus camera-based localization, and camera-based.

LIDAR-based localization methods depend solely on LIDAR sensors, which offer accuracy and easiness for processing. However, LIDAR will cost more than the camera. Industries are trying to reduce the cost of the LIDAR. In LIDAR plus camera-based localization methods, LIDAR is used to build the map, and camera is used to predict the car's position relative to the map. Camera-based localization methods are cheap and convenient, even though it is less precise and/or reliable.

Some methods rely mainly on camera data to localize self-driving cars. Brubaker proposed a localization method based on visual odometry. They use the OpenStreetMap, extracting from it all crossing and all roads connecting them in the area of interest. A recursive Bayesian filtering algorithm is used to perform inferences in the graph and the model of how the car moves as measured by the visual odometry. This algorithm is able to pinpoint the car's position in the graph by increasing the probability that the current pose lies in a point of the graph that is correlated. Ziegler describes the localization methods used by vehicle Bertha. Two complementary vision-based localization techniques were developed, named point feature-based localization (PFL) and lane feature-based localization (LFL). In PFL, the current camera image is compared with images of a sequence of camera images that is acquired previously during mapping. In LFL, the map, computed semi-automatically, provides a global geometric representation of road marking features. The current image is matched against the map by detecting and associating road marking features extracted from a camera image. Based on the above contributions proposed by the respective authors to provide optimal solutions for localization of the self-driving car, the gaps identified are failure in feature extraction through camera, and the algorithm used for robust method for localization using the predefined landmarks in the offline

maps. To overcome this issue, a particle filter algorithm is proposed with the low variance resampling of the weights.

The rest of the paper is organized as follows. The overview of the particle filter algorithm is given in Sect. 2. The proposed methodology used in localization of the self-driving car is discussed in Sect. 3. The experimental results are discussed in Sect. 4. The paper is summarized and concluded in Sect. 5.

## 2 Overview of Particle Filter

The particle filter realized as a set of mathematical equations that provides an efficient computational means to implement the Bayes filter. In a particle filter, we randomly create particles throughout, and we assign a weight to every particle. The weight of the particle represents the probability that our car is at the location of the particle. Unlike the Kalman filter [7], we have probabilities that are not continuous values but discrete values. Here, we talk about the weights.

The implementation of the algorithm is according to the following scheme. We distinguish four stages: (1) Initialization, (2) Prediction, (3) Update, (4) Re-sampling with the help of several data such as global positioning system, IMU, speeds, and measurements of the landmarks [8].

### 2.1 Initialization

We use an initial data from the GPS and add noise to it because of sensor inaccuracy. Each particle has a position  $(x, y)$  and an orientation. This gives us a particle distribution throughout the GPS area with equal weights. We are interested in predicting the state of the car at the current time  $k$ , given the initial state and all measurements  $Z^k = \{z_k, i = 1, 2, \dots, k\}$ . We will work on 3D state vector that is  $X = [x, y]^T$ , position and orientation of the car. The posterior density of the Bayesian filter problem is  $p(x_k | Z^k)$  of the current state on all measurements [9]. The probability density function is taken to represent all the knowledge we know about the state  $X_k$ , from that we can predict the current position. This density function will be multi-modal, and computing a single position is not appropriate. To localize, we need to recursively calculate the density  $p(x_k | Z^k)$  at every time step. These are done in following phases [10–13].

## 2.2 Prediction

Once particles initialized, we make a first prediction taking into account the speed and rotation of the car. In every prediction, our movements will be taken into consideration. We use the motion model to estimate the current position of the car in the form of probability density function  $p(x_k|Z^k)$ , by taking motion into consideration. The assumption here is that the current state  $X_k$  is only dependent on the previous state  $X_{k-1}$  and a given input control  $u_{k-1}$ . The predictive density over  $x_k$  is then obtained by integration.

$$p(x_k|Z^{k-1}) = \int p(x_k|x_{k-1}, u_{k-1})p(x_{k-1}|Z^{k-1})dx_{k-1}$$

## 2.3 Update

In this phase, we used measurement model to take the information from the sensors to obtain the probability density function  $p(x_k|Z^k)$ . The assumption here is that the measurement  $z_k$  is conditionally independent of earlier measurements  $Z^{k-1}$  given  $x_k$ , and the measurement model is given by  $p(z_k|x_k)$ . This term gives that the car is at location  $x_k$  given that  $z_k$  was observed. The probability density function is obtained using Bayes theorem:

$$p(x_k|Z^k) = \frac{p(z_k|x_k)p(x_k|Z^{k-1})}{p(z_k|Z^k)}$$

This process is repeated recursively. At time  $t$ , the knowledge about the initial state  $x$  is known in the form of  $p(x)$ . The initial position is given as the mean and co-variance of a Gaussian centered around  $x$ .

## 3 Proposed Methodology

The proposed methodology for the localization of the self-driving car using the particle filter includes initialization, prediction, update, and resampling as shown in Fig. 1. In sampling, step one represents the density as the probability function  $p(x_k|Z^k)$  by a set of  $N$  random samples drawn from it. Then, recursively calculate at each time-step  $k$  the set of samples  $S_k$  that is drawn from  $p(x_k|Z^k)$ .

In parallel with the formal filtering problem as explained in Sect. 2, the algorithm proceeds in two phases.

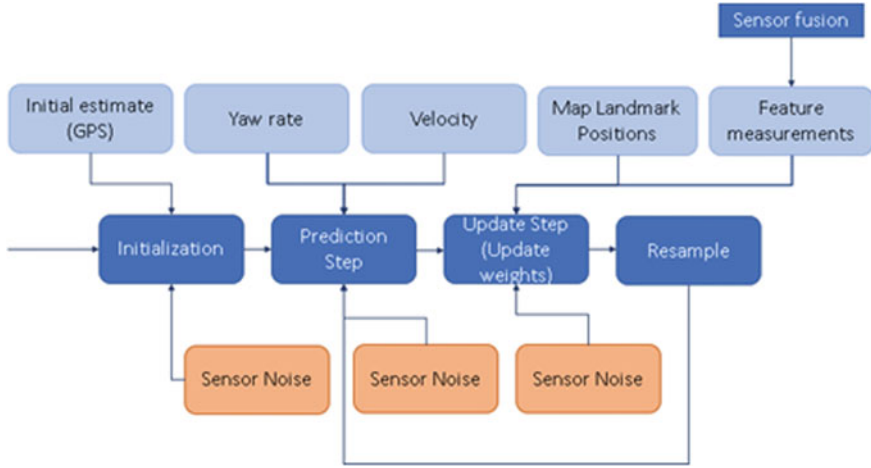


Fig. 1 Methodology

**Prediction phase:** In this phase, we have the set of particles  $S_{k-1}$  calculated in the previous iteration and apply the motion model to each particle by sampling from the probability density function [14].

A new set  $[s_k^i]$  is obtained that approximates a random sample from the empirical predictive density  $[p(x_k|Z^{k-1})]$ .

$$p(x_k|Z^{k-1}) = \sum_{i=1}^N p(x_k|S_{k-1}^i, u_{k-1})$$

This describes a density approximation to  $[p(x_k|Z^{k-1})]$  consisting of equally weighted combination of  $p(x_k|S_{k-1}^i, u_{k-1})$  per sample  $[s_{k-1}^i]$ . From stratified sampling, we draw one sample  $[s_k^i]$  from each of the  $N$  to obtain  $[s_k^i]$ .

Motion model equations:

$$x_f = x_0 + \frac{v}{o} [\sin(\theta_0 + o(dt)) - \sin(\theta_0)]$$

$$y_f = y + \frac{v}{o} [\cos(\theta_0) - \cos(\theta_0 + o(dt))]$$

$$\theta_f = \theta_0 + o(dt)$$

**Update phase:** In this phase, we have the measurements  $[z_k]$  and weights each of the samples in  $[s_k^i]$  by the weight  $[m_k^i = p(z_k/s_k^i)]$  that is the likelihood of  $s_k^i$  given  $[z_k]$ .

Then, calculate by resampling from this set for  $[j = 1, 2, 3 \dots N]$  draw one  $[S_k]$ . sample  $[s_k^j]$  from  $[\{s_k^i, m_k^i\}]$ .

We use the measurement model from the prediction phase and sample from the empirical posterior density:

$$\hat{P}(x_k/Z^k) \propto p(z_k/x_k) \hat{P}(x_k/Z^{k-1})$$

**Resampling phase:** To accomplish the above phase, we use a statistics technique called importance sampling. In a correction action, each of the sample is then updated with other weights by attaching the importance weight  $[w = p(x)/f(x)]$ .

We sample from  $[p(x) = \hat{p}(x_k/Z^k)]$  and take the importance function  $[f(x) = \hat{p}(x_k/Z^{k-1})]$ . In this phase, it selects the particles with the higher probability samples that have the high likelihood associated with them, and a new set is obtained that approximates a random sample from. This resampling algorithm performs efficiently in  $O(N)$  time [15, 16].

We re-weight the obtained samples by:

$$m_k^i = \frac{g(x)}{f(x)} = \frac{p(z_k|x_k) \hat{p}(x_k|Z^{k-1})}{\hat{p}(z_k|Z^{k-1})} = p(z_k|x_k)$$

Subsequent resampling of the samples is required to convert the non-random samples back into a set of equally weighted samples:

$$S_k = \{s_k^i\}$$

The steps are repeated recursively. To initialize, we start at time  $k = 0$  with a random sample from the previous values [17, 18].

## 4 Experimental Results and Discussion

The particle filter algorithm for the localization of the self-driving car with the resampling method was implemented, and the experimental results obtained are discussed in this section. The algorithm is tested for the cases where there is curvature as well as the linear motion of the car with the dataset of the known GPS location of the poles on the virtual simulation platform are discussed in detail.

**Test case 1: Straight Line Path:** The location of the poles with the actual GPS location is marked on the virtual simulation platform which is shown in the below diagram. The map containing the landmarks, initial location of the car with a big uncertainty, noisy landmarks observation while the vehicle is moving which is provided as the input to the filter. In this test case, the vehicle is heading in the map in a straight path

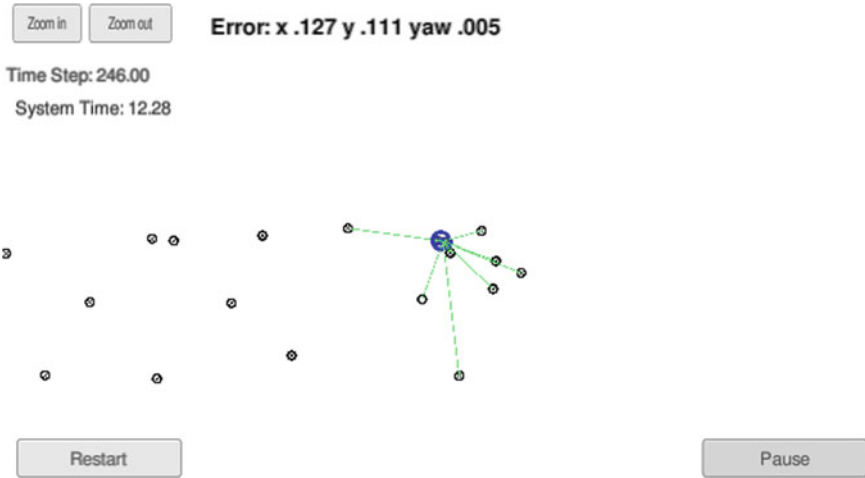


Fig. 2 Test case for straight path

without the significant change in the heading angle. The maximum error between the actual path and the predicted path is less than 1 m, i.e., 0.75 m (Fig. 2).

**Test case 2:** Curved Path: One of the key advantages of the particle filter algorithm with the resampling methods over the Kalman-filter-based approaches is the ability to represent the multi-model probability distribution. Where there will be nonlinear data, i.e., the path where the vehicle is heading is through the curved path, and there is a nonlinear data from the observation model as well as the motion model from the algorithm. In this test case, the vehicle is heading in the map in curved path with a varying change in the heading angle. The maximum error between the actual path and the predicted path is less than 1 m, i.e., 0.83 m (Fig. 3).

**Test case 3:** Entire Scenario: Figure 4 presents the entire scenario or the environment where the car will travel for testing of the particle filter. The particle filter posed a better accuracy and reduced the GPS error from 5 to 10 m to an error rate less than 1 m.

## 5 Conclusion

In this work, we introduced an approach to localization of self-driving car, the particle filter also known as Monte Carlo localization method. Instead of approximating the probability density function, we represent the samples randomly drawn from it. By using the resampling type methods, we have combined the advantages of grid-based Markov localization with the efficiency and accuracy of Kalman-filter-based approaches. The algorithm is able to deal with ambiguities and thus can globally

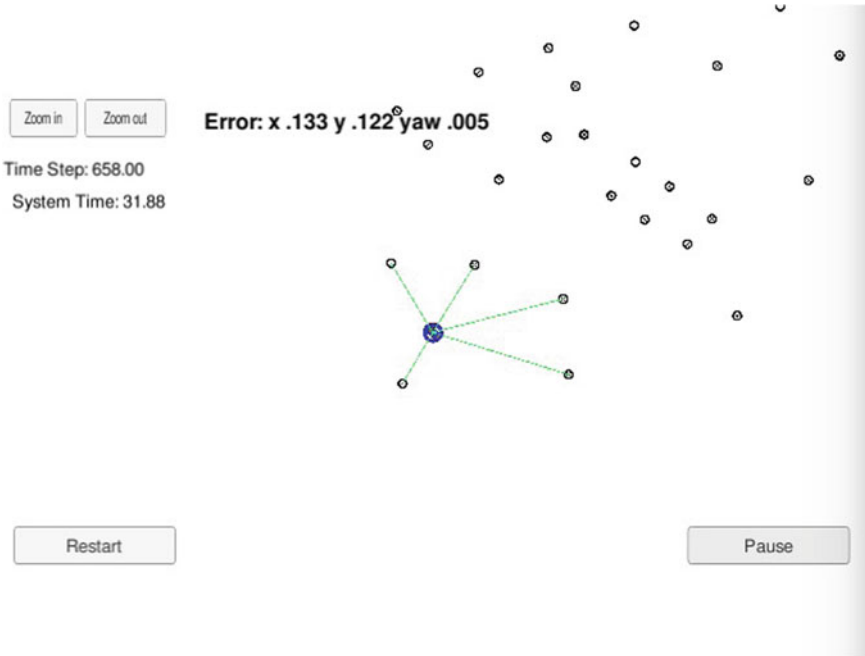


Fig. 3 Test case for curved path

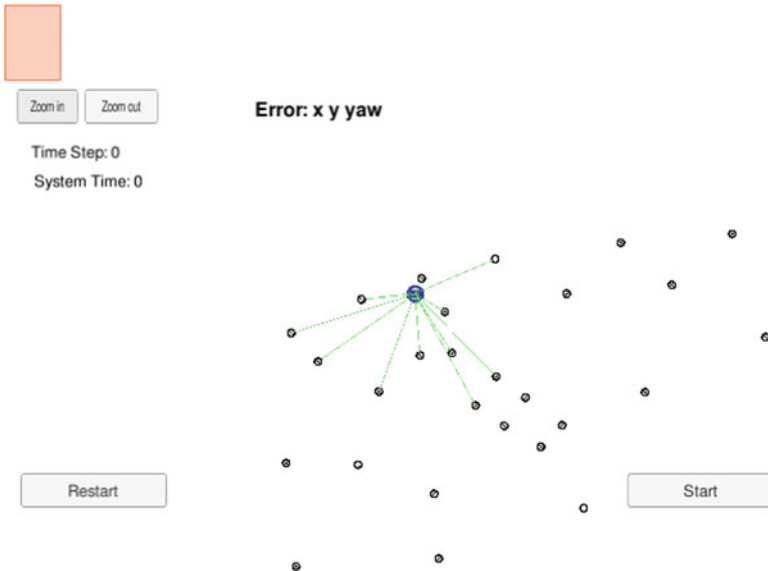


Fig. 4 Entire map

localize a car along with a nonlinear data. By concentrating on the resources and cost on the relevant parts of the state space, this method can efficiently and accurately estimate the position of the vehicle. When compared to other methods, this approach has significantly reduces memory requirements while at the same time data acquisition in real time is considerably higher frequency. Even though this algorithm gives the promising results with the current resampling methods, there are still drawbacks where we can work on, one potential issue with the specific algorithm is that the step used in the resampling step, where the weight with the higher value will be selected multiple times, resulting in the loss of diversity.

## References

1. J.J. Leonard, H.F. Durrant-Whyte, *Directed Sonar Sensing for Mobile Robot Navigation* (Kluwer Academic, Boston, 1992).
2. R.S. Bucy, Bayes theorem and digital realization for nonlinear filters. *J. Astronaut. Sci.*
3. A.F.M. Bucy, A.E. Gelfand, Bayesian statistics without tears: a sampling-resampling perspective. *Am. Stat.* (1992)
4. J.E. Handschin, Monte Carlo techniques for prediction and filtering of non-linear stochastic processes. *Automatica* (1970)
5. G. Kitagawa, Monte Carlo filter and smoother for non-Gaussian nonlinear state space models. *J. Comput. Graph. Stat.* (1996)
6. I. Nourbakhsh, R. Powers, S. Birchfield, DERVISH an office navigating robot. *All Mag.* **16**, 53–60 (Summer 1995)
7. R.E. Kalman, A new approach to linear filtering and prediction problems. *J. Basic Eng.* **82**(1), 35–45 (1960)
8. R. Simmons, S. Koenig, Probabilistic robot navigation in partially observable environments. in *Proceedings of the International Joint Conference on Artificial Intelligence* (1995)
9. M. Isard, A. Blake, *Stochastic Models, Estimation and Control*, vol. 1 (Academic, New York, 1979).
10. A. Doucet, *On Sequential Simulation-Based Methods for Bayesian Filtering* (Department of Engineering, University of Cambridge, 1998)
11. J. Carpenter, P. Clifford, P. Fernhead, *An Improved Particle Filter for Non-Linear Problems* (Department of Statistics, University of Oxford, 1997)
12. M.K. Pitt, N. Shephard, *Filtering via Simulation: Auxiliary Particle Filters* (Department of Mathematics, Imperial College, London, 1997).
13. M. Isard, A. Blake, A mixed-state Condensation tracker with automatic model-switching. in *European Conference on Computer Vision* (1997)
14. L.P. Kaelbling, A.R. Cassandra, J.A. Kuerien, Acting under uncertainty: discrete Bayesian models for mobile-robot navigation. in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems* (1996)
15. S. Thrun, Bayesian landmark learning for mobile robot localization. *Mach. Learn.* (1998)
16. N.J. Gordon, D.J. Salmond A.F.M. Smith, Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proc.* (1993)
17. M. Isard, A. Blake, Contour tracking by stochastic propagation of conditional density. in *European Conference on Computer Vision* (1996)
18. M. Isard, A. Blake, Condensation—conditional density propagation for visual tracking. *Int. J. Comput. Vis.* (1998)



# Convex Combination of Maximum Versoria Criterion-Based Adaptive Filtering Algorithm for Impulsive Environment



S. Radhika, A. Chandrasekar, and K. Ishwarya Rajalakshmi

**Abstract** This paper elaborates convex combination approach of two maximum versoria criteria-based adaptive filters for impulsive environment. The maximum versoria criteria-based adaptive filter performs better than minimum mean square error and maximum correntropy criteria under impulsive environment. The main drawback with the current approach is that there is trade-off in the speed of convergence and steady-state mean square error. In order to overcome this trade-off, convex combination method is adopted in this paper. A new update rule is also proposed to make the algorithm to have more robustness. Experiments were conducted for echo cancellation and system identification applications to validate the performance improvement of the proposed approach.

**Keywords** Adaptive filter · Maximum versoria criterion · Convex combination · Impulsive environment · Convergence · Robustness

## 1 Introduction

Means square error (MSE)-based adaptive filters are found to be suitable for Gaussian models and linearity assumption [1]. These algorithms do not perform well

---

S. Radhika (✉)

Department of Electrical and Electronics Engineering, School of Electrical and Electronics Engineering, Sathyabama Institute of Science and Technology, Chennai, India  
e-mail: [radhika.eee@sathyabama.ac.in](mailto:radhika.eee@sathyabama.ac.in)

A. Chandrasekar

Department of Computer Science Engineering, St. Joseph's College of Engineering, Chennai, India  
e-mail: [drchandrucse@gmail.com](mailto:drchandrucse@gmail.com)

K. I. Rajalakshmi

Department of Electronics and Communication Engineering, St. Joseph's College of Engineering, Chennai, India  
e-mail: [Irkirk99@gmail.com](mailto:Irkirk99@gmail.com)

under non-Gaussian and nonlinear assumptions. Generally, lower-order moment-based adaptive filters are used for situation where heavy tailed (Laplace, Alpha-stable) interferences occur. Some of the well-known examples are the family of sign algorithms (SA) [2] and mixed norm algorithms [3]. On the other hand when the interference is said to be made of light tailed noise like uniform binary distribution, higher-order moment-based algorithms such as the least mean fourth (LMF) is used [3].

Recently, information theory criteria-based adaptive filters enjoy robustness against non-Gaussian impulsive interference. The well-known algorithm is the maximum correntropy criterion (MCC) algorithm and its variants [4, 5]. The maximum versoria criterion (MVC) proposed in [6] indicates that they have better performance than MCC as well as reduced computational complexity. The performance analysis made in [6] indicates that it mainly depends on the radius of the circle that generates versoria. Also it is found from the simulation results that lower value of steady-state MSE can be obtained with radius chosen as smaller at the cost of lesser convergence speed and vice versa. Therefore, it is required to remove this trade-off in the performance of MCC algorithm.

It is known from the literatures that convex combination is found to be more suitable method to improve the overall performance of algorithms [7–10]. Thus, in this paper, an attempt has been made to propose a robust adaptive filter by convex combination of standard MVC based filter with complementary values of radius of the generating circle of versoria in order to obtain good convergence speed together with lesser steady-state MSE. Therefore, the main objective of the paper is as follows: (1) To propose a novel robust adaptive filter based on convex combination approach which can combine the best feature of the combining filter so as to remove the trade-off in the performance. (2) To propose the logarithmic function of error-based mixing strategy so as to maintain the robustness as the mixing parameter based on MSE will not be suitable for non-Gaussian interference. (3) A new weight transfer method is also proposed to further improve the performance. The remaining of the paper is distributed as follows. Section 2 describes the maximum versoria criteria, and the proposed method is introduced in Sect. 3. In Sect. 4, the simulations are performed to validate the proposed approach. Conclusions are made in Sect. 5.

## 2 Maximum Versoria Criterion-Based Adaptive Filter

Let us consider the problem of identification of unknown system. Using linear regression model, the desired signal  $d_n$  is given as:

$$d_n = \mathbf{x}_n^T \mathbf{w}_o + v_n \quad (1)$$

where  $w_o$  is the unknown optimal weight.  $x_n$  denotes the input given by  $\mathbf{x}_n = [x_n, x_{n-1}, x_{n-2}, \dots, x_{n-N+1}]$ . Let  $v_n$  be the noise term which is constituted by both impulsive and background noise. If the estimated output is defined as  $y_n = \mathbf{x}_n^T \mathbf{w}_n$ , then

the error signal is  $e_n = d_n - y_n$  where  $w_n$  is the estimated weights. Let  $n$  be the time index. The cost function for MSE based algorithm is given by:

$$f(e_n) = E[e_n^2] \quad (2)$$

The cost function MVC algorithm [6] is expressed as:

$$f(e_n) = \frac{e_n}{((1 + \tau|e_n|)^p)} \quad (3)$$

Here  $E$  is the expectation operator, and  $\tau$  is given by  $(\frac{1}{2a})^p$  where ‘ $a$ ’ is the radius of circle that generates versoria function and  $p$  represents is the shape. The original versoria function is obtained when  $p = 2$  using stochastic gradient descent rule, the update weight recursion of the standard MVC algorithm is written as:

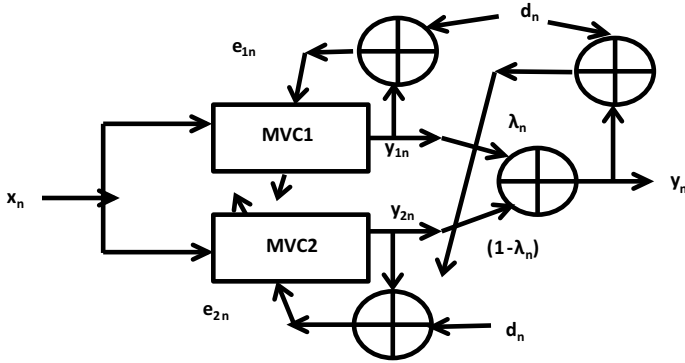
$$w_{n+1} = w_n + \mu \frac{e_n}{((1 + \tau e_n^2)^2)} x_n \quad (4)$$

where  $\mu$  is step size. From [6], it is found that even though the standard MVC is robust against impulse interference than MSE based algorithm, it still suffers from trade-off in performance problem like MCC algorithm. An adaptive MVC algorithm in which the adaptive algorithm based on changing values of ‘ $a$ ’ is proposed in [6]. The simulations indicates that when the value of ‘ $a$ ’ is selected as small value, then the steady-state MSE is found to be lower at the cost of slower convergence rate, and larger value of ‘ $a$ ’ gives higher value of steady-state MSE with greater speed of convergence rate. This conflicting trade-off between greater convergence speed and lower steady-state MSE is the major bottleneck of the MVC based adaptive algorithm.

### 3 Proposed Convex Combination Approach

The convex combination approach is found to be providing promising results in several adaptive filters whenever trade-off in performance occurs [7]. The convex combination scheme is given in Fig. 1. Motivated by the convex combination approach, we propose to combine two MVC based adaptive algorithms with different values of ‘ $a$ ’ using convex combiner. The first component MVC adaptive filter called MVC<sub>1</sub> is made to operate with larger value of radius of generating circle called  $a_1$ , and the second one called MVC<sub>2</sub> operates with smaller value of radius called  $a_2$ . Thus, the update recursion of component 1 filter is given by:

$$w_{1n+1} = w_{1n} + \mu \frac{e_{1n}}{((1 + \tau_1 e_{1n}^2)^2)} x_n \quad (5)$$



**Fig. 1** Convex combination of two MVC adaptive filters

Similarly, the recursive equation of component 2 filter is given by:

$$w_{2n+1} = w_{2n} + \mu \frac{e_{2n}}{((1 + \tau_2 e_{2n}^2)^2)} x_n \tag{6}$$

The output of the combined filter is as follows:

$$y_n = \lambda_n y_{1n} + (1 - \lambda_n) y_{2n} \tag{7}$$

where  $\lambda_n$  denotes the mixing parameter. The overall weight of the final filter is given by:

$$w_n = \lambda_n w_{1n} + (1 - \lambda_n) w_{2n} \tag{8}$$

The overall final output error is thus given as:

$$e_n = \lambda_n e_{1n} + (1 - \lambda_n) e_{2n} \tag{9}$$

In order to remove trade-off, it is required to make the mixing parameter nearer to 1 at the initial stage and then nearer to 0 when the algorithm reaches the final stage. The generally used update of mixing parameter is by use of sigmoidal function [8] which is written as:

$$\lambda_n = \text{sgm}(\alpha_n) = \frac{1}{(1 + e^{-\alpha_n})} \tag{10}$$

If the updation of is based on MSE, then it is not able to work in impulsive environment; hence, the adaptive rule is modified using versoria function and using stochastic positive gradient approach which is given by:

$$\alpha_{n+1} = \alpha_n + \mu_n \frac{\partial}{\partial \alpha_n} [\ln(1 + \tau e_n^2)] = \alpha_n + \mu_n \lambda_n (1 - \lambda_n) (y_{1n} - y_{2n}) \frac{e_n}{(1 + \tau e_n^2)} \quad (11)$$

where  $\mu_n$  represents the step size. In order to prevent the very slow adaption near the extremes, the value of  $\alpha_n$  is restricted in the range  $[-4, +4]$ . Further, this can be further enhanced by the use of weight transfer rule. Thus, the modified weight transfer rule is given by

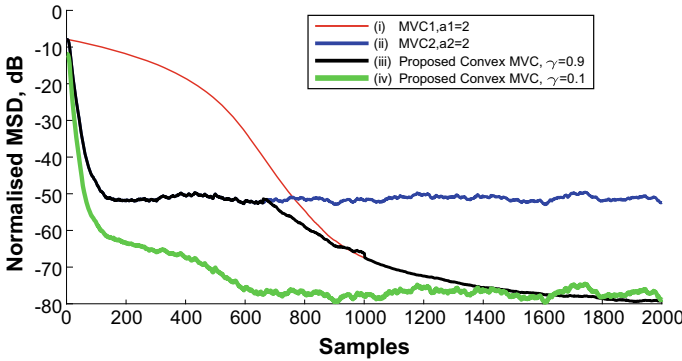
$$w_{2n} = \gamma_n w_{2n} + (1 - \gamma_n) w_{1n} + \mu \frac{e_{2n}}{(1 + \tau_2 e_{2n}^2)} x_n \quad (12)$$

## 4 Simulation Results

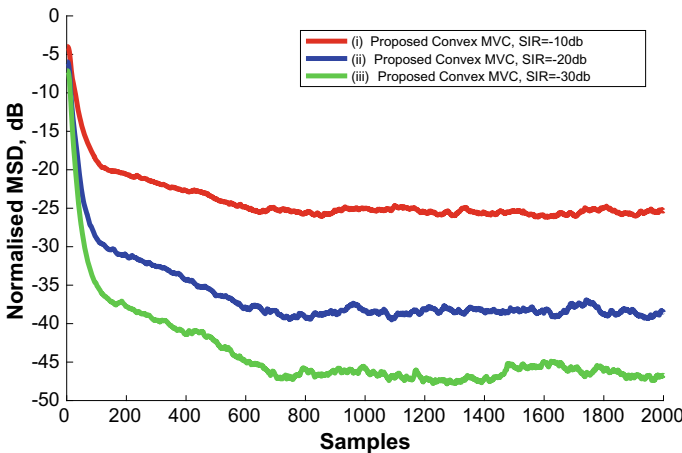
Here, the experiments were conducted to validate the improvement in performance of the proposed convex combination approach. The simulations are performed in context of unknown plant identification problem. For this purpose, the randomly generated filter coefficients of 16 sequence length are chosen. Let us adopt that both the unknown system and filter are made of same number of filter coefficients. The input is Gaussian with unity variance and null mean. The background noise is also said to be Gaussian so as to obtain the required signal-to-noise ratio (SNR). The impulsive noise is generated by using relation  $bn * Bn$  where  $bn$  is Bernoulli trail with a probability of success as  $pr$ .  $Bn$  is zero mean white Gaussian noise which is selected so as to generate different signal-to-interference ratio (SIR). The metric used to evaluate the performance is normalized MSD (NMSD) given by  $NMSD = 20 \log_{10}(\tilde{w}\tilde{n}(n)_2 / W_{02})$ . All the simulation results are obtained by ensemble averaged over 100 independent runs.

The first experiment is conducted to prove the improvement in the convergence speed and proposed convex combination approach. For this, two different values of  $a$  are considered,  $a_1$  is chosen to be 20 and  $a_2$  as 2. The step size for both the filters is selected as 0.01, and SNR is fixed as 30 db. The SIR is selected as  $-30$  db. Figure 2 illustrates the performance of the proposed scheme. It can be seen from Fig. 2 that when the algorithm is in the initial stages, it has faster convergence speed, and while the algorithm is in the steady state, it achieves lower steady-state MSE. Also the proposed weight transfer scheme achieves the desired response. Thus, the proposed scheme achieves the desired performance by combining the best performance of the component filters.

The next work is the evaluation of proposed approach for different values of SIR which are set as  $-10$  db,  $-20$  db, and  $-30$  db, respectively. Figure 3 illustrates the performance. As seen from Fig. 3, it is found that the proposed scheme is robust against impulsive interference like standard MVC, and additional advantage is claimed that propose scheme performs better than MVC in terms of achieving both



**Fig. 2** Performance of the proposed convex combination approach with SNR = 30 db, SIR = -30 db, and step size = 0.01



**Fig. 3** Convex combination approach for different values of SIR with SNR = 30 db, step size = 0.01, and  $\rho_r = 0.01$

faster convergence and lower steady state for different values of impulsive interference. The same experiment is repeated for different values of  $\rho_r$  as shown in Fig. 4 and for different values of SNR as shown in Fig. 5.

Thus, from Figs. 4 and 5, it can be concluded that the algorithm can outperform its component filters at all environment conditions. The performance of the combiner is illustrated in Fig. 6. As seen from Fig. 6, it can be concluded that the mixing parameter initially works with maximum value so as to work similar to MVC1 and as it reaches steady state and value of mixing parameter is nearer to 0 and the proposed scheme works similar to MVC2 as desired.

Next for echo cancellation application, a system with 512 filter coefficient is taken, and the performance of the proposed is evaluated as shown in Fig. 7. The

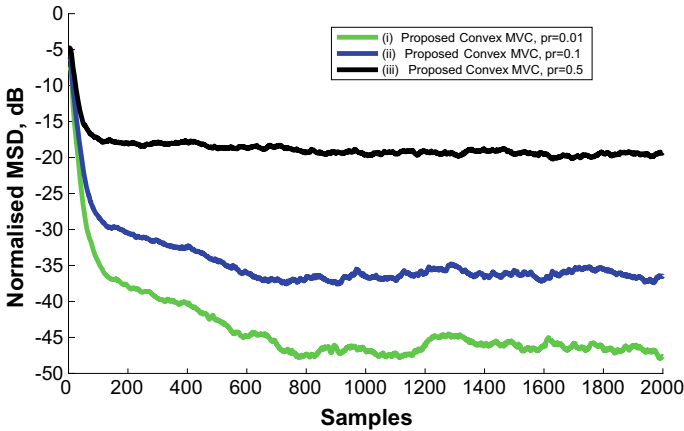


Fig. 4 Evaluation of proposed convex combination approach for different values of pr with SNR = 30 db, SIR = -30 db, and step size = 0.01

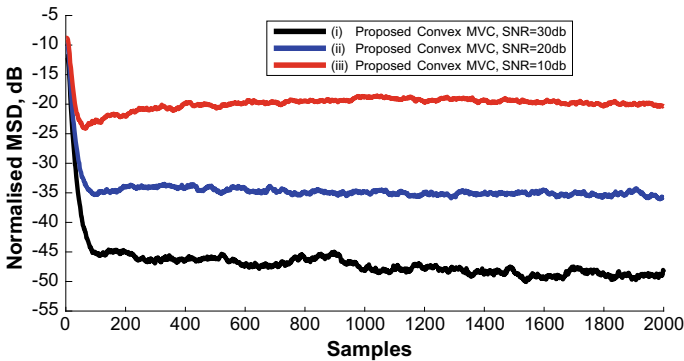
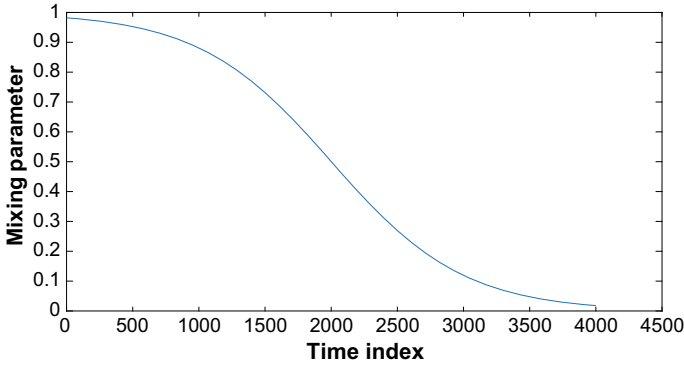


Fig. 5 Proposed convex combination approach for different values of SNR with SNR = -30 db, step size = 0.01

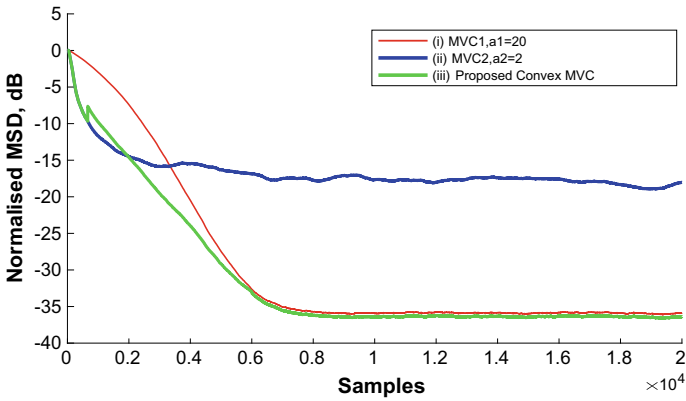
weight transfer method with  $\gamma_n$  is chosen as 0.7, the step size is selected as 0.05, and all other parameters are same as first experiment. Thus, from the simulation results, it is proved that the proposed convex combination approach is suitable for echo cancellation approach.

### 5 Conclusion

Convex combination of two MVC adaptive filter of complementary nature is proposed in this paper. A new update rule is also proposed. Further, a new weight



**Fig. 6** Time evolution of mixing parameter



**Fig. 7** Convex combination of MVC for echo cancellation application

transfer rule is also proposed to obtain good tracking nature. Finally, simulation in the context of unknown plant identification problem proves that the proposed scheme can able to achieve good performance in impulsive environment. Further, this work can be extended for other type of adaptive algorithms.

## References

1. S.S. Haykin, *Adaptive Filter Theory* (Pearson Education, India, 2008).
2. S. Radhika, A. Sivabalan, Steady-state analysis of sparsity-aware affine projection sign algorithm for impulsive environment. *Circ. Syst. Signal Process.* 1–14 (2016)
3. E. Walach, B. Widrow, The least mean fourth (LMF) adaptive algorithm and its family. *IEEE Trans. Inf. Theory* **30**(2), 275–283 (1984)



4. B. Chen et al., Steady-state mean-square error analysis for adaptive filtering under the maximum correntropy criterion. *IEEE Signal Process. Lett.* **21**(7), 880–884 (2014)
5. S. Radhika, A. Sivabalan, Variable step size maximum correntropy criteria based adaptive filtering algorithm. *Eng. Technol. Appl. Sci. Res.* **6**(2), 923–926 (2016)
6. F. Huang, J. Zhang, S. Zhang, Maximum versoria criterion-based robust adaptive filtering algorithm. *IEEE Trans. Circuits Syst. II Express Briefs* **64**(10), 1252–1256 (2017)
7. J. Arenas-García et al., Mean-square performance of a convex combination of two adaptive filters. *IEEE Trans. Signal Process.* **54**(3), 1078–1090 (2006)
8. L. Shi, Y. Lin, Convex combination of adaptive filters under the maximum correntropy criterion in impulsive interference. *IEEE Signal Process. Lett.* **21**(11), 1385–1388 (2014)
9. S. Radhika, A. Sivabalan, Steady state mean square analysis of convex combination of ZA-APA and APA for acoustic echo cancellation. *Intell. Syst. Technol. Appl.* 437–446 (2016)
10. W. Wu, Z. Liang, Y. Bai, W. Li, Multi-convex combination adaptive filtering algorithm based on maximum versoria criterion (workshop). in *International Conference on Communications and Networking in China* (Springer, Cham, 2019), pp. 297–306

# Verifying Mixed Signal ASIC Using SVM



H. R. Aishwaraya, Saroja V. Siddamal , Aishwaraya Shetty,  
and Prateeksha Raikar

**Abstract** KLEEL2020 is an in-house developed event logger. The ASIC is implemented in TSMC 0.18  $\mu\text{m}$  CMOS mixed signal technology, 3.3/1.8 V. The focus of this paper is to achieve functional precision of the design before the tape-out. The process of verification is critical stage in the design flow because any bug not detected at earlier stage will lead to overall failure of the design process. In this paper, the authors present a framework for the complete verification of KLEEL2020 using System Verilog Methodology (SVM). The proposed SV environment allows I2C protocol as communication means with DUT. Different test scenarios are developed, and reused to verify the ASIC. Event logger is verified for various test cases. This verification attempt helped identify 05 RTL bugs in the design.

**Keywords** SVM · Verification · Event logger · Testbench

## 1 Introduction

Event recorder-KLEEL2020 is the test ASIC designed and developed by team of students and faculty of KLE Technological University. This ASIC has similar features of maximum DS1683. The DS1683 [1] is an elapsed time recorder. Based on the time, the EVENT pin is held high, at the falling edge, the time is tracked and accumulated. The application is to track the power cycle of the device. A similar lower version maximum DS 1682 [2] identifies and records events. It also calculates the total

---

H. R. Aishwaraya · S. V. Siddamal (✉) · A. Shetty · P. Raikar  
KLE Technological University, Hubballi, India  
e-mail: [sarojavs@kletech.ac.in](mailto:sarojavs@kletech.ac.in)

H. R. Aishwaraya  
e-mail: [aishwarayahr26@gmail.com](mailto:aishwarayahr26@gmail.com)

A. Shetty  
e-mail: [aishvshetty@gmail.com](mailto:aishvshetty@gmail.com)

P. Raikar  
e-mail: [prateeksharaikar13@gmail.com](mailto:prateeksharaikar13@gmail.com)

collective event time since it was last reset to 0. The authors in paper [3] have designed a test ASIC NKETC2019 low power of 250 ns using on-chip oscillator. The ASIC finds its application as elapsed time counter. The counters count at every 1 s. The oscillator generates a frequency of 32 kHz which is further downscaled to 1 Hz to run the elapsed time counters. To set up the complete flow from design to silicon, the university has introduced verification course System Verilog. To understand the verification process, the authors use Hardware Verification Language, System Verilog to verify the design. There are more advanced HVL like UVM, OVM, etc., but authors have used SVM as it is the first learning stage of verification and due to its benefit as it allows the user to construct trustworthy, repeatable verification environments.

To determine the precision of the Design under Test (DUT) a testbench is built in System Verilog. This is done first by generating the stimulus, pertain stimulus to the DUT, capture the response, check for correctness. Most of the previous work focuses on UVM. The books [4, 5] explain System Verilog (SV). SV provides a complete verification environment, with coverage, assertion constrained random test case generation. Open Verification Methodology (OVM) for functional verification is explained in [6, 7]. This verification is used for complex designs which are the latest verification methodology. The authors in [8] have built structures for verification environment which meets all the requirements of SoC. A standardized testbench is developed which maintains consistency and maintains the quality gap of testbenches. The authors [9] in their patent describe verification method for IC containing more the one core. The authors claim their verification methodology reduces software required and time to incorporate the software. In [10], verification of SoC and writing SoC testbenches is discussed. The authors discuss about simulation-based verification. SoC verification requires integration of blocks. The bugs are due to poor integration of various blocks. Reusing IP verification for SoC verification is discussed in [11]. IP verification and IP testbenches are designed using UVM. The authors claim to reduce the verification time by two times, and resources are reduced by twice.

The rest of paper is organized as follows. Overview of KLEEL2020 is presented in Sect. 2. Section 3 provides the detailed proposed SVM architecture for the core. Coverage and assertion are discussed in Sect. 4. Test cases and issues identified are discussed in Sect. 5. Conclusion of this work is discussed in Sect. 6.

## 2 Overview of KLEEL2020-Event Logger

The architecture of the proposed event logger KLEEL2020 is as shown in Fig. 1. KLEEL2020 keeps log of events including duration and number of events occurred. Information is available to the outside world through I2C interface.

This product runs on single supply of 3.3 V and internally generates 1.8 V using on-chip regulator for the usage of core circuits. The main objective of this work is the verification of the KLEEL2020, the availability of analog blocks like oscillator, LDO and POR would be irrelevant. Verification architecture in Fig. 2 shows all subsystems that are interfaced for verifying the event logger. I2C master is used

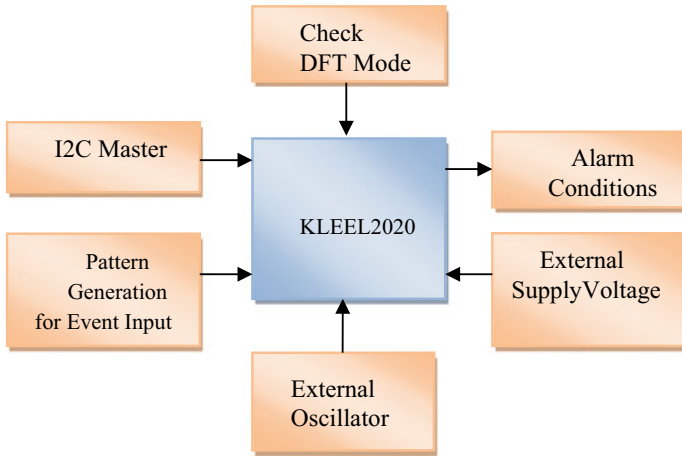


Fig. 1 Verification architecture

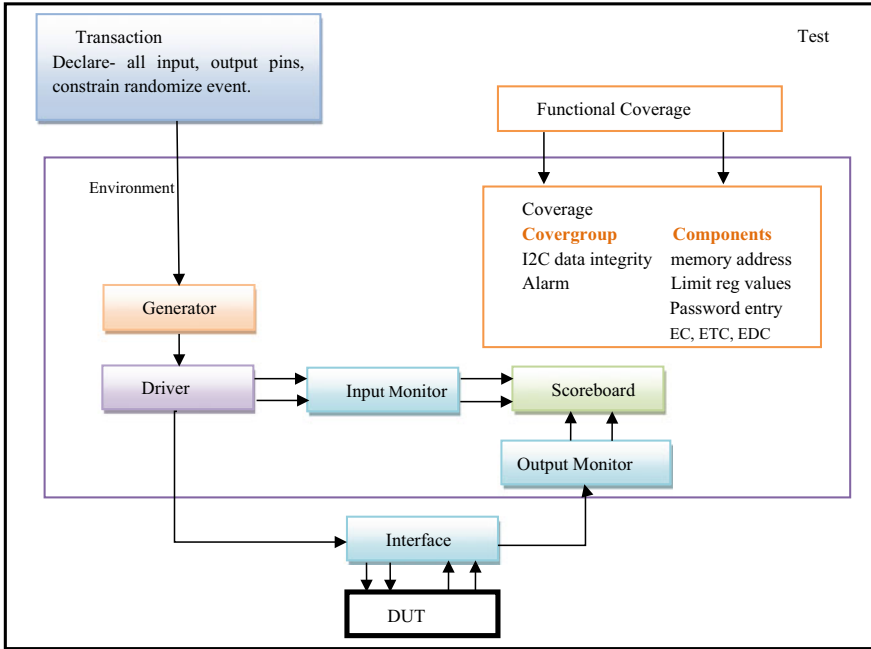


Fig. 2 SVM layered testbench architecture for KLEEL2020

for communication with slave event logger, pattern generation for event input and generation of external clocks (which are generated by oscillator internally in the design). Testing the alarm conditions and verifying the design in functional mode.

### 3 Proposed SVM Verification for KLEEL2020

SVM layered testbench architecture is shown in Fig. 2. Brief explanation for each component will be discussed in this section.

- **VM Virtual Interface:** The Device under Test (DUT) is connected to testbench through interface. The design uses a single interface. The interface is a group comprising input–output, event, data to SDA.
- **SVM Agents:** As the design is simple, one agent is used to inject stimuli’s to DUT. The agent is active agent which receives the stimuli and gives it to interface.
- **SVM Monitor:** Two monitors are used for every agent, one for monitoring input stimuli to DUT and second for output response from DUT. The data from the monitor is further sent to scoreboard.
- **SVM Drivers:** Each agent has one driver. The generator sends the stimuli to the driver and forces the information to interface and monitor. The driver communicates to DUT via interface.
- **SVM Transaction:** Transaction holds all the data that is used in driving DUT. All the received inputs, outputs and their randomization, events data to SDA are declared in transaction.
- **SVM Scoreboard:** Scoreboard present in the environment collects the transaction from monitor through driver. The scoreboard evaluates the way DUT does evaluation. The evaluated output from scoreboard is compared with output from DUT.

### 4 Coverage and Assertion

The coverage metric is developed in parallel with environment [12]. The model has code and functional coverage modules. In code coverage, each line in the DUT is checked if it is exercised at least once. This is done to verify that all test cases reach the blocks. Functional coverage checks the functionality covered. This checks the DUT is free from all hazards. The test cases have covered most of the functionality. The code coverage results are given in Table 1.

**Table 1** Code coverage results

Block (%)	State (%)	Toggle (%)	Transition (%)	Assertion (%)
98.4	97.5	94.87	100	100

## 5 Test Cases Passed and Design Issues in KLEEL2020

Various test cases were verified. Few design issues were found in the core by running test cases on KLEEL2020 testbench. Test cases are summarized in Table 2. Few

**Table 2** Test cases passed

Sl. No.	Test case	Description	Test result
<i>POR</i>			
1	Reset	When POR goes low, even though the event is high, there is no alarm	Pass
2	SE	When scan enable is high, the design does not function in functional mode	Pass
3	Event HIGH	if event pin is high for 7 positive edges	Pass
4	Event LOW	if event pin is low for 7 positive edges	Pass
5	Initialize I2C slave controller	The slave controller should stay in idle state and wait for master communication	Pass
6	Check READ/WRITE sent by master	Based on the W/R bit sent by slave controller should read or write	Pass
7	ED, ETC, EC limit set	Send ED, ETC, EC limit to comparator	Pass
8	Set latch	The latch for the alarm is set(default low)	Pass
9	Config [1] == 1	ETC alarm enable set	Pass
10	Config [2] == 1	ED alarm enable set	Pass
11	Config [3] == 1	Alarm polarity set (1–alarm output high, 0–alarm output low)	
12	Password comparator generates write access signal	The password comparator sets the write access only if correct password is provided, [through I2C slave controller]	Pass
13	Limit register update	If the limit values are updated, then the values must be sent to comparator	Pass
14	Updated ED limit set	Send updated ED limit to comparator	Pass
15	Updated ETC limit set	Send updated ETC limit to comparator	Pass
16	Updated Evt limit set	Send updated Evt limit to comparator	Pass

(continued)

**Table 2** (continued)

Sl. No.	Test case	Description	Test result
17	Updated Overflow limit set	Send updated overflow limit to comparator	Pass
18	Simultaneously send data to SRAM	While the event pin is high, the count values are written into SRAM simultaneously	Pass
<i>Alarm</i>			
19	Based on the configuration register bits and comparator output flags	If ((E_alarm_en && E_alarm_flg == 1) && (alarm_pol == 1)) If ((ET_alarm_en && ET_alarm_flg == 1) && (alarm_pol == 1)) If ((ED_alarm_en && ED_alarm_flg == 1) && (alarm_pol == 1))—high If ((ED_alarm_en && ED_alarm_flg == 1) && (alarm_pol == 0))—low	Pass
20	ED alarm flag set	The flag is set high if ED count is equal or exceeds the ED limit values	Pass
21	SRAM write	Check if valid/reliable data transfer takes place at different address	Pass
22	SRAM read	Check if valid/reliable data transfer takes place at different address	Pass

design issues were fixed in RTL. Logic bugs found in the core are described below.

- The slave address, limit register address and values are sent using I2C communication cycle. The values are reflected in the respective memory's address for only one cycle, for next cycle, the values in the same address change to 'X State', and hence, no limits are set.
- Alarm condition is not being met even though respective conditions are met.

## 6 Conclusion

The authors have presented verification environment for in-house developed ASIC. This is an Institutional Research Project done by team of faculty and students to give complete experience from design to silicon. The environment is developed using SVM. The verification environment is built using object-oriented capabilities available in SVM classes. Exhaustive list of test cases are used to verify the functionality of the design. The bugs identified were fixed in the RTL.

## References

1. <https://datasheets.maximintegrated.com/en/ds/DS1683.pdf>
2. <https://datasheets.maximintegrated.com/en/ds/DS1682.pdf>
3. S.V. Siddamal, S.B. Shirol, S. Hiremath, N.C. Iyer, Design and physical implementation of mixed signal elapsed time counter in 0.18  $\mu\text{m}$  CMOS technology. In: A. Sengupta, S. Dasgupta, V. Singh, R. Sharma, V.S. Kumar (eds.) *VLSI Design and Test. VDAT 2019*. Communications in Computer and Information Science, vol. 1066 (Springer, Singapore, 2019)
4. RM, *SystemVerilog 3.1a Language Reference Manual Accellera's Extensions to Verilog®* (Accellera Organization, Inc., 2004)
5. C. Spears, 2006. *SystemVerilog for Verification, A Guide for Learning the Testbench Language Features*, 2nd edn. (Springer, 2006)
6. R. Edelman, A. Crone, et al., *Improving Efficiency, Productivity, and Coverage Using SystemVerilog OVM Registers*, courtesy of Mentor Graphics Corporation (2014)
7. M. Glasser, UVM: the next generation in verification methodology. in *Methodology Architect*, February 4, Courtesy of Mentor Graphics Corporation (2011)
8. Y.-N. Yun, J.-B. Kim, N.-D. Kim, B. Min, Beyond UVM for practical SoC verification. in *SoC Design Conference (ISOCC), 2011 International* (17–18 Nov 2011), pp. 158–162
9. R.J. Devins, J.R. Robinson, Automated system-on-chip integrated circuit design verification system. U.S. Patent 6,658,633, issued 2 Dec 2003
10. G. Mosensoson, Practical approaches to SoC verification. in *Proceedings of DATE User Forum* (Citeseer, 2002), pp. 05–08
11. Z. Hu, A. Pierres, S. Hu, F. Chen, P. Royannez, E.P. See, Y.L. Hoon, Practical and efficient SOC verification flow by reusing IP testcase and testbench. in *2012 International SoC Design Conference (ISOCC)* (IEEE, 2012), pp. 175–178
12. Valtrix Technologies Pvt. Ltd., RISC-V CPU test plan, revision 1.0. [Online] (October 2017). Available: <https://valtrix.in/announcements/riscv-test-plan>



# Design of High-Speed Turbo Product Code Decoder



Gautham Shivanna, B. Yamuna, Karthi Balasubramanian,  
and Deepak Mishra

**Abstract** In the field of digital communication, there has always been a requirement for an efficient, low complex, and high-speed error control encoder and decoder. Many such encoders and decoders for different error control codes have been proposed in the literature by researchers. However, developing such CODECs whose performance can be suitable for the requirements of modern communication systems is still an open research problem. In this paper, one such decoder, namely fast Chase decoder proposed in the literature, has been studied. The hardware design of the decoder has been done and verified with results from MATLAB simulations. An attempt has been made to improve the speed by replacing the ripple carry adder in the design with a fast adder. The hardware architecture is implemented in Xilinx XC7A35T platform, and an increase in computation speed of 5% has been achieved.

**Keywords** Turbo product code · Chase Pyndiah decoder · BER · Fast adder

## 1 Introduction

Turbo codes [1] have been receiving extensive importance in various communication systems since its inception in 1993. The idea of soft-input soft-output (SISO) iterative decoding is used in both turbo convolutional codes and turbo product codes. Turbo product codes (TPCs) have performance approaching the Shannon limit, provide high coding gain, and have highly parallelizable structures [2]. The iterative Chase Pyndiah SISO [3] algorithm that evolved in 1994 brought an acceptable compromise

---

G. Shivanna · B. Yamuna (✉) · K. Balasubramanian  
Department of Electronics and Communication Engineering, Amrita School of Engineering,  
Amrita Vishwa Vidyapeetham, Coimbatore, India  
e-mail: [b\\_yamuna@cb.amrita.edu](mailto:b_yamuna@cb.amrita.edu)

D. Mishra  
Digital Communication Division (DCD), Space Application Center (SAC),  
ISRO, Ahmedabad, India  
e-mail: [deepakmishra@sac.isro.gov.in](mailto:deepakmishra@sac.isro.gov.in)

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2021  
S. M. Thampi et al. (eds.), *Advances in Computing and Network Communications*,  
Lecture Notes in Electrical Engineering 735,  
[https://doi.org/10.1007/978-981-33-6977-1\\_15](https://doi.org/10.1007/978-981-33-6977-1_15)

between complexity and performance. Because of the reasonable complexity of the decoder and its usage in high data rate transmissions, TPCs have been used in many applications, such as IEEE 802.16, IEEE 802.20, satellite communication systems, and digital storage systems.

A typical two-dimensional TPC structure is built by using two component codes that handle a block of data, as rows and columns that can be decoded iteratively, one at a time [4]. The Chase Pyndiah algorithm [3], a derivation from the Chase II decoding algorithm, is widely used because of its reasonable decoder complexity.

Following the introduction of Chase Pyndiah decoding algorithm, many algorithms were introduced over the years to reduce the decoder complexity further. In [2], parallel decoding of the received data was introduced. Complexity reduction was realized by reducing the need for storing the syndrome table. In [5], a hybrid decoding algorithm was introduced, that coupled hard-input hard-output (HIHO) and SISO decoding algorithms together. In this algorithm, SISO was used for the initial iterations, and HIHO for later iterations, thus reducing the complexity. In [6], the fast Chase decoder was introduced which presented the concept of reordering the TPCs to reduce the complexity of the decoder and at the same time increase the computation speed. The hardware architecture used has shown a reduced decoding computation time. In [7], the complexity of the fast Chase algorithm was further brought down by reducing the number of real additions required for extrinsic information calculation. This low complexity decoder has negligible performance degradation when compared with the fast Chase decoder proposed in [6].

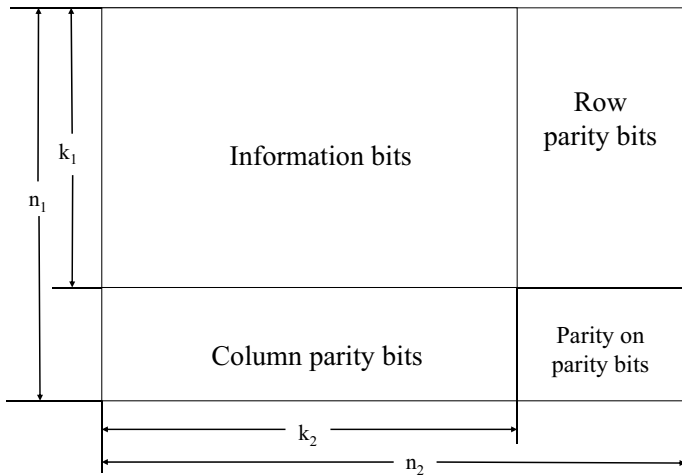
In this work, the low complexity fast Chase decoding algorithm proposed by Wang et al. in [7] is simulated and the performance is compared with respect to Chase Pyndiah algorithm. The hardware proposed in [7] has been used in this work. The hardware architecture has been simulated and its results verified. We propose to replace the ripple carry adder that has been used for additions in [7] by a carry save adder, a type of fast adder, with the objective of further reducing the computation time.

The paper content is as follows. In Sect. 2, the construction of TPC, the Chase Pyndiah decoding, the fast Chase decoding and the low complexity fast Chase decoding algorithms are discussed. This is followed by the description of the hardware design of the decoder and the introduction of fast adder in the hardware design in Sect. 3. In Sect. 4, the simulation and synthesis results of the decoder are presented and the paper concludes in Sect. 5.

## 2 Turbo Product Codes

### 2.1 Construction

A two-dimensional TPC is created by using two linear block codes. The widely used component codes used are Reed–Solomon (RS) [8], Single Parity Check (SPC) [9],



**Fig. 1** Construction of TPC [3]

Hamming [10] and Bose–Chaudhuri–Hocquenghem (BCH) [11] codes. The linear block codes are represented by  $C(n, k, d_{min})$ , where  $n, k, d_{min}$  denote codeword length, number of information bits and the minimum Hamming distance. The two-dimensional matrix is constructed by placing the information bits in a  $k \times k$  array. Following this, the rows and columns are encoded using their respective component codes. Figure 1 shows the construction of a TPC [3].

### 2.2 Chase Pyndiah Algorithm

The received signal  $R = (r_1, r_2, \dots, r_n)$  that is input to the decoder is called intrinsic information. The steps involved in Chase Pyndiah algorithm [3] are as follows:

1. The received signal is used to determine the hard decision values  $Y = (y_1, y_2, \dots, y_n)$ , where  $i = (1, 2, \dots, n)$ .

$$y_i = \begin{cases} 1, & \text{if } r_i > 0 \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

2.  $p$  least reliable bit (LRB) positions are determined from  $R$ .
3.  $2^p$  test patterns ( $T$ ) are generated by placing 0/1 at the least reliable positions and 0 in the remaining positions.
4. Test sequences,  $Z_i$  are to be generated by using (2).

$$Z_i = Y \oplus T_i \tag{2}$$

5. Syndrome  $S$  is calculated by using (3).

$$S_i = Z_i \cdot H^T \quad (3)$$

where  $H^T$  is the parity check matrix. Using the computed syndrome, hard decision decoding (HDD) is performed on  $Z_i$  to obtain the candidate codewords  $C_i = (c_1, c_2, \dots, c_n)$ .

6. The squared Euclidean distance is calculated for each of the valid codewords with respect to the received signal.

$$|R - C_i|^2 = \sum_{j=1}^{n-1} r_j - (2c_j - 1)^2 \quad (4)$$

7. The maximum likelihood (ML) codeword is determined as the codeword with the least squared Euclidean distance.
8. Following this, the extrinsic information ( $w_j$ ) is computed using (5), where  $d_j$  represents the received data bits.

$$w_j = \beta \cdot (2d_j - 1) \quad (5)$$

9. The extrinsic information calculated is used to update the received data,  $R$  using (6).

$$R = R + \alpha(w_j) \quad (6)$$

$\alpha$  and  $\beta$  represent weighing factor and reliability factor respectively. Pyndiah et al. in [3] has proposed the use of  $\alpha$  and  $\beta$  values as

$$\alpha = [0.0, 0.2, 0.3, 0.5, 0.7, 0.9, 1.0, 1.0]$$

$$\beta = [0.2, 0.4, 0.6, 0.8, 1.0, 1.0, 1.0, 1.0]$$

for practical considerations. The same has used in this work as well.

### 2.3 Fast Chase Algorithm

The complexity involved in the Chase Pyndiah algorithm is reduced by using the fast Chase algorithm [6]. The test patterns obtained are reordered in gray-code format. Hence, each of the test pattern differs from its predecessor by 1 bit only and thus can be derived from one another. The algorithm comprises of three steps, namely pre-processing, main-processing loop, and post-processing [6].

## 1. Pre-processing:

- (a) From the hard decision values,  $Y = y_1, y_2, \dots, y_n$ .  $p$  least reliable bits are selected.
- (b) The first test pattern  $T^0 = [00 \dots 0]$  is created and the corresponding test sequence  $Z^0$  is found. The syndrome  $s(0) = Z^0 \cdot T^0$  is calculated. The corresponding metric  $m'(0)$  is set to  $[00 \dots 0]$ .

## 2. Main-processing loop:

- (a) The error positions are determined using hard decision decoding. If the error position falls in one of the least reliable positions, the codeword need not be processed again and hence discarded. Else, the codeword is included into a list of candidate codewords.
- (b) Calculation of the analog weight,  $m(i)$ :

$$m(i) = \begin{cases} m'(i) + |r(i)| + |r_0|, & \text{if } p(i) \neq 0, \\ m'(i) + |r(i)|, & \text{otherwise} \end{cases} \quad (7)$$

where  $m'$  is the unamended analog weight and  $i = (0, 1, \dots, 2^p - 1)$

- (c) Generation of  $(i + 1)^{th}$  test pattern.
- (d) Generation of the syndrome for next sequence using (8).

$$s(i + 1) = s(i) \oplus h_l \quad (8)$$

where  $l$  is the index of nonzero bit between the test patterns.

- (e) Generation of  $m'$  for the next test pattern:

$$m'(i + 1) = \begin{cases} m'(i) + |r_l|, & \text{if } t(i + 1) > t(i), \\ m'(i) - |r_l|, & \text{if } t(i + 1) < t(i), \end{cases} \quad (9)$$

- 3. Post-processing: The list of analog weights and candidate codewords generated in the main processing loop are used to calculate the extrinsic information as given by (5) and (6).

## 2.4 Low Complexity Fast Chase Decoding Algorithm

In low complexity fast Chase decoding proposed in [7], the focus is on reducing the complexity involved in generating the candidate codeword list and its corresponding analog weight. This is done by reducing the complexity involved in the computation of extrinsic information that involves many comparison operations and real number additions. The complexity reduction is done by replacing  $\beta$  in (5) with the following:

$$\beta = (m_2 - m_1) \quad (10)$$

where  $m_2$  and  $m_1$  are the corresponding smallest analog weights of the candidate codewords. As we are using Hamming code, each component code has one error position. Hence, number of error positions will be  $5 \cdot 2^p + 1 \cdot p$  [7], where  $p$  represents number of error positions. When using (5) for the extrinsic information calculation, the total number of additions required is  $5 \cdot 2^p + 2 \cdot p$ . Upon modifying the extrinsic information calculation in (5) with (10), only one real addition is being performed per component code. Hence, total number of additions required will be  $3 \cdot 2^p + 1$ . As a result, there is a reduction in the number of additions and comparisons when compared to the fast Chase decoder proposed in [6].

### 3 Hardware Architecture

The hardware architecture of the fast Chase algorithm and the improvements proposed for TPC  $(31, 26)^2$  Hamming code are discussed in this section.

#### 3.1 Top Level Architecture

The top level module includes the memory module, the input module, the controller module, and the main TPC decoder as shown in Fig. 2.

The received data is stored in a text file as floating point numbers represented using 32-bit single precision format as shown in Fig. 3 [12].

This is sent to the decoder module that performs row decoding, column decoding, and extrinsic information calculation for the number of iterations provided. When the iteration count is reached, the results of the decoder are taken as the output and updated in the memory.

#### 3.2 Decoder Architecture

For our work, we use the decoder architecture that has been proposed by Wang et al. in [7]. Figure 4 shows the block diagram of the same.

The decoder architecture is divided into four major parts.

1. LRB, hard decision and syndrome calculation.
2. Candidate codewords and analog weights generation.
3. Extrinsic information calculation.
4. Received data updation using extrinsic information.

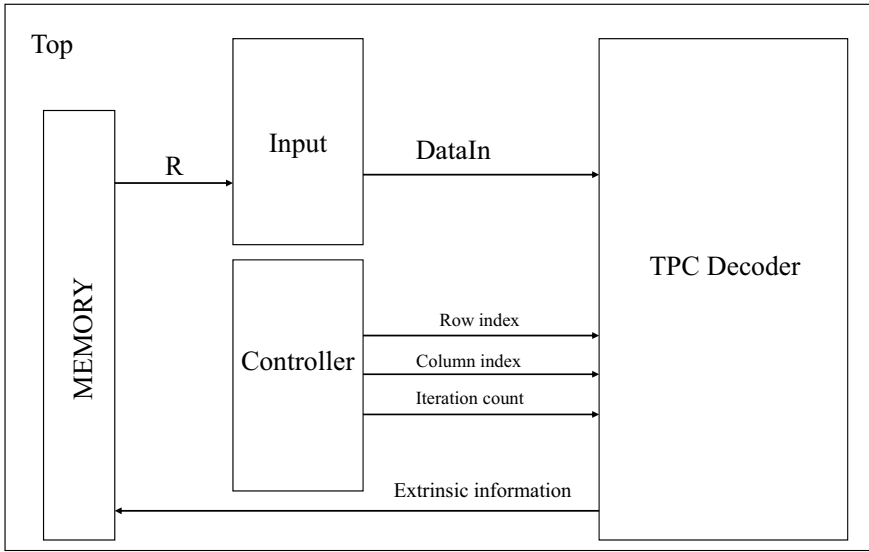


Fig. 2 Top level architecture

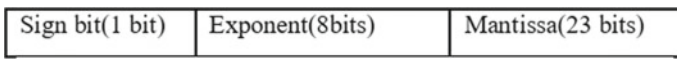


Fig. 3 IEEE 754 format of storing floating point data

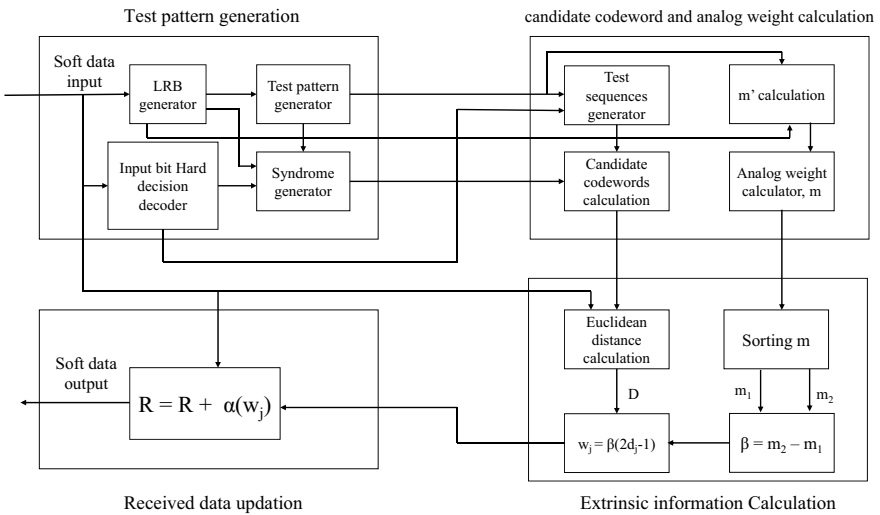


Fig. 4 Hardware design of the decoder [7]

The LRB generator module receives the soft inputs and generates the location of the least reliable bits by sorting the received data. These locations are sent to the test pattern generator module that converts the location information in to a gray code format. This data forms the test patterns that are passed on to the syndrome calculation module. Simultaneously, hard decision values of the received data are also generated. The test patterns and the hard decision values are sent to the syndrome generator module, and the syndromes are generated. This process takes thirty-two clock cycles (one for each bit in the data), and the output is stored in registers.

Both the input bit hard decision values and the test patterns are given to the test sequence generator module to generate the test sequences. Single error correction is performed on these test sequences using the syndrome generated to form candidate codewords. Along with this, the module also calculates the analog weights of the codewords simultaneously. These candidate codewords and the analog weights are then sent to Euclidean distance calculation module which would calculate the Euclidean distance between  $R$  and  $D$ . The candidate codeword  $D$ , with the minimum Euclidean distance is called the decoded codeword. Generating and decoding the codewords are performed using registers to store the intermediate values in the generation process and hence takes eight clock cycles. Reading the memory and finding the LRB requires one clock period each.

The decoded codeword and the analog weights are used for calculating the extrinsic information. The extrinsic information is calculated as in (10). This process using combinational modules for multiplication and addition; sequential modules for storing the intermediate values takes a total of twenty two clock cycles for completion. Hence, the total number of clock cycles necessary in one half iteration is—the sum of the clock cycles needed for: syndrome calculation for each bit in the data (32), storing intermediate values in the codewords generation and decoding (8), reading the memory and finding the LRB (2) and extrinsic information calculation (22) - 64.

### 3.3 Modified Euclidean Distance Calculation

In the architecture proposed in [7], Wang et al. use a ripple carry adder (RCA) to add two numbers in extrinsic information calculation. A major disadvantage of using RCA is that the time delay is linearly proportional to the bit length [13]. This is calculated as in (11).

$$t_{\text{total}} = (n - 1)t_{\text{carry}} + t_{\text{sum}} \quad (11)$$

where  $t_{\text{carry}}$  is the time delay of the full adder carry calculation, and  $t_{\text{sum}}$  is the time delay of computing the sum in the last bit. Thus, the performance of RCA becomes rather restricted when number of input bits increases.

To overcome this challenge, we propose a faster adder in the architecture as a replacement to RCA. Various fast adders are possible, including carry save adder (CSA), carry select adder, advanced carry select adder, parallel prefix adder and Ling adder, to name a few. In this paper, we propose to use CSA, aiming towards reducing



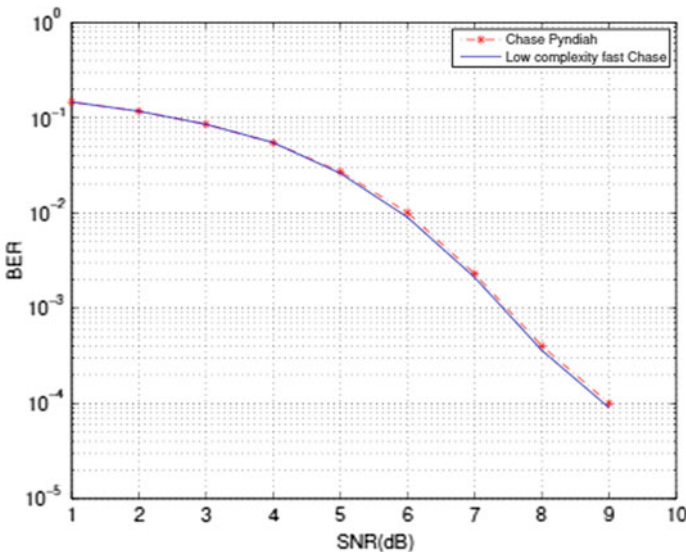
the number of additions. Compared to other adders, CSA reduces the addition of three numbers to only two. The carry-save unit is made from full adders, where each adder would compute a sum and carry bit based only on the input bits. The propagation delay is fixed to three gates irrespective of the total bits to be added because each full adder only computes sum and carry bit corresponding to its three inputs provided. Thus, sum is calculated, without any need for intermediate carry propagation [14].

## 4 Simulation and Synthesis Results

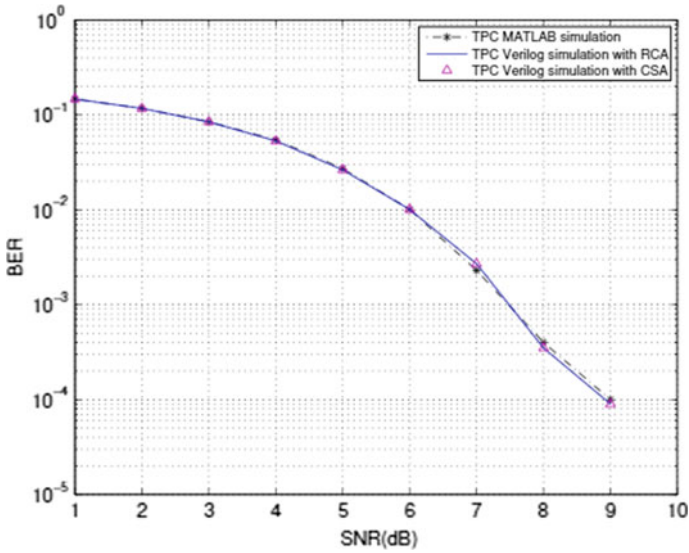
The simulation and the synthesis results of the low complex fast Chase algorithm are analyzed in this section.

### 4.1 Simulation Results

The component codes used in the TPC construction are Hamming codes. The encoded message is generated using the simulation setup in MATLAB. It is modulated using BPSK scheme and passed through an AWGN channel. It is then provided as an input for simulating the designed decoder in Verilog and MATLAB. The BER curve is plotted and analyzed to check the correctness of the decoder design [15]. This performance analysis is done in both MATLAB and Verilog.



**Fig. 5** BER plots of  $(31, 26)^2$  of the low complexity fast Chase algorithm and the conventional Chase Pyndiah algorithm to show the correctness of the design



**Fig. 6** BER plot of TPC decoder using the fast Chase algorithm with RCA and CSA adders. It can be seen that there is negligible difference between the plots, thus validating the correctness of the proposed change in the hardware

Figure 5 shows the comparative results of the BER performances of decoding TPC  $(31, 26)^2$  Hamming code by using the low complexity fast Chase and the conventional Chase Pyndiah decoding algorithms.

The comparative plot shows that there is negligible performance degradation in the fast Chase algorithm, thus validating the correctness of the design.

Figure 6 shows the BER plots of the Verilog simulation of the fast Chase algorithm with both RCA and CSA adders. MATLAB simulation of the fast Chase algorithm is also plotted as a reference.

It can be seen from the plots that the performance of Verilog designs of the fast Chase algorithm with RCA and CSA adders match with the MATLAB design of the same. Thus logical correctness of the proposed hardware change has been established.

## 4.2 Synthesis Results

The decoder is simulated using ModelSim and synthesized on to Artix-7-based FPGA (XC7A35T). The delay from the input to the output for the decoder with RCA was found to be 25.95 ns while the proposed design with CSA adder had a delay of 24.86 ns. Thus, the proposed design achieves a speed faster than the original low complex fast Chase decoder.

## 5 Conclusions

In this paper, the low complexity fast Chase decoding algorithm proposed by Wang et al. has been simulated and the performance was compared with the conventional Chase Pyndiah decoder. The BER performance analysis of the same is done for TPC  $(31, 26)^2$  Hamming code in MATLAB. The hardware design of the decoder with ripple carry adder used by Wang et al. has been done using Verilog. Based on this simulation study, a carry save adder-based hardware architecture has been proposed and Verilog simulation of the same is done. The simulation results have shown no performance degradation. Following this, the decoder has been synthesized and a comparison is performed with the usage of carry save adder by replacing the ripple carry adder. The synthesized results show that the use of carry save adder in the low complexity decoder achieves a reduction in time delay as compared to that of the use of ripple carry adder. This work can be extended to an increased number iterations in hardware and the VLSI realization of the same can be done on a FPGA evaluation board.

## References

1. C. Berrou, A. Glavieux, P. Thitimajshima, Near Shannon limit error-correcting coding and decoding: Turbo-codes. 1, in *Proceedings of ICC '93 - IEEE International Conference on Communications*, vol. 2, pp. 1064–1070 (1993)
2. C. Xu, Y. Liang, W.S. Leon, A low complexity decoding algorithm for extended turbo product codes. *IEEE Trans. Wireless Commun.* **7**(1), 43–47 (2008)
3. R.M. Pyndiah, Near-optimum decoding of product codes: block turbo codes. *IEEE Trans. Commun.* **46**(8), 1003–1010 (1998)
4. S.N. Vaniya, N. Kumar, C. Sacchi, Performance of iterative turbo coding with nonlinearly distorted OFDM signal, in *IEEE Annual India Conference (INDICON)*, vol. 2016, pp. 1–5 (2016)
5. B. Ahn, S. Yoon, J. Heo, Low complexity syndrome-based decoding algorithm applied to block turbo codes. *IEEE Trans. Commun.* **6**, 26693–26706 (2018)
6. S.A. Hirst, B. Honary, G. Markarian, Fast Chase algorithm with an application in turbo decoding. *IEEE Trans. Commun.* **49**(10), 1693–1699 (2001)
7. Y. Wang, J. Lin, Z. Wang, A low-complexity decoder for turbo product codes based on extended hamming codes, in *2018 IEEE 18th International Conference on Communication Technology (ICCT)*, pp. 99–103 (2018)
8. R. Zhou, R. Le Bidan, R. Pyndiah, A. Goalic, Low-complexity high-rate Reed-Solomon block turbo codes. *IEEE Trans. Commun.* **55**(9), 1656–1660 (2007)
9. D.M. Rankin, T.A. Gulliver, Single parity check product codes. *IEEE Trans. Commun.* **49**(8), 1354–1362 (2001)
10. C. Xu, Y. Liang, W.S. Leon, Shortened turbo product codes: encoding design and decoding algorithm. *IEEE Trans. Veh. Technol.* **56**(6), 3495–3501 (2007)
11. H. Mukhtar, A. Al-Dweik, M. Al-Mualla, A. Shami, Adaptive hybrid ARQ system using turbo product codes with hard/soft decoding. *IEEE Commun. Lett.* **17**(11), 2132–2135 (2013)
12. S. Nikhila, B. Yamuna, K. Balasubramanian, D. Mishra, FPGA based implementation of a floating point multiplier and its hardware trojan models, in *2019 IEEE 16th India Council International Conference (INDICON)*, pp. 1–4 (2019)

13. P.K. Kssrb, S. Pravallika, V. BhaskaraRaju, S. Ramesh et al., A low delayarchitecture for logarithmic multiplication, in *2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184)* (IEEE, New York, 2020), pp. 70–74
14. T. Kim, W. Jao, S. Tjiang, Circuit optimization using carry-save-adder cells. *IEEE Trans. Comput.-Aided Des. Integrated Circuits Syst.* **17**(10), 974–984 (1998)
15. V. Kakkara, K. Balasubramanian, B. Yamuna, D. Mishra, K. Lingasubramanian, S. Murugan, A Viterbi decoder and its hardware Trojan models: an FPGA-based implementation study. *PeerJ Comput. Sci.* **6**, e250 (2020)

# **Data Analytics**

# Extraction and Analysis of Facebook Public Data and Images



Bala Gangadhara Gutam, D. Subhash Chandra Mouli, and Sudhakar Majjari

**Abstract** Social networks play vital role in human communication and to improve the business applications. The Facebook is one of the most popular social networking application. However, the Facebook generates huge amount of data in the form of text, advertisements, posts, images, and videos. By analyzing Facebook data, we can find the location where our business promising in the real world. While sharing personal data, users demand security. Trust becoming an essential parameter in social networking. In this paper, a new technique is presented to identify duplicate logo images and profile pictures to prevent the fraud in business by keeping secret information in the profile picture or logo without distraction along with theoretical description of Facebook.

**Keywords** Social network · Privacy preserving · Facebook

## 1 Introduction

Data is collecting from the social networks and transforming into digitized data that able to analysis with machines. Information is all around us. It is in everything we touch, hear, taste, see and smell. Data mainly classifies into three types like structured, unstructured, and semi-structured. We live in an increasingly rich world of data. The amount of data currently exists doubles every 18 months. In real world,

---

B. G. Gutam (✉)

Department of Computer Science and Engineering, Siddhartha Educational Academy Group of Institutions, Tirupati, India

e-mail: [balabgangadhar@gmail.com](mailto:balabgangadhar@gmail.com)

D. Subhash Chandra Mouli · S. Majjari

Department of Computer Science and Engineering, MRR Institute of Technology and Science, Udayagiri, India

e-mail: [dschandramouli@gmail.com](mailto:dschandramouli@gmail.com)

S. Majjari

e-mail: [majjarisudhakar@gmail.com](mailto:majjarisudhakar@gmail.com)

sources of large data generation are like administrative data, online commercial or transactional data, data provided by sensors like satellite, climate analysis, data provided by tracking devices, behavioral data and data provided by social media. Social media are categorized into six major categories of each with their own advantages and drawbacks. They are bookmarking sites, social news, microblogging, media sharing, blog forums, and social network. Book marking sites are providing services like organizing, managing, and save links to various resources and Web sites around the Internet. Most popular book marking sites are StumbleUpon and Delicious. Social news are provide services like posting various links and news to outside articles and then allow users to vote on them. Most popular social news sites are Reddit and Digg. Media sharing is a service, and it provides users to share and upload various media such as videos and pictures. Most popular media sharing services are YouTube and Flickr. Microblogging is a service provided to users to post short updates that are pushed out to anyone subscribed to receive the updates. Most popular microblogging service site is Twitter. Blog comments and forums are services to allow users to hold conversations by posting messages and discussions around the topics of the blog. There are many forums and blogs one popular blog is blogger by Google. Social networking provides services that allow users to connect with other users of similar background and interests. In this paper, we mainly focus on social networking a category of social media. Most popular social networking sites are Facebook and LinkedIn. According to the survey, Facebook has nearly 1, 590 millions of active user accounts as up to April 2016. Facebook is free and a best medium to communicate with the world. It is more than all the remaining social networking sites like Whatsapp, Facebook Messenger, QQ, WeChat. Facebook establishes a rich place for researchers attentive in the affordances of social networks outstanding to its thick usage technological and patterns capacities that tie offline and online connections. We trust that Facebook signifies an understudied offline to online in that it primarily aided a geographically-bound the campus [2, 4]. The use of social networking (SN) among teenagers has grown fast in current days. Reports show that 92% of European teenagers report being a fellow of at least one social network [7]. Of these SN, Facebook ruins the most commonly used [5]. Due to the huge amount of time over on Facebook [3, 8, 14], concerns have been outstretched about the probable outcomes of Facebook use on teens well-being. Social networking tools are may be of specific utility for personalities whose have difficulties founding and preserving both strong and weak bonds. Some research has shown that the Internet might service individuals with small psychological well-being due to few bonds to friends and neighbors [3, 11]. The training of social networking services (SNS) like Facebook presents a number of challenges and concerns that makes the intellectual investigation of these facilities, and the several methods of content they hold significantly different from the training of the open Web [13].

Nowadays compared to text, more image data is generated in Facebook and it is using for several applications. By analyzing Facebook data, we can find the location where our business promising in the world [7]. By using Facebook previous posts, we can post the appropriate advertisements to improve business. In Facebook, we know the updates posted by the friends, pages liked, and groups. It is useful to

reconnect with old school or college friends. Facebook offers a very customizable advertisements placement facility, which are very easy to use and effective charges. Facebook is a place where we can chat with others, share our ideas, ask any questions to other, comment on other peoples status, we also add our own status, make friends, we can find market places of any items via advertisements by others, market the business, advertise and sell promote products or services for business purpose. Facebook is well designed for user, and it allows user to bookmark any Web site, to create or manage any applications, free gaming facility, most of the Facebook users are engaged Facebook through playing games. In Facebook, we also store our photos in albums, chatting information, videos. Facebook is used to identify the location of the presence of the user if he or she used Facebook in their mobile devices with GPS option [1, 3, 6, 9, 14]. Public images in Facebook are accessed by any Facebook user. Sometimes, users may misuse our personal, branded, advertised, or any other public images posted in Facebook. In this paper, we proposed a technique with invisible digital marking on Facebook images for avoiding misuse and finding fraud images.

## 2 Forms of Facebook Data

In Facebook, the data will store in different formats based on our posts, likes, groups, etc. Facebook is accept various kinds of data in the form of text data, image data, videos, and smiles. Each category of Facebook data is described below. Profile: Facebook profile contains the personal information about user. In this, Facebook Profile maintains the information like Facebook link address, email id used to register in Facebook, Date of Registration, Date of Birth, Gender, Names displayed on Facebook (all previously used names also stored), Current City, Home Town, Relationship Status, Family relations, Education details, Employers details, Languages spoken, Interests, Movies information, Television shows, Books interested, Favorite Cricket Team, Favorite athletes, Games and other interests, Involved Groups and maintained Networks, Apps Used and owned, and finally pages a user admin.

**Contact Info:** Facebook maintains the contact information of the user. It mainly stores the information like complete address of the user, different mail ids owned by the user, mobile phone numbers both verified and not verified contacts, screen names of LinkedIn, YouTube, etc., and it maintains and stores the contacts of the user like Google contacts. It synchronizes all mobile contacts along with the names stored in mobile devices.

**Timeline:** Facebook maintains timelines information with activity along with date and time. In Facebook, whatever activity we performed is stored in timeline. For example, if a person played a game Thursday, August 18, 2016, at 8:54 pm UTC+05:30, PraveenKumar Donta played Farm Heroes Saga. here the person played a game Farm Heroes Saga on Aug 18, 2016, like information it manages. For example, if our sent request is accepted by our friend, it maintains information like Monday, May 23, 2016, at 10:49 am UTC+05:30, PraveenKumar Donta and Nene Sri are



now friends. For example, if we shared a link Friday, July 22, 2016, at 12:09 pm UTC+05:30 PraveenKumar Donta shared a link, etc., information is maintained in Facebook Timeline.

**Photos:** Facebook maintains photos in different forms like Cover photos, profile pictures, timeline pictures, mobile uploads, and other albums. Recent days most of the users are like to post pictures rather than textual conversations in Facebook. In Facebook, whatever the picture we posted is one of the above categories. Along with the photos, Facebook stores metadata about the image like height, width, resolution, camera make, camera model, Orientation, Exposure, F-shot, Original Width, Longitude, Latitude, ISO Speed, Focal length, modified, uploaded date and time, picture taken time and date, upload IP address, likes and comments, etc. Facebook also maintains a copy of all comments and the commented person name. Along with uploaded photos, we also synchronize the photos to Facebook. We synchronize the photos to Facebook from Dropbox, Mobile camera photos automatically synchronize to Facebook, and Google Picasa pictures.

**Videos:** All uploaded videos in Facebook maintained along with the metadata about the video same as images. If we upload any video in Facebook, it stores video along with date and time of uploading, the upload IP address and some properties of video like length in seconds and in quality. Each and every video creates a link that specifies the location where exactly our video is stored.

**Friends:** It contains all the friends list of Facebook. In this module, it maintains the information about the deleted list of the friends and the requests we sent to others and they are not at accepted. We can find the list of details with the names of friends.

**Messages:** It stores all chatting information like sent and received messages. Each message along with sender or receiver and time and date of sent or receive information is available. But the deleted messages are not available. Pokes: A lot of reasons to poke a friend or friend of friends in Facebook. It is just to say Hi, Hello like attention messages to friends. When we poke them a notification is sent.

**Events:** A list of events along with posted time date, event date and times, Event posted with event name, location of the event with full address, whether we attend or not, etc. This information is maintained by the Events section.

**Security:** The Facebook stores the login and logout information. In the login session, it store the details of date time. IP address of login system, Browser used, and cookies information. Termination session it stores either terminated by logout or Web session termination (close the browser without logout the Facebook). In the Facebook, it identifies and stores the Operating system browser information, Login security information with cookies, estimated location inferred from IP address, Data authentication Cookie information. Administrative records like changing date of birth, change of user name or name, check point completed, removed profile pictures, security question responds change and password change, etc.

**Ads:** This section of Facebook maintains the advertisements displayed on our wall, advertisements we clicked and advertisements posted by the user. It maintains the history of ads along with date and time.

**Apps:** Apps are created by the user for a specific purpose. There are many apps available in Facebook for different purposes. Apps are like Gaming, educational, promoting, etc. App center a place it contains all apps created in Facebook.

**Pages:** Pages are used for brands, businesses, organizations, and public figures to generate a presence on Facebook, although profiles represent individual people. Anyone with an account can create a Page or help manage one, if they have been given a role on the Page like admin or editor. People who like a Page and their friends can get updates in News Feed. Admin or editor may post any updates in a Facebook page. Page information is set to public all Facebook user view the page posts. If a page is private, only the users liked that page can view the posts. In page, we may post text or image data.

**Groups:** A Group is used to grouping a like-minded people in Facebook. Group is not directly accessed by the users, admin of the group must add the users to see the posts in a group.

Data analysis is mainly about four questions, they are (1) what happened? (2) Why did it happen? (3) What will happen? And (4) what is the best that can happen? By using analysis we will give the answers to above four questions. By analyzing Facebook data, we may conclude a sentiment of a person, business analysis for posting ads, misbehaves of any users, spam detection, location tracking, identify the location to host a business, spam detection, etc. There are different tools used for data analysis like Microsoft excel, IBMs SAS and R programming.

### 3 Methods to Extract Facebook Data

To perform the analysis on Facebook, first we need to extract the data set. There are several tools and techniques to extract Facebook data, and some of them are listed below with their advantages and limitations.

In Facebook, there is an option to download a copy of Facebook data. This option is available in settings in Facebook. If we want to get a copy of Facebook data, it is available in a zip format. In that zip contains a web-based code contains the data of Facebook up to the time we downloaded. It contains the information about personal details, contact information, photos, shared photos, videos, security information, pokes, events, friends, ads, timeline information, apps used and liked are available in html code format. Each and every module it generates an html file. Photos and videos are separately stored in a folder. Each album in our Facebook account is extract to separate folders. This technique is little bit complex to perform the analysis why because the data is in the form of html files. We may need to use some other algorithms to separate the data from the html pages. If we want to perform the analysis on the

images or videos, it is a good approach why because images are available in separate folders. One more limitations with this approach is we get only the past data, and it is not possible to analysis the streaming data. In this technique, we can extract only the login user data, not even others Facebook users public data. One more drawback with this approach is it does not gives the information about groups and pages we maintained or liked. So we cannot perform the analysis on the page and groups posts. This is a limited but an easy technique, with in less time we get the Facebook training data. No tools and algorithms are required.

One of the traditional methods is used to collecting the social networking data through some software based tools, before that they recruited some researcher from Facebook organization and ask the person and collect the data from the persons by asking questioning and then they analyse the data [10]. It is traditional and time-consuming task, and some additional data may be added during the collection rather than the Facebook data.

Some of the APIs are provided by the third-party developers to the interactions with social networking to extracting the data with limited well-structured data. Facebook itself provided a graph API tool to extract the information about a valid user. But it also provides only textual data and links of images and videos. Here the data is provided with attributes and index values.

NameGenWeb is developed by the research scholars of Oxford Internet Institute provided the facility to extract the Facebook friends network and connections. Network Importer, a plug-in for the NodeXL visualization and network analysis toolkit developed by an international group of scholars, affords similar functionality for downloading personal networks, but also a means to extract extensive data from Facebook pages, including monopartite networks for users and posts, based on co-like or co-comment activities, and bipartite networks combining the two in a single graph [13].

Netvizz is an application developed by Bernhard Rieder from University of Amsterdam to extract the Facebook data and also performs the analysis and gets the pictorial representation of the result. He developed this to extract the Facebook data like posts, groups, friend network, newsfeed; in this data will be extracted into some text files with a provided graphical use interface (GUI). In this approach, he able to extract only the text information and analysis is also in the form of text data analysis. He does not provide approaches like extracting images from Facebook and analysis of images and videos. Here we can download data in a text file format. We can get only textual data of a particular users liked pages, groups, etc. We cannot get any images of user or others from this tools. It is limited to only extracting textual data and its analysis reports [13].

In R programming, a package called Rfacebook contains many functions to extract various kinds of public data like pages, groups, posts, etc., by linking with the Graph API Explore tool [12] from authenticated users in Facebook. Graph API explore is a tool and helps us to authenticate a user. In Rfacebook, we authenticate by using the access token. We can generate access token in two ways one is using Graph API Explore and another is using a Facebook app. When we generate access token in Graph API Explore it is valid up to 120min after that its session is closed, again we

need to generate the access token. When we go with the Facebook app, it is possible to generate access token for long-lived and is valid almost two months. Once we generated access token, it is very easy to access the Facebook data. In R, there are several functions to perform the different operations like get posts, pages, groups, timeline data, user personal information, searching for a post, find friends list, friends network, etc. Following are the functions in Rfacebook package.

- `getUsers()` function is useful to get the public and information of a authenticated user and public information of the other user friends used our app.
- `getFriends()` is used to get the friends details of authenticated user.
- `getGroup()` function is used to extract the information and posts of a public group.
- `getInsights()` is used to extract the insights metric of a public Facebook page
- `getLikes()` is used to extracts the friends likes.
- `getNetwork()` function is used to extracts authenticated users friends network details.
- `getNewsfeed()` function is used to extract the recent posts from the authenticated users newsfeed.
- `getPage()` function is used to extract the Facebook public pages post and their information.
- `getPost()` function is used to extract the Facebook public post and its details.
- `getReactions()` function is used to extract the count of reactions of a Facebook public post.
- `getShares()` function is used to extract a list of users who publicly shared any public Facebook post.
- `searchFacebook()` function is used to search any public Facebook post by specifying a string.
- `searchGroup()` function is used to search a Group ID of a Facebook public group.
- `searchPages()` function is used to search a Facebook public page by specifying a string.
- `updateStatus()` function is used to update our Facebook status from R programming.

All above functions are useful to extract the textual data. The data can be stored in some files for further processing. There is no specific function to extract Facebook images in Rfacebook package, and there is no specific tool give the images of Facebook.

## 4 Extract Facebook Public Pictures

Here we proposed a method to extract a group of images, profile pictures from Facebook for analysis. One image worth of thousand words. Compare to text data, Images may give more information. Nowadays compared to text more image data is generated in social networking like Facebook. So, image processing on social networking images gives the best result than textual analysis. For performing the

image analysis, dataset is required. Here we proposed a method to extract Facebook profile picture and all public images of a particular user in Facebook.

Here we proposed a method to extract a group of images, profile pictures from Facebook for analysis. Facebook image extraction is done with six steps are as follows:

Get Query from the Graph API Explore Tool: Graph API Explore Tool with the provided link as <https://developers.facebook.com/tools/explorer> for user authentication to access the different forms of the Facebook public data. This app generates a token to authenticate a user for accessing Facebook public data. From Graph API tool we also get the link that contain the Data about pictures. That link is [https://graph.facebook.com/v2.6/me?fields=photos.limit\(100\)%7Bpicture%7D&access\\_token=EAACEdXXXX](https://graph.facebook.com/v2.6/me?fields=photos.limit(100)%7Bpicture%7D&access_token=EAACEdXXXX).

Here access token is truncated. Figure 1 shows the Graph API Tool. We repeat the above steps until get all the images of Facebook authenticated user. Execute Query in callAPI() along with Authentication Token: We execute above query in the function callAPI() along with the access token. We also get query along with the access token. When we get query along with access token, there is no need to provide any external access token. CallAPI() returns a JSON formatted output contains the details of image id, image link, comments, likes, posted date and times, etc.

Tokenize and extract Image links from JSON file: From the JSON output, we tokenize the data and extract only the image links. We store each link into an array for further processing.

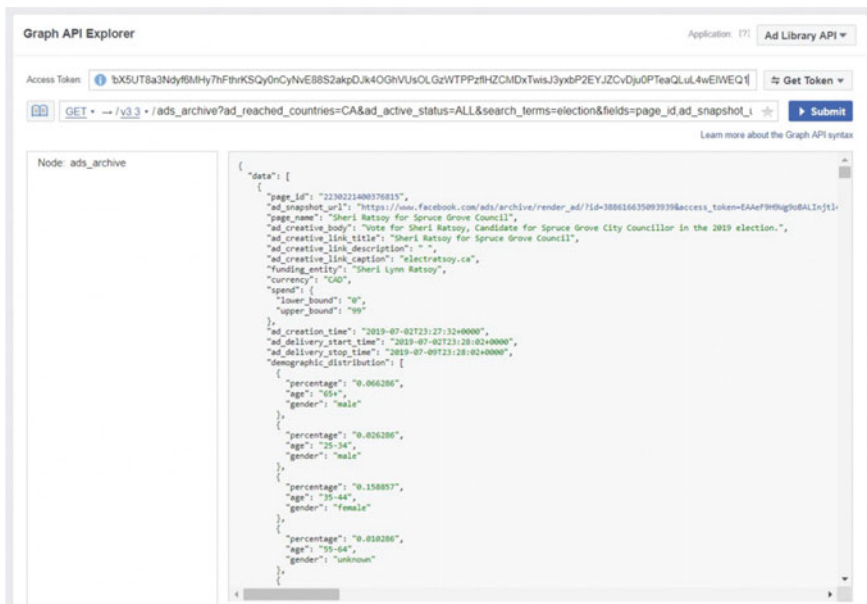


Fig. 1 Graph API tool



**Fig. 2** Facebook public image extraction steps

Execution of the image links in GET(): Here we pass the image link in GET() along with access token. It gets a raw data of the web page that contains data:URL, status code, headers, all headers, cookies, content, date, times, request, handle. In that content field contains the image data in one dimensional array format.

**Extract content of the image data from the raw data**

From the raw data generated by GET, we can separate the image content and store into one temporary variable. When it is separated, it generates a two-dimensional matrix format of image data. Convert matrix into Image: After getting the two-dimensional matrix, we can easily convert matrix into Image. Converted image can be stored into one directory for further use or directly to perform the image processing operations on it.

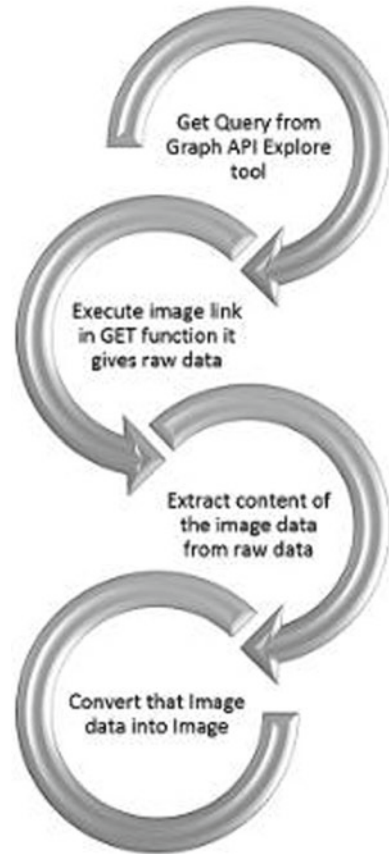
Figure 2 shows the Facebook public images extractions steps.

After extraction of the images, we may perform some image processing operations on them. In the next section, we performed one image processing technique on the extracted images.

**4.1 Extraction Facebook Profile Picture**

In this section, we described about the extraction of Face-book profile picture. The extraction is done only with four steps: Get query from Graph API Explore tool, Execute the image link and get raw data, Extract the content of the image from the raw data, and Convert that data into Image. And we skip the two steps (executing with callAPI() function and Tokenization) of above-discussed method. The following figure shows the procedure to extract Facebook Profile picture. Grayscale image extracted from the Facebook is in Figs. 4 (Fig. 3).

**Fig. 3** Steps to extract Facebook profile picture



#### ***4.2 Extraction Facebook Public Pictures***

All the Facebook images are extracted into a directory we specified. On extracted images, we can perform the operations of image processing (Figs. 5 and 6).

### **5 LSB Operation on Facebook Images**

The image steganography is the method in which we can hide the data within an image so that there will not be any perceived visible change in the original image. In the LSB technique, we convert image into shaded grayscale image. This image will act as reference image to hide the original image. In a grayscale image, each pixel is represented with 8 bits. The least significant bit as its value will affect the pixel value only by 1. So, this property is used to hide the data in the image. Here we have



Fig. 4 Extracted Facebook profile picture



Fig. 5 Extracted Facebook Public pictures



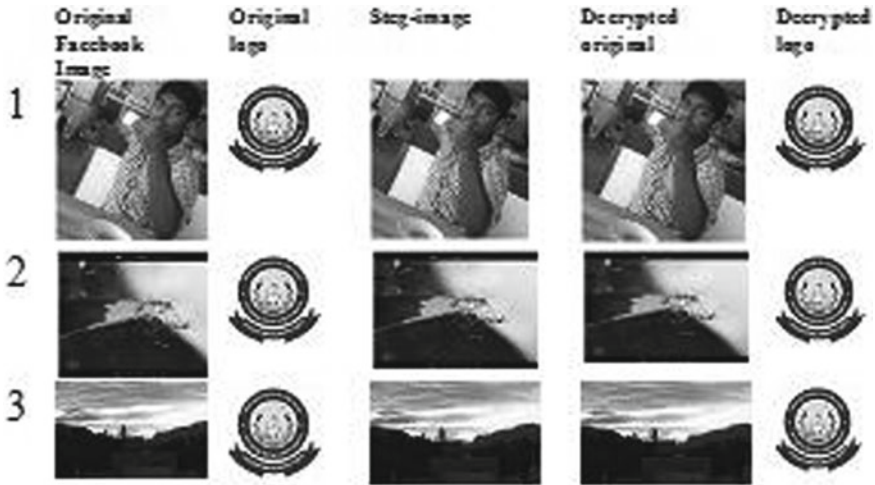


Fig. 6 Encrypted and decrypted output images

to consider last two bits as the least significant bits as they will affect the pixel value only by 3 to store extra data. The least significant bit (LSB) steganography is one such technique in which least significant bit of the image is replaced with secure image data bit. As this method is vulnerable to steganalysis so as to make it more secure we encrypt the raw image data before embedding it in the image. The encryption method increases the time complexity, but at the same time provides greater security also. This approach is very simple. In this method, the LSB bits of some or all of the bytes inside an image is replaced with a bits of the secret image bits.

### 5.1 Algorithm and Results

This section explains the proposed algorithm and its results.

#### Encryption Algorithm

1. Read Image from Facebook through R language named as Orig.
2. Read the logo image specified directory and named as logo.
3. Convert both images pixels into binary format.
4. Now take the 2 bits of a pixel from logo and store it in the two LSB of Orig. Repeat it to 8 bits of logo image pixel in four pixels of Orig image.
5. Now convert the binary values of the resulting image (Stego-image) to decimal.
6. After decimal conversion result could be in column matrix and so it has to be converted to the size of the image Orig.
7. Convert the matrix into Image. The Stego-image is obtained.

**Table 1** Jaccard and PSNR values

Image	Jaccard	PSNR
1	0.002168367	0.7494915
2	0.0007699275	0.439036
3	0.001766417	0.7285485

Note: This algorithm stores the logo image into the last bit of the Facebook image along the Column pixels.

#### **Decryption Algorithm**

1. Read the Stego-Image and named as Orig.
2. Convert Orig into a matrix and convert each pixel into binary values.
3. Take the LSB of each pixel column wise, and append these bits depending on the depth of the logo image to form pixels of the logo image.
4. Convert this binary values into Decimal values.
5. Convert the column matrix into the size of the logo image.

Note: Step5 can be done successfully only by the prior knowledge of the size of the secret image. Following are the results (Table 1).

## **5.2 Limitations**

Images of Facebook can be extracted only through online, not in offline. So Internet connection is required. In LSB operation, the logo image must be four times smaller than the Facebook image. Otherwise, some of the logo content is skip while encryption. While decryption process, we must know the size of the logo image.

## **6 Conclusion And Future Enhancement**

In this paper, we introduced different formats of the Facebook collected data from the user. After that, we presented different tools used to extract Facebook data of different forms which are inspired us to create a new method to Extract Image data of the Facebook. After that we applied LSB image steganography operation on extracted Facebook images with a logo image. In future, we extend this work to detection of spams, apply the digital watermarking techniques to secure our images in Facebook, Based on the images posted in Facebook, we can analyze the behavior of a user, and business intelligence.

## References

1. X. Bai, J.R. Marsden, W.T. Ross Jr., G. Wang, Relationships among minimum requirements, facebook likes, andgroupon deal outcomes. *ACM Trans. Manage. Inform. Syst. (TMIS)* **6**(3), 1–28 (2015)
2. J.A. Bargh, K.Y. McKenna, The internet and social life. *Annu. Rev. Psychol.* **55**, 573–590 (2004)
3. A. Błachnio, A. Przepiorka, M. Benvenuti, D. Cannata, A.M. Ciobanu, E. Senol-Durak, M. Durak, M.N. Giannakos, E. Mazzoni, I.O. Pappas et al., An international perspective on facebook intrusion. *Psychiatry Res.* **242**, 385–387 (2016)
4. N.B. Ellison, C. Steinfield, C. Lampe, The benefits of facebook “friends:” social capital and college students’ use of online social network sites. *J. Comput.-Mediated Commun.* **12**(4), 1143–1168 (2007)
5. R. Junco, Comparing actual and self-reported measures of facebook use. *Comput. Hum. Behavior* **29**(3), 626–631 (2013)
6. C. Kumar, A.K. Singh, P. Kumar, Improved wavelet-based image watermarking through SPIHT. *Multimedia Tools Appl.* **79**(15), 11069–11082 (2020)
7. A. Lenhart, M. Duggan, A. Perrin, R. Stepler, H. Rainie, K. Parker et al., Teens, social media & technology overview (2015)
8. J. Lin, R. Oentaryo, E.P. Lim, C. Vu, A. Vu, A. Kwee, Where is the goldmine? finding promising business locations through facebook data analytics, in *Proceedings of the 27th ACM Conference on Hypertext and Social Media*, pp. 93–102 (2016)
9. D. Praveen Kumar, A. Tarachand, A.C.S. Rao, Machine learning algorithms for wireless sensor networks: a survey. *Inform. Fus.* **49**, 1–25 (2019)
10. D. Quercia, R. Lambiotte, D. Stillwell, M. Kosinski, J. Crowcroft, The personality of popular facebook users, in *Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work*, pp. 955–964 (2012)
11. B. Rieder, Studying facebook via data extraction: the Netvizz application, in *Proceedings of the 5th Annual ACM Web Science Conference*, pp. 346–355 (2013)
12. A. Shahmohammadi, E. Khadangi, A. Bagheri, Presenting new collaborative link prediction methods for activity recommendation in facebook. *Neurocomputing* **210**, 217–226 (2016)
13. A.K. Tsitsika, E.C. Tzavela, M. Janikian, K. Ólafsson, A. Iordache, T.M. Schoenmakers, C. Tzavara, C. Richardson, Online social networking in adolescence: patterns of use in six European countries and links with psychosocial functioning. *J. Adolescent Health* **55**(1), 141–147 (2014)
14. A.S. Vairagade, R.A. Fadnavis, Automated content based short text classification for filtering undesired posts on facebook, in *2016 World Conference on Futuristic Trends in Research and Innovation for Social Welfare (Startup Conclave)* (IEEE, New York, 2016), pp. 1–5

# Subspace Clustering Using Matrix Factorization



Sandhya Harikumar and Shilpa Joseph

**Abstract** High-dimensional data suffers from the curse of dimensionality and sparsity problems. Since all samples seem equidistant from each other in high-dimensional space, low-dimensional structures need to be found for cluster formation. This paper proposes a top-down approach for subspace clustering called projective clustering to identify clusters in low-dimensional subspaces using best low-rank matrix factorization strategy, singular value decomposition. The advantages of this approach are twofold. First is to obtain multiple low-dimensional substructures using the best low-rank approximation, thereby reducing the storage requirements. Second is the usage of the obtained projective clusters to retrieve approximate results of a given query in time-efficient manner. Experimentation on six real-world datasets proves the feasibility of our model for approximate information retrieval.

**Keywords** High-dimensional data · Subspace clustering · Matrix factorization · Singular value decomposition

## 1 Introduction

With voluminous and high-dimensional data generated in almost every domain such as finance, health care, genomics, and signal processing, it has become necessary to eliminate redundant features and rows [1]. This not only solves the problem of curse of dimensionality but also helps in exploring patterns in low-dimensional subspaces. A single low-dimensional subspace from high-dimensional data will not give hidden relationships existing between subsets of features and subsets of rows. Therefore, a matrix factorization method is proposed on multiple subsets of data that identifies a subset of important attributes to form multiple and relevant low-dimensional subspaces [2, 3]. Projecting each of the samples on the most relevant subspace forms

---

S. Harikumar (✉) · S. Joseph  
Department of Computer Science and Engineering,  
Amrita Vishwa Vidyapeetham, Amritapuri, India  
e-mail: [sandhyaharikumar@am.amrita.edu](mailto:sandhyaharikumar@am.amrita.edu)

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2021  
S. M. Thampi et al. (eds.), *Advances in Computing and Network Communications*,  
Lecture Notes in Electrical Engineering 735,  
[https://doi.org/10.1007/978-981-33-6977-1\\_17](https://doi.org/10.1007/978-981-33-6977-1_17)

203

projective clusters and maintains the semantics of the data. The advantage of forming such projective clusters is reduced storage space, minimum information loss, and time-efficient approximate information retrieval.

For example, microarrays, that measures the expressions of thousands of genes on hundreds of different patients can be realized in terms of a matrix consisting of rows as genes and attributes as patients, and the value as the gene expression value for each patient [4]. In order to identify different groups of genes with similar expression profiles indicating homogeneous functions, it is necessary to find multiple low-dimensional substructures. In such scenarios, projective clusters are necessary to be explored.

As the dimensions grow, data become very sparse and Euclidean distance measure becomes meaningless as each of the samples seem to be equidistant from each other [3, 5, 6]. So, dimensionality reduction or feature selection techniques are used to reduce the dimensions [7]. Projective clustering aims at finding clusters in subspaces of data. Thus, cluster formation and dimensionality reduction are achieved simultaneously in this type of clustering. It usually follows top-down approach to simultaneously cluster the data and finding the relevant subspaces. One approach is to initially cluster the data and then evaluate the attributes in the context of the formed cluster [8]. The clusters so obtained are often hyperspherical due to the use of cluster centers to represent groups of similar instances. The clusters that are obtained give non-overlapping partitions of the dataset [9, 10].

The research objective of this paper is to find the dimensions in low space that can represent various clusters to hold most of the information in the dataset and to retrieve information effectively in less time.

The contributions of this paper are as follows.

1. Apply matrix factorization on multiple subsets of data to form  $K$  relevant subspaces.
2. Leverage the clusters formed using the subspaces for approximate information retrieval in time-efficient manner
3. Prove empirically the feasibility of this model using real-world datasets.

## 2 Related Works

Various types of clustering algorithms exist to group data into various clusters such that the intracluster similarity is high and intercluster similarity is low [8]. Basically, subspace clustering is done using either top-down approach or bottom-up approach. Hybrid approaches also exist. In the bottom-up approach, first subspaces are found and then clusters are formed. In CLIQUE, a histogram for each dimension is created and then those bins which have densities above the given threshold are selected. It is based on the downward closure property of density [11]. That is, if there are dense units in  $k$  dimensions, there are dense units in all  $(k - 1)$ -dimensional projections. Thus, subspaces are formed using only those dimensions that contained dense units.

Other bottom-up approaches are ENCLUS [12], MAFIA [13], cell-based clustering (CBF) [12], CLTree [12], and DOC [12]. In the top-down approach, first clusters are formed in full feature space and then relevant subspaces are explored. For subspace formation, each dimension is assigned a weight for each of the clusters. The updated weights are then used in the next iteration to regenerate the clusters. This approach needs multiple iterations of clustering algorithm in the full set of dimensions. Top-down algorithms usually create disjoint partitions thereby assigning each instance to only one cluster. The most critical parameters for top-down algorithms are the number of clusters ( $K$ ) and the size of the subspaces,  $l$ , on an average, which are often very difficult to determine. Some of the top-down approaches are PROCLUS [14], ORCLUS [15], FINDIT [16], and  $\delta$ -Clusters [17], COSA. In this paper, we adapt the clustering approach of Proclus and later appropriate subspaces are found using matrix factorization [4, 18].

Matrix factorization reduces a matrix into constituent parts which make it easier to calculate more complex matrix operations. It is the foundation of linear algebra in computers. These are the basic operations for solving systems of linear equations, calculating the inverse, and calculating the determinant of a matrix [19]. One of the best matrix factorization methods that gives the best low-rank approximation is singular value decomposition (SVD). SVD is used to reduce the number of features of a dataset by reducing space dimensions from  $d$  to  $l$  [20–22]. Various models such as prediction models for student performance and healthcare text classification have been developed using Matrix factorization [23, 24].

We are interested in extracting meaningful information from the data for a given query in reduced time. Hard-coded rules or feature-based models are available to retrieve information [7, 25]. In machine learning, the end goal is to learn good models of reality to regress, classify, or describe the data. The only way information retrieval is related to machine learning is that it makes use of ML models. Queries are formal statements of the information needed, for example, search strings in Web search engines. In information retrieval, a query does not uniquely identify a single object in the collection. Instead, several objects may match the query, perhaps with different degrees of relevancy.

### 3 Subspace Clustering and Information Retrieval

Subspace clustering is an unsupervised learning technique which groups a set of data points or objects into a cluster such that the objects in the same cluster are similar pertaining to a subset of dimensions. Matrix factorization models map the items and concept into a space of dimensionality  $d$ , such that the interactions are modeled as inner products in that space.

In this paper, singular value decomposition [SVD] is used to find the dimensions that can be used for subspace formation in different groups of instances. When a cluster consisting of a subset of samples from the original data is given to SVD, it



**Initial Clustering** This phase is to find a good set of medoids which can form good representatives for subspace clusters. Medoids chosen are farthest from each other in high-dimensional space. This is ensured by using  $L1$  distance metric due to its characteristics like robustness to outliers and sparsity. To form  $K$  subspace clusters, a set of maximum distant medoids,  $M$ , of size greater than  $K$  is computed. These medoids help for forming initial clusters in full-dimensional space as shown in 1 taken from [14].

---

**Algorithm 1: Initialization Phase**

---

$C_i$  is the  $i^{th}$  cluster  
 $D_i$  is the set of dimensions associated with cluster  $C_i$   
 $M_{current}$  is the set of medoids in current iteration  
 $M_{best}$  is the best set of medoids found so far  
 $N$  is the final set of medoids with associated dimensions  
 $A, B$  are constant integers  $S = \text{random sample of size } A * K$   
 $M = \text{Greedy}(S, B * k)$

---



---

**Algorithm 2: Iterative Phase**

---

1. BestObjective =  $\infty$
2.  $M_{current} = \text{random set of medoids } m_1, m_2 \dots m_k$
3. for each medoid  $m_i$  in  $M_{current}$  do
  - Let  $\delta_i$  be distance to nearest medoid from  $m_i$
  - $Lo_i = \text{Points in sphere centered at } m_i \text{ with radius as } \delta_i$
4.  $Lo = \text{Localityset i.e., } [Lo_1, Lo_2, \dots, Lo_k]$
5. Dimensions = FindDimensions( $k, l, Lo$ )
6. Clusterset = AssignPoints(Dimensions)
7. ObjectiveFun = EvaluationCluster(Clusterset, Dimensions)
8. if ObjectiveFun < Bestobjective
  - begin
  - BestObjective = Objectivefun
  - $M_{best} = M_{current}$
  - Find the bad medoid in  $M_{best}$  and replace them
  - end

---

**Iterative Phase** In this phase, randomly  $K$  medoids are chosen from set of  $M$  medoids obtained from initial phase. These  $K$  medoids are used to form  $K$  clusters in full-dimensional space as shown in Algorithm 2. Then matrix factorization, SVD, is applied on these clusters, to find a good substructure by finding the dimensions. The algorithm to find dimensions is as shown in Algorithm 3.

**Cluster Refinement** Dimensions and cluster obtained from iterative phase are passed through refinement phase to improve the quality of the final cluster. For each medoid, and its corresponding dimension, we find least distance  $\delta_i$  to one of the medoids



---

**Algorithm 3: FindDimensions (k,l,Lo)**


---

1. Apply SVD on each locality  $L_i$  and find  $U_i, V_i^T, \Sigma_i$  for the same.
  2. Then identify the optimal set of dimensions based on  $\Sigma_i$  values.
  3. If  $V^T$  matrix is used then , each row in  $V_i^T$  is the projected axis and we project the original data to these axis to form a subspace.
  4. If U matrix is used, then each column is the projected axis and the data is projected to each  $U_i$
- 

corresponding to the dimension. Points outside the sphere of influence are considered as outliers for the respective clusters. The algorithm is as shown in Algorithm 4.

---

**Algorithm 4: Refinement phase**


---

1. With the medoid set  $M_{best}$  obtained from the iterative phase , form a new locality  $L$
  2. Find new dimensions and form clusters  
 $NewDimensions = FindDimension(Dimensions)$   
 $(C_1, C_2, \dots, C_k) = AssignPoints(NewDimensions)$
  3. Find the optimal clusters  $C = \{C_1, \dots, C_k\}$  with best medoid set  $M_{best}$  and corresponding dimensions ( $M_{best}, NewDimensions$ )  
return  $C$
- 

These algorithms are adapted from [14].

**Query processing** After the formation of the subspace clusters, information can be retrieved using the obtained clusters. For a given query  $q$ , the similarity is found between  $q$  and each of the subspace clusters  $C_i$ . The most similar subspace is used to retrieve the information.

---

**Algorithm 5: Query Processing to retrieve information for query  $q$** 


---

1. Find the cluster  $C_q$  that is closest to the  $q$ .
  2. For each sample  $dp_i$  in the Cluster  $C_q$  do  
Distanceset = find the distance between  $q$  and  $dp_i$ .  
qSimilar = min(Distanceset)
  - end
  4. return qSimilar
-

## 4 Experimental Analysis

This section briefs our experimental results and comparison of our subspace clustering algorithm with another subspace clustering algorithms, PROCLUS, and CLIQUE to show the effectiveness of our proposed work.

**Performance Parameters** We used the following parameters in our study: the number of instances  $n$ , number of subspace clusters  $K$ , and average dimensionality of each subspace cluster  $l$  to demonstrate the effectiveness of our proposed algorithm. The proposed approach was then applied on the datasets to determine how the cluster quality varied with  $n$ ,  $K$ , and  $l$ .

**Datasets** To evaluate the performance of our proposed model, using six datasets as shown in the table. These six datasets are high-dimensional with dimensionality varying from 9 to 7200 and the number of instances from 767 to 10,000 and other parameters are as shown in below table.

Here, average number of dimensions per cluster is taken as ( $l$ ).

## 5 Results and Analysis

After the cluster was formed, we evaluated the **cluster quality** using **Davies–Bouldin index**. Its validation of how well the obtained clustering is made using quantities and features inherent to the dataset. Lower the DB index value, better is the clustering.

$$DBindex(U) = 1/k \sum_{i=1}^k \max \frac{\Delta(X_i) + \Delta(X_j)}{\delta(X_i, X_j)} \tag{1}$$

where,

$\delta(X_i, X_j)$  is the intercluster distance.

$\Delta(X_k)$  is the intracluster distance of cluster  $X_k$ .

Another evaluation metric we used is :

$$\sum_{i=1}^k (C_i * w_i) / N \tag{2}$$

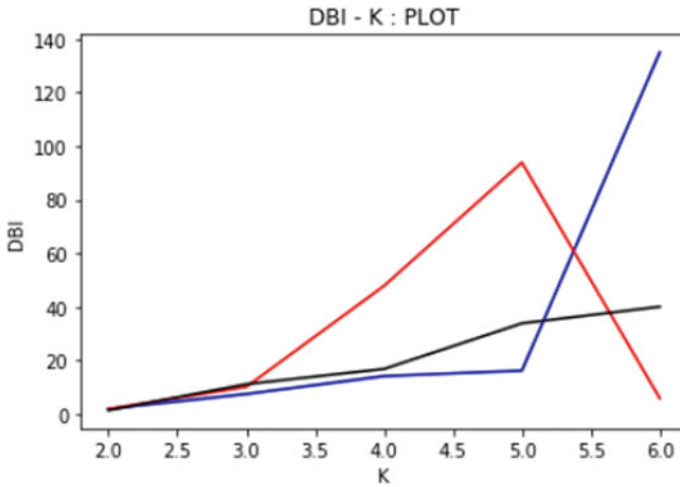
where.

$C_i$  is the  $i_{th}$  cluster

$w_i = \sum_j Y_{i,j} / |D_i|$

$Y_{i,j}$  is average distance of points in  $C_i$  to centroid of  $C_i$  along dimension  $j$ ,  $j \in D_i$  and  $D_i$  is the set of dimensions

$N$  is the number of instances in the dataset.



**Fig. 2** Cluster evaluation using DBI metric for diabetes dataset by varying number of clusters  $K$  with number of instances  $n$  and number of attributes  $d$  constant as given in Table 1. Average dimensions per cluster  $l$  are taken to be 5

Also, the execution time against number of cluster.

We have compared our proposed model with PROCLUS and CLIQUE and the results are shown in Figs. 2, 3, 4, 5, 6, 7, and 8.

The experiments are conducted by using different the parameters, as one varying and the other constant and the details are given in Table 1 .

*[Red line represents PROCLUS, blue line represents SVD-based subspace clustering (proposed model), and black line is CLIQUE.]*

The following aspects were considered for evaluation,

- (1) Effect of the number of instances,  $N$  on DBI
- (2) Effect of average dimensions per cluster,  $l$  on DBI.
- (3) Effect of number of clusters( $K$ ) on DBI and.
- (4) Execution time while varying the number of clusters  $K$ .

**Table 1** Dataset

Dataset	Instances ( $n$ )	Attributes ( $d$ )	Avg. dim. ( $l$ )	Clusters ( $k$ )
diabetes.csv	767	9	5	6
leukemia.csv	36	7130	2	4
Glass.csv	214	10	4	5
bio_test.csv	10,000	77	15	7
phy_train.csv	900	80	9	12
GCM.csv	144	1600	10	4

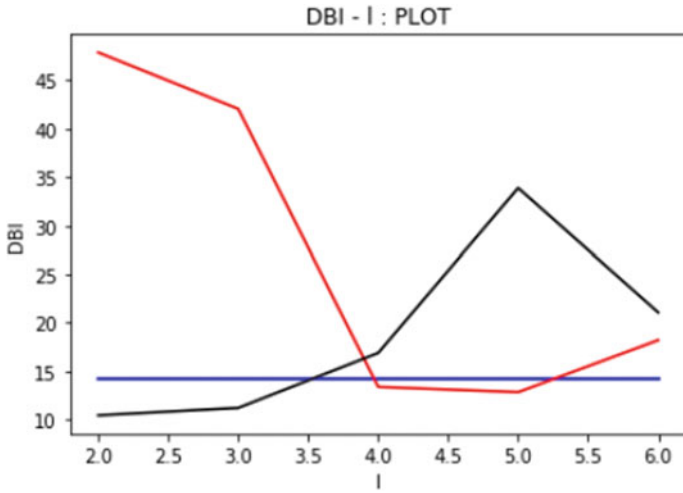


Fig. 3 Cluster evaluation using DBI metric for diabetes dataset by varying  $l$  with number of instances  $n$  and attributes  $d$  constant as given in Table 1.

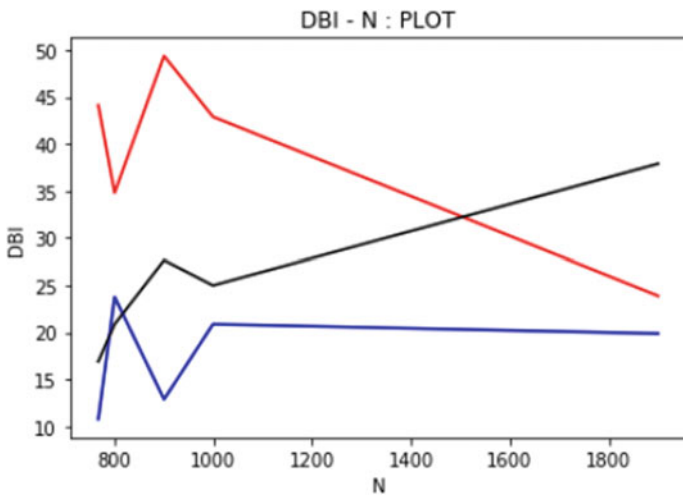
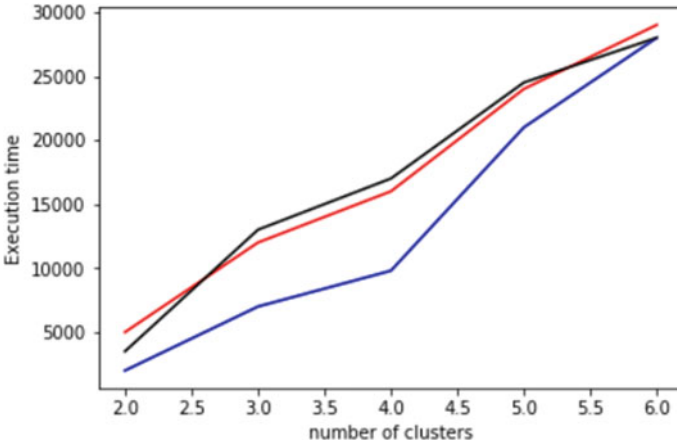
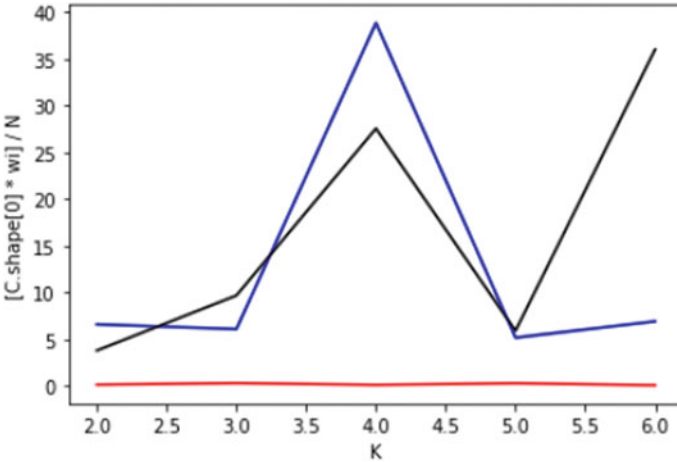


Fig. 4 Cluster evaluation using DBI metric for each dataset in Table 1 by number of instances  $n$  in it



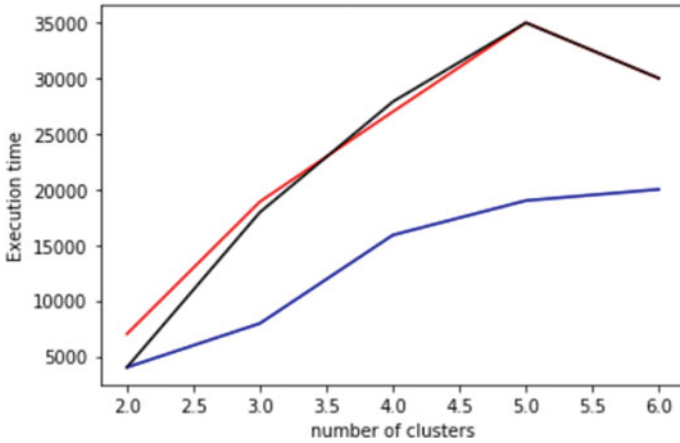
**Fig. 5** Evaluation time represents clustering and query processing time by varying number of clusters  $k$  for diabetes dataset. [Execution time is represented in seconds]



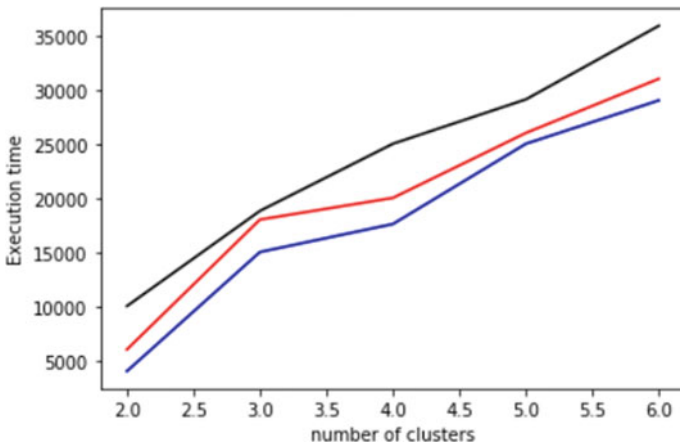
**Fig. 6** Cluster evaluation using Eq. (2) for diabetes dataset by varying number of clusters  $K$  with number of instances  $n$  and attributes  $d$  constant as given in Table 1.

The results are shown in the below graphs. We observe that, while the number of features, average number of dimensions per cluster, and cluster numbers increases, the approach shows better results while considering cluster quality(DBI).

From Figs. 2, 3, and 4, we can see that our proposed model gives better cluster compared to PROCLUS and CLIQUE as we vary different parameters like, number of clusters formed, as the average number of dimensions per cluster  $l$  varies by keeping the number of clusters as constant ( $K = 5$ ) and also for each iteration the number of instances ( $N$ ) vary.



**Fig. 7** Evaluation time represents clustering and query processing time by varying number of clusters  $k$  for GCM dataset and remaining parameters as constant as in Table 1. [Execution time is represented in seconds]



**Fig. 8** Evaluation time represents clustering and query processing time by varying number of clusters  $k$  for bio\_train dataset and remaining parameters as constant as in Table 1. [Execution time is represented in seconds]

In Figs. 5, 7, and 8, we can see that PROCLUS and CLIQUE take more time for clustering and query processing, while our approach takes less execution time. Here, we can see that as the number of clusters increases the execution time taken by our model is less compared to the other two models.

By looking at the results, we can see that our approach gives better cluster and execution time compared to other two models.

## 6 Conclusion

We have proposed an alternative approach toward subspace clustering using matrix factorization technique called singular value decomposition (SVD). This aids not only in computing stable subspace clusters but also retrieves semantic information for a given query. SVD is used to find the low-dimensional substructure in a local neighborhood leading to a subspace. Thus, multiple subspaces and clusters corresponding to the obtained subspaces are formed by projecting the data onto an optimal subspace. The proposed model for subspace clustering gives better performance than algorithms such as PROCLUS and CLIQUE. Further, the usage of SVD in subsets of data has improved the computing time of subspace clustering significantly and the features or attributes that are used for cluster formation have been retained in its original form. This model discovers interesting patterns in subspaces of high-dimensional data spaces. It allows the selection of different sets of dimensions for different subsets of the data with minimal information loss.

## References

1. S. Harikumar, A.S. Akhil, Semi supervised approach towards subspace clustering. *J. Intell. Fuzzy Syst.* **34**, 1619–1629 (2018). <https://doi.org/10.3233/JIFS-169456>
2. R. Ng, J. Han, Efficient and effective clustering methods for spatial data mining, in *Proceedings of the 20th VLDB Conference*, pp. 144., 155 (1994)
3. C. Aggarwal, A. Hinneburg, D. Keim, On the surprising behavior of distance metrics in high dimensional space, in *Database Theory-ICDT 2001*, Lecture Notes in Computer Science (Springer, Berlin, Heidelberg, 2001), pp. 420–434
4. S. Chitra Nayagam, Comparative study of subspace clustering algorithms. *Int. J. Comput. Sci. Inform. Technol.* **6**(5), 4459–4464 (2015)
5. T. Gonzalez, Clustering to minimize the maximum intercluster distance. *Theor. Comput. Sci.* **38**, 293–366 (1985)
6. R. Lee, Clustering analysis and its applications, in *Advances in Information Systems Science*, ed. by J. Toum, vol. 8 (Plenum Press, New York, 1981), pp. 169–292
7. L. Yu, H. Liu, Feature selection for high-dimensional data: a fast correlation-based filter solution, in *Proceedings of the Twentieth International Conference on Machine Learning*, pp. 856–863 (2003)
8. A. Jain, R. Dubes, *Algorithms for Clustering Data* (Prentice Hall, Englewood Cliffs, NJ, 1998)
9. S. Harikumar, P.V. Surya, K-medoid clustering for heterogeneous datasets, in *4th International Conference on Eco-friendly Computing and Communication Systems (ICECCS)*
10. S. Harikumar, M. Shyju, M.R. Kaimal, SQL-mapreduce hybrid approach towards distributed projected clustering, in *2014 International Conference on Data Science & Engineering (ICDSE)*
11. C.C. Aggarwal, P.S. Yu, Finding generalized projected clusters in high dimensional spaces, in *Proceedings of the 2000 ACM SIGMOD international conference on Management of data* (ACM Press, 2000), pp. 70–81
12. L. Parsons, E. Haque, H. Liu, Subspace clustering of high dimensional data: a review, in *ACM SIGKDD Explorations Newsletter* (2004)
13. S. Goil, H. Nagesh, A. Choudhary, MAFIA: Efficient and scalable subspace clustering for very large data sets, Technical Report CPDC-TR-9906-010 Northwestern University (1999)

14. C.C. Aggarwal, C. Procopiuc, J.L. Wolf, P.S. Yu, J.S. Park, Fast algorithms for projected clustering, in *SIGMOD '99*, Philadelphia PA Copyright, ACM, 1999 1-581 13-084-8/99/05
15. P. Pore, Must-Know: What is the curse of dimensionality? <https://www.kdnuggets.com/2017/04/must-know-curse-dimensionality.html>
16. K.G. Woo, J.H. Lee, FINDIT: A Fast and Intelligent Subspace Clustering Algorithm using Dimension Voting. PhD thesis, Korea Advanced Institute of Science and Technology, Taejon, Korea (2002)
17. Yang et al., 'It'-clusters: capturing subspace correlation in a large data set, in *ICDE* pp. 517–528 (2002)
18. M. Hund, M. Behrisch, I. Farber, M. Sedlmair, T. Schreck, T. Seidl, D. Keim, Subspace nearest neighbor search—problem statement, approaches, and discussion position paper, in *International Conference on Similarity Search and Applications SISAP 2015: Similarity Search and Applications*, pp. 307–313
19. T.F. Chan, Rank Revealing QR Factorizations. Department of Mathematics University of California at Los Angeles, Los Angeles, CA
20. W. Xu, X. Liu, Y. Gong, Document clustering based on non-negative matrix factorization, in *SIGIR'03* July 28–August 1, 2003, Toronto, Canada. Copyright 2001 ACM 1-58113-646-3/03/0007
21. S. Harikumar, S.S. Thaha, MapReduce model for K-Medoid clustering, in *2016 IEEE International Conference on Data Science and Engineering (ICDSE)*
22. Wikipedia, Clustering high-dimensional data. [https://en.wikipedia.org/wiki/Clustering\\_high-dimensional\\_data](https://en.wikipedia.org/wiki/Clustering_high-dimensional_data)
23. N. Prema, T.K. Smruthy, Personalized multi-relational matrix factorization model for predicting student performance, in *Intelligent Systems Technologies and Applications*, ed. by S. Berretti, S.M. Thampi, P.R. Srivastava (Springer, Cham, 2016), pp. 163–172
24. B. Barathi, H. Ganesh, M.A. Kumar, K.P. Soman, Distributional semantic representation in health care text classification, in *CEUR Workshop Proceedings, Volume 1737, 2016, Pages 201-204, Forum for Information Retrieval Evaluation, FIRE 2016, Kolkata, India*
25. W. Li, C. Chen, J. Wang, An efficient clustering method for high-dimensional data, *Conference: Proceedings of The 2008 International Conference on Data Mining, DMIN 2008*, 2 vol., July 14–17, 2008, Las Vegas, USA
26. N. Lal, S. Qamar, S. Shiwani, Information retrieval system and challenges with dataspace. *Int. J. Comput. Appl.* **147**(8), 23–28 (2016)



# A Data-Driven Approach for Peer Recommendation to Reduce Dropouts in MOOC



Manika Garg and Anita Goel

**Abstract** Massive open online course (MOOC) is an online mode for learning aimed at unlimited participation. A characteristic feature of MOOC is reduced availability of social interaction, which is often responsible for learners feeling isolated. Although to facilitate interaction, MOOC has functionalities like discussion forum, group assignment and peer grading; however, to use these functionalities, the learner has to extensively search for the right person to interact from a large pool of learners. The isolation among learners is one of the significant factors contributing to high learner dropout rate, a major concern for MOOC. In this paper, we present an approach to reduce the dropout rate of MOOC by solving the problem of isolation. A potential solution to this problem is to encourage peer learning, by supporting learners to find other learners for interaction purposes. In this paper, we propose a user similarity-based peer recommendation approach that makes use of learners' scores and their demographic attributes, to provide recommendations on potential learning peers. To date, however, the main focus of traditional approaches for peer recommendations is on providing recommendations to all learners, including the ones who were not feeling isolated. Furthermore, these approaches provide peer recommendations to learners without considering their actual cause of isolation. To overcome these limitations, we use adaptive interventions to first identify the isolated learners and then recommend peer learners based on their cause of isolation. The proposed approach for peer recommendation is evaluated on the basis of scalability and coverage. The publicly available MIT Harvard database has been used for experimental purpose.

**Keywords** MOOC · E-learning · Dropout · Meta-data · Peer recommendation

---

M. Garg (✉)

Department of Computer Science, University of Delhi, New Delhi, India  
e-mail: [manikagarg2007@gmail.com](mailto:manikagarg2007@gmail.com)

A. Goel

Department of Computer Science, Dyal Singh College, University of Delhi, New Delhi, India  
e-mail: [goel.anita@gmail.com](mailto:goel.anita@gmail.com)

# 1 Introduction

The rapid evolution of MOOC as a distance learning method has dramatically transformed the education sector. Various MOOC platforms, such as edX and Coursera, provide access to a large number of courses across various fields. The open access feature of MOOC attracts large number, often hundreds of thousands of participants. However, in spite of the high enrolment, the percentage of learners actually completing any particular MOOC often falls below 10% [1–3]. While various factors contribute to these high dropout rates, the problem of feeling of isolation among many learners is an important factor that directly affects learners' dropout rate of MOOC [4–7].

An innate feature of online learning is reduced availability of face-to-face interactions, which in turn is often responsible for students feeling isolated [8]. Learners involved in distance learning, frequently, feel the lack of personal assistance by course authors and fellow students, which mostly leads to learners' loss of motivation [9]. All these causes can create a feeling of isolation among MOOC learners.

To facilitate social interaction among learners, MOOC has included features like discussion forums, group assignments and peer assessments. Interaction among learners in MOOC is primarily dependent on discussion forums. But according to Chiu and Hew [10], the discussion forums posts are written by only 5–12% of learners, and more than 75% of learners read the forum posts only once. One of the issues with discussion forums is the difficulty of a MOOC learner who has to extensively search for a right person to interact, from a large pool of learners. Also, according to Labarthe et al. [11], many learners do not necessarily know how to initiate, have meaningful conversations within this community and feel shy or inhibited in such crowded places.

One potential way of addressing the problem of isolation is to encourage peer learning in MOOC, by supporting learners to find other learners for interaction and cooperation purposes. Several studies [5, 12–14] have shown that stimulating interactions among learners is a key to foster learner engagement and mitigate dropouts. Traditional approaches for peer recommendations are mainly based on either learners' profile and/or their behavioural attributes. These approaches neither identify the isolated learners nor determine their cause of isolation. The recommendations are for all learners including the ones who are not even feeling isolated.

Here, we present an approach for peer recommendation to reduce dropouts in MOOC. We use adaptive interventions to identify the target learners having the feeling of isolation and their cause of isolation, during the course. We propose a user similarity-based peer recommendation approach that makes use of the learners' score and their demographic attributes. For the isolated learner, we provide suggestions on potential learning peers.

The peer recommendation approach presented here is used to assist learners in selection of the most suited learner for interaction. Moreover, it is used to enhance engagement and social tutoring among learners, by reducing their feeling of isolation

and thus decreasing the probability of dropout. From the experimental results, it is seen that the proposed approach is well balanced between scalability and coverage. Since we identify the target learners, when using our approach, the recommendation is for the learners who have a feeling of isolation during the course.

The remainder of the paper is organized as follows. Section 2 presents related work in the field of peer learning, peer recommendation systems and adaptive interventions in e-learning platforms. In Sect. 3, the proposed approach for peer recommendation is described. Section 4 presents the experimental results. Lastly, Sect. 5 concludes the paper and presents future work.

## 2 Related Work

This section briefly describes related work in the field of peer learning, peer recommender systems and adaptive interventions used in online learning.

### 2.1 *Peer Learning in MOOC*

Peer learning is a method of learning, where students learn by interacting with one another. Many experts from prestigious universities have incorporated peer learning in their main methods. When a student learns with a peer, they learn their ideas, their thoughts and their views on a certain topic. Past studies [13, 15, 16] have shown that peer learning in MOOC results in better learning and increased performance of the learners. It brings them an opportunity to share thoughts and helps in the development of their intellectual skills [17, 18]. It has been observed that peer learning has resulted in more engagement in terms of attendance in the course and discussion forums [11] and thus has helped in decreasing the dropout rate of MOOC.

Many MOOC platforms offer features like discussion forums, group assignments and peer reviews to promote the idea of peer learning [19, 20]. However, the overcrowding of participants from diverse cultural and physical backgrounds [21] often makes it extremely difficult for learners to find potential peer learners for collaboration purposes.

### 2.2 *Peer Recommender Systems*

Recommender systems in online learning platforms [22] are being widely developed to provide personalized recommendations in various categories such as course recommender systems [23, 24], learning path recommender systems [25] and peer

recommender systems [11]. Peer recommendation systems (PRS) in e-learning platforms are an evolving research field and have scope for many development opportunities. Peer recommender systems are built on the concept of collaborative learning and help in enhancing the learning experience of learners.

Different approaches [8, 11, 18, 19, 26–31] have been used for peer recommendation in the context of online learning. Xu and Yang [26] have categorized learners into three groups: questioning learner ( $Q$ -learners), answering learner ( $A$ -learners) and normal learner ( $N$ -learners) on the basis of messages posted by them on the course discussion forum and computed the similarity over topics among forum learners using topic modelling. Answering learners were recommended to questioning learners with high topic similarity. In [11], a recommender system is designed to suggest relevant chat contacts on the basis of demographics and progression criteria (in terms of number of quizzes replied to). In another study, Rothkrantz [27] presents an approach to create balanced groups of students, by considering the personal characteristics and abilities of learners, and the requirements fixed by course authors.

Elghomary and Bouzidi [28] presents a dynamic peer recommendation system (DPRS) based on trust management system (TMS) considering the influence of the trust relationships among MOOC learners that impact strongly the selection of the suitable partner. In another work, Hu et al. [29] presented a framework for recommending learning peers using a tripartite graph by modelling dynamic interaction behaviours of learners. Lalingkar et al. [19] and Chounta [30] make use of principle of zone of proximal development (ZPD) for the recommendation of peers for the formation of study groups.

Another approach for recommendation of peer in online setting is the use of reciprocal recommender systems [18]. In their work, Prabhakar et al. [31] built a recommender system based on learners' profile that recommends mutually interested learners who can possibly interact with each other. Thanh et al. [8] implemented a learning partner recommender system (LPRS) that provides students with suggestions on learning partners based on their individual characteristics, what they look for in peers, and preferences in learning partners.

All the above approaches for peer recommendations are mainly based on either learners' profile and/or their behavioural attributes. We observe that these approaches provide no method to identify isolated learners, and hence, recommend peers to all the learners of the course, including the ones who are not feeling isolated.

### ***2.3 Adaptive Intervention***

Intervention are alterations made to the learning environment or learner's experience of it, like providing extra learning resources, prompting learners to return to the course or varying current course content [32]. Adaptive interventions are not new to education and have been extensively used in web-based adaptive educational systems. Various approaches have been used to implement adaptive interventions which can adapt based on the type of learner and change as a learner progresses

through the course. For example, Davis et al. [33] and Alevan et al. [34] designed an adaptive retrieval practice intervention by delivering quiz questions from course content accessed previously by the learner. Similarly, NeCamp et al. [32] proposed an adaptive intervention of sending weekly email reminders to motivate learners to engage with course content. In this paper, we propose adaptive interventions to identify the isolated learners, during the course.

### 3 Peer Recommendation Approach

This section presents our peer recommendation approach. Our approach has three steps: (1) using adaptive intervention to identify the learners having the feeling of isolation, (2) determining the cause of their feeling of isolation and (3) proposing the peer recommendation algorithm to generate potential peers for the isolated learner. The three steps are described in the following subsections, in detail.

#### 3.1 Identification of Target Learner

For our purposes, we define the target learner as the one who feels isolated during the pursuing of the course. To identify the target learners, we intervene during the course after the learner has attempted the quiz of the first module, to ask the learners about their feeling of isolation. The intervention is repeated after each module attempted by the learners (Fig. 1). The learners may either select that they have a feeling of

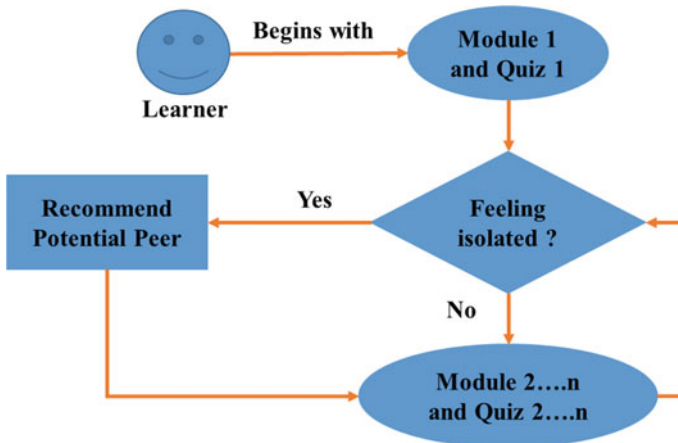


Fig. 1 Adaptive intervention for identification of target learner

isolation or that they do not feel isolated. The learners who select that they have a feeling of isolation are identified as target learners.

### 3.2 *Determination of Cause of Isolation*

We focus on the two common causes for the feeling of isolation among MOOC learners: (1) lack of one-to-one interaction with peer learners and (2) lack of personal assistance provided to learners while pursuing of MOOC. To determine the cause of isolation of the identified target learner, we consider the score of the latest quiz attempted by the target learner. If the latest quiz score is greater than or equal to a threshold value (set by the course author), we categorize the target learner as a *Good Scorer*, and else, the target learner is categorized as a *Bad Scorer*. For *Bad Scorer*, we consider the cause of isolation as lack of personal assistance needed in understanding of the course. On the other hand, in the case of *Good Scorer*, we consider the cause of isolation as lack of one-to-one interaction with peer learners.

Based on the cause of isolation, different recommendation parameters are used for both cases.

- For *Good Scorer*, we recommend peers based on demographic attributes, as the cause of isolation is due to lack of interaction.
- For *Bad Scorer*, we recommend peer learners based on the quiz score as well as demographic attributes. This is because the better scoring learner may provide personal assistance to the target learner.

Demographic attributes play an important role in enhancing peer interaction [35]. According to French [36], students from all age group tend to interact with similar age peers as they feel more connected with them. Likewise, same gender peers are likely to communicate more as they share similar interest [37]. Learners who share same level of education commonly have similar level of understanding and hence can effectively collaborate with each other. According to Loh and Teo [38], the learners who belong to same location share similar culture and also share similar learning style which, in turn, can facilitate more effective peer learning. Therefore, we identify the demographic attributes to be considered for peer recommendation as—Age, gender, geographical location and level of education.

### 3.3 *Peer Recommendation Algorithm*

In this section, a peer recommendation algorithm is presented for the selection of suitable peer learners for the target learner. The algorithm is executed every time a target learner is identified. As discussed in Sect. 3.2, the algorithm selects recommendation parameters based on the type of learner (*Good Scorer* or *Bad Scorer*).

Accordingly, different computations are performed for both cases to provide peer recommendation to the target learner.

Algorithm 1 presents the peer recommendation algorithm for our approach. The algorithm takes the learner database  $D$ , the value of  $k$  (number of peer to be recommended) and  $val$  (min score required to be a *Good Scorer*) as input and generates a list of  $k$  peer recommendations for the target learner  $T$ .

---

**Algorithm 1.** Generating a List of Peer Recommendations for Target Learner

---

```

2D Database of learners =  $D$ 
Similarity measure =  $SM$ 
Similarity score =  $sim\_score$ 
Number of recommendations to be provided for Target learner  $T = k$ 
Threshold value =  $val$ 
Score of the learner  $L_i = score[L_i]$ 
Input:  $D, T, k, val$ 
Output:  $k$  peer recommendations
1:    $FiltrD \leftarrow []$ ;
2:    $PeerD \leftarrow []$ ;
3:   if  $score[T] < val$  then //Case of Bad Scorer
4:     for each  $L_i \in \{D - T\}$  do
5:       if  $score[L_i] > score[T]$  then
6:          $FiltrD.append(L_i)$ ;
7:       end if
8:     end for
9:     for each  $L_i \in FiltrD$  do
10:       $sim\_score[L_i] = SM(L_i, T)$ ;
11:    end for
12:  else //Case of Good Scorer
13:    for each  $L_i \in \{D - T\}$  do
14:       $sim\_score[L_i] = SM(L_i, T)$ ;
15:    end for
16:  end if
17:   $PeerD$  – a sorted list of learners based on decreasing similarity scores
18:  return top  $k$  most similar learners from  $PeerD$ 

```

---

The algorithm uses the *score* of the latest quiz  $Q$  attempted by the target learner. On the basis of *score* and *val*, the target learner is first categorized as a *Good Scorer* or a *Bad Scorer*. If the target learner is a *Bad Scorer*, the scores and the demographic attributes are considered for recommendation. The learners who have scored greater than or equal to the target learner in the quiz  $Q$  are appended to *FilterD*. Next, the demographic similarity between each learner  $L_i$  of *FilterD* and the target learner is computed and stored in  $sim\_score[L_i]$ . On the other hand, if the target learner is a *Good Scorer*, the demographic similarity is directly computed between the target learner and all the other learners of the database  $D$ .

There exist several similarity measures, like cosine similarity, Euclidean distance, Pearson coefficient and Jaccard index. Any measure can be used for the calculation of the similarity scores as we find that there is compatibility in the results. The similarity measure is applied on the learners' demographic attributes (age, gender,

location, and level of education). Once all the similarity scores have been computed, learners are sorted in the decreasing order of their similarities. Top  $k$  similar learners are recommended to the target learner.

### 3.4 A Simple Example

In this section, we illustrate the working of the proposed algorithm using an example of ten learners (Table 1). We aim to generate three peer recommendations ( $k = 3$ ) for target learner  $B$ , with threshold value of the *score* as 50% ( $val = 5$ ). The data is taken from the release of de-identified data from the first year of MITx and HarvardX courses on the edX platform [39]. We selected records with attributes about age, location, qualification and gender. Also, we augmented this information with synthesized data about learners' quiz score.

As required by our algorithm, we pre-process the data of the learners to transform it into a format that can be used for the calculation of similarity scores. Next, the peer recommendation algorithm applied to the pre-processed data is discussed.

**Pre-processing.** Each learner has six attributes—*Name*, *location*, *level of education (LoE)*, *gender*, *age* and *score*. The database takes *age* as a numeric attribute while *gender* and *LoE* as categorical attributes. We convert *LoE* on a scale of 1 to 5 by categorizing it into five levels: less than secondary (1), secondary (2), bachelors (3), masters (4) and doctorate (5). *Gender* is assigned a value of 0 for male and 1 for female. For finding the *location* similarity of the learners with the target learner, we use Eq. (1), i.e. if the *location* is same as that of the target learner, then the value is 1, else 0. After this, we normalize the data using min–max scalar to scale all the attribute values between 0 and 1.

**Table 1** Sample database of ten learners

Name	Location	LoE	Gender	Age	Score
A	India	Bachelor's	M	23	2
B	United States	Secondary	M	19	4
C	United States	Bachelor's	M	24	7
D	Australia	Secondary	M	20	2
E	Australia	Master's	F	32	5
F	Mexico	Bachelor's	M	22	10
G	Brazil	Bachelor's	M	23	1
H	Other South Asia	Master's	F	32	6
I	India	Bachelor's	M	20	8
J	United States	Bachelor's	M	26	9



**Table 2** Demographic similarity of filtered learners with target learner *B*

Name	Cosine	Euclidean	Pearson	Jaccard
C	0.84578	0.684592	0.855106	0.3333
J	0.805838	0.632597	0.799908	0.3333
I	0	0.439662	-0.40075	0
F	0	0.429199	-0.51202	0
E	0	0	-1	0
H	0	0	-1	0

$$\text{sim}(\text{Loc}_{L_i}, \text{Loc}_T) = \begin{cases} 1, & \text{Loc}_{L_i} = \text{Loc}_T \\ 0, & \text{Loc}_{L_i} \neq \text{Loc}_T \end{cases} \quad (1)$$

**Peer recommendation.** As described in Sect. 3.3, since, the score (*score* = 4) of the target learner is less than *val*, therefore, the target learner is the case of a *Bad Scorer*. Accordingly, first, all the learners who have scored greater than the target learner *B* are selected. Six out of nine learners (excluding the target learner) in Table 2 have scored greater than the target learner. These six learners (*C*, *E*, *F*, *H*, *I*, and *J*) are appended to a list of filtered learners. After this, the proposed algorithm calculates the similarity scores between each of the filtered learners and the target learner.

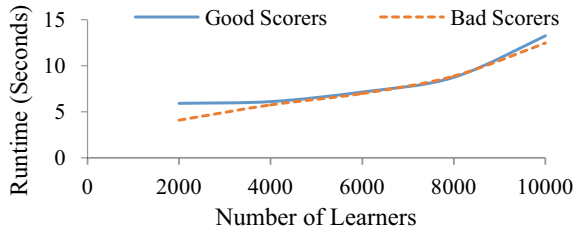
We calculated similarity scores by using four different types of similarity measures—Cosine similarity, Euclidean distance, Pearson coefficient and Jaccard Index (Table 2). For this example, the algorithm recommends three potential peers. We see that all the measures include *C* and *J* as the top two recommended peers. However, for the recommendation of third peer, Euclidean and Pearson measure recommends learner *I*, whereas Cosine and Jaccard measure recommends learners *I*, *F*, *E* and *H*, as they all obtain the same similarity score. We infer that the peers recommended are compatible across all similarity measures. Furthermore, we recommend the usage of Euclidean and Pearson measures as they provide more detailed values of similarity, with lesser chances of overlap, as compared to Cosine and Jaccard measure.

Thus, for the target learner *B*, peer learners *C*, *J* and *I* are recommend as potential learners for interaction and collaboration purposes.

## 4 Evaluation

To evaluate the proposed algorithm, we employ two standard measures—Scalability and coverage. We apply our approach on the data of 10,000 learners obtained from the first year of MITx and HarvardX courses on the edX platform.

**Fig. 2** Runtime of algorithm 1 with respect to number of learners

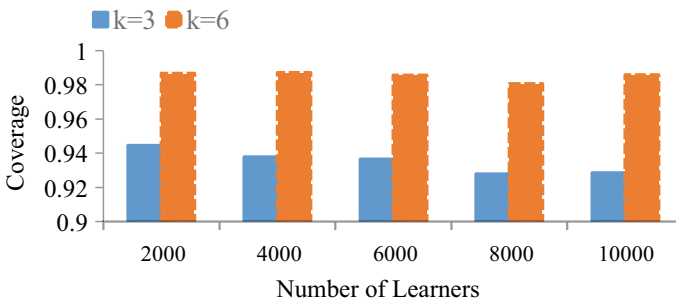


## 4.1 Scalability

The running time of algorithm plays an important role in our approach as the peer recommendation approach is on demand and real time. Therefore, we evaluate our proposed algorithm for scalability to see the time it takes for the peer recommendation. The scalability of our algorithm is tested by varying the total number of learners in a range of 2000–10,000 learners. The runtime of algorithm with respect to different number of learners belonging to both the cases, keeping the total number of recommended peers as three ( $k = 3$ ), is reported in Fig. 2. As expected, the runtime of the algorithm increases as number of learners increased. We see that even in the current setting, the recommendation in a course with 10,000 learners can be provided within a few seconds.

## 4.2 Coverage

We evaluate the coverage of our algorithm to ensure that the same learners are not repeatedly recommended to all the isolated learners. We calculate the coverage by finding out the proportion of learners who were recommended at least once to other learners. To test the coverage our algorithm, we varied the total number of learners in a range of 2000–10,000 learners. Figure 3 shows the coverage with respect to



**Fig. 3** Coverage versus number of learners

different number of learners for different number of peer recommendations ( $k = 3$  and  $k = 6$ ). Let us denote  $rec(L_i)$  as the recommendation list of learner  $L_i$ . Let  $n$  be the set of all learners in the course. Then, the *coverage*, as defined by [40], can be measured as follows:

$$\text{coverage} = \frac{|\cup_{i=1\dots n} rec(L_i)|}{|n|} \quad (2)$$

We observe that under all the settings, the coverage is close to 1, which shows that the same learners are not repeatedly recommended in our approach. However, it must be noted that the calculated coverage depends upon the demographic attributes and scores of learners at any given point in time. The coverage shown may not be universally correct.

## 5 Conclusion and Future Work

Learning is a social process, and thus, encouraging interactions among learners is an effective way to keep them engaged and subsequently improves the completion rate of MOOC. An approach for peer recommendation in MOOC, based on the learners' scores and their demographic attributes, is presented in this paper. The approach facilitates to identify the target learners who are feeling isolated in the course and also helps in determining the probable cause of their feeling of isolation. Accordingly, suggestions on potential learning peers are provided to the target learner.

Our approach eases the decision-making process to select relevant peer by MOOC learners and encourages the idea of peer learning. An experimental validation is carried out to determine the scalability and coverage of the proposed algorithm. From the results, it can be seen that the proposed algorithm is well balanced between scalability and coverage. However, there are some limitations of our work. The scalability is calculated by considering the runtime of single target learner only, though there may be large number of target learners. Also, although we provide peer recommendations to isolated learners, but the successful interaction among learners is subject to reciprocity factor [41] between the target learner and the recommended learner. In future, we are considering to use mathematical modelling to evaluate our recommendation algorithm. Moreover, we plan to integrate some more learner attributes for the process of recommendation, like learners' interests, engagement levels and their conversation skills.

## References

1. K.M. Alraimi, H. Zo, A.P. Ciganek, Understanding the MOOCs continuance: the role of openness and reputation. *Comput. Educ.* **80**, 28–38 (2015)

2. T.R. Liyanagunawardena, A.A. Adams, S.A. Williams, MOOCs: a systematic study of the published literature 2008–2012. *Int. Rev. Res. Open Dist. Learn.* **14**(3), 202–227 (2013). ISSN: 1492-3831
3. S. Yin, Q. Shang, H. Wang, B. Che, The analysis and early warning of student loss in MOOC course. in *ACM TURC '19 Proceedings of the ACM Turing Celebration Conference—China* (2019), pp. 1–6. <https://doi.org/10.1145/3321408.3322854>
4. F. Dalipi, A.S. Imran, Z. Kastrati, MOOC dropout prediction using machine learning techniques: review and research challenges. in *2018 IEEE Global Engineering Education Conference (EDUCON)* (2018), pp. 1007–1014
5. F. Bouchet, H. Labarthe, K. Yacef, R. Bachelet, Comparing peer recommendation strategies in a MOOC. in *Adjunct Publication of the 25th Conference on User Modeling, Adaptation and Personalization* (2017)
6. Y. Pang, W. Liu, W., Jin, H. Peng, T. Xia, Y. Wu, Adaptive recommendation for MOOC with collaborative filtering and time series. *Comput. Appl. Eng. Educ.* **26**, (2018). <https://doi.org/10.1002/cae.21995>
7. K.S. Hone, G.R. El Said, Exploring the factors affecting MOOC retention: a survey study. *Comput. Educ.* **98**, 157–168 (2016)
8. T.N. Thanh, M. Morgan, M. Butler, K. Marriott, Perfect match: facilitating study partner matching. in *Proceedings of the 50th ACM Technical Symposium on Computer Science Education* (2019)
9. J.M. Galusha, Barriers to learning in distance education. *Interpers. Comput. Technol. Electron. J. 21st Century* **5**(3/4), 6–14 (1998)
10. T.K.F. Chiu, T.K.F. Hew, Factors influencing peer learning and performance in MOOC asynchronous online discussion forums. *Australas. J. Educ. Technol.* **34**(4), 16–28 (2018). <https://doi.org/10.14742/ajet.3240>
11. H. Labarthe, F. Bouchet, R. Bachelet, K. Yacef, Does a peer recommender foster students' engagement in MOOCs? in *9th International Conference on Educational Data Mining* (Raleigh, United States, 2016), pp. 418–423
12. H. Labarthe, R. Bachelet, F.R. Bouchet, K. Yacef, Increasing MOOC completion rates through social interactions: a recommendation system. in *EMOOCs 2016 Conference. Fourth European MOOCs Stakeholders Summit* (University of Graz (Austria), Graz, Austria 2016), pp. 471–480
13. J.W. Peltier, W. Drago, J.A. Schibrowsky, Virtual communities and the assessment of online marketing education. *J. Mark. Educ.* **25**(3), 260–276 (2003). <https://doi.org/10.1177/0273475303257762>
14. M. Martínez-Núñez, O.B. Gené, Á.F. Blanco, Social community in MOOCs: practical implications and outcomes. *TEEM'14* (2014)
15. C. Reidsema, L. Kavanagh, E. Ollila, S. Otte, J. McCredden, Exploring the quality and effectiveness of online, focused peer discussions using the MOOCchat tool. in *27th Australasian Association for Engineering Education Conference* (2016)
16. Q. Tang, A personalized learning service for MOOCs. *J. World Trans. Eng. Technol. Educ.* **14**(1), 140–145 (2016)
17. B. Kieslinger, J. Tschank, T. Schaefer, C.M. Fabian, Working in increasing isolation? How an international MOOC for career professionals supports peer learning across distance. *Int. J. Adv. Corp. Learn.* **11**, 23–30 (2018)
18. B.A. Potts, H. Khosravi, C. Reidsema, A. Bakharia, M. Belonogoff, M.K. Fleming, Reciprocal peer recommendation for learning purposes. in *8th International Conference on Learning Analytics and Knowledge* (2018)
19. A. Lalingkar, S. Srinivasa, P. Ram, Characterization of technology-based mediations for navigated learning. *Adv. Comput. Commun.* **3**(2), 33–47 (2019)
20. S. Kellogg, S. Booth, K. Oliver, A social network perspective on peer supported learning in MOOCs for educators. *Int. Rev. Res. Open Distrib. Learn.* **15**, 263–289 (2014)
21. D. Leris, M.L. Sein-Echaluce, M. Hernández, A. Fidalgo-Blanco, Relation between adaptive learning actions and profiles of MOOCs users, in *Fourth International Conference on Technological Ecosystems for Enhancing Multiculturality*. TEEM 2016. (ACM, New York, 2016), pp. 857–863

22. N. Manouselis, H. Drachslar, R. Vuorikari, H. Hummel, R. Koper, Recommender systems in technology enhanced learning. in *Recommender Systems Handbook* ed. by F. Ricci, L. Rokach, B. Shapira, P. Kantor (Springer, Boston, MA 2011). [https://doi.org/10.1007/978-0-387-85820-3\\_12](https://doi.org/10.1007/978-0-387-85820-3_12)
23. Y. Pang, Y. Jin, Y. Zhang, T. Zhu, Collaborative filtering recommendation for MOOC application. *Comput. Appl. Eng. Educ.* **25**(1), 120–128 (2017)
24. F. Bousbahi, H. Chorfi, MOOC-Rec: a case based recommender system for MOOCs. *Procedia Soc. Behav. Sci.* **195**, 1813–1822 (2015)
25. Z.A. Pardos, S. Tang, D. Davis, C.V. Le, Enabling real-time adaptivity in MOOCs with a personalized next-step recommendation framework. in *Proceedings of the Fourth ACM Conference on Learning@Scale* (ACM, Cambridge, MA, USA, 2017), pp. 23–32
26. B. Xu, D. Yang, Study partners recommendation for xMOOCs learners. *Comput. Intell. Neurosci.* **2015**, 1–10 (2015)
27. L. Rothkrantz, How social media facilitate learning communities and peer groups around MOOCs. *Int. J. Human Capital Inf. Technol. Professionals* **6**, 1–13 (2015)
28. K. Elghomary, D. Bouzidi, Dynamic peer recommendation system based on trust model for sustainable social tutoring in MOOCs. in *1st International Conference on Smart Systems and Data Science (ICSSD)* (2019), pp. 1–9
29. Q. Hu, Z. Han, X. Lin, Q. Huang, X. Zhang, Learning peer recommendation using attention-driven CNN with interaction tripartite graph. *Inf. Sci.* **479**, 231–249 (2019)
30. I. Chounta, Using learning analytics to provide personalized recommendations for finding peers. *ArXiv, abs/1910.07381* (2019)
31. S. Prabhakar, G. Spanakis, O.R. Zaiane, Reciprocal recommender system for learners in massive open online courses (MOOCs). (Springer International Publishing, Cham, 2017), pp. 157–167
32. T. NeCamp, J. Gardner, C. Brooks, Beyond A/B testing: sequential randomization for developing interventions in scaled digital learning environments. in *Proceedings of the 9th International Conference on Learning Analytics and Knowledge* (2019)
33. D. Davis, R.F. Kizilcec, C. Hauff, G.-J. Houben, The half-life of MOOC knowledge: a randomized trial evaluating the testing effect in MOOCs. in *Proceedings of the 8th International Conference on Learning Analytics and Knowledge (LAK)* (2018).
34. V. Alevan, J. Sewall, J.M. Andres, R.A. Sottolare, R.A. Long, R. Baker, Towards adapting to learners at scale: integrating MOOC and intelligent tutoring frameworks. in *Proceedings of the Fifth Annual ACM Conference on Learning at Scale* (2018)
35. M. Rivera, S.B. Soderstrom, B. Uzzi, Dynamics of dyads in social networks: assortative, relational, and proximity mechanisms. *Rev. Soc.* **36**, 91–115 (2010)
36. D.C. French, Children’s social interaction with older, younger, and same-age peers. *J. Soc. Pers. Relat.* **4**(1), 63–86 (1987)
37. C.M. Mehta, J. Wilson, Gender segregation and its correlates in established adulthood. *Sex Roles* **83**, 240–253 (2020). <https://doi.org/10.1007/s11199-019-01099-9>
38. C. Loh, T. Teo, Understanding Asian students’ learning styles, cultural influence and learning strategies. *J. Educ. Soc. Policy* **7**(1), 194–210 (2017)
39. A.D. Ho, J. Reich, S.O. Nesterko, D.T. Seaton, T. Mullaney, J. Waldo, I. Chuang, HarvardX and MITx: the first year of open online courses, fall 2012-summer 2013 (harvardX and MITx working paper no. (1). SSRN J. (2014)
40. M. Ge, C. Delgado-Battenfeld, D. Jannach, Beyond accuracy: evaluating recommender systems by coverage and serendipity. in *Proceedings of the fourth ACM conference on recommender systems (RecSys ‘10)* (ACM, New York, NY, USA, 2010), pp. 257–260
41. M. Saqr, J. Nouri, H. Vartiainen, J. Malmberg, What makes an online problem-based group successful? A learning analytics study using social network analysis. *BMC Med. Educ* **20**, 80 (2020). <https://doi.org/10.1186/s12909-020-01997-7>

# Bag of Science: A Query Structuring and Processing Model for Recommendation Systems



Prakash Hegade, Vibha Hegde, Sourabh Jain, Rajaram M. Joshi,  
and K. L. Vijeth

**Abstract** Technological advancements and the changing needs drive the process workflow, meeting the need-of-the-hour requirements, and calibrating system components. While the perception evolves, the fundamental principles stay put and wrap around generational disparities. In a changing scenario of the physical market to an e-commerce site, the recommendation systems have had substantial roles. The present systems customarily use the item or user profiles for recommendations. The existing recommendation systems rely heavily on data and learning algorithms. An improved recommendation system given by considering the query's semantics rather than using only historical data of numerous worldwide queries can create a paradigm shift in the technologies involved in computer recommendations. Bag of science attempts to take on this challenge. The model dwells on inferring a query's meaning in all contexts to create an order in which the words relate. By constructing a word definition graph, the methodology explores the possibility of enhancing the recommendation systems to improve the e-commerce platforms' business. The paper presents the model's architecture with its chief components, including a parser and scraper, graph generator, graph traversal, and results. The model presents the traversal results and analysis of the constructed e-commerce graphs using hops as the threshold metric. The paper also presents the model's abstract data type to make it applicable and extend to other domain contexts that involve query engines and need recommendations.

**Keywords** Bag of science · Graphs · Recommendations

## 1 Introduction

The business and consumer subtleties regarding the producer–consumer association have seen structural and behavioral changes since their inception. The barter system allowed the producers to trade their surplus to surfeit, and the common meeting place for the barter evolved into market places [1]. Further, the need to equate the values

---

P. Hegade (✉) · V. Hegde · S. Jain · R. M. Joshi · K. L. Vijeth  
KLE Technological University, Hubballi, Karnataka, India  
e-mail: [prakash.hegade@kletech.ac.in](mailto:prakash.hegade@kletech.ac.in)

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2021  
S. M. Thampi et al. (eds.), *Advances in Computing and Network Communications*,  
Lecture Notes in Electrical Engineering 735,  
[https://doi.org/10.1007/978-981-33-6977-1\\_19](https://doi.org/10.1007/978-981-33-6977-1_19)

231

to a common denomination led to the invention of currency. While businesses grew from trading, business places coined markets. Modern-day markets have transformed from shops to supermarkets [2]. With the advent of technology, markets have evolved into virtual shops under the envelope of e-commerce and adaptive progression [3].

As paradigm shifts have led the world into a digital transformation, trade principles have engaged new facades, nurturing over the classics, to name one—recommendations. To be precise, despite the transaction's mode and nature, the component elemental to any purchase is a good recommendation. While a shopkeeper talks to the customer into buying products based on personal contact about day-to-day life and needs, a supermarket uses strategic arrangements based on what products can be purchased together [4]. However, these do not translate well, yet on to the e-commerce sites.

Online shopping recommendations rely majorly on collaborative filtering and context-based filtering [5]. They are outlying from what is precise and accurate concerning the customer's needs despite their success. The field presents itself with ample scope for improvement [4]. The quest for a good recommendation must begin at the query and stretch beyond the data available in the history. A query on any present-day search engine goes through a combination of semantic [6] and syntactic [7] layers to rank pages as results. Some search engines also use cookies and search history to promote sponsored content [8] and make the search results akin to a user's activity rather than what is accurate. These results may or may not have any correlation to the user query, specifically in scientific data. For the purpose of bringing the underlying semantics of the query to light, user queries are treated as scientific data.

In case that a user query is more than just a word, that is, when it represents any conceptual idea or a science [9], the query results need to be processed differently. A recommendation system's emblematic components include query structuring and processing, data processing, candidate generation, scoring, and analysis. The present models work on improvising one or related sub-components in the process, masking the holistic picture. The gaps present in the system necessitate a need to work on the model's science, establishing the correlation between the sub-components working toward a condescending goal. The domain knowledge from web parsing and scraping, graphs traversals, data structures [10], scientific data stores, statistics, and inferences can aid the betterment process.

This paper proposes a query structure, processing, and analysis model for recommendation systems—Bag of Science (BoS). The paper is further divided into the following sections: Sect. 2 presents the literature survey. Section 3 presents the model, design principles, and architecture. Section 4 presents the results and discussion, and Sect. 5 concludes the paper along with the future scope.

## 2 Literature Survey

We have sewed together the relevant principles to build a BoS model from search engines, recommendation systems, web parsing, web scraping, e-commerce, and query processing systems. This section presents a brief survey on each of the pointed out components. Alan Emtage, Bill Heelan, and J. Peter Deutsch created the very first automated searching tool “Archie” in 1990. Archie stored all the files located on public anonymous FTP, creating a searchable database of file names. From Archie, we have to-the-time steadied to Google, amidst the prevailing challenges [11]. The search engine capability, which was once limited to string match, now uses natural language processing and semantic algorithms to improvise the search [12, 13]. Although search engines provide better recommendations to the user, they lack the generation of definitions for various scientific queries. There are also generic search engine models conceived to offer a parent–child fork model suiting to current needs [14].

Search engines are vital to the e-commerce domain. The procedure of shopping online starts with a search. Buying anything online has a concrete and established stage-wise process. The user first searches for the item he/she wants, adds a specific item to the cart, and then checks out by making the payment. The recommendation system has a huge role to play in what the customer buys. Recommendation systems have been designed for various tasks and preferences based on data models and system institutional designs [15]. The principles, evaluations, and design have been domain specific [16]. A recommendation has a vital role in marketing and e-commerce and one of the major marketing strategies. Machine learning has been used as a maneuver to make better recommendation systems using historical data [17]. They have also been designed for personalized promotions [18]. Systems usually also recommend based on factors like best-selling products, or based on the consumer’s demographics, or analysis of the customer’s purchase history and behavior [19]. The process usually is based on text mining and utility matrix operations. Although intelligence and machine learning are mainly responsible for the operation of today’s recommendation systems, they blemish in providing connotation of patterns and identifying contributory associations.

On the other hand, search and recommendation engines process user queries. Any query processing engine needs data to be operated on and with. The data can be from the stored database or mined from the web of relevant sciences. Web mining is the process of discovering potentially useful and previously unknown information or knowledge from the web data [20]. Web content mining is a type of web mining technique used by search engines to search the web via content, i.e. discover useful information from web content such as text, images, videos, etc. [21]. Scraping the web plays an essential role in web mining. The process includes finding and parse web pages from a set of defined web links returned by a search engine query [22].

Graphs and graph data structures have been exhaustively used in query processing. Graph query languages and operative measures have been designed as well [23]. Another advantage that graphs offer is that user queries’ intermediate results can be efficiently managed with graph data structures. Graphs offer traversal mechanisms based on breadth and depth [10]. Content-based recommendations have been hence designed using knowledge graphs [24].



Bag of science seeks motivation from the design principles of bag of words, a popular representation used for object categorization. In this model, a text is represented in the form of words, irrespective of order and grammar but with respect to multiplicity [25]. BoS is designed considering each of the mentioned systems' principles and advantages by putting them together, buoyantly, to address the gaps present in the recommendation models. BoS attempts to address the problem by giving a unified data structure that starts from a user query and ends by generating the results putting all the intermediate processing steps into the graph as nodes.

### **3 Bag of Science: Model Design and Deliberations**

The language is a universal web of words. A human being begins to understand the connections and interactions between the words as he grows to learn the world's ways. Similarly, the existence of any object in the universe is a relative space in the bag of science. There is a way that each existence, each concept, and each science are related, and unification of this has been a collective goal. BoS derives this goal, in principle, to attempt to understand the context of a query in terms of the definition of a science. The closer the sciences connect in the data structure, the greater the contextual similarity. A chair can be related to a table. BoS quantifies this relationship in terms of the number of nodes that need to be traversed to reach one from the other. This completely disregards the need to analyze the data around how many individuals think that the table and chair are related or need to be. In a utopian universe, the BoS connects every science there is, despite the query. A context is built through the definition of different concepts and sciences to build one enormous web, a relevance measure of a universal dictionary. This section walks through the BoS model, design, and deliberations.

#### ***3.1 Design Goals***

The design goals of the model are as follows: Firstly, to realize a structure to hold the scraped and parsed web data, secondly, to spawn and clamp semantically connected nodes in the graph structure, and thirdly, generate meaningful inferences from the user query to enrich recommendations.

#### ***3.2 BoS Architectural Components***

The BoS model includes a query processing, web scraping, graph formation, and traversal modules (Fig. 1).

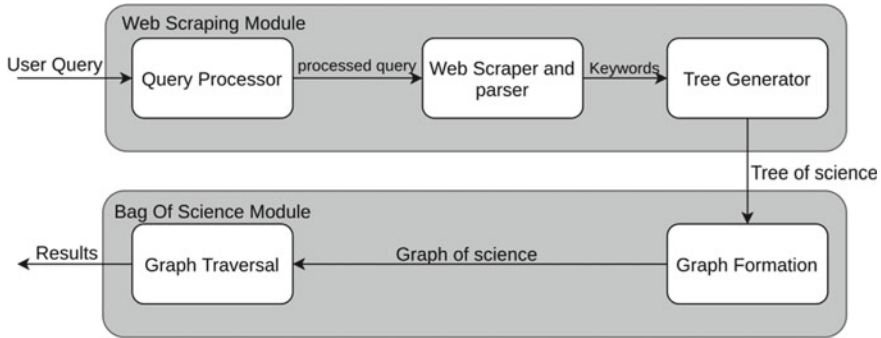


Fig. 1 Bag of science architecture

**Web Scraping.** This module implements a web scraper that retrieves scientific definitions and related terminologies to user queries from selected web spaces. Of all the sciences available, the module semantically chooses five definitions that are relevant to the queried word (based on ranking on the Google search). It then processes these selections using natural language processing and returns the list of  $n$  most occurring keywords across all sciences to the bag.

A tree data structure is built with the user query as the root node and the chosen words after scraping become the children nodes of the root. Iteratively, each of these keywords is then scraped for, returning a list of words of their own, which constitute the secondary children nodes and so on. Thus, a tree of  $m$  levels of relevant sciences is built after scraping. The computational constraints limit us to stop the build at six levels. Hence, at the end of the scraping, we have a complete five-ary six-level tree.

$$\text{Number of keywords} = n^0 + n^1 + n^2 \dots + n^{m-1} \tag{1}$$

Equation (1) can be rewritten as:

$$\text{Number of keywords} = \sum_{i=0}^{m-1} n^i \tag{2}$$

Since no scraping is made on leaf nodes, the total number of queries can be seen below in Eq. (3), where  $m > 1$ .

$$\text{Total queries} = \sum_{i=0}^{m-2} n^i \dots \tag{3}$$

The web scraping module, as explained above, is pictorially presented in Fig. 2.

**Graph Module.** This module converts the scraping module’s tree structure into a graph data structure by semantically comparing the words and drawing edges between

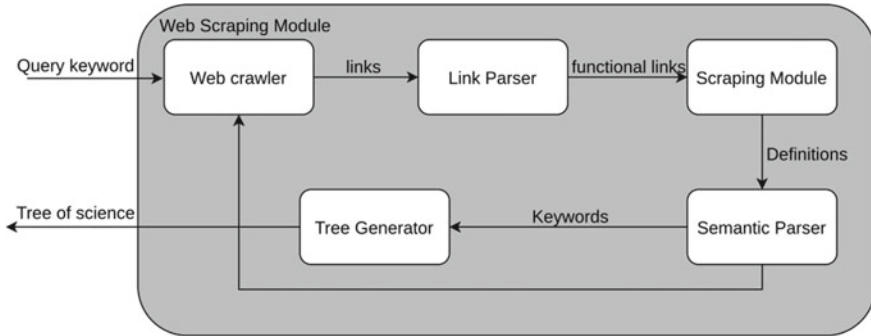


Fig. 2 Web scraping module

the same/similar words. Once cycles are formed by the new edges generated on the tree, the sub-tree connected to the node at the lower level is terminated. This eliminates redundancy in the structure. We then build a graph data structure that connects all relevant sciences for the queried keyword.

As seen in Fig. 3, the tree rooted at “r” has three pairs of identical words, namely  $(a, b)$ ,  $(p, q)$ , and  $(x, y)$ . Note that, these nodes need not be leaf nodes of the tree. If identical words are present on the internal nodes, the sub-tree connected to the node on the lower level is pruned. If the nodes are on the same level, either of the sub-trees are pruned away.

Once the sub-trees are pruned, the node at the lower level among the similar nodes is discarded, and an edge is added between the nodes at the higher level of the tree and the parent of the node present at the lower level. Here, the edges are formed between  $(a, c)$  and  $(p, s)$ , where  $c$  is parent of  $b$  and  $s$  is parent of  $q$  and finally, between  $(x, z)$  where  $z$  is parent node of  $y$ . This abridged graph is shown in Fig. 4.

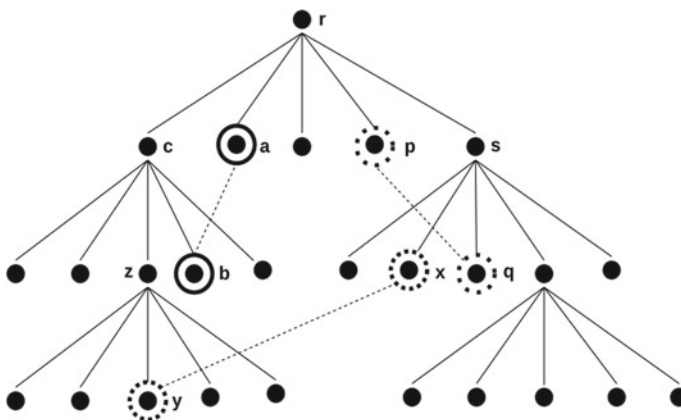


Fig. 3 Generated keyword graph

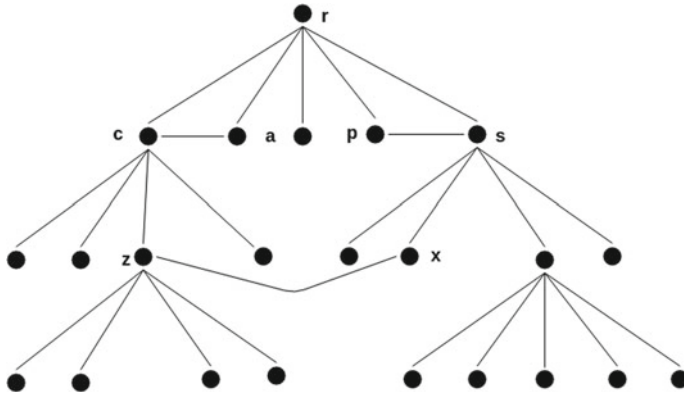


Fig. 4 Pruned graph of keywords

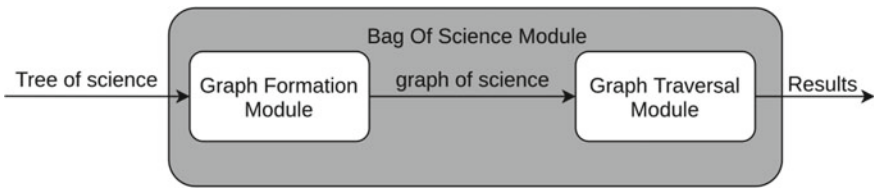


Fig. 5 Bag of science—graph module

The module further works to generate the results via tree traversals. We use depth-first search [26] and Floyd–Warshall algorithms [27]. A depth-first search is performed on the graph data structure, starting with the root node. Depth-first search gives nodes’ reachability in a single traversal of the graph by ranking the nodes contextually from the most relevant to the least relevant of the sciences. It allows us to recognize the relationship between the keywords present in the graph data structure. The Floyd–Warshall algorithm generates the shortest path between all pairs of keywords present in the graph data structure. This is used to measure the relation between keywords by counting the number of hops required to reach one keyword from another. It quantifies the closeness of any two words in the bag. The module can be summed up, as seen in Fig. 5.

### 3.3 Algorithms

This section presents the algorithm of the various modules discussed in Sect. 3.2. The summary of all algorithms is presented in Table 1 below.

**Algorithm:** ScrapeKeyDefinitions (key)

**Table 1** Algorithms overview

Algorithm	Description
ScrapeKeyDefinitions	Returns five relevant keywords for a given key by scraping definitions from the web
AddEdge	Adds a bi-directional edge between two nodes in a graph
AddTreeNode	Forms a tree of science by recursively obtaining relevant keywords
TransformToGraph	Converts the tree to a graph of science by adding edges between similar/same nodes
TraverseGraph	Performs depth-first search traversal and applies Floyd–Warshall’s algorithm on the graph
BagOfScience	Builds a graph of science on the basis of a user query and performs traversals to provide meaningful results

```

//Input:      key - keyword
//Output:    a list of keywords relevant to the query keyword
//Description: uses the input keyword and returns the top five keywords from the
//            scraped definitions
definition_list ← ∅
keywords ← ∅
links ← top five search engine links on key
for all the links li do
    definition ← scrape definition from li
    definition_list ← {definition_list} U {definition}

for all definition from definition_list do
    keys ← get_keys()
    keywords ← {keywords} U {keys}
countDict ← {}
for key in keywords do
    if key not in countDict then
        countDict[key] ← 0
    else
        countDict[key] ← countDict[key]+1
sort countDict values in descending order
len ← 0
while len < 5
    keywords ← {keywords} U countDict[len]
    len←len+1
return keywords

```

**Algorithm:** AddEdge (graph, key1, key2)

```

//Input:      graph, key1, key2
//Output:     graph
//Description: adds edge between two nodes
Graph[key1] ← {Graph[key1]} U {key2}
Graph[key2] ← {Graph[key2]} U {key1}
return graph

```

**Algorithm:** AddTreeNode(graph, keywords)

```

//Input:      graph, keywords - which is a list of keys
//Output:     the corpus fits into the tree data structure
//Description: generates tree by recursively finding keywords for the given keys
//            in keywords
wordCount ← 1
for keyword in keywords do
  if wordCount >= countLimit then
    break
  relevantKeywords ← ScrapeKeyDefinitions(keyword)
  for key in relevantKeywords do
    graph ← AddEdge(graph, key, keyword)
    keywords ← {keywords} U {key}
    wordCount ← wordCount + 1
return graph

```

**Algorithm:** TransformToGraph (graph, keywords)

```

//Input:      graph (a tree to be specific), keywords - list of keys
//Output:     the corpus fits into the graph data structure
//Description: generates graph by connecting common keywords
n ← len(keywords)
for keyIndex1 ← 1 to n do
  for keyIndex2 ← keyindex1+1 to n do
    if keywords[keyIndex1] = keywords[keyIndex2] then
      graph ← AddEdge(graph, keywords[keyIndex1], keywords[keyindex2])

```

**Algorithm:** TraverseGraph(graph)

```

//Input:      generated graph
//Output:     graph traversal results
//Description: it applies Depth First Search traversal and Floyd Warshall's algorithm
//            on the graph to derive relevant inferences
Traversal ← dfs (graph)
distanceMatrix ← floyd(Graph, Traversal)
print distanceMatrix

```

**Algorithm:** BagOfScience()

```

//Input:      nil
//Output:     graph traversal results
//Description: it builds a graph data structure from the user input keyword and applies
//             traversals
key ← user input
keywords ← {key}
graph ← {key ← {}}
AddTreeNode(graph, keywords)
TransformToGraph(graph, keywords)
TraverseGraph(graph)

```

**3.4 BoS Abstract Data Type**

This section presents the Abstract Data Type (ADT) for the designed data structure. This can help to customize and apply to different domain models and applications. The ADT abstracts the operations to customize the functionality to the required domain and maneuvers.

```

abstract typedef <<eltype>> BOS(eltype);
abstract eltype isCyclePresent(bos)
BOS (eltype) bos;
postcondition isCyclePresent == (numberOfCycles >= 1)
abstract eltype numberOfNodes(bos)
BOS (eltype) bos;
postcondition numberOfNodes == len(keywords)
abstract createTree(bos)
BOS (eltype) bos;
postcondition createTree == (numberOfNodes(bos) > 0)
abstract createGraph(bos)
BOS (eltype) bos;
precondition isCyclePresent(bos) == FALSE
postcondition createGraph == (isCyclePresent(bos) == TRUE)
abstract eltype traversal(bos)
BOS (eltype) bos;
precondition isCyclePresent(bos) == TRUE
postcondition traversal == DFSTraversal
abstract eltype allPairShortestPaths(bos)
BOS (eltype) bos;
precondition isCyclePresent(bos) == TRUE
postcondition allPairShortestPaths == distanceMatrix

```

### 4 Results and Discussion

The BoS was applied to e-commerce data, and this section discusses the results and analysis. The model was tested by giving keywords as inputs, and one such result graph for the keyword—Television is presented in Fig. 6. The tree is presented only till level 2 for the sake of simplicity and presentation. The graph was generated after taking an input keyword and then recursively obtaining the relevant keywords by processing the definitions scraped from the web, as explained in Sect. 3.

The model was run with several keywords, and the data was collected for analysis. The code implemented in python was added with the condition to crawl and parse only those pages that were permitted through bot.txt. Figure 7 presents the average of common keyword elimination count when run over a hundred different keywords. The figure can be read as, for example, in fourth hop, there was an average of at least 25 common keywords (Level 0 is hop 1).

For further analysis, a graph was plotted (Fig. 8) between hops and the number of keywords reachable from the root. It was evident from the graph that though the similar keywords at each level were eliminated, the number of reachable keywords does not vary much until four hops and only drops significantly at the fifth hop (Level 0 in the tree is treated as hop 1 and so on). This is because new paths were formed

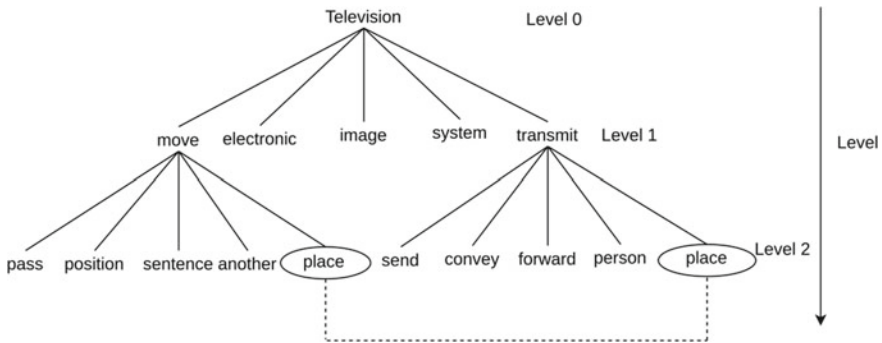
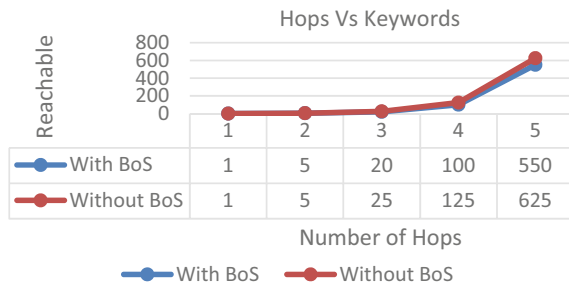


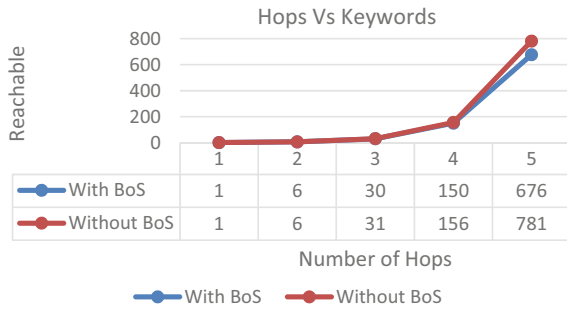
Fig. 6 Graph structure for keyword—television

Fig. 7 Hops versus keyword results





**Fig. 8** Hops versus number of reachable keywords



**Table 2** BoS recommendations

Keyword	Recommended keywords
Television	living, expression, action, etc
Mobile phone	cell, complex, connected, etc
Shirt	part, section, divided, etc

between keywords during the elimination of the common keywords leading to more reachable keywords at each hop. This was an effect of applying the traversal and shortest path algorithm.

These results were further used as an aid for recommendation engines on which words to be considered as related words based on keywords reachability and hop count. This process gave new keywords that otherwise are not related to a traditional recommendation engine. The table below (Table 2) presents the words recommended by BoS for sample keywords. These were the keywords obtained after three hops on the constructed tree having connected with the degree of the node with at least three. The graph keywords are helpful in finding the related and new perception of the given keyword, adding to science.

The table (Table 3) gives important metrics or data structures and their functionality. The table presents the related metrics of BoS, their description, and the functionality implication toward building a recommendation engine.

Alternately, a static approach for storing sciences was also attempted for the sake of experimentation. In this approach, definitions of around 60,000 words were collected from the wordnet module in NLTK [28]. A graph consisting of all these words was generated by adding a directed edge between two nodes if the definition of one word contains the other. Later, depth-first search and Dijkstra’s algorithms were used to find paths between two queried keywords. This approach was discarded because of insufficient data relating to the e-commerce domain.

**Table 3** BoS metrics and description

Metric	Description	Functionality
Distance matrix	2D matrix containing the smallest number of hops required to reach one node from another	Quantifies how closely the words are associated with each other
Traversal	Depth-first search traversal of the graph starting from the root (user query) node	Gives the reachability of nodes in a single traversal of the graph, by ranking the nodes contextually from the most relevant to the least relevant of the sciences
Average hops	The average number of hops required to reach one node from another	Gives the average degree of association between nodes
Reachable nodes	The nodes that are reachable from the root within a given number of hops	Allows to choose only those words which are closely related to the root

## 5 Conclusion and Future Scope

While efforts have been made to build intelligent crawlers and parsers [29], so is the requisite for recommendation engines. The bag of science model scrapes the web for relevant sciences for the queried keyword, realizes the keywords in the definitions, identifies them as intermediate sciences, and repeats this process until a general layer criterion is met. BoS encapsulates all the sciences generated in a graph data structure by forming edges between these sciences using implicational logic and generates results by performing traversals. From the given intermediate keywords obtained on the traversals, the bag can vouch for the fact that the presence of these words has a definitive purpose to it and that these sciences are related. The implications can give rise to better recommendations. The bag mimics the age-old wisdom of a shopkeeper who can talk any customer into buying anything just based on what he is buying right now.

The reason for how or why these certain sets of words appear together in the bag needs a computational explanation. The traversal results of the BoS need to be further systematically analyzed to improve inference quality. The scraping and traversal algorithms also need to be more efficient to be made feasible. The number of constraints can be increased by the keyword formation module. These recommendations based on sciences are definitely more reliable than recommendations based on data, as the data does not predict the indecisive human behavior accurately.

**Acknowledgements** We would like to thank Knit Arena Software Research and Services Private Limited, Hubballi, for the guidance and support in carrying out this work.

## References

1. P. Hegade, *See, Say, Market Recommendations*, 1st edn. (Smashwords, 2017)
2. R. Buzzell, Market functions and market evolution. *J. Mark.* (1999)
3. A. Noguev, S. Mohseni, R. Yazdanifard, B. Samadi, M. Menon, The evolution and development of e-commerce market and e-cash. (2011)
4. G.D. Pires, J. Aisbett, The relationship between technology adoption and strategy in business-to-business markets: the case of e-commerce. *Ind. Mark. Manage.* **32**(4), 291–300 (2003)
5. A. Felfernig, M. Jeran, G. Ninaus, F. Reinfrank, S. Reiterer, M. Stettinger, Basic approaches in recommendation systems. in *Recommendation Systems in Software Engineering* (Springer, Berlin, Heidelberg, 2014), pp. 15–37
6. H. Bast, B. Buchhold, E. Haussmann, Semantic search on text and knowledge bases. (2016)
7. K. Forster, I. Olbrei, Semantic heuristics and syntactic analysis. *Cognition* (1972)
8. H. Bhargava, J. Feng, A model of sponsored results in intelligent recommenders and search engines. *SSRN Electron. J.* (2008)
9. H.S. Jensen, L. Ricard, M. Vendelø, The evolution of scientific knowledge. (2003)
10. T.H. Cormen, C.E. Leiserson, R.L. Rivest, *Introduction to Algorithms*, 1st edn. (MIT Press and McGraw-Hill, 1990)
11. A.D. Vanberg, From Archie to Google: search engine providers and emergent challenges in relation to EU competition law. *Eur. J. Law Technol.* **3**(1), (2012)
12. W.B. Croft, D. Metzler, T. Strohman, *Search Engines: Information Retrieval in Practice* (2009)
13. G. Madhu, D.A. Govardhan, D.T. Rajinikanth, *Intelligent Semantic Web Search Engines: A Brief Survey* (2011)
14. B. Banavalikar, A. Bhat, A. Joshi, P. Talavar, P. Hegade, Anveshan—a model for search. *Procedia Comput. Sci.* **171**, 2362–2371 (2020)
15. D. Jannach, M. Zanker, A. Felfernig, G. Friedrich, *Recommender Systems: An Introduction* (2010)
16. F.O. Isinkaye, Y.O. Folajimi, B.A. Ojokoh, Recommendation systems: principles, methods and evaluation. *Egypt. Inf. J.* **16**(3), 261–273 (2015)
17. S.B. Aher, L.M.R.J. Lobo, Combination of machine learning algorithms for recommendation of courses in E-learning system based on historical data. *Knowl.-Based Syst.* **51**, 1–14 (2013)
18. Q. Zhao, Y. Zhang, D. Friedman, F. Tan, E-commerce recommendation with personalized promotion. in *Proceedings of the 9th ACM Conference on Recommender Systems* (2015), pp. 219–226
19. M. Dauod, S.K. Naqvi, A. Ahamed, Opinion observer: recommendation system on e-commerce website. *Int. J. Comput. Appl.* (2014)
20. R. Kosala, H. Blockeel, Web mining research: A survey. *ACM SIGKDD Explor. Newsl.* **2**(1), 1–15 (2000)
21. F. Johnson, S. Gupta, Web content mining techniques: a survey. *Int. J. Comput. Appl.* **47**, 44–50 (2012)
22. R. Mitchell, *Web Scraping with Python: Collecting More Data from the Modern Web* (O'Reilly Media, Inc. 2018)
23. L. Sheng, Z.M. Ozsoyoglu, G. Ozsoyoglu, A graph query language and its query processing. in *Proceedings 15th International Conference on Data Engineering* (IEEE, 1999), pp. 572–581
24. K. Joseph, H. Jiang, Content based news recommendation via shortest entity distance over knowledge graphs. in *Companion Proceedings of the 2019 World Wide Web Conference* (2019), pp. 690–699
25. Y. Zhang, R. Jin, Z. Zhou, Understanding bag-of-words model: a statistical framework. *Int. J. Mach. Learn. Cybern.* **1**, (2010)
26. R. Tarjan, Depth-first search and linear graph algorithms. in *IEEE Annual Symposium on Foundations of Computer Science* (1971)
27. D. Hochbaum, *IEOR* 266, Fall 2003 Floyd-Warshall algorithm for all pairs shortest paths. *Graph Algorithms Netw. Flows* **41**, (2011)

28. E. Loper, S. Bird, *NLTK: The Natural Language Toolkit* (2002)
29. P. Hegade, R. Shilpa, P. Aigal, S. Pai, P. Shejekar, Crawler by inference. in *2020 Indo-Taiwan 2nd International Conference on Computing, Analytics and Networks (Indo-Taiwan ICAN)* (IEEE, 2020), pp. 108–112

# A Novel Design Approach Exploiting Data Parallelism in Serverless Infrastructure



Urmil Bharti, Deepali Bajaj, Anita Goel, and S. C. Gupta

**Abstract** Serverless computing has emerged as a new application design and execution model. The serverless application is decomposed into granular logical functional units that run on small, low cost, and short-lived compute containers. These containers are dynamically managed by FaaS service providers. Users are charged only for the compute and storage resources needed for the execution of their piece of code. Cloud functions have restrictions on memory usage and execution time-out as imposed by their service providers. Due to this limitation, compute intensive tasks time-out before their completion and hence unable to harness the power of serverless computing. In this paper, we propose a design approach for serverless applications. It exploits data parallelism in embarrassingly parallel computations. Using our approach, compute bound tasks that are implemented in conventional design and fail in serverless environment can get executed successfully without worrying about the limitations imposed by serverless platforms. For this, several extensive experimentations using Amazon's AWS Lambda service have been performed. Further, a serverless application designed using our approach exploits the auto-scalability feature of serverless computing to achieve faster execution benefit.

**Keywords** Embarrassingly parallel computations · AWS Lambda · Scalability in serverless infrastructure · FaaS · Apache JMeter · Apache Bench

---

U. Bharti · D. Bajaj (✉)

Department of Computer Science, Shaheed Rajguru College of Applied Sciences for Women, University of Delhi, Delhi, India  
e-mail: [deepali.bajaj@rajguru.du.ac.in](mailto:deepali.bajaj@rajguru.du.ac.in)

U. Bharti

e-mail: [urmil.bharti@rajguru.du.ac.in](mailto:urmil.bharti@rajguru.du.ac.in)

A. Goel

Department of Computer Science, Dyal Singh College, University of Delhi, Delhi, India  
e-mail: [goel.anita@gmail.com](mailto:goel.anita@gmail.com)

S. C. Gupta

Department of Computer Science, Indian Institute of Technology, Delhi, India  
e-mail: [scgupta@cse.iitd.ac.in](mailto:scgupta@cse.iitd.ac.in)

## 1 Introduction

Today, serverless computing has become an evolving area in the world of cloud computing. It aims to abstract away low-level server management choices from application architect. Here, allocation of infrastructure resources is managed by the serverless provider and developers are relieved from the burden of managing or scaling servers. As an outcome, big cloud vendors are bringing their own versions of serverless computer platforms such as AWS Lambda [1], Google Cloud Functions [2], Microsoft Azure [3], and IBM Cloud Function [4] (based on Apache Openwhisk). Fewer developer logistics, dynamic auto-scaling, sub-second billing, built-in availability, and fault tolerance have enticed many developers to embrace FaaS platforms for their applications like microservices, IoT, Chatbots, Scheduled tasks, and machine learning [5].

Functions, executing on these FaaS platforms, have hard limits on memory, disk space, CPU, I/O resources, and execution time. Any serverless function consuming more resources than its specified limits is abruptly terminated by the service provider [6]. This is a major limiting ground when performing some heavy computational task. The class of applications that have generally long running processes like big data analytics, video/graphics processing, transforming bulk data, very long synchronous requests, and statistical computations is typically challenging to run on serverless environments. In essence, these hard limits imposed by providers prove to be a major stumbling block and prevents FaaS from being adopted in several use cases [7].

Many of these applications make use of embarrassingly parallel algorithms also known as naturally parallel algorithms. These are simple and efficient class of algorithms where the initial problem is split into a large number of independent sub-problems and each sub-problem is solved on different instance of cores of machines in parallel [8]. In our work, we have utilized this essential quality of splitting a large problem into sub-problems and map it to the strengths of serverless environments so as to make these environments more generally applicable and acceptable in parallel distributed computing scenarios [9].

Here, we propose an application design approach in which a large computation can be broken into small serverless functions that can be executed in parallel in serverless environment. This approach performs better than a traditional design approach where a single serverless function fails to handle a very large-scale computation. Our approach shall be a tipping point where a single machine/container is not big enough to perform a big computation and a serverless function fails for this reason. Using our novel proposed design approach, it shall be feasible to execute applications having embarrassing parallel task and large data-at-scale by leveraging the scalability benefit of FaaS.

To this end, our research presents the following main contributions:

1. Proposed a serverless application design approach that can be used in development of a large-scale computebound and memory intensive applications where data parallelism is possible.

2. Based on the proposed design approach, we developed a running prototype for distributed matrix multiplication use case. Implementation was done using Amazon's popular AWS Lambda and AWS Backend-as-a-Service (BaaS) services.
3. We also validated our results by load testing and performance comparison done using Apache JMeter and Apache Bench (ab) tools.

The remainder of this paper is structured as follows. In Sect. 2, we present related works and compared our work with the existing research literature articles. In Sect. 3, we present important concepts related to problem and solution domain to explain our methodology, while Sect. 4 provides a detailed description of the proposed design approach and its application on a case study. Section 5 discusses the experimentation and results. Final section discusses concluding remarks and limitations.

## 2 Related Work

Harnessing the power of serverless infrastructure for parallel processing applications is still in its infancy. Thus, there is a need to concretely identify issues and challenges in this direction. Few researchers have already started to explore serverless infrastructures for data intensive and long running applications. Sotlani et al. [10] proposed a distributed migration approach wherein long-duration serverless functions when reaches execution timeout limit, function is transferred to some other FaaS platform where it is further executed. So, they exploited multi-cloud paradigm features for long running applications. Developing and running an application on multi-cloud paradigm could be complex and difficult to manage. Shankarv et al. [11] designed a system called *numptywren* for linear algebraic algorithms to exploit benefits of serverless architecture. Authors also introduced *LAmbdaPACK* that is a domain-specific language, specially designed for this task. Work performed by Werner et al. [12] has used Strassen's algorithm to perform distributed matrix multiplication. Their work is close to our work, but their technique depends on AWS S3 and AWS Step function for coordination of parallel jobs. Kehrer et al. [13] designed serverless skeletons for parallel programming in cloud infrastructure. They investigated their approach on widely used farm skeletons. They presented a prototype runtime framework and proved their idea on numerical integration and hyper-parameter applications.

Few researchers have also experimented parallel computations on grid architectures. Stockinger et al. [14] proposed a system to exploit an embarrassingly parallel concepts for applied bioinformatics algorithms profile-hidden Markov model (HMM) which is a CPU-intensive task on grid infrastructure. Their approach outperformed mid-size clusters for large-scale problems despite added latencies due to grid infrastructures. Neiswanger et al. [15] implemented embarrassingly parallel algorithm on *Markov Chain Monte Carlo (MCMC)* problem. Both [14, 15] lack their implementation on serverless frameworks.

Though few approaches are present that suggest how serverless frameworks can be used for large scale applications, none of them have exploited data parallelism for running their applications on serverless infrastructures. We could not find any research article that shows the tremendous power of dynamic scaling of serverless functions for this class of applications.

With our research work, we have tried to exhibit that large-scale parallel processing applications can leverage the benefits of serverless frameworks. Experimentally, we have shown that our proposed design strategy can be used for long running parallel computations without surpassing the maximum execution limits posed by serverless frameworks.

### 3 Background

Serverless architecture is an emergent paradigm wherein users create stateless functions and deploy them to FaaS platform [16]. These platforms abstract all operational complexities and relieve its users from the burden of infrastructure configuration and management. FaaS providers facilitate dynamic auto-scaling of these stateless short-lived cloud functions; thus, making its hype truly explicable. Serverless functions get executed in response to defined triggering events. Each trigger invokes a serverless function to execute it on-demand. Concurrent triggers may invoke multiple functions in parallel. Every function invocation is atomic and idempotent in nature and does not have any relations with other invocations running concurrently. In case, if demand declines for a function, the FaaS platform dynamically scales down the function to zero and terminates all its instances to avoid extra billing. This dynamic and elastic auto-scaling characteristic is the most enticing attribute for users to embrace serverless infrastructure for a wide variety of applications instead of server centric infrastructure [17].

#### 3.1 *Embarrassingly Parallel Computations*

In this section, we have reviewed a special class of applications having embarrassingly parallel computations use cases. These computations are characterized by (i) negligible effort is required to divide a big computation into a number of small parallel tasks, (ii) no dependency among parallel tasks, and (iii) little or no communication of results among tasks. Figure 1 shows decomposition and composition of embarrassingly parallel computations [18]. Some of the use cases of embarrassingly parallel computations are matrix multiplication, Monte Carlo simulations, 3D video rendering, password cracking, and external sorting.

The inherent characteristics of these computations indicate that all decomposed sub-problems can be executed concurrently to get the final result. Consequently, these use cases can be aligned perfectly with the scalability feature of serverless



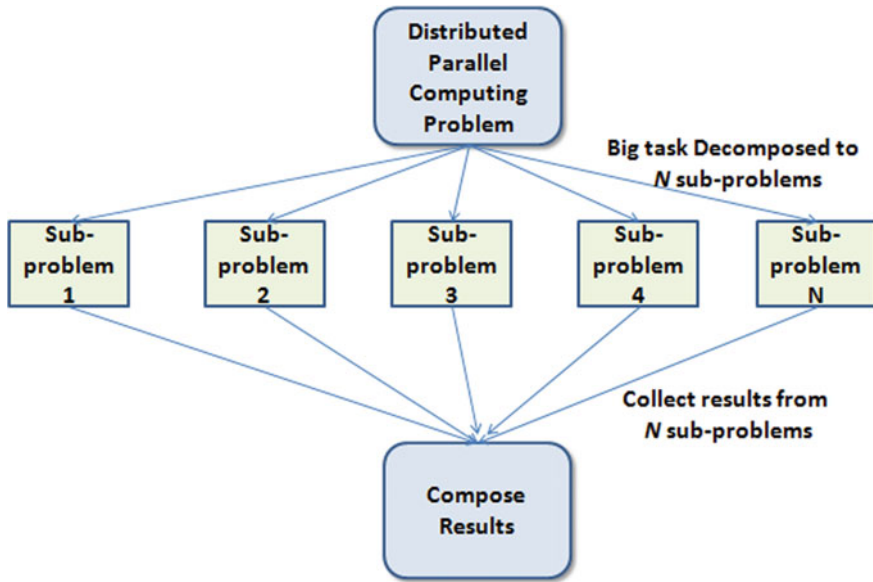


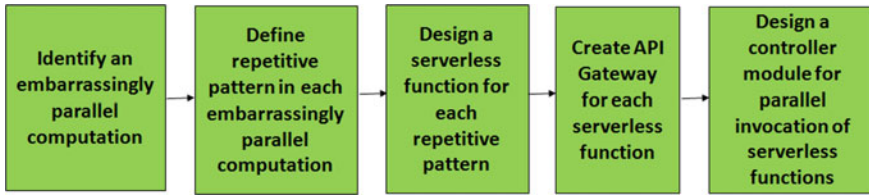
Fig. 1 Decomposition and composition of embarrassingly parallel computations

computing. In serverless cloud computing model, multiple stateless tasks can be scheduled as independent cloud functions and invoked concurrently. Thus, embarrassingly parallel computations of an application can be efficiently designed to run on serverless frameworks which would otherwise need special server farms/ fleets for their execution.

#### 4 Proposed Design Approach and Its Application on Matrix Multiplication

Here, we shall discuss our design approach and suggest how large-scale embarrassingly parallel computations can take benefits of serverless infrastructure.

Our systematic approach has following well defined steps: (i) Identification of embarrassingly parallel computation. (ii) Define repetitive pattern in each embarrassingly parallel computation. (iii) Design a serverless function for each such pattern. It should be noted that repetitive patterns using different data sets are the right candidates for serverless functions in an application design as they can be executed independently in parallel. (iv) Create API Gateway for each serverless function. (v) Design a controller module for each embarrassingly parallel computation that is responsible for parallel invocation of serverless functions with appropriate input and collate their output to achieve the final result. These steps will be applied for



**Fig. 2** Flow graph of proposed design approach

all embarrassingly parallel computations existing in an application. A flow graph describing the proposed design approach is illustrated in Fig. 2.

Auto-scalability feature of serverless framework can easily handle the execution of any number of function invocation in parallel. This design will be more efficient in terms of execution time and memory usage in comparison to conventional design approach.

To demonstrate the results of our proposed design approach, we use the matrix multiplication case study as an example and prove our results by experimental evaluation. Matrix multiplication is a classical benchmark for demonstrating the effectiveness of our new design approach. Let us assume two square matrices, Matrix  $A$  and Matrix  $B$  of dimension  $N$ . Product of these matrices  $A$  and  $B$  gives the resultant Matrix  $C$ , i.e.,  $C = A \times B$ .

Using conventional approach, computing each element  $C_{ij}$  of Matrix  $C$  requires computing the dot product of  $i$ th row vector in Matrix  $A$  is multiplied by the  $j$ th column vector in Matrix  $B$ . This dot product computation requires  $N$  multiplication operations and  $(N - 1)$  addition operations. Since there are  $N^2$  elements in Matrix  $C$ , the dot product operation must be computed  $N^2$  times. The total number of mathematical operations will be  $[N^2 * (N + (N - 1))] = (2N^3 - N^2)$ , i.e., the overall time complexity would be  $O(N^3)$ . Overall matrix multiplication computation is a memory and compute intensive problem [19].

Our serverless application design approach for matrix multiplication exploits data parallelism which involves distribution of data across multiple computing nodes on serverless frameworks such that each computation can be handled in parallel. This can be achieved by partitioning Matrices  $A$  and  $B$  into blocks of rows and columns, respectively. As a result, every element of Matrix  $C$  is a scalar product of two vectors. This repetitive pattern enables us to calculate every element of Matrix  $C$  independently by executing serverless function designed for this task in parallel. Computation done by the function involves multiplying one row of Matrix  $A$  with a column of Matrix  $B$ . As the dimension of matrices increase, single function can be executed in parallel making complexity of  $O(N)$  for any matrix dimension. Figure 3 shows how the repetitive pattern along with data parallelism can be exploited in a  $3 \times 3$  matrix multiplication by running nine independent computations as a serverless function on a serverless framework.

This design completely matches with the auto-scalability feature of serverless platforms where multiple tasks can be executed independently and asynchronously

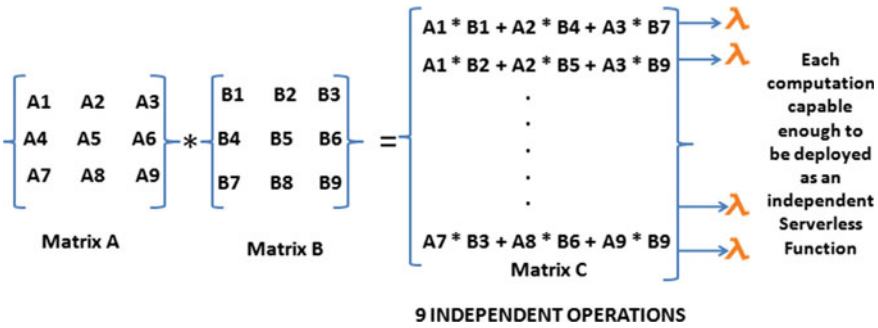


Fig. 3 Exploiting repetitive pattern with data parallelism in a 3 × 3 matrix multiplication

in parallel and have no dependency and communication amongst each other. These types of computations can take true advantage of dynamic scalability of serverless computing. Our design model is generic and may be applied to compute and memory intensive parallel algorithms that otherwise cannot get executed in a serverless environment if implemented using conventional design approach. This design approach widens the scope of applications that could be handled by serverless computing.

## 5 Experimentation and Evaluation

Amazon Web Services is the most mature and stable serverless platform. Worldwide hundreds of companies are opting AWS for their compute and other requirements as it provides a plethora of FaaS and BaaS services. So, in order to prove our results, we performed benchmarking using AWS Lambda [20] and AWS API Gateway service [21] provided by Amazon. We compared the results of the proposed design approach with that of conventional design approach on Matrix Multiplication case study [22].

### 5.1 Test Bed

To support our design model, we developed Lambda functions in Python 3.8 runtime for both the design, i.e., (i) Conventional design approach ( $\lambda C$ ) and (ii) Proposed design approach ( $\lambda P$ ). AWS Lambda functions' settings were set at default values, i.e., Timeout = 3 s and Memory = 128 MB [23]. Maximum function invocation payload (JSON input limit) that includes both request and response for AWS Lambda functions is 6 MB. It indicates that functions that process huge data-at-scale (i.e., input more than 6 MB) cannot be executed if designed according to conventional design approach. We argue that matrix multiplication problem also suffers from this limitation and we cannot execute  $\lambda C$  for progressively increasing matrix size  $N$  as  $\lambda C$

takes values for complete Matrix  $A$  and Matrix  $B$  as its input. Input to both functions ( $\lambda C$  and  $\lambda P$ ) was supplied as JSON string. A sample of input string for  $3 \times 3$  matrix multiplication has been shown in Table 1 along with algorithms for conventional ( $\lambda C$ ) and proposed ( $\lambda P$ ) design.

Here, Num key is used to indicate dimension of the matrix, i.e., Num = 3 means  $3 \times 3$  matrix multiplication and total  $3 \times 3 = 9$  elements will be there in Matrices  $A$  and  $B$  each. For  $\lambda C$ , input payload consists of all the elements of both matrices, i.e., eighteen elements. Functions designed using conventional approach will be invoked only once with complete input data. Hence, Number of Requests for  $\lambda C$  will always be one. For  $\lambda P$ , payload consists of elements of only one row and one column of Matrix  $A$  and Matrix  $B$ , respectively, i.e., six elements. Here, Number of Requests will vary with matrix dimension. For  $3 \times 3$  matrix multiplication, Number of Requests will be nine. As the dimension of matrix grows progressively, complexity of matrix multiplication computation increases. Table 2 shows Number of Requests and Number of Input Parameters for a progressively increasing matrix dimension  $N$  in  $\lambda C$  and  $\lambda P$ . In  $\lambda C$ , for matrix dimension  $90 \times 90$ , 16,200 input parameters will be sent in JSON format as a payload. In  $\lambda P$ , for same matrix dimension, only 180 input parameters will be sent in JSON format as a payload, but 8100 parallel invocations will require.

AWS Lambda stores Lambda function's code in a private Amazon Simple Storage Service (S3) bucket which is not accessible to users for security reasons. For our experiments, special policies like AWS Xray Full Access, *Cloud Watch Logs Full Access* and *Amazon API Gateway Push To Cloud Watch Logs* were attached to these Lambda functions via AWS Identity and Access Management (IAM) console so as to monitor and view execution traces using AWS Cloud Watch and AWS X-Ray services. In order to make these functions accessible from the front-end application via HTTP Post requests, we designed APIs for both Lambda functions ( $\lambda C$  and  $\lambda P$ ) using AWS API Gateway service. API Gateway enables us to create, publish, maintain, and monitor RESTful APIs that are HTTP-based and allows stateless real-time client-server communication. It basically acts as a front door for serverless applications. Users via some front-end tool invoke these Lambda functions asynchronously using APIs. Figure 4 shows comprehensive experimental bed involving AWS Lambda, AWS S3, AWS API Gateway, AWS X-Ray and AWS Cloud Watch services.

## 5.2 Experimental Evaluation

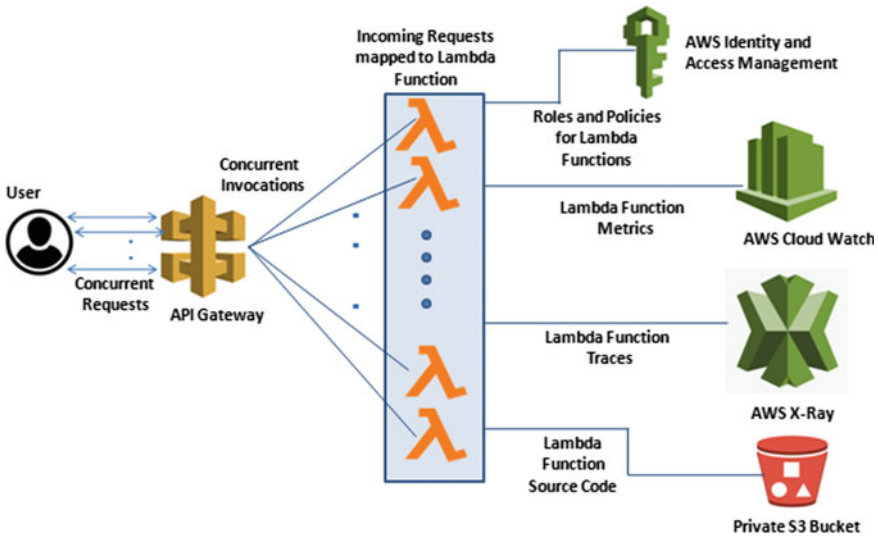
To perform our experiment, we used Apache JMeter [24] and Apache Bench [25] that are open source load testing and workload generator tools used for measuring the performance of Web applications. Requests were generated for both  $\lambda C$  and  $\lambda P$  and their concurrency limits were followed as per Table 1. For benchmarking, we sent our HTTP requests from Ubuntu 18.04 LTS machine having AMD A6-5350 M APU 64-bit processor and 8 GB RAM.

**Table 1** JSON input format and algorithm for  $\lambda C$  and  $\lambda P$  Lambda functions implemented in AWS Lambda

Conventional design ( $\lambda C$ )	Proposed design ( $\lambda P$ )
<p><i>Input Format</i></p> <pre>{   "Num": 3,   "Jstring": [{"N0":1,"N2":1,"N3":1,"N4":1,"N5":1,"N6":1,"N7":1,"N8":1,     1,"M0":1,"M1":1,"M2":1,"M3":1,"M4":1,"M5":1,"M6":1,"M7":1,"M8":1}] }</pre>	<pre>{   "Num": 3,   "Jstring": [{"N0":2,"N1":3,"N2":4,"N3":5,"N4":3,"N5":4}] }</pre>
<p><i>Algorithm</i></p> <ol style="list-style-type: none"> <li>1. Create two Num dimensional arrays <math>A</math> and <math>B</math> from <math>N_i</math> values where <math>i = 0</math> to <math>(\text{Num} * \text{Num}) - 1</math></li> <li>2. Row <math>J</math> (<math>J</math> indicates row number) of Matrix <math>A</math> will take <math>N_i</math> values according to formula: From <math>i = (\text{Num} * J) - \text{Num}</math> to <math>(\text{Num} * J) - 1</math></li> <li>3. Row <math>j</math> (<math>j</math> indicates row number) of Matrix <math>B</math> will take <math>M_i</math> values according to formula From <math>i = (\text{Num} * j) - \text{Num}</math> to <math>(\text{Num} * j) - 1</math></li> <li>4. Multiply matrix <math>A</math> and <math>B</math> in conventional way where recurrence equation is <math>C[i, j] = C[i, j] + A[i, k] \times B[k, j]</math></li> <li>5. Convert the resultant matrix in JSON format and return</li> </ol>	<ol style="list-style-type: none"> <li>1. Create two single dimensional arrays <math>A</math> and <math>B</math> from <math>N_i</math> values where <math>i = 0</math> to <math>(\text{Num} + \text{Num}) - 1</math></li> <li>2. Array <math>A</math> will take <math>N_i</math> Values according to formula: <math>i = 0</math> to <math>(\text{Num} - 1)</math></li> <li>3. Array <math>B</math> will take <math>N_i</math> Values according to formula: <math>i = \text{Num}</math> to <math>(\text{Num} + \text{Num} - 1)</math></li> <li>4. Multiply and add corresponding elements of <math>A</math> and <math>B</math> <math>\sum A_i * B_i</math> where <math>i = 0</math> to <math>(\text{Num} - 1)</math></li> <li>5. Convert the result in JSON format and return</li> </ol>

**Table 2** Number of Requests and Number of Input Parameters for progressively increasing matrix dimension  $N$

Matrix dimension	Conventional design ( $\lambda C$ )		Proposed design ( $\lambda P$ )	
	Number of requests	Number of input parameters in JSON string (Matrix $A$ + Matrix $B$ )	Number of requests	Number of input parameters as JSON string (Matrix $A$ + Matrix $B$ )
4	1	16 + 16	16	4 + 4
10	1	100 + 100	100	10 + 10
20	1	400 + 400	400	20 + 20
30	1	900 + 900	900	30 + 30
40	1	1600 + 1600	1600	40 + 40
50	1	2500 + 2500	2500	50 + 50
60	1	3600 + 3600	3600	60 + 60
70	1	4900 + 4900	4900	70 + 70
80	1	6400 + 6400	6400	80 + 80
90	1	8100 + 8100	8100	90 + 90



**Fig. 4** Test bed for running  $\lambda C$  and  $\lambda P$  in AWS serverless platform

### 5.3 Test Results

Our experiments indicate that using a conventional design approach, Lambda function ( $\lambda C$ ) fails and time-out beyond matrix dimension  $N = 80$ . It suggests that with

**Table 3** Mean and maximum running time for both  $\lambda C$  and  $\lambda P$  using Apache JMeter

Matrix dimension	Conventional design ( $\lambda C$ )	New design ( $\lambda P$ )	
	Mean running time (ms)	Mean running time (ms)	Max running time (ms)
4	303	310	352
10	329.5	324	492
20	967	330	927
30	1456.5	324	1100
40	3074.5	315	1092
50	3030	310	1175
60	4028.5	349	1284
70	6799.5	344	1392
80	7103.5	353	1524
90	Task timed out after 3 s	339	1518
100	Task timed out after 3 s	343	1385

progressively higher dimensions, the amount of computation increases which could not be finished in default execution timeout limits of an atomic Lambda function. At the same time,  $\lambda P$  (Lambda function designed on the proposed approach) executed successfully beyond  $N = 80$  and even for further higher dimensions. Additionally, in the new design approach, mean and maximum running time of each function call is significantly lower as compared to conventional design approach since a big computation is divided into multiple parallel invoked smaller invocations. Table 3 shows the time measurements collected using Apache JMeter for conventional design and proposed design approach.

To collect measurements using Apache Bench tool, we invoked API endpoints for both  $\lambda C$  and  $\lambda P$  as per Table 2. This tool exhibits two interesting parameters (i) Time taken for tests—It is the time taken when the first socket connection is created to the time when the last response is received (ii) Total transferred—It is the total number of bytes received from the server. This parameter basically indicates the number of bytes transferred over the network channel. Figure 5 shows our results on ab tool which clearly indicate that performance of  $\lambda P$  is better than  $\lambda C$ . Furthermore, matrices of higher dimensions, i.e., above  $80 \times 80$  fail in this experiment also which is consistent with our Apache JMeter results. Figure 6 shows total bytes transferred in both the design approaches and presented graph depicts that value of this parameter is evidently less in  $\lambda P$  as compared to  $\lambda C$ .

We argue that for higher dimensions in matrix multiplication, users either require access to high performance infrastructure like computational grids or clusters or may use our proposed design approach on a serverless framework. Use of clusters/grids

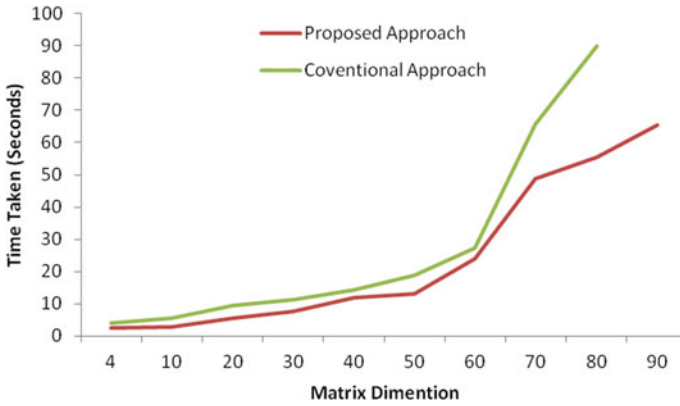


Fig. 5 Time taken for  $\lambda$ C and  $\lambda$ P

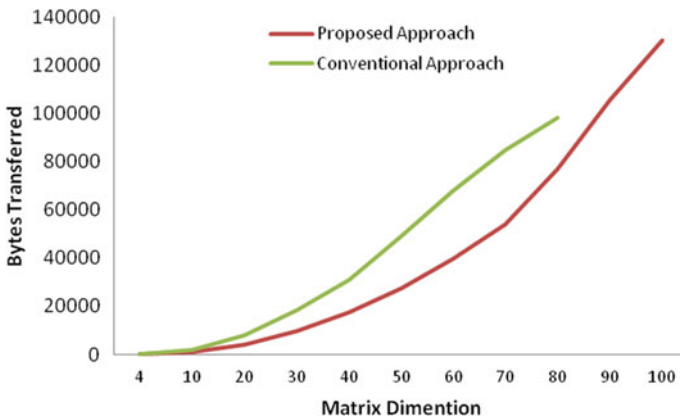


Fig. 6 Total byte transferred to server for both  $\lambda$ C and  $\lambda$ P

may impose additional configuration and management challenges. Hence, our serverless design approach renders new opportunities for serverless frameworks that can be adapted to achieve elastic scalability with a great ease of management. Our approach tackles the execution time-out limit of the serverless platforms for executing the compute and memory intensive tasks that are inherently parallel in nature. So, our results convincingly show that serverless computing can be utilized for parallel large-scale applications also without explicitly applying parallel programming coding paradigms. This enables users to run their customized code for those set of problems that are beyond the capabilities of a single serverless function. Our proposed approach hence widens the envelope of serverless technology and equipped them with novel opportunities and abilities to handle an even broader range of applications.



## 6 Conclusion

Applicability of parallel workloads to FaaS is indeed a recurrent and widely debated issue in the serverless community. For many workloads where function execution time limit is within providers limit, serverless can serve as a most relevant candidate. But, for number crunching algorithms, these timeout limits can prove to be challenging. To alleviate these issues, we have proposed a novel serverless application design approach for scenarios that are embarrassingly parallel in nature. Our idea is to divide a big computation into multiple independent tasks that can be invoked concurrently and get executed on serverless infrastructure. Our experiments have proved that serverless platforms can be used cost effectively for large-scale parallel processing applications while also preserving resilience and high performance. We were able to get the results where conventional style of Lambda function failed but at the same time Lambda function based on proposed approach performed fairly well without users worrying about parallelism and resource management issues. With this research, emerging serverless frameworks can be used for a larger class of parallel processing applications. Limitation of our approach is not applicable to those applications that cannot be subdivided among many concurrent running serverless functions. In the future, we shall conduct more statistical tests for benchmark comparison to validate or improve our design approach.

## References

1. <https://aws.amazon.com/lambda/>. Accessed on 10 June 2020
2. GCF. [Online]. Available: <https://cloud.google.com/functions>. Accessed on 10 June 2020
3. Azure. [Online]. Available: <https://azure.microsoft.com/en-in/>. Accessed on 10 June 2020
4. IBM. [Online]. Available: <https://cloud.ibm.com/functions>. Accessed on 10 June 2020
5. J. Spillner, C. Mateos, D.A. Monge, Faaster, better, cheaper: the prospect of serverless scientific computing and HPC. *Commun. Comput. Inf. Sci.* **796**, 154–168 (2018). [https://doi.org/10.1007/978-3-319-73353-1\\_11](https://doi.org/10.1007/978-3-319-73353-1_11)
6. H. Lee, K. Satyam, G. Fox, Evaluation of production Serverless computing environments. in *IEEE International Conference on Cloud Computing (CLOUD)*. (July 2018), pp. 442–450. <https://doi.org/10.1109/CLOUD.2018.00062>
7. J. Kuhlenkamp, S. Werner, M.C. Borges, D. Ernst, D. Wenzel, Benchmarking elasticity of FaaS platforms as a foundation for objective-driven design of serverless applications. in *Proceedings of the ACM Symposium on Applications Computing* (2020), pp. 1576–1585. <https://doi.org/10.1145/3341105.3373948>
8. J.-C. Régim, M. Rezgui, A. Malapert, Embarrassingly parallel search. in *International Conference on Principles and Practice of Constraint Programming* (2013), pp. 596–610
9. J.M. Hellerstein et al., Serverless computing: one step forward, two steps back. in *CIDR 2019—9th Bienn. Conference on Innovations Data System Research*, vol. 3 (2019)
10. B. Soltani, A. Ghenai, N. Zeghib, Towards distributed containerized serverless architecture in multi cloud environment. *Procedia Comput. Sci.* **134**, 121–128 (2018). <https://doi.org/10.1016/j.procs.2018.07.152>
11. V. Shankar, K. Krauth, Q. Pu, E. Jonas, S. Venkataraman, I. Stoica, ... J. Ragan-Kelley, Numpywren: Serverless linear algebra. (2018) arXiv preprint [arXiv:1810.09679](https://arxiv.org/abs/1810.09679)

12. S. Werner, J. Kuhlenkamp, M. Klems, J. Müller, S. Tai, Serverless big data processing using matrix multiplication as example. in *Proceedings of 2018 IEEE International Conference on Big Data, (Big Data 2018)* (2019), pp. 358–365. <https://doi.org/10.1109/BigData.2018.8622362>
13. S. Kehrer, J. Scheffold, W. Blochinger, Serverless skeletons for elastic parallel processing. in *2019 IEEE 5th International Conference on Big Data Intelligence Computing* (2019), pp. 185–192. <https://doi.org/10.1109/DataCom.2019.00036>
14. H. Stockinger, M. Pagni, L. Cerutti, L. Falquet, Grid approach to embarrassingly parallel CPU-intensive bioinformatics problems. in *2006 Second IEEE International Conference on e-Science and Grid Computing (e-Science'06)* (2006), p. 58
15. W. Neiswanger, C. Wang, E. Xing, Asymptotically exact, embarrassingly parallel MCMC. arXiv Prepr. arXiv1311.4780, (2013)
16. Martin Fowler Serverless. [Online]. Available: <https://martinfowler.com/articles/serverless.html>. Accessed on 28 May 2020
17. E. Van Eyk, Addressing Performance Challenges in Serverless Computing. in *Proceedings of ICT. OPEN* (2018), pp. 6–7
18. Embarrassingly Parallel Computation. [Online]. Available: <https://cs.boisestate.edu/~amit/teaching/530/handouts/ep.pdf>. Accessed on 13 June 2020
19. T. Back, V. Andrikopoulos, Using a microbenchmark to compare function as a service solutions. in *Lecture Notes in Computer Science (including Subser. Lecture Notes in Artificial Intelligence. Lecture Notes in Bioinformatics)*, vol. 11116 (LNCS, 2018), pp. 146–160. [https://doi.org/10.1007/978-3-319-99819-0\\_11](https://doi.org/10.1007/978-3-319-99819-0_11)
20. <https://docs.aws.amazon.com/lambda/latest/dg/lambda-invocation.html>. Accessed on 18 June 2020
21. welcome @ docs.aws.amazon.com. Available: <https://docs.aws.amazon.com/lambda/latest/dg/lambda-dg.pdf>. Accessed on 15 July 2020
22. I. Bermudez, S. Traverso, M. Munafò, M. Mellia, A distributed architecture for the monitoring of clouds and cdns: Applications to amazon aws. *IEEE Trans. Netw. Serv. Manag.* **11**(4), 516–529 (2014)
23. AWS Limits. [Online]. Available: <https://docs.aws.amazon.com/lambda/latest/dg/gettingstarted-limits.html>. Accessed on 11 June 2020
24. D. Nevedrov, Using JMeter to Performance Test Web Services. in *Publ. dev2dev* (2006), pp. 1–11
25. Apache Bench, ab-Apache HTTP server benchmarking tool. Available: <https://httpd.apache.org/docs/2.4/programs/ab.html>. Accessed on 18 June 2020

# A LoRa-Based Data Acquisition System for Wildfire Early Detection



Stefan Rizanov, Anna Stoynova , and Dimitar Todorov

**Abstract** A new original LoRa-based data acquisition system for wildfire detection is developed and presented. The emphasis in the paper is pointed towards hardware design concepts, physical system architecture, and implementation as well as embedded firmware structural details and algorithms. The main purpose of the proposed design is to propose techniques whose goals are to improve upon existing WSN fire hazard detection system designs by reducing the end-device power consumption. These techniques and the data analytical steps are described in detail, and evaluation on the basis of testing of their overall system performance improvement has been shown.

**Keywords** Data acquisition · Environmental monitoring · Forestry · Simultaneous localization and mapping · Wireless sensor networks

## 1 Introduction

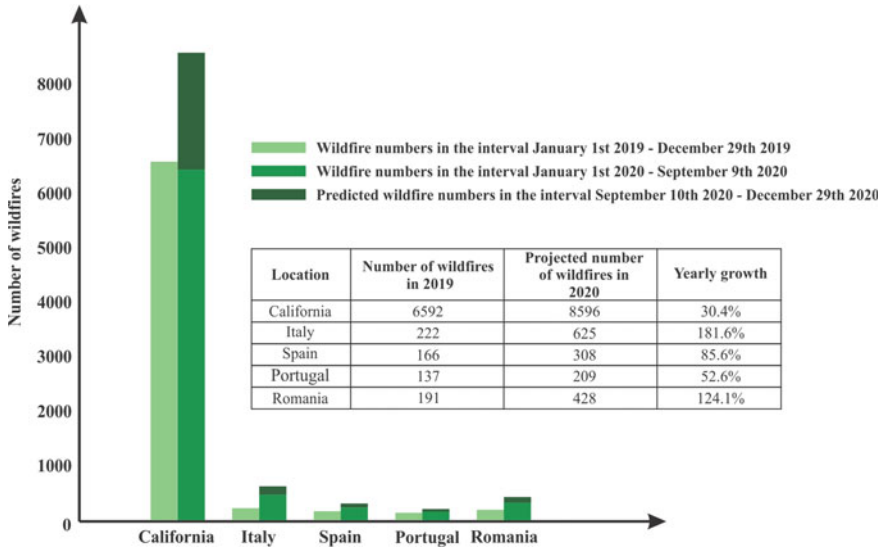
Wildfire occurrences and, related to them, inflicted damages and effects have been gradually increasing on a yearly basis. Figure 1 shows a statistical excerpt of the number of wildfire hazard events in the year 2019 and the predicted number of wildfires by the end of 2020 [1, 2]. A European Commission JRC report outlines and lists some non-material and material damages, caused by fires in 2018, for example, the loss of 100 human lives only in Greece, over 50,000 ha burned land representing 36% of the Natura 2000 network, etc. [3] These statistics show the severity of the

---

S. Rizanov · A. Stoynova (✉) · D. Todorov  
Faculty of Electronic Engineering and Technologies, Technical University of Sofia, 8, Kliment  
Ohridski Blvd, 1000 Sofia, Bulgaria  
e-mail: [ava@ecad.tu-sofia.bg](mailto:ava@ecad.tu-sofia.bg)

S. Rizanov  
e-mail: [stefan\\_tu@hitori97.com](mailto:stefan_tu@hitori97.com)

D. Todorov  
e-mail: [dgt@tu-sofia.bg](mailto:dgt@tu-sofia.bg)



**Fig. 1** Statistical excerpt of the number of wildfires in 2019 and 2020

issue and the tendencies for its annual growth in significance. Different analyses performed on sets of people, of different age groups, living in close proximity to areas with increased wildfire activity showed a correlation between these hazards and the increased risk of development of cardiovascular and respiratory health issues.

Throughout the years, multiple technologies and approaches have been proposed, developed, and implemented and aimed towards providing a solution to the problem of wildfire hazard early detection. The early detection and the possibility of monitoring prerequisites and early signs of potential wildfire activity in certain areas allow limiting drastically the damages inflicted by them.

In order for such systems to be effective in tackling such an issue, their design process has to be centered towards increasing their reliability, improving their responsiveness, optimizing their resource usage, taking advantage of modern technologies for improving their system-level applicability, and expanding their area coverage. In times when off the shelf hardware and software modules are readily available, constructing a simple system is easy and cheap, but in order to build a fully comprehensive and optimized design, taking into account all of the previously mentioned objectives, this can only be performed by creating a complete system design from scratch, utilizing the latest technological means and methods. With currently available battery technologies, there are hard limitations as to what energy density and capacity are physically achievable in a single cell structure—this is why implementing techniques for the reduction of energy consumption of each battery-powered device is crucial in today’s technology-dominated world.

Within this work, we present a comprehensive data acquisition system design, aimed towards the early detection of wildfires. Hardware and software details are

discussed. Novel techniques, such as the adaptive sample rate algorithm, for the reduction of power consumption of each end device, are presented, and their effectiveness is evaluated and shown via excerpts from performed tests. A detailed analysis of existing technologies and related works is performed in the next section.

## 2 Related Works

We will now present some of the different system designs and try to highlight some of their advantages and deficiencies. Broadly, the types of systems for wildfire detection can be split into two distinct classes: camera-based systems and wireless sensor nodes (WSNs).

Camera-based wildfire detection systems are constructed around the utilization of infrared (IR) thermal imaging sensors and implementing digital image processing and spectral analysis methods for the detection of fire hazards. These types of systems typically perform energy measurements in the 1–15  $\mu\text{m}$  spectral band, where radiance peaking can be observed due to the wildfire burning temperatures of 120–400  $^{\circ}\text{C}$  [4, 5]. Such system designs can be observed in the works of Alkhatib [6], Jones et al. [7], and also the FireHawk, ForestWatch, EYEfi platforms [8]. Some of the advantages of these types of platforms are as follows: Smoke emissions are more transparent and less dense when performing energy measurements within this IR spectral band; nighttime detection is possible [9]; large open areas can be analyzed using this methodology (typically around 10–40 km in radius). But camera-based hazard detection systems possess several deficiencies such as dynamic range issues; due to atmospheric absorption and reflected sunlight effects, the 3–5  $\mu\text{m}$  and 5–8  $\mu\text{m}$  spectral bands are difficult to use and performing analysis within them may cause false-positive detections [10]; their utilization of complex imaging sensors and powerful image processing modules increases their design complexity, cost of production and implementation, support costs, power consumption and decreases their reliability, possibility for mass quantity allocation, and necessitates the need for their human operator-based operational observance. Large-power consumption is one of the biggest problems looming over these types of systems—as a result, each imaging station has to be fitted typically with a battery pack with a large capacity and a powerful energy harvesting module, which in turn presents a regular replacement expenditure for their operational lifetime support.

The other massively adopted technological methodology is to construct wireless sensor-based network systems (WSNs). The structure behind the WSN technology is based on the creation of a system containing multiple end-devices organized in either a mesh [11] or clustered topology. Whichever topological scheme is used depends mainly on the used wireless communication technology. Typically, a graphical representation of such types of system organizations is done through the usage of signal graphs, with each end-device being denoted as a graph node. Each end device is fitted with an embedded sensory block, enabling it to measure environmental parameters such as temperature, humidity, carbon monoxide, and carbon dioxide concentrations.

After gathering the sensed parameter information, the end device in turn more often calculates deterministic fire hazard event probability result based on the measurement data which then transmits wirelessly upstream via the established communication channel link. Over the past several years, these types of the system gained significant ground in being more massively adopted and being of interest to both types of research and private companies—mainly because of their relative hardware design simplicity, improvements made in the sphere of wireless communications, and the accessibility to more precise and low-power sensory elements. The dominant wireless technologies, used for these types of systems, were for a long time GSM/GPRS and ZigBee, with examples of such systems being the works of Önal et al. [12] and Varunkumar et al. [13], who proposed constructing a ZigBee + Raspberry Pi mesh of Internet-connected sensors and the work of Toledo-Castro et al. [14], who proposed a fuzzy logic and GSM-based WSN system. These two technologies possess intrinsic limitations, mainly the ZigBee technology imposes a maximum connectivity distance between two end devices of 100–200 m, essentially necessitating that large areas of land can be analyzed only by systems comprised of numerous nodes; utilizing a GSM/GPRS wireless connectivity, though not possessing the upwards-mentioned distance limitation, increases the overall device power consumption, increases packet transmission times due to the latency when using third-party cellular networks, expands upon the production and support costs, and limits the applicability of such systems to only areas having cellular coverage. A large disadvantage in today's more massively proposed utilization of 3G and 4G GSM transceiver modules in WSN systems [15] is that they provide an unnecessarily large communication channel bandwidth, with sensory devices typically broadcasting data packets of size often no more than 20–30 bytes. One of the bigger technological leaps forward within these types of system designs came with the proposition of utilizing the LoRa technology for wireless communication. LoRa gradually became the preferred connectivity technology for WSN fire-hazard detection platforms, with it being a central point in many recent research works such as that of Rizanov et al. [16], Sendra et al. [17], Anitha et al. [18], Gaitan et al. [19], and others, with emphasis within them being pointed more strongly towards the broader system architecture, the usage of off the shelf modules for constructing the hardware platform and analyzing and evaluating the RF capabilities and performance of the LoRa communication link and data exchange over it. Brito et al. [20] describe through their experimental results, and the correlation between environmental parameter changes when a wildfire event occurs in close proximity to the WSN end device. The usage of off the shelf modules limits the possibilities for more complex hardware techniques to be implemented aimed towards the reduction of the overall power consumption of the sensory device. A topic overlooked in most WSN wildfire systems related research is power consumption optimization. Within this work, we will try and expand on this very import subject by presenting a developed LoRa WSN system, utilizing several novel methods for energy consumption optimization, and simple data analysis algorithms which eliminate the necessity for each end-device to evaluate and provide a deterministic result as to whether a fire hazard event has occurred. This way each node device has the

sole function of just collecting and cleaning up the sensory measurement data in an energy-efficient manner.

### 3 Sensor Node Hardware Structure

The constructed system consists of LoRa sensory nodes, localized LoRa gateway, a central server, and a PC for user-based monitoring. The hardware structure of the prototyped LoRa sensor node is shown in Fig. 2.

Figure 3 shows photographs of the developed WSN system, consisting of a battery-powered sensory node device, a LoRa gateway utilizing both a LoRa and GSM transceiver modules, and a system administrator PC, connected via USB to the gateway for system testing purposes and bypassing the connection to a central

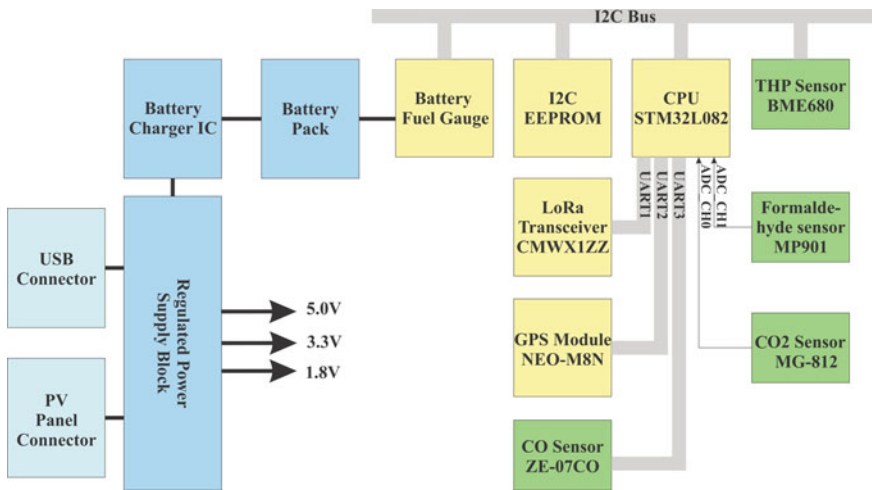


Fig. 2 Sensor node hardware structure diagram

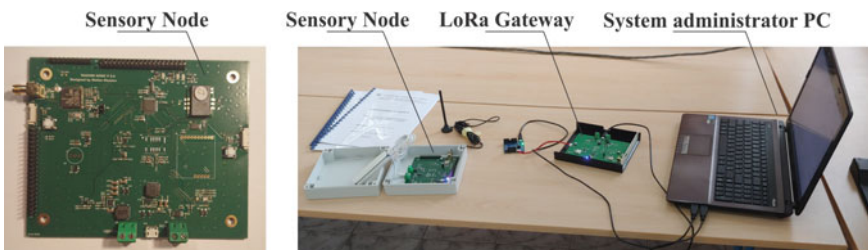


Fig. 3 Developed WSN system

server. The node design consists of a CPU—STM32L082xx; LoRa transceiver—CMWX1ZZABZ; GPS module—NEO-M8N; THP sensor—BME680; CO sensor—ZE-07CO; CO<sub>2</sub> sensor—MG-812; formaldehyde sensor—MP901; I2C EEPROM; battery fuel gauge; battery pack; battery charger IC; regulated power supply block; USB connector; photovoltaic panel connector. The default configuration of the CMWX1ZZABZ is 3 dBm output power, SF7. Higher transmission ranges can be achieved by increasing the output power initially, before raising the spreading factor [21]. Based on the work of Yokelson et al. [22], capturing quantitatively the gas emissions during a forest fire, the designed LoRa based system implements sensors for measuring the gasses with the highest concentrations: CO<sub>2</sub> concentration of 1655 g/kg; CO concentration of 83 g/kg; HCHO concentration of 2.37 g/kg. Forest fires emit a number of other gasses such as HCN, propane, HCOOH, etc., but comparatively, their concentrations are much lower, thus complicating their remote measurement and increasing the complexity and overall cost of the system. The system is battery-powered, with battery charging done via USB or the connected PV panel. Utilizing a low-power CPU, LoRa transceiver, and GPS modules, constructing the embedded firmware, and utilizing the methods discussed later allow the sensory node to have a large operational lifetime. The CPU connects to the I2C bus as the master device, and the I2C communication is configured in standard mode, 7-bit device addressing. Via the bus, it communicates with the connected slave devices—THP sensor, I2C EEPROM, and battery fuel gauge. The CPU interfaces with the LoRa and GPS module via a UART link. The MCU connects to the CO sensor via UART and also reads the analog output signals from the formaldehyde and CO<sub>2</sub> sensor via two ADC channels. The developed LoRa gateway device consists of a CPU—STM32L082xx, LoRa transceiver—CMWX1ZZABZ, and a GSM transceiver—SARA R510S.

For the application of such a system construct, multiple LoRa sensory nodes are placed throughout an area of interest in predefined locations and clustered. They perform regular sensory measurements on a timely basis of a group of parameters. Upon collecting the raw sensory data, each LoRa node performs basic data filtering, analysis, averaging, and transmits the compacted data upstream via the LoRaWAN communication channel to the nearest local LoRa gateway. The LoRa gateway is able to acquire data from multiple sensory nodes in proximity to its part of the localized cluster via the 4G communication channel link connects to the central system server and as the cluster's head device transmits a large amount of collected sensory data. The system administrator can then analyze the measurement data, stored locally on the server, and monitor in real-time signs of potential wildfire occurrence in a certain area and/or early detect and localize the formation of a forest fire. Due to the system structure and the implementation of low-power techniques, it is possible to construct a complex multi-threaded sensory system, monitoring large areas of land with the limited necessity of human operator local terrestrial oversight. The reduction of operator oversight drastically reduces the implementation and support costs needed for the system to be used effectively. For testing purposes, as shown in Fig. 3 the gateway device is connected via a USB interface to the PC. In order to verify if the system functions properly and evaluating its capabilities for detecting



the presence of a nearby fire hazard, open fire tests in a controlled environment were performed. These controlled tests were aimed also towards understanding whether the developed adaptive sample rate algorithm is effectively reducing the average power consumption of the sensory node device. The results from these tests will be described in further detail within Sect. 5.

## 4 Software Structure

The sensory node will collect data for a group of parameters—temperature, humidity, air pressure, CO, CO<sub>2</sub>, and formaldehyde concentrations. Based on the chosen sensors, measurements with a limited accuracy could be performed within the defined operational conditions: 0–65 °C, RH0–100%; AP 300–1100 hPa; CO 0–500 ppm; CO<sub>2</sub> 350–10000 ppm; Formaldehyde 1–50 ppm. The measurement accuracy is as following—Temperature: ±1%; Humidity: ±3%; Air pressure: ±0.6 hPa; CO: 0.1 ppm; CO<sub>2</sub>: 0.5 ppm; Formaldehyde: 1 ppm.

The CPU performs regularly timed measurement samples of all parameters. Several considerations have to be made when choosing the optimal and appropriate sample rate:

- (1) The sample rate must be chosen such that the average device power consumption must not exceed a predefined value, due to the requirement for battery-powered operation and long exploitation lifetime;
- (2) The sample rate must be chosen such that enough data is collected in order to perform precise analysis onto it;
- (3) In consideration must be taken the fact that temperature and humidity changes are typically slow processes throughout the day—averaging at speeds of 2.275 °C/h and 5.042% Rh/h in urban environments [23].

Taking into account, this a reasonable initial time delay between samples of 20 min would be appropriate. During its operation, the device enters three different states—SLEEP, SENS, and TRANSMIT. In order to conserve energy when the device is in SLEEP mode, the power supply lines of the LoRa transceiver, GPS transceiver, and the environmental sensors are cut. In SENS mode, the power supply lines of the LoRa and GPS modules are cut. In TRANSMIT mode, the power supply lines of the sensors are cut. After each performed measurement sample, the device will go in SLEEP mode, and the configured internal timer module generates an interrupt event at each new time step; when this event occurs, the device goes into SENS mode, and within the interrupt handler routine, the data collection subroutine is performed, after it has been completed the device goes through RTI and back to SLEEP mode. Upon collecting a number of data points, the device performs basic exploratory data analysis (EDA) over the gathered information. EDA proposes that data should be first explored without assumptions about probabilistic models, error distributions, etc. As a result, EDA allows exploration of the gathered information to reveal patterns and features that help in the understanding, analysis, and modeling of data [24]. Implementing

a Tukey method [25, 26] allows identifying data outliers by finding and comparing all possible pairs of means and finding those that are drastically different from each other. After outliers have been identified, they are marked as invalid readings and removed from the dataset. This simplified methodology of each device being solely a raw information entry point, part of a larger data mining framework consisting of all networked devices, and for it performing only data filtration and outlier removal and not performing embedded level deterministic hazard classifications has been shown to reduce the power consumption of the sensory device. The resulting data after the filtering and outlier removing steps have been complete is passed through a simple analytical subroutine. Figure 4 shows graphically the different steps of the performed data analysis method onto the sensory measurement readings. Each stage of it will be described in further detail.

Upon a timed wake-up event, the sensory node device performs measurements of each of the parameters. The first graph in Fig. 4 shows an example of collected valid raw measurement data of ten temperature samples.

After collecting  $N$  samples (in this case 10), the device then normalizes the read values with respect to Sample 1. The second graph of Fig. 4 shows the normalized raw measurement data. The node device then calculates in which sample has the largest absolute value within the remaining  $(N-2)$  samples, excluding the last one. The CPU then packs Sample 1, the maximum valued sample, and Sample  $N$ . The third graph of Fig. 4 shows the packed and transmitted sample values from the dataset.

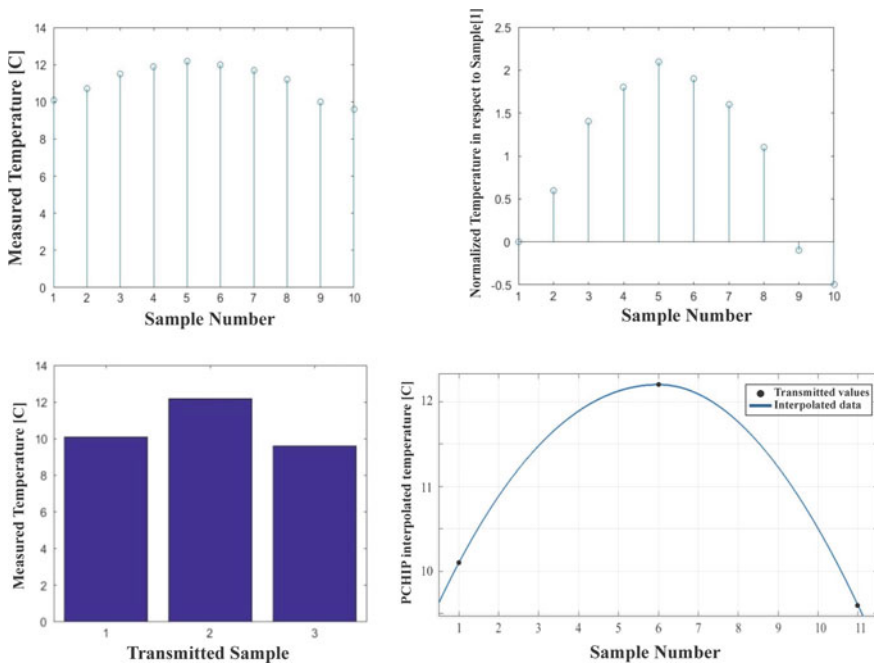


Fig. 4 Data analysis steps performed on the sensory measurement data

This procedure is performed for each measurement parameter. Upon compacting all data measurement packs, the MCU forwards them to the LoRa transceiver. On the receiving end, the central server collects this sensory data and by means of PCHIP or some other curve fitting algorithm interpolates the data for the missing sample values. The fourth graph of Fig. 4 shows a PCHIP interpolation being performed onto the received data in order to superimpose values to the missing data points. This approach drastically decreases the necessary bandwidth for transferring all the sensory data via the communication channel. This effective reduction in transferred data quantity reduces additionally the average power consumption of the device. An additional novel software method that was developed and implemented is the adaptive sample rate algorithm which will now be discussed. The purpose of this algorithm is to eliminate the fixed value of the time step, at which timer interrupts are generated, and to generate the reconfigure and adjust it dynamically based on the values of previously measured environmental parameter values. After performing  $N$  samples of each one of the parameters, the device calculates the average value of all the readings. The device stores the average values of each  $N$  samples within its internal memory. It then calculates the derivative of this mean value and uses it to evaluate what the time step should be. Monitoring the rate of change and sign allows for the detection of sudden changes in the typically slowly changing environmental parameters, thus indicating that there may be a fire hazard prerequisite. This algorithm is centered on a fixed derivative threshold value comparison. In order to verify that such a prerequisite exists and it is not just a random fluctuation, the algorithm imposes a change in the time step only if three consecutive derivatives are comparatively larger than the fixed threshold values. Figure 5 shows a pseudo code representation of the implemented adaptive sample rate algorithm, and Fig. 6 shows a block diagram of the adaptive sample rate algorithm.

Using the adaptive sample rate algorithm allows nodes with comparatively small computational resources to evaluate the sensing data without performing complex fire weather index calculations or as some researchers have proposed—usage of ANNs [27] for producing a fire hazard event probabilistic determination. On the receiving end, the server can detect if one or more end devices are in close proximity to a wildfire by monitoring their broadcasting rates and detecting their increases.

The CPU of the LoRa sensor node communicates with the embedded GPS transceiver module via a NMEA 0183 V4.0 communication protocol. Once a day, the MCU powers on the GPS module, waits a certain time interval, due to the cold start of the module and the required data acquisition time to fill in the almanac, and receives from it information regarding its location—longitude, latitude, and altitude. The CPU stores this information within the I2C EEPROM IC. The CPU also receives accurate GPS GMT time, which is used to synchronize its internal RTC.

The CPU prepares two types of data packets, which are required to be transmitted over the air through the LoRaWAN channel—an identification data packet and a measurement data packet. The identification data packet is sent first, and it is sent only once a day from each LoRa sensory node to the nearest LoRa gateway. Its length is 14 bytes and consists of 2 bytes for LoRa node Device ID number, 6 bytes of the device's GPS coordinates, 2 bytes of data regarding the internal battery status, 2 bytes

```

# create blank array of type env_s
env_s_raw_data_array[] = {}

# TIMER_STEP_DEFAULT_VALUE defined in header file
int timer_step = TIMER_STEP_DEFAULT_VALUE
int timer_step_change = 0
bool meas_status_flag = False
int sample_cnt = 0 #sample count
# raw data derivative over sliding window of size RAW_SAMPLE_ARRAY_MAX_NUM
int raw_data_der = 0
int der_cnt = 0 # derivative number count
int der_array [3] = {0,0,0}
bool der_flag = False
int avg_value = 0
int i = 0
int avg_array[] = {}
if (Device_Wakeup()): # Device has entered SENS state
    # new measurement is performed and a new array
    entry is included
    # return true if complete successfully
    meas_status_flag = Perform_Envi_Par_Sens(raw_data_array)
    if (meas_status_flag):
        sample_cnt ++ #increment ready sample number

    # if max value is reached
    if (sample_cnt==RAW_SAMPLE_ARRAY_MAX_NUM):
        avg_value = Raw_Data_Avg(raw_data_array)
        avg_array[i] = avg_value
        i++
        if (i == AVG_MAX_VAL):
            i = 0
    # get the derivative value over a sliding window
    raw_data_der = Get_Data_Derrivative(avg_array)
    # add new array entry
    der_array[der_cnt] = raw_data_der
    #increment derrivative counter
    der_cnt ++
    if (der_cnt==2):
        for i in range (0,2):
            if (der_array[i] >= DERIVATIVE_THRESHOLD):
                #if 3 consecutive derivatives are above threshold
                der_flag = True
            if (der_array[i] < DERIVATIVE_THRESHOLD):
                der_flag = False
        der_cnt = 0
    if (der_flag):
        #find the maximum value of the der_array
        # return the ratio Max_Value/Threshold_Value

        timer_step_change = Max_Der_Val(der_array)
        timer_step = timer_step -
(timer_step_change * (DERIVATIVE_THRESHOLD / 5))
        # regonfigure the timer module
        Adjust_Timer_Step(timer_step)

    sample_cnt=0 # null sample counter
    Clear_Raw_Data_Array(raw_data_array)
    Device_Sleep() #enter device back in sleep mode

```

**Fig. 5** Adaptive sample rate algorithm pseudo code

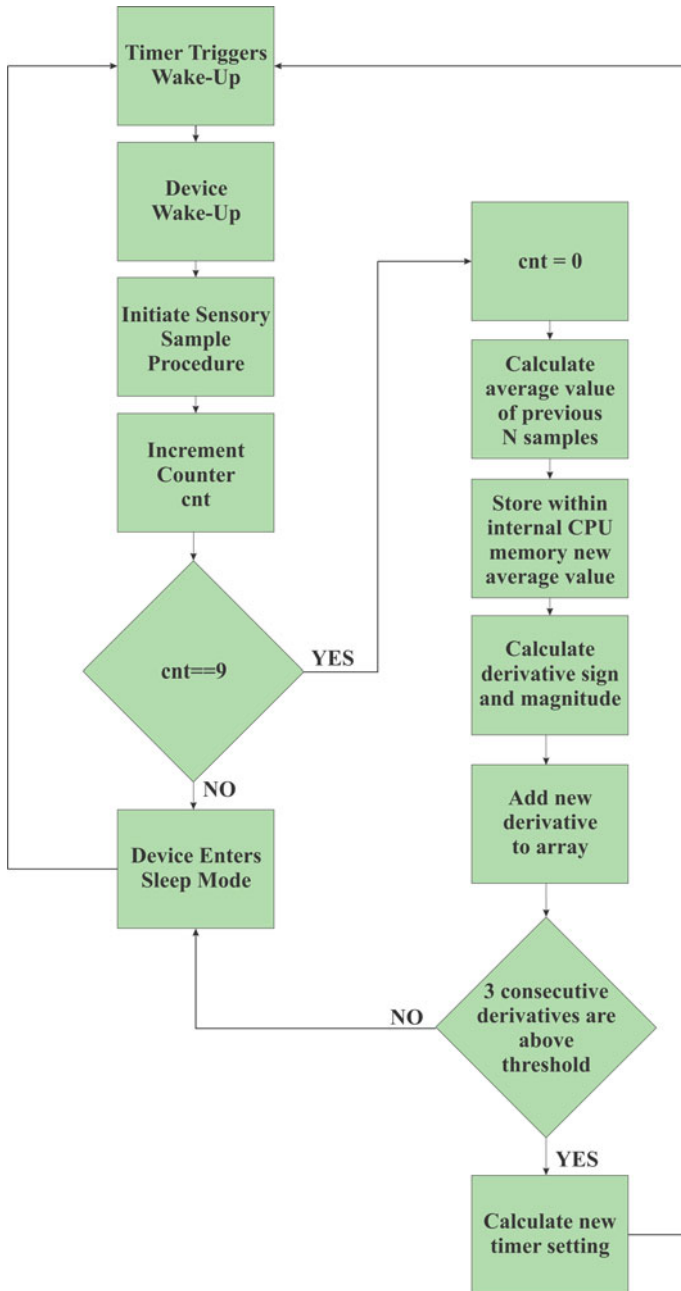


Fig. 6 Adaptive sample rate algorithm workflow diagram

Device ID		Device GPS Coordinates						Battery Status		Synch Time		CRC16	
Byte 13	Byte 12	Byte 11	Byte 10	Byte 9	Byte 8	Byte 7	Byte 6	Byte 5	Byte 4	Byte 3	Byte 2	Byte 1	Byte 0

Identification Data Packet

Fig. 7 Identification data packet format

Device ID		Packet Number		Tx Time		Temperature Data			
Byte 31	Byte 30	Byte 29	Byte 28	Byte 27	Byte 26	Byte 25	Byte 24	Byte 23	Byte 22

Measurement Data Packet

Humidity Data				Air Pressure Data				CO Data			
Byte 21	Byte 20	Byte 19	Byte 18	Byte 17	Byte 16	Byte 15	Byte 14	Byte 13	Byte 12	Byte 11	Byte 10

Measurement Data Packet

Co2 Data				Formaldehyde Data				CRC16	
Byte 9	Byte 8	Byte 7	Byte 6	Byte 5	Byte 4	Byte 3	Byte 2	Byte 1	Byte 0

Measurement Data Packet

Fig. 8 Measurement data packet format

of current GMT synchronized with the GPS received time, and a CRC16 performed over the whole data packet. Figure 7 shows the identification data packet format.

The measurement data packet is sent each time a successful sensory information acquisition step is performed. The data packet has a length of 32 bytes, and it consists of 2 bytes for LoRa node Device ID number; 2 bytes for packet number in respect to the initial transmission of the identification data packet; 2 bytes of TX time containing information about the time delay between the first transmitted packet and the current one;  $6 \times 4$  bytes of sensory data for all listed measured parameters, and a CRC16 performed over the whole data packet. Figure 8 shows the measurement data packet format.

## 5 System Testing

In order to evaluate the efficiency of the implemented power conservation methods for the developed WSN wildfire detection system, tests in a controlled environment were performed. These tests were aimed towards monitoring the power consumption

of the system comparatively when it implements and when it does not implement the proposed techniques. The testing environment is presented in Fig. 9. The testing setup consists of simulating a wildfire and placing the sensing node device a different distance from it—four different zones were used at distances of 5, 10, 15, and 20 m. The power consumption of the device was monitored using an R&S HM01202 scope.

The results from the experimental measurements are shown in Fig. 10. In the shown test results, the sensory device is placed in zone C. In the upper graph, the device utilizes the proposed adaptive sample rate algorithm, and in the bellow graph, the device operates with a fixed time step. The battery power consumption of the device in SLEEP mode is measured to be around  $0.33 \mu\text{A}$ ; when the device enters SENS mode, the consumption jumps initially to around  $713 \mu\text{A}$ ; then, during the measurements when all sensors are enabled, the power consumption peaks at around

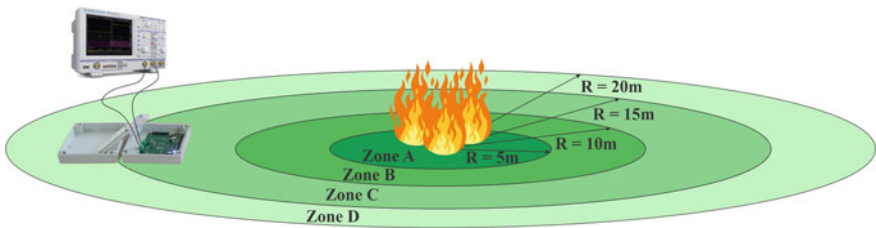


Fig. 9 Testing setup

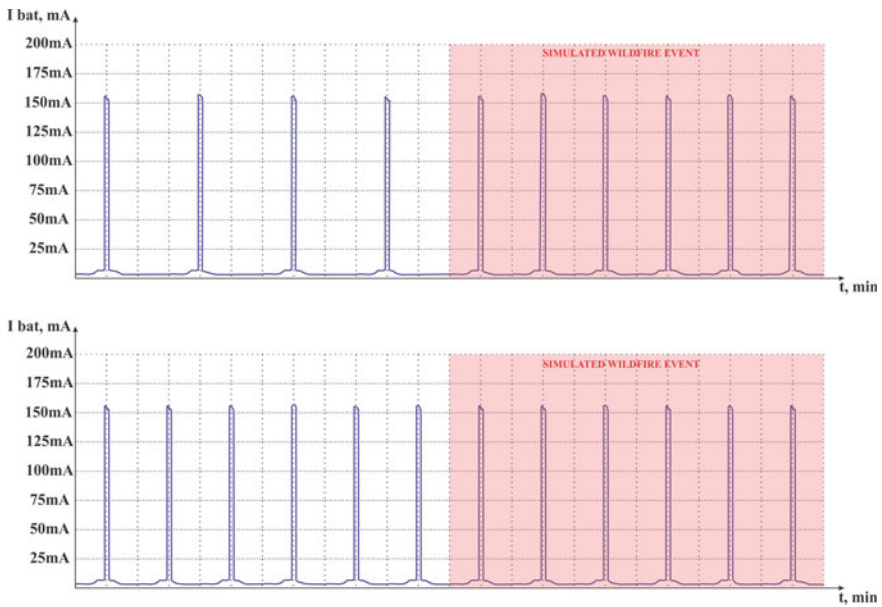


Fig. 10 Experimental results

154.9 mA. For the tests, the fixed step was fixed at 5 min, while the default time step value using the adaptive sample rate algorithm was fixed to a value of 7.5 min. The number of samples over which analysis is performed was fixed to  $N = 5$  for both cases. These experimental results show that implementation of the adaptive sample rate algorithm reduces the average consumption of the system, depending on what parameter initial values are chosen for the default time step, the time step change, and the size of the sliding window of  $N$  number of samples.

## 6 Conclusion

An original data acquisition system, utilizing the LoRa technology, for early detection of wildfires, has been developed and presented. The proposed method has the advantages of being constructed as a stand-alone device from scratch and implementing: a LoRa long-range low-power wireless communication; multiple different types of on-board sensors, allowing for a more complex model to be constructed, based on the measurement data; a novel adaptive sample rate algorithm aimed toward the decreasing of power consumption of the device; EDA and Tukey methods for filtering the raw data and elimination of data outliers, hence reducing the necessary transmission bandwidth and increasing the energy efficiency of the system.

**Acknowledgements** This research was funded by a grant (No. DN-17/16) from the National Science Fund of Bulgaria.

## References

1. EFFIS STATISTICS. <https://effis.jrc.ec.europa.eu/static/effis.statistics.portal/effis-estimates/EU>
2. CALL FIRE. <https://www.fire.ca.gov/stats-events/>
3. JRC Technical Reports, Forest Fires in Europe, Middle East and North Africa 2018, European Union (2019). <https://doi.org/10.2760/1128>
4. U.S. Forest Service. [www.firelab.org/resource/thermal-imaging](http://www.firelab.org/resource/thermal-imaging)
5. M. Modest, *Radiative Heat Transfer* (Academic Press, Elsevier Science, 2013) ISBN: 9780123869449
6. A. Alkhatib, A review of forest fire detection techniques. *Int. J. Distrib. Sens. Networks*, **2014** (2014). <https://doi.org/10.1155/2014/597368>
7. S. Jones, K. Reinke, S. Mitchell, F. McConachie, C. Holland, *Advances in the Remote Sensing of Active Fires*. Report no. 336.2017, Melbourne: Bushfire and Natural Hazards CRC, (2017)
8. D.K. De et al., Twenty-first century technology of combating wildfire, in *IOP Conf. Series: Earth and Environmental Science*, vol. 331 (2019). <https://doi.org/10.1088/1755-1315/331/1/012015>
9. B. Mohanta, R.K. Mohanta, B. Sethi, Application of wireless sensor network to monitor forest fire. *IJATER* **1**, 53–57 (2018) ISSN: 2250-3536
10. R. Allison, J. Johnston, G. Craig, S. Jennings, Airborne optical and thermal remote sensing for wildfire detection and monitoring. *Sensors* **16**(8), 1310 (2016) ISSN 1424-8220



11. A.E.U. Adnan, Salam, A. Arifin, M. Rizal, Forest fire detection using LoRa wireless mesh topology, in *2nd East Indonesia Conference on Computer and Information Technology (EIConCIT)*, Indonesia, pp. 184–187 (2018). <https://doi.org/10.1109/EIConCIT.2018.8878488>
12. A.F. Önal, B. Ülver, A. Durusoy, B. Erkmen, Intelligent wireless sensor networks for early fire warning system. *Electrica* (2019). <https://doi.org/10.26650/electrica.2019.19019>
13. S. Varunkumar, P.V. Yokeshraj, V. Vignesh, S. Tamilselvan, Implementation of wireless sensor network and IoT for real time forest fire warning system. *Int. J. Eng. Tech.* **4**(1), 150–153 (2018). ISSN: 2395-1303
14. J. Toledo-Castro, I. Santos-González, et al., Forest fire prevention, detection, and fighting based on fuzzy logic and wireless sensor networks. *Complexity* **2018**, (2018). <https://doi.org/10.1155/2018/1639715>
15. E.A. Kadir, S.K.A. Rahim, S.L. Rosa, Multi-sensor system for land and forest fire detection application in Peatland area. *IJEEI* **7**(4) (2019). <https://doi.org/10.11591/ijeei.v7i4.1604>
16. S. Rizanov, A. Stoyanova, D. Todorov, System for early warning and monitoring of wildfires, in *Presented at the XXVIII International Scientific Conference Electronics*, Sozopol, Bulgaria, 12–14 Sept., (2019). <https://doi.org/10.1109/ET.2019.8878653>
17. S. Sendra, L. Gracia, J. Lloret, I. Bosch, R. Vega-Rodriguez, LoRaWAN network for fire monitoring in rural environments. *Electronics* **9**(3) (2020). <https://doi.org/10.3390/electronics/9030531>
18. P. Anitha, V.S. Deepa, R. Divya, K. Sobana, An automatic forest fire detection using LoRa wireless mesh topology. *IRJET* **7**(2) (2020). ISSN: 2395-0072
19. N. Gaitan, P. Hojbot, Forest fire detection system using LoRa technology. *IJACSA* **11**(5) (2020). <https://doi.org/10.14569/IJACSA.2020.0110503>
20. T. Brito, A.I. Pereira, J. Lima, A. Valente, Wireless sensor network for ignitions detection: an IoT approach. *Electronics* **9**(893) (2020). <https://doi.org/10.3390/electronics9060893>
21. E. Bäumker, A. Miguel Garcia, P. Woias, Minimizing power consumption of LoRa and LoRaWAN for low-power wireless sensor nodes. *PowerMEMS* (2018). <https://doi.org/10.1088/1742-6596/1407/1/012092>
22. R. Yokelson, S. Urbanski, E. Atlas, et al., Emissions from forest fires near Mexico City. *Atmosph. Chem. Phys.* **7** (2007). ISSN: 5569-5584
23. H. Yang, Y. Jiang, J. Liu, Z. Gong, The field investigation on thermal comfort of tent in early autumn in Tianjin, in *MATEC Web of Conferences*, vol. 61, (2016). <https://doi.org/10.1051/mateconf/20166101001>
24. W. Martinez, A. Martinez, *Computational Statistics Handbook with MATLAB* (Chapman & Hall/CRC, New York, 2015). ISBN 1-58488-229-8
25. D. Hoaglin, F. Mosteller, J. Tukey (eds.), *Understanding Robust and Exploratory Data Analysis* (Wiley, New York, 2000). ISBN: 978-0-471-38491-5
26. J. Chambers, W. Cleveland, B. Kleiner, P. Tukey, *Graphical Methods for Data Analysis* (CRC Press, 2018). ISBN-10: 0412052717
27. L. Yu, N. Wang, X. Meng, Real-time forest fire detection with wireless sensor networks, in *Proceedings of the International Conference on Wireless Communications, Networking and Mobile Computing (WCNM'05)*, pp. 1214–1217 (2005). <https://doi.org/10.1109/WCNM.2005.1544272>

# Announcer Model for Inter-Organizational Systems



Prakash Hegade, Nikhil Lingadhal, Usman Khan, Tejaswini Kale,  
and Srushti Basavaraddi

**Abstract** From barter systems to shopping online, markets have evolved with institutional design characteristics with the objective of providing a platform for buying and selling. The technological investment portfolio has brought in significant changes to market dynamics. Though there are apprehensions to migrate the offline features online, along with online benefits, there are also inherent challenges to be managed. In supply chain management, which manages from raw materials to customers, inventory management has a substantial role and acts as a key player affecting the entire chain directly or indirectly. Nevertheless of automation, inventory management is a tedious task and could use a computational helping hand. The announcer model proposes an alternative to computationally solve resource management between the business cycle's various stages through this paper. It attempts to establish a strong relationship between the intermittent by announcing the iterative status flags via tags and further utilizing it to improve work efficiency. The model eases the interaction and provides an automated channel for communication. This paper proposes the model and discusses its architecture and a sample workflow from a simulated industry transaction. The announcer space can also be integrated to live web data, making the system dynamic and self-learning to current market needs. The learning capability of the announcer contemplates modern challenges. The system attempts to achieve a natural order by balancing the system components' workflow through the announcer model. The announcer model promises to provide an intellectual space for coordination and collaboration.

**Keywords** Announcer · Computation · Inventory

## 1 Introduction

The accelerated shifts in the business environment have contributed to the development of complex supply-chain networks [1]. The storage of unprocessed materials

---

P. Hegade (✉) · N. Lingadhal · U. Khan · T. Kale · S. Basavaraddi  
KLE Technological University, Hubballi, Karnataka, India  
e-mail: [prakash.hegade@kletech.ac.in](mailto:prakash.hegade@kletech.ac.in)

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2021  
S. M. Thampi et al. (eds.), *Advances in Computing and Network Communications*,  
Lecture Notes in Electrical Engineering 735,  
[https://doi.org/10.1007/978-981-33-6977-1\\_22](https://doi.org/10.1007/978-981-33-6977-1_22)

277

and goods, work-in-progress inventory, processed goods storage, and end-to-end requirement fulfillment are crucial parts of the supply chain [2, 3]. Inventory stands as an intermediary in the process, a holding and transaction space. The management of each of these components along with holistic enterprise design has seen timely changes fulfilling the business needs. While management is the desired goal, it unsurprisingly must also blend in the optimization challenge [4]. The recent digital transformations have integrated the enterprise components easing several communication channels and introducing coordination challenges in the digitally connected network [5, 6]. The process of digitization has reduced delivery time and loss of goods and has increased the effective product life cycle time [7].

A system is a complex organization of sub-systems connected to function as a single network. The sub-systems which comprise the system traditionally contribute either as the interacting sub-systems or toward the systemic end results. The following properties of the system can be considered for discussion. Firstly, the nature of the sub-system interaction with respect to other sub-systems and its contribution to the system. Secondly, the category of information that is communicated across the system. Thirdly, the topology of sub-system connections and interactions. Fourthly, the sub-system features and tasks identifying the pattern communications and relationships.

While the system behavior can be observed and monitored as above, it also provides a scope for improvement in various regards. The current outcomes of the system can be used to improvise the future operations. The operating status of the sub-systems can be optimized by utilizing the status of other sub-systems. The interconnected systems can communicate in predefined frequency to optimize the operational compartment. The critical intermediate results of one sub-system can assist in the functional operations of other sub-systems. While sub-systems might be working on the dedicated assigned tasks, they might as well also depend on task completion of other sub-systems.

We have advanced computational capabilities and have been providing various tools and technologies to manage these systems. The comprehensive automation, however, questions the feasibility and maintainability of the process [8]. Even though there are ample challenges yet to be solved and of what an automated system could do, a better computational model is certainly need-of-the-hour—a model whose sum is larger than its individual contributing parts.

In this paper, we propose the announcer model. The announcer model aims to solve resource management across the components of the supply chain computationally. The paper is further divided into the following sections. Section 2 presents the literature survey. Section 3 presents the model, design principles, and architecture. Section 4 presents the results and discussion. Section 5 concludes the paper along with the future scope.

## 2 Literature Survey

The rise and fall mannerism, be it economy or within an organization, is a common phenomenon. The objective is usually to minimize the fall and its after-effects if so ever. The different bodies in an organization works toward a common goal. Different sections of an organization are designed to interact and achieve the intent of an organization [9]. The factors that stand as a reason for the improvements or deterioration in an organization usually form a pattern [10]. The complexity of these patterns is directly proportional to the complexity of the social order in an organization. These patterns can be analyzed to predict the direction an organization is heading at [11].

The division of labor in the society is a long-known phenomenon [12], so is its adaptation for system design. Work units can be designed to increase productivity. The sharing of related knowledge can increase the scope for improvement and experimental opportunities for the groups. This has had a positive impact on performance [13]. Entities that are heterogeneous and self-adaptive collaborate with each other forming collective adaptation. Implementing collective adaptation requires mutation, and also care should be taken to maintain the coherence of the system [14]. The work outcome is effective if the work is shared with a specialized set of workers. This division creation makes the individuals of a particular group specialized in a particular task. This helps the social groups evolve and perform a given task sophisticatedly, leading to the evolution of the group [15].

Considering e-commerce as a specific domain, e-commerce involves the use of communication technology in business transactions to build the relationship between the organizations and the individuals or among the organizations [16]. Improvements in the performance at any stage can contribute to the improvement of the entire cycle. With the advent of technology, supply chain management also has information-flow as a core component [17].

Though the supply chain in the system is automated, it has its own shortcomings. Automation usually intents toward process automation and quality management [18]. Monitoring and tracking capabilities have their own challenges dependent on the domain [19]. Using computational intelligence approaches to minimize the supply chain cost along with embedded risk has been more than a decade old [20]. With emerging industry 4.0 standards, integration with IoT devices is also another prominent challenge [21]. Security is another concern that is also under discussion [22]. The evolving software tools and technologies demand appropriate information architecture for the supply chain quality management [18]. Enterprise architectural frameworks have also been designed for supply chain integration [23].

Among different retailers, Wal-Mart was the one to experiment with point-of-sale and storage levels, also implementing a central database. With the help of information technology, the operators could know the real-time information about the commodities stored in centers for distribution [24]. Wal-Mart's task was to reduce the goods' storage to an extent, as possible [25]. They used a cross-docking method, in which the commodities were not stored anywhere between transportation [26].

Though there are computational optimization models for production planning in supply chain and quantitative models for the supply chain, likewise, these models target a specific component [27, 28]. A need for a unified model is, hence, justified and can bridge the existing gaps, also being in-line with current and future vision. There have been gaps in an organization, supply chain, and, to be specific, e-commerce domain. We attempt to address these gaps using a computational model—announcer.

### 3 Announcer Model

This section presents the announcer model with its design principles, architecture, algorithms, abstract data type, and also a self-learning architecture.

#### 3.1 *Design Principles*

The design principles of the model are as follows: Firstly, to design a computational model to realize the interactions between the system components. Secondly, to design a collaboration methodology to balance and restore the natural order in the system. The design principles are derived from the foundations of computational thinking. The algorithm designed for the announcer rests on the principles of decomposition, pattern recognition, and abstraction. The system is decomposed into sub-systems, challenges, and interactions are abstracted after pattern recognition.

#### 3.2 *System Architecture*

The announcer adapts from a client–server architecture model where the sub-systems are connected to the announcer space. The announcer space has permission, structure of information, tag validation, and view as essential components (See Fig. 1). The sub-systems are represented as sections in the architecture space. Though the sections might communicate with each other in organizational space, the information pertinent to announcer are not communicated across and only via shared announcer space.

An announcement has a predefined format, and only the specified format messages are recognized. The formatted structure consists of an id, tag and an announcement message (See Fig. 2).

Each announcement is identified by the ‘id’ field. Each sub-system or section in the announcer model has a unique id to distinguish announcements between various sub-systems. ‘Tags’ are announcement tags indicating the message category and define the metadata of announcement as a collection of words. Tag sets are predefined in the system. The content of the announcement is added to the ‘Message’ field. When the sub-system makes an announcement, the announcer space appends

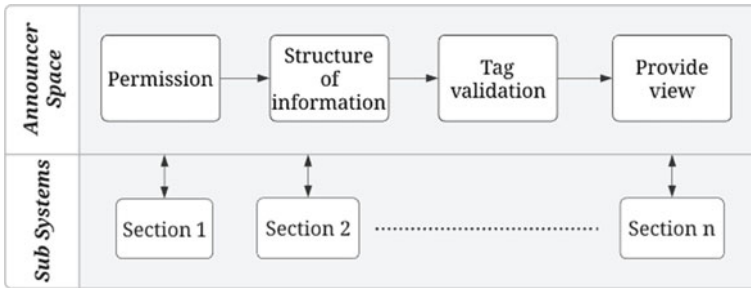


Fig. 1 Announcer architecture

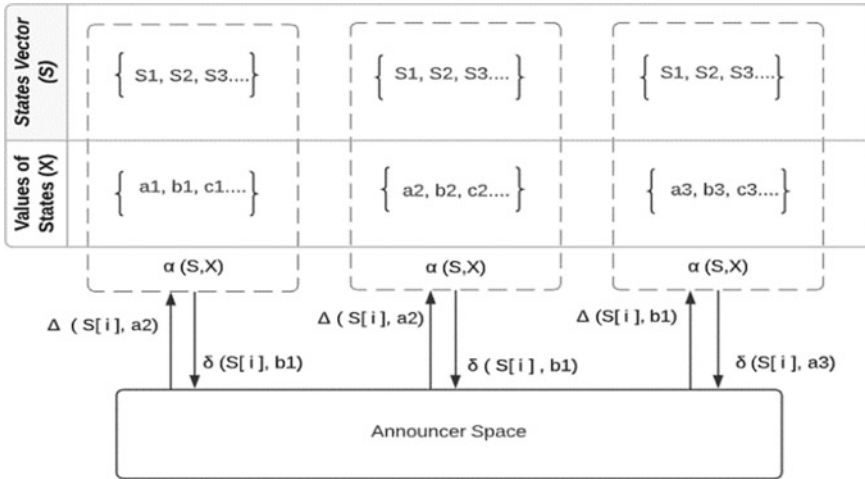
Fig. 2 Announcement format



the announcement to its collection of announcements and maintains its count. This triggers a change in the announcer space. The announcer model notes the tag field of the recent announcement and matches it with the tags subscribed by the sub-system. If the tags match with the subscription tags of the sub-system, the announcement is sent to that sub-system.

Consider for example a beverage manufacturing company that is a part of the supply chain system. The manufacturer, retailer, and raw materials supplier are its sub-systems (sections). Let us consider a scenario to understand the working of the announcer. If there is a spike in the sales of type A beverage, the retailer announces the number of sales in the announcer space. The announcement format consists of id as '1,' tag as 'type A' and message as '70% of beverages sold.' The manufacturer and the raw materials supplier being subscribed to the tag type A receive this announcement from the announcer space. For the manufacturer, this is a repercussion that they need to increase the production of type A beverages. If the manufacturer used to produce 1000 units of type A beverage, then the announcer model suggests the manufacturer improve its production by 70%, i.e., 1700 units of type A. This increase in the production will eventually trigger another announcement by the manufacturer in the announcer space with id as '2,' tag as 'type A' and message as '70% increase in production.' Both messages with id '1' and '2' are indication to the raw materials supplies to increase the supplies required for type A beverage. The extensive interactions between the sub-systems help improve resource management in the supply chain network and provide a computational benefit.

The contents of the sub-system change according to the changes in other sub-systems. These changes are made by reading the variations in other sub-systems through the announcer. The sub-system increments or decrements its production to stabilize the system. The balance of the complete system depends on the stability of



**Fig. 3** Announcer interaction model

the individual sub-system. The method for data manipulation in the sub-system is mathematically represented below (See Fig. 3).

Description of the states of the model is given as follows:

- $\alpha$ : Current state of the sub-system.
- $S$ : Vector consisting of different states in the sub-system.
- $X$ : Vector consisting of values in the states.
- $\delta$ : Change in the sub-system state with the given value.
- $\Delta$ : Change in the announcer space propagated to sub-system.
- $i$ : Index.

The natural order in the system is about balancing the supply and demand. The state of a sub-system is changed based on the announcement read to balance the natural order. After verifying the inventory, if the quantity of the stock is less than the requested read, the balance of the sub-system is calculated based on the previous records and the current state. The history of the system is closely tracked and monitored. The system tracks the following three:  $\hat{E}$ : previous record,  $\zeta$ : current state, and  $\hat{O}$ : optimal state. The previous record ( $\hat{E}$ ) is the variable consisting of the results obtained before using the current and optimal state. The current state ( $\zeta$ ) is the present state of a system. An optimal state ( $\hat{O}$ ) is the situation that a system should be after reading the announcement. The overall balance is calculated as:

$$\beta = \left( \hat{E} * \frac{\zeta_{i-1}}{\hat{O}_{i-1}} + (1 - \hat{E}) * \frac{\zeta_i}{\hat{O}_i} \right) * 100 \tag{1}$$

$$\hat{E} = \beta / 100 \tag{2}$$

where  $\beta$  is the system balance. The changes are made in order to balance the systems. The balance ( $\beta$ ) is maximized by changing the current state ( $\zeta$ ).

### 3.3 Algorithms

This section presents the algorithm for the various major components of the model. Table 1 summarizes the description of all the algorithms. The detailed algorithms follow ahead. Longer variables names are used to make the process self-explanatory.

```

ALGORITHM subscriptionLimit
input:  subscription details
output: registration details
for i  $\leftarrow$  0 to number_of_subsystems do
    if subscription_of_subsystems in subscriptionList then:
        return subscriptions
    else
        return not_subscribed
    
```

**Table 1** Algorithm description

Algorithm	Description
subscriptionLimit	Takes subscription details as input and returns the subscription if exists else returns an appropriate message
createAnnouncement	Takes message details as input and structures it into the announcer tuple
fetchAnnouncement	Fetches the announcements from the announcer space
getDetails	Acts as an interface between the announcer and the sub-system to add and retrieve announcements
createSubsystem	Adds a new sub-system by assigning appropriate permissions
getSubscription	Checks the validity of the announcer tuple for the given subscription list of the sub-system
calculateBalance	System balance is calculated considering the new demand generated as well as the current system balance. Equal weightage is given for both the current and previous system balance



**ALGORITHM** createMessage

input: message details  
output: structured message  
dict['id']  $\leftarrow$  id  
dict['message']  $\leftarrow$  message  
**for** i  $\leftarrow$  0 to length of data[lengthofIndex+1] **do**:  
    Add tag value to dict['tag']  
**return** dict

**ALGORITHM** fetchAnnouncements

input: nil  
output: nil  
**for** j  $\leftarrow$  lengthofRecievedAnnouncements to lengthOfAnnouncerSpace **do**:  
    **for** i  $\leftarrow$  0 to lengthofSubscriptionLimit **do**:  
        **if** j is not in EnteredList **then**:  
            add the value from announcerSpace to receiveAnnouncement

**ALGORITHM** getDetails

input: list of user details  
output: send a message to the announcer  
**if** newUser = 'yes' **then**  
    get username and subscriptions  
    add sub-system to the announcer and the sub-system list  
    check if a user wants to send message  
**if** True **then**:  
    get the message details  
    check if a user wants to receive the message  
    **if** True **then**:  
        validate tag and send the message to the sub-system

**ALGORITHM** createSubsystem

input: user input of details  
output: nil  
validate new user  
get user name  
add privilege of send and receive of message

**ALGORITHM** getSubscription  
input: required subscription List  
output: nil  
validate tag subscription  
validate message subscription  
validate id subscription

**ALGORITHM** calculateBalance  
input: demand and type of material  
output: balance of subsystem in percentage  
check type of user  
new\_change  $\leftarrow$  get the new available amount of stock  
prev\_bal  $\leftarrow$  previous balance of the sub-system  
curr\_bal  $\leftarrow$  current balance of the sub-system  
**if** demand = 'high' **then**  
    curr\_bal  $\leftarrow$  (prev\_bal\*(curr\_bal/new\_change)+(1 -prev\_bal) \*  
(curr\_bal/new\_change))\*100  
    print curr\_bal  
    curr\_bal = curr\_bal/100  
**else**  
    curr\_bal  $\leftarrow$  prev\_bal  
    print curr\_bal  
    curr\_bal  $\leftarrow$  curr\_bal/100  
repeat the above steps for all sub-systems

### ***3.4 Announcer Abstract Data Type***

This section presents the abstract data type (ADT) for the designed data structure. This can help to customize and apply to different domain models and applications. The ADT abstracts the operations to customize the functionality to the required domain and manoeuvres.

```

abstract typedef <<eltype>> ANNOUNCER_TUPLE(eltype);

abstract eltype createTuple(at)
ANNOUNCER_TUPLE (eltype) at;
precondition:  createTuple == (eltype == NULL );

abstract typedef <<ANNOUNCER_TUPLE>> ANNOUNCER_SPACE (eltype);
abstract checkAnnouncements(a)
ANNOUNCER_SPACE (ANNOUNCER_TUPLE) a;
postcondition:  checkAnnouncements == (len(a) == 0);

abstract addAnnouncement(a, elt)
ANNOUNCER_SPACE (ANNOUNCER_TUPLE) a;
eltype elt;
postcondition:  a == <elt> + a';

abstract eltype fetchAnnouncement(a)
ANNOUNCER_SPACE (ANNOUNCER_TUPLE) a;
precondition:  checkAnnouncements == FALSE;
postcondition:  fetchAnnouncement(a) == ANNOUNCER_TUPLE;

```

### 3.5 *Self-Learning Model*

The announcer model can also interact with the web for dynamic updates. The web can be treated like any other section or sub-system interacting with the announcer (See Fig. 4). Because the communications are formally defined, it makes minimal difference to the announcer on who is interacting as a section. Having web wrapper tailors, the system with changing market trends and needs. An organization can not only be updated with internal transactions, but also obtain the market trends from across the globe.

## 4 Results and Discussion

This section presents a sample implementation simulation. The model was implemented and tested for the company 'Knit Arena.' This section abstracts out the operations to provide a simple and working explanation of the model.

Consider a model of a spoon manufacturing company producing two types of spoons, one composed of brass and another of steel. The system consists of a retailer, manufacturer, and two suppliers producing spoons depending on the market need. The increase and decrease in the threshold are directly proportional to the

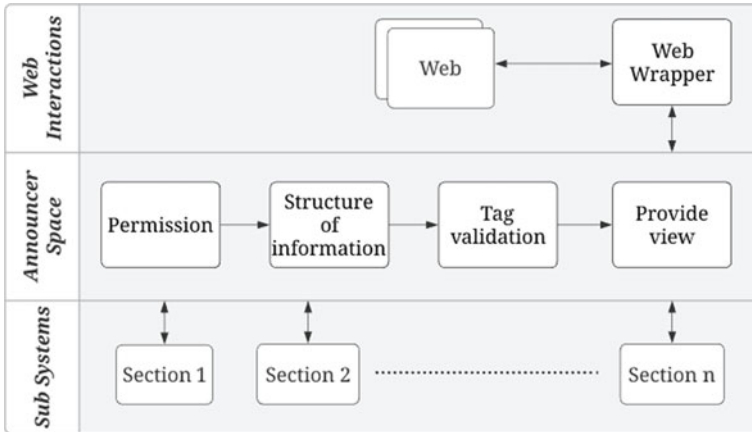


Fig. 4 Announcer with web wrapper

demand. The manufacturer checks the stock of spoons reading the demand of spoons, announced by the retailer. If the stocks remain sufficient, the manufacturer does not make any announcement. If the stock is less than the threshold, production is enhanced, and an announcement is made, stating that the requirement is high (see Fig. 5).

The supplier makes the changes accordingly to provide adequate materials to the manufacturer. Internally, computation is performed to calculate the stability of the system. To sustain the natural order, the percentage of balance calculated internally should be larger than the threshold (a threshold is defined by the organization). The states of supplier and manufacturer are collected to analyze and find the patterns in the system. The data generated by applying the model to multiple circumstances gives a precise pattern and can be used in the long run.

We further present the model with sample examples and cases. The recorded information is the interaction between a manufacturer and a retailer. The values tabularized

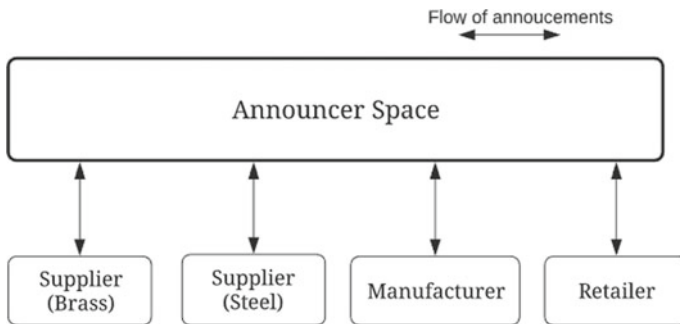


Fig. 5 Announcer for spoon manufacturing unit

**Table 2** Manufacturer readings

ID	Message	Tag
2	High	Steel
2	Low	Steel
3	High	Brass
3	High	Brass
2	High	Steel

**Table 3** Manufacturer announcements

ID	Message	Tag
1	High	Steel
1	High	Steel
1	High	Brass
1	High	Brass

are the interactive messages between the manufacturer and a retailer. The information advertised by the retailer is the demand for the product, and the manufacturer reads this information and makes appropriate changes. Then, the manufacturer announces if the stock is high or low in the announcer.

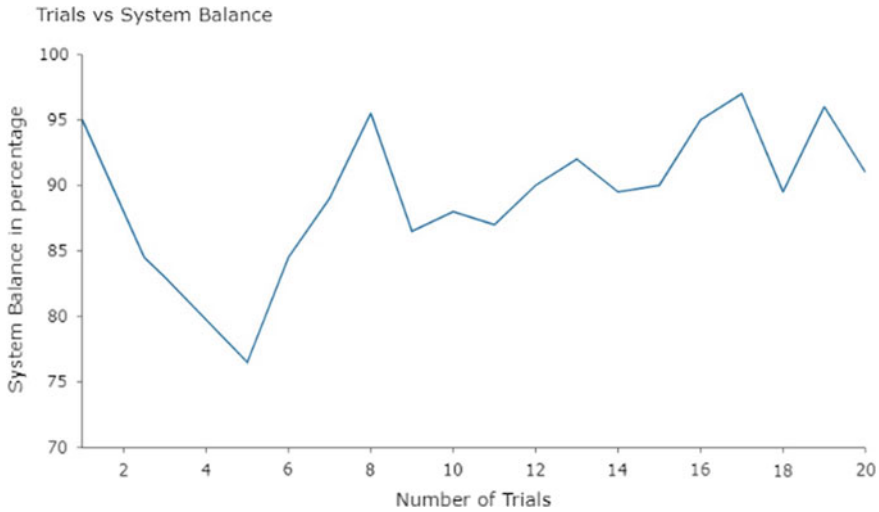
The retailer provides suggestions based on the information stated by the manufacturer. The table below shows the interaction between the retailer and the manufacturer. The results obtained as presented in the table are of one run only of the program (Similar behavior continues in a dynamic ever-running model. The presentation is snapshot only). For sake of simplicity, the messages are either ‘High’ or ‘Low’ indicating the demand status. Refer Tables 2, 3, 4, and 5 for the specific cases and samples of the interaction.

**Table 4** Retailer readings

ID	Message	Tag
1	High	Steel
1	High	Steel
1	High	Brass
1	High	Brass

**Table 5** Retailer announcements

ID	Message	Tag
2	High	Steel
2	Low	Steel
3	High	Brass
3	High	Brass
2	High	Steel



**Fig. 6** Number of trials versus percentage balance graph

The overall balance of the system was calculated using Eq. (1), and the value obtained for the run was 95.875. Figure 6 presents the timeline of stabilization.

As seen in the graph, as the trials increase, the system attempts to balance to maintain the natural order. The graph depicts the system behavior and balance pattern. The announcement made is used to tune the internal calculation of the sub-systems. The balance of the sub-system is calculated by the availability of the materials and the required amount. Calculated balance is used to indicate the status of the sub-system—the status which changes after every announcement. The past result records are stored to improve the future predictions and system balance.

Announcer, unlike the existing systems, captures every intermediate action of each sub-system and announces it to the system. These interactions which are often neglected can provide a structured means to manage and operate the systems in restored and efficient manner.

## 5 Conclusion and Future Scope

Computational capabilities are not only meant to automate a system, but the same can also be used to build models and capture the inherent characteristics of the system. The announcer is a mathematical model that universalizes all the sub-systems connected to it as a packet of three elemental structures: id, tag, and message. Considering the behavior of each system components, the interaction, feedback, and the nature of the system, the announcer attempts for a natural order in an iterative fashion.

The model promises to be a framework working with any-time, any-format plug-in components.

With the model being put to practice at Knit Arena, the immediate observations that were made are as follows. The model needs to be strengthened with formal properties and the tuple definition defined. The characteristics and hooks need to be defined to capture the additional constraints of the system. A framework needs to be built to automatically decompose a system, pattern recognize, and abstract the organization model. For this, the internal details and working of every sub-system in a system needs to be logged and analyzed. If the system is implemented with self-learning model, it will serve as a greater benefit for the mission of an organization. A state-of-art crawler can assist in building a self-learning model [29]. The system needs to be improved with respect to security aspects as well with a model like Po-Miner [30]. With computational models in demand, the announcer can serve the purpose to optimize the organization goals.

**Acknowledgements** We would like to thank Knit Arena Software Research and Services Private Limited, Hubballi, for the guidance and support in carrying out this work.

## References

1. C. Avery, P. Resnick, R. Zeckhauser, The market for evaluations. *Am. Econ. Rev.* **89**(3), 564–584 (1999)
2. G.P. Cachon, M. Fisher, Supply chain inventory management and the value of shared information. *Manage. Sci.* **46**(8), 1013–1169 (2000)
3. I. Dv, M. Ep, *Statistical Models for Constructing Mathematical Models* (Science Book Publishing House LLC, Yelm, 2014).
4. R. Wadhwa, R. Kaur, S. Bhagi, Major problems and future challenges towards e-commerce market In India. *J. Adv. Schol. Res. All. Educ.* **15**(4), 268–272 (2018)
5. S. Burt, L. Sparks, E-commerce and the retail process: a review. *J. Retail. Consum. Serv.* **10**(5), 275–286 (2003)
6. D.M. Lambert, M.C. Cooper, J.D. Pagh, Supply chain management: implementation issues and research opportunities. *Int. J. Logistics Manage.* **9**(2), 1–20 (1998)
7. P.C. Hong, T.K. Kallarakal, M. Moina, M. Hopkins, Managing change, growth and transformation. *J. Manage. Develop.* **38**(4), 298–311 (2019)
8. J.U. Min, H. Bjornsson, Agent based supply chain management automation. in *Proceedings of the Eighth International Conference on Computing in Civil and Building Engineering (ICCCBE-VIII)*, vol. 1006 (2000)
9. A. Abhijit, E-commerce in India- a review. *Int. J. Market. Finan. Serv. Manage. Res.* **2**(2):126–132 (2013)
10. N. Canak, *Wal Mart Business Case Study* (GRIN Verlag, Munich, 2006).
11. P.J. Jacques, R. Thomas, D. Foster, J. McCann, M. Tunno, Wal-Mart or world-mart? A teaching case study. *Rev. Radical Politi. Econ.* **35**(4), 513–533 (2003)
12. E. Durkheim, *The Division of Labor in Society*, 1st edn. (Simon and Schuster, New York, 2014).
13. S. Jamshed, M. Nazri, R. Abu Bakr, N. Majeed, The effect of knowledge sharing on team performance through lens of team culture. *Arabian J. Bus. Manage. Rev.* **7**(3), 64–80 (2018)
14. L. Marengo, M. Faillo, G. Dosi, Organizational capabilities, patterns of knowledge accumulation and governance structures in business firms: an introduction. *Organiz. Stud.* **29**(8–9), 1165–1185 (2008)

15. G. Cooper, S. West, Division of labour and the evolution of extreme specialization. *Nat. Ecol. Evolut.* **2**, 1161–1167 (2019)
16. S.D. Bhat, K. Kansana, J. Majid, *A Review Paper on E-Commerce* (2019).
17. N. Viswanadham, The past, present, and future of supply-chain automation. *IEEE Robot. Autom. Mag.* **9**(2), 48–56 (2002)
18. L.D. Xu, Information architecture for supply chain quality management. *Int. J. Prod. Res.* **49**(1), 183–198 (2011)
19. F. Casati, U. Dayal, M.C. Shan, E-business applications for supply chain management: challenges and solutions. in *Proceedings 17th International Conference on Data Engineering* (IEEE, Heidelberg, Germany, 2011), pp. 71–78
20. S.K. Kumar, M.K. Tiwari, R.F. Babiceanu, Minimisation of supply chain cost with embedded risk using computational intelligence approaches. *Int. J. Prod. Res.* **48**(13), 3717–3739 (2010)
21. S. Cisneros-Cabrera, A. Ramzan, P. Sampaio, N. Mehandjiev, Digital marketplaces for industry 4.0: a survey and gap analysis. in *Working Conference on Virtual Enterprises* (Springer, Cham, 2017), pp. 18–27
22. L. Urciuoli, J. Hintsu, Adapting supply chain management strategies to security—an analysis of existing gaps and recommendations for improvement. *Int. J. Logist. Res. Appl.* **20**(3), 276–295 (2017)
23. C. Chandra, Kumar, S: Enterprise architectural framework for supply-chain integration. *Indust. Manage. Data Syst.* **101**(6), 290–304 (2001)
24. K. Mark, *Half a Century of Supply Chain Management at Wal-Mart* (Harvard Business School Press, Boston, 2012).
25. T. Ha, T. Nguyen, Wal-Mart’s successfully integrated supply chain and the necessity of establishing the triple-a supply chain in the 21st century. *J. Econ. Manage.* **29**(29):102–117
26. C. Chiles, M. Dau, *An Analysis of Current Supply Chain Best Practices in The Retail Industry With Case Studies of Wal-Mart and Amazon.com.* (Massachusetts Institute of Technology, Master. MA, 2005).
27. S. Voß., D.L. Woodruff, *Introduction to Computational Optimization Models for Production Planning in a Supply Chain*, vol. 240 (Springer Science & Business Media, Berlin, Heidelberg, 2006)
28. S. Tayur, R. Ganeshan, M. Magazine (eds) *Quantitative Models for Supply Chain Management*, vol. 17. (Springer Science & Business Media, Berlin, Heidelberg, 2012).
29. P. Hegade, R. Shilpa, P. Aigal, S. Pai, P. Shejekar, Crawler by inference. in *2020 Indo–Taiwan 2nd International Conference on Computing, Analytics and Networks (Indo-Taiwan ICAN)* (IEEE 2020), pp. 108–112
30. P. Hegade, Po-Miner: random poem generator and security model. in *2020 Indo–Taiwan 2nd International Conference on Computing, Analytics and Networks (Indo-Taiwan ICAN)*. (IEEE, 2020), pp. 81–85



# Evaluation of Attributed Network Embedding Algorithms for Patent Analytics



Jinesh Jose  and S. Mary Saira Bhanu 

**Abstract** Patent analytics is a specialized branch of data analytics where patent documents are analysed to understand behavioural information. Citation network analysis is one of the common techniques to examine the importance of a patent by studying its citations. Typical patent citation network (PCN) will have millions of attributed nodes and edges. Inferencing on such a large network necessitates the use of attributed network embedding (ANE) techniques to bring down the computational requirements by reducing the dimensionality of the network data. Identifying the suitable ANE algorithm for PCN analytics is the purpose of this study. Multiple ANE algorithms are applied on the patent dataset to create low-dimensional embeddings, and these embeddings are used as the input for performing the innovation value prediction using linear regression model. Mean square error (MSE) is calculated between the predicted innovation values and the actual innovation values. MSE values obtained with different ANE algorithms are analysed to identify the most suitable ANE algorithm for patent analytics. GraphSAGE with mean-based aggregator resulted in the least MSE compared to all other ANE algorithms evaluated for patent analytics.

**Keywords** Patent analytics · Citation network analysis · Attribute network embedding

## 1 Introduction to Patent Analytics

Patent is the exclusive rights conferred to the legal owner of an invention for making public disclosure of the invention through patent document. This makes the patent document as one of the major sources of information in this technology era [2].

---

J. Jose (✉)  
Government Engineering College, Idukki, Kerala, India  
e-mail: [jinesh@gecidukki.ac.in](mailto:jinesh@gecidukki.ac.in)

S. Mary Saira Bhanu  
National Institute of Technology, Tiruchirappalli, Tamilnadu, India  
e-mail: [msb@nitt.edu](mailto:msb@nitt.edu)

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2021  
S. M. Thampi et al. (eds.), *Advances in Computing and Network Communications*,  
Lecture Notes in Electrical Engineering 735,  
[https://doi.org/10.1007/978-981-33-6977-1\\_23](https://doi.org/10.1007/978-981-33-6977-1_23)

293

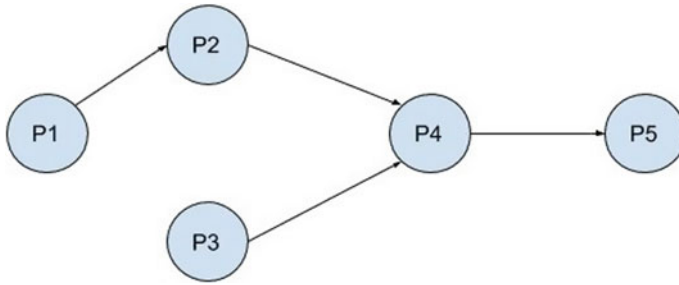
Patent analytics is a specialized branch of data analytics where patent documents are analysed to extract patterns and behavioural information [2]. Every patent document contains several features, including structured items like author and assignee, unstructured items like abstract, description, claims and backward citations [20]. Patent citation network (PCN) analysis is one of the common techniques to examine the importance of a patent by studying its citations [25]. There are many published researches to demonstrate various data analytic operations performed on PCN [17, 20]. PCN networks are extremely large and require the application of dimensionality reduction techniques to reduce the computational requirements.

Most of the PCN analytic operations have used a dimensionality reduction technique known as attributed network embedding (ANE) [6]. ANE is an active area of research, and several innovative ANE algorithms are available in the literature [7, 12, 14, 19]. All these algorithms create a low-dimensional equivalent of the network data, which is popularly known as embedding. The embeddings created by different ANE algorithms have different properties. The attainment level of higher-order proximity preservation for nodes and attributes is different in each of these algorithms. This non-uniform behaviour of the embeddings by different ANE algorithms opens a challenge in the selection of the suitable ANE algorithm for each problem domain. Published PCN analytic works have arbitrarily used any one of the ANE algorithms to reduce the dimensionality of the PCN. Reasoning behind the selection of the particular ANE algorithm for their work is undocumented. In this paper, the performance improvement in PCN analytic problem by using a different ANE algorithm which is matching with PCN requirements is demonstrated.

PCN used in this work is built from the US's patent dataset. Patent's innovation value [24] prediction is selected as the analytic operation for the performance measurement of ANE algorithms. Different ANE algorithms are applied on the PCN to create low-dimensional embeddings. These embeddings are used as the input for performing the patent innovation value prediction using linear regression. Mean square error (MSE) is calculated between the predicted innovation value and the actual innovation value of the training dataset. MSE values obtained with different ANE algorithms are analysed [1] to identify the most suitable ANE algorithm for patent analytics.

## ***1.1 Patent Citation Network***

A patent citation is a reference made to a previously published work (patent or non-patent literature) that is considered relevant to a current patent application. In every patent document, there is a reference section which contains multiple citations to other patents and non-patent literatures. Citation dataset used for this evaluation has only patents citations, and hence, non-patent citations are ignored in the PCN used for ANE evaluation. Citations are of two types, namely forward citation and backward citation.



**Fig. 1** Illustration of a patent citation network

**Forward Citation:** Forward citations are patents or non-patent literatures that cite a particular patent. This information is not directly available from the patent document. To find out the forward citations of a particular patent, all the patents citing that particular patent has to be identified [13] by means of rigorous search across the entire set of patents and non-patent literatures.

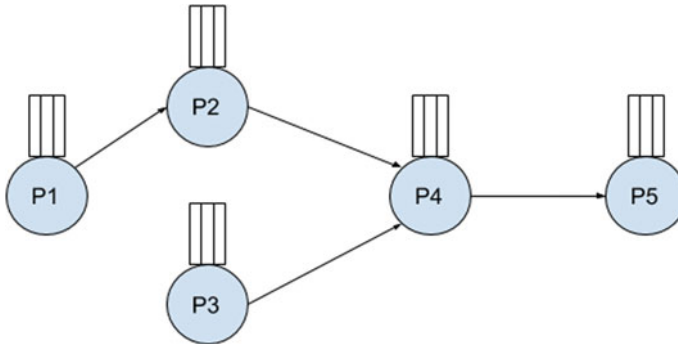
**Backward Citation:** Backward citations are patents or non-patent literatures that are cited by a specific patent document. So, all references present in a patent document are examples of backward citations of that patent. Backward citation data of any patent is directly available from its reference section. Figure 1 is an example of a simple PCN. In this example, there are five patents numbered P1 to P5. Patents are represented as the nodes, and the citation relationship among the patents is represented as the edges connecting the nodes. An incoming edge represents a backward citation, and an outgoing edge represents a forward citation. Consider the patent P4. This patent has two backward citations P2 and P3. P5 is a forward citation for P4.

## 1.2 Representation of Patent Citation Network

Attributed network is the one where the nodes or edges are having attributes. PCN is an attributed network where every node (patent) is having attributes such as number, classification type and title. Figure 2 is an illustration of a PCN with node attributes. PCN can be modelled as an attributed graph  $G$  with directed edges  $E$ .

Patent Citation Network  $G = (V, E, F)$ , where

- $V$  is the set of  $N$  nodes (patents).
- $E$  is the set of edges (citations) between  $N$  nodes.
- $F$  is a set of  $N$   $m$ -dimensional feature vectors. Every patent in  $V$  has its feature vector with its attributes such as number, classification type, title etc.



**Fig. 2** Attributed patent citation network illustration

### ***1.3 Applicability of Attributed Network Embedding***

In a typical PCN, there will be millions of attributed nodes (patents) and millions or even billions of edges (citations). Execution of analytic operations on such a huge network requires extremely high computational capacity, which is very difficult in most cases. Solution to overcome this computational barrier is the application of some dimensionality reduction techniques like network embedding [3]. Attributed network embedding (ANE) is a specialized form of network embedding where the nodes in the network have attributes. The basic idea of network embedding is to reduce the dimensionality of the network by mapping it into a low-dimensional vector space [3]. Embeddings created using ANE supports recreation of the original network from its embedding. The proximity between the nodes and attributes is partially captured with some information loss. First-order and second-order proximities are preserved in most of the ANE algorithms. When it comes to higher-order proximity preservation, each ANE algorithm performs differently with different levels of accuracy. Selection of an appropriate ANE algorithm for PCN analytics shall be made only after considering the general requirements of PCN problem domain.

### ***1.4 Important Requirements of Patent Citation Network***

Patent offices across the globe are granting millions of patents every year. The volume of patent documents available is extremely high and continuously growing. Hence, the identification of all the forward citations of any patent document is a practically unachievable target. PCN created at any time is expected to have missing forward citations, that is the PCN is expected to be an incomplete representation of the citation network. The ANE algorithms for PCN shall anticipate this incompleteness of the PCN while creating its embedding. The continuously growing nature of the patent data set adds another important patent domain specific ANE requirement,

which is termed as *inductive learning*. Inductive learning is the ability to predict the embeddings for new nodes (patents) without requiring model re-training. In other words, *inductive learning* is the ability of the algorithm to create node embeddings for previously unseen nodes [11]. This reduces the need for frequent training of the embedding creation model. All ANE algorithms are designed to produce embeddings of large networks, and hence, large size of the PCN is not a special requirement. ANE algorithms with capability to handle incomplete network structure and support inductive learning shall be selected for PCN problems.

## 2 Related Works

OpenANE is an open-source initiative to consolidate all the ANE algorithms into a single framework [4]. This project is hosted in GitHub repository. OpenANE implementations of the selected ANE algorithms are used for the performance evaluation with PCN. Three categories of ANE algorithms, namely normal, incomplete and inductive are used for this evaluation. They are summarized in Table 1.

### 2.1 Normal ANE Algorithms

These are the traditional ANE algorithms which focuses on the lower-order proximity preservation of the nodes and its attributes. Attributed social network embedding

**Table 1** Summary of ANE algorithms evaluated

Type of ANE algorithm	OpenANE implementation [4]	Supported PCN requirements	
		Incomplete network	Inductive learning
Normal ANE Algorithm	ASNE—Attributed Social Network Embedding	–	–
ANE algorithms for incomplete network	AttriPure	Yes	–
	AttriComb—AttriPure with Deep Walk	Yes	–
	ABRW—Attributed Biased Random Walks	Yes	–
ANE algorithms with inductive learning support	GraphSage—mean based aggregator	Yes	Yes
	GraphSage—GCN based aggregator	Yes	Yes

(ASNE) is a remarkable ANE algorithm belongs to this category. Social network is very similar to PCN in many terms. The work done by Dr. Lizi Liao and team for ASNE is highly equatable to a PCN [18]. Their ASNE algorithm performs embedding of attributed network data into low-dimensional vector space. ASNE is a deep neural network-based model which considers structural proximity and attribute proximity. This algorithm does not emphasize on the inductive learning and incompleteness of the attributed network.

## ***2.2 ANE Algorithms for Incomplete Network***

The research work done by Dr. Chengbin Hou and team is identified as the basic work to process large networks with incomplete structure [12]. This ANE method accepts network structure and node attributes as the input and produces unified low-dimensional node embeddings for every node in the network. The resulting node embeddings are isolated low-dimensional data points in euclidean space. Number of dimensions is proportional to the number of attributes considered for each node, but the dimensionality shall be uniform for all the nodes in the embedding. The node embeddings can be used for various different downstream tasks like node classification and link prediction. In OpenANE, three algorithms are implemented based on the above-mentioned work [12]. AttriPure is the standard implementation of the published work. Other two are the extensions of AttriPure. Attributed Biased Random Walks (ABRW) are the extension of the AttriPure with the concepts of Random Walk. The extension with the addition of Deep Walk in AttriPure is named as AttriComb. All three variants are used in the evaluation with PCN data.

## ***2.3 ANE Algorithms with Inductive Learning Support***

The research work done by Dr. William L. Hamilton and team for the inductive representational learning [11] is implemented in OpenANE as GraphSage. This algorithm creates low-dimensional embeddings for large graphs. Sage stands for sample and aggregate. This algorithm avoids the need for repeated training during each change to the network structure. Once it is trained with a network, it can produce node embeddings for the network even after including new nodes into the network. Instead of training individual embeddings for each node, this algorithm learns a function that generates embeddings by sampling and aggregating features from a node's local neighbourhood [11]. Based on the selection of aggregators, there are two implementations of GraphSage i.e., mean-based and GCN [8]-based. Incomplete network is one which is having undetected or unseen network portions. GraphSage algorithms can work with unseen nodes, and hence, it can process the incomplete networks to a certain extent. Both variants of GrapSage are evaluated for the PCN embedding creation.

### 3 ANE Evaluation Setup

The overall design of the evaluation setup to measure the performance improvement of PCN analytic operation by using different ANE algorithms is represented in Fig. 3. All the ANE algorithms are reused from the OpenANE framework. This section elaborates on the functional modules of this evaluation setup.

#### 3.1 Dataset

This work has used the publicly available dataset from PatentsView.org. Patents View ([www.patentsview.org](http://www.patentsview.org)) is a patent data visualization and analysis platform for US patent data [22]. Office of Chief Economist in the United States Patent and Trademark Office (USPTO) is the primary supporter for PatentsView platform. PatentsView has published several datasets to encourage the research activities. Data files from PatentsView are in *tab separated value* (TSV) format. Extracted data from the following files are used in this work.

1. Patent.tsv—contains the information of granted patents by USPTO from 1976
2. Uspatentcitation.tsv—contains the information of citations made to US granted patents by US patents.
3. cpccurrent.tsv—contains the information of current Cooperative Patent Classification (CPC) [21] data for all US patents.

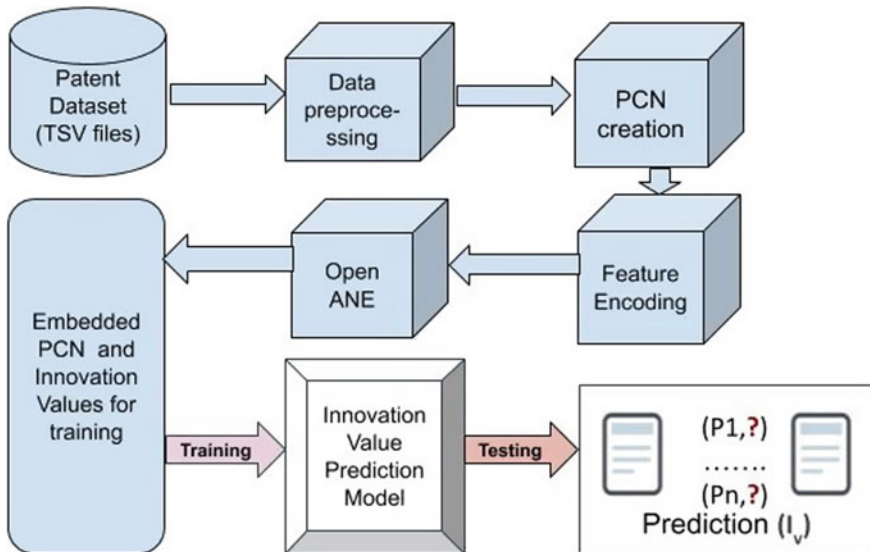


Fig. 3 ANE evaluation setup

### 3.2 PCN Creation

Some of the records in the TSV files are having formatting issues which demands cleaning steps. Partial and erroneous records are eliminated during the data cleansing stage. Information elements are spread across three TSV files. Appropriate fields from different files are selected and combined to get the desired record with all the necessary fields. Since the contents are in raw text format, appropriate encoding techniques are also necessary. After preprocessing [9], PCN is created with the chronologically filtered records. To ensure smooth execution of the experiments, *year of grant* is used as the filtering parameter to control the number of records in the processed dataset. NetworkX [10] python library is used to perform PCN creation. Patents are added as nodes and citations are added as the edges in the NetworkX instance. Number of claims and CPC-type information are added to each node as node attributes. One-hot encoding [23] is performed for the categorical node attribute. End of PCN creation has resulted in the creation of two text files, an edge list of the network and an encoded node attribute list.

### 3.3 Performance Evaluation

The evaluation setup is using different ANE implementations from OpenANE framework. The edge list and encoded node attribute list are fed to six different ANE algorithms and created six different embeddings. Each embedding is fed to a prediction model which is trained for the patent innovation value prediction. Linear regression is used in the innovation value prediction model. Let  $X$  be the embedded features and  $Y$  be the label file containing the set of actual innovation values for training. Using  $X$  and  $Y$  a linear regression model is created to predict innovation values. Once the model is built, the model is used to predict the innovation values ( $\hat{Y}$ ) of patents during the testing phase. Correctness of the predicted innovation values are measured by calculating the MSE. The MSE calculation is represented as (1).

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \quad (1)$$

where

- $Y$  is the actual innovation value of the patent from training data.
- $\hat{Y}$  is the predicted innovation value of the patent.
- $n$  is the total number of patents in the dataset.

The innovation value prediction is performed through linear regression operation. The formula [16] to compute the predicted innovation value ( $\hat{Y}$ ) is represented as (2)

$$\hat{Y} = \beta_0 + \beta_1 x_1 + \dots + \beta_k x_k + \epsilon \quad (2)$$



**Table 2** MSE values obtained with different ANE algorithms

Test–Train ratio	MSE with different ANE Algorithms					
	ASNE	AttriPure	ARBW	AttriComb	GraphSage—mean	GraphSage—GCN
20–80	2.168860	2.158519	2.137909	2.084982	2.015381	2.113457
30–70	2.101938	2.091925	2.074990	2.026121	1.954258	2.044731
40–60	2.148981	2.138447	2.123574	2.068092	2.006083	2.090283

where

- $\beta_0$  to  $\beta_k$  are the coefficients.
- $x_0$  to  $x_k$  are the encoded patent attributes.
- $\epsilon$  is the error.

Regression operation is repeated with different test–train splits [5] on the dataset. This step is to identify the optimal test–train split for performing the innovation value prediction. Suitability of the ANE algorithm for PCN analytic operation (innovation value prediction) under evaluation is decided by analysing the MSE between the predicted innovation value and the actual innovation value used for the training.

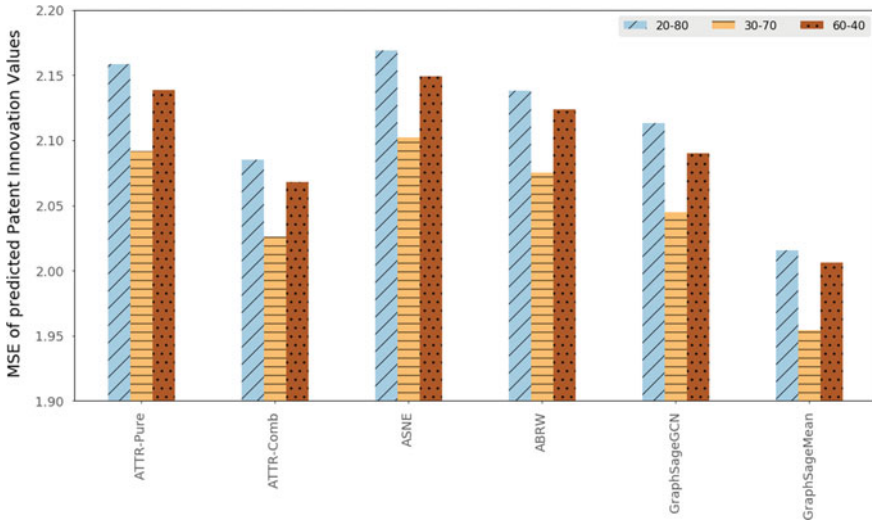
### 3.4 Innovation Value Prediction Results

Linear regression is performed with all the six embeddings. To perform this scikit-learn [15] python package is used. Regression operation is repeated with different test–train ratio to figure out the optimal test–train ratio for the PCN analytic operation with the given ANE algorithm. The accuracy of the innovation value prediction is computed by calculating the MSE. MSE values obtained with each embedding are given in Table 2.

Graphical representation of this data is presented in Fig. 4. As per this result GraphSage with mean-based aggregator gives the minimal MSE value compared to all other methods. This result is achieved with 30–70 split, i.e. 30% testing data and 70% training data

## 4 Conclusion and Future Work

In citation network-based patent analytics, there are different choices for ANE algorithms. This paper has evaluated the suitability of the well-known ANE algorithms by using them in a patent innovation value prediction problem. Multiple ANE techniques are evaluated and GraphSage with mean-based aggregator is identified as the ANE technique which produces the least MSE in patent innovation value prediction.



**Fig. 4** Performance results of different ANE algorithms, lower MSE indicates better result

This algorithm supports inductive learning and incomplete network structure. Hence, it becomes a recommendable choice for the PCN analytic problems. MSE values are analysed for different test–train split of the dataset. As per the results 30–70% is found to be the optimal test–train split. At this stage, six existing ANE techniques are evaluated. Future work is to device a novel ANE algorithm for improved performance with PCN-based patent analytic problems.

## References

1. D.M. Allen, Mean square error of prediction as a criterion for selecting variables. *Technometrics* **13**(3), 469–475 (1971)
2. L. Aristodemou, F. Tietze, *Exploring the Future of Patent Analytics* (Cambridge, 2017)
3. E.M. Bergman, Embedding network analysis in spatial studies of innovation. *Annals Regional Sci.* **43**(3), 559 (2009)
4. Z.D. Chengbin Hou, Openane: the first open source framework specialized in attributed network embedding. <https://github.com/houchengbin/OpenANE> (2018)
5. P.S. Crowther, R.J. Cox, *A method for optimal division of data sets for use in neural networks, in International Conference on Knowledge-Based and Intelligent Information and Engineering Systems* (Springer, Berlin, 2005), pp. 1–7
6. P. Cui, X. Wang, J. Pei, W. Zhu, A survey on network embedding. *IEEE Trans. Knowl. Data Eng.* **31**(5), 833–852 (2018)
7. H. Gao, H. Huang, Deep attributed network embedding. *IJCAI* **18**, 3364–3370 (2018)
8. H. Gao, Z. Wang, S. Ji, Large-scale learnable graph convolutional networks, in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1416–1424 (2018)
9. S. García, J. Luengo, F. Herrera, *Data Preprocessing in Data Mining* (Springer, Berlin, 2015)

10. A. Hagberg, P. Swart, D. Schult, Exploring network structure, dynamics, and function using networkx. Tech. rep., Los Alamos National Lab. (LANL), Los Alamos, NM (United States) (2008)
11. W.L. Hamilton, R. Ying, J. Leskovec, Inductive representation learning on large graphs, in *NIPS* (2017)
12. C. Hou, S. He, K. Tang, Attributed network embedding for incomplete attributed networks. arXiv preprint [arXiv:1811.11728v2](https://arxiv.org/abs/1811.11728v2) (2019)
13. X. Hu, R. Rousseau, J. Chen, On the definition of forward and backward citation generations. *J. Inform.* **5**(1), 27–36 (2011)
14. X. Huang, J. Li, X. Hu, Accelerated attributed network embedding, in *Proceedings of the 2017 SIAM International Conference on Data Mining* (SIAM, 2017), pp. 633–641
15. O. Kramer, *Scikit-learn*, in *Machine Learning for Evolution Strategies* (Springer, Berlin, 2016), pp. 45–53
16. M.H. Kutner, C.J. Nachtsheim, J. Neter, W. Li et al., *Applied Linear Statistical Models*, vol. 5 (McGraw-Hill, Irwin, NY, 2005)
17. J.O. Lanjouw, M. Schankerman, The quality of ideas: measuring innovation with multiple indicators (Tech. rep., National Bureau of Economic Research, 1999)
18. L. Liao, X. He, H. Zhang, T.S. Chua, Attributed social network embedding. arXiv preprint [arXiv:1705.04969](https://arxiv.org/abs/1705.04969) (2017)
19. L. Liao, X. He, H. Zhang, T.S. Chua, Attributed social network embedding. *IEEE Trans. Knowl. Data Eng.* **30**(12), 2257–2270 (2018)
20. H. Lin, H. Wang, D. Du, H. Wu, B. Chang, E. Chen, Patent quality valuation with deep learning models, in *International Conference on Database Systems for Advanced Applications* (Springer, Berlin, 2018), pp. 474–490
21. M. Palumbo, Commentary: cooperative patent classification: a new era for the world's intellectual property offices. *Technol. Innov.* **15**(2), 125–127 (2013)
22. PatentsView: Patents view. <https://www.patentsview.org/>. Accessed 17 Dec 2019
23. K. Potdar, T.S. Pardawala, C.D. Pai, A comparative study of categorical variable encoding techniques for neural network classifiers. *Int. J. Comput. Appl.* **175**(4), 7–9 (2017)
24. G. Silverberg, B. Verspagen, The size distribution of innovations revisited: an application of extreme value statistics to citation and value measures of patent significance. *J. Econ.* **139**(2), 318–339 (2007)
25. I. Von Wartburg, T. Teichert, K. Rost, Inventive progress measured by multi-stage patent citation analysis. *Res. Policy* **34**(10), 1591–1607 (2005)

# A Comparative Analysis of Garbage Collectors and Their Suitability for Big Data Workloads



Advithi Nair, Aiswarya Sriram, Alka Simon, Subramaniam Kalambur,  
and Dinkar Sitaram

**Abstract** Big data applications tend to be memory intensive, and many of them are written in memory managed languages like Java/Scala. The efficiency of the garbage collector (GC) plays an important role in the performance of these applications. In our paper, we perform a comparative analysis of Java garbage collectors for three commonly used big data workloads to check the choice of the garbage collector for each of the workloads. The garbage collectors under scrutiny are garbage first, parallel and ConcurrentMarkSweep. We demonstrate (a) the relative difference between existing Java workloads that are used to study garbage collectors and big data workloads and (b) the selection of the right garbage collector for a given workload. We find that the garbage first collector gives a performance uplift of up to 15% in certain workloads.

**Keywords** Big data · ConcurrentMarkSweep (CMS) · Dacapo · Garbage collections · Garbage collector (GC) · Garbage first (G1) · Java development kit (JDK) · Java virtual machine (JVM) · Parallel · Pause time

---

A. Nair · A. Sriram (✉) · A. Simon · S. Kalambur · D. Sitaram  
PES University, BSK 3rd Stage, Bengaluru, India  
e-mail: [aiswarya.spaa@gmail.com](mailto:aiswarya.spaa@gmail.com)

A. Nair  
e-mail: [advithi.nair@gmail.com](mailto:advithi.nair@gmail.com)

A. Simon  
e-mail: [alkasimon23@gmail.com](mailto:alkasimon23@gmail.com)

S. Kalambur  
e-mail: [subramaniamkv@pes.edu](mailto:subramaniamkv@pes.edu)

D. Sitaram  
e-mail: [dinkar.sitaram@gmail.com](mailto:dinkar.sitaram@gmail.com)

## 1 Introduction

Efficient memory management is a requirement for applications dealing with massive amounts of data and long run times. In languages like C/C++, the programmer is responsible for allocation and deallocation of objects. However, Java has automatic memory management which eliminates the need for explicit object allocation and deallocation by the programmer. The unused objects are cleared by the garbage collector. This process of automatic garbage collection is important for big data applications, which are written in languages like Java.

The goal of our study is to recommend a particular GC for big data workloads following the evaluation of various metrics. These workloads are a part of the Big-DataBench suite [1] and run on Apache Spark [2]. Every JDK version comes with a default GC. The Java version configured on our machine is OpenJDK 12.0.1 [3], where the default GC is G1 GC [4]. In addition to G1 GC, we have considered two other garbage collectors—ConcurrentMarkSweep GC [5] and parallel GC [6]. Although there are big data specific garbage collectors like NumaGiC [7] and Yak [8], since they are not open-source GC's, we have decided to explore the de facto standard GC's. The contributions of this paper are as follows:

- Comparison of big data workloads with DaCapo benchmarks [9]: It is seen that big data applications have different characteristics and work differently with Java garbage collectors. They put a much larger load on the garbage collector. This causes greater performance overheads and these differences, therefore, merit study.
- A big data workload behaves differently with different GCs. We find that although G1 GC gives the best application run time for sort and grep workloads, there exist interesting anomalies when other GC metrics are considered. For sort workload, G1 GC is faster than CMS and parallel GCs by 12.57% and 15.31%, respectively. For grep workload, G1 GC is faster than CMS and parallel GCs by 5.25% and 6.78%, respectively.

This paper is structured as follows: Section 2 presents related work. Section 3 provides an overview of garbage collection. Section 4 describes our methodology, test environment and benchmarks chosen. Sections 5 and 6 consist of our results and conclusions. Section 7 outlines future work that can be undertaken.

## 2 Related Work

With big data applications occupying the forefront of technology these days, automatic memory management for programs that handle sizeable amounts of data has become more important. Over the past few years, a considerable amount of work has been done with regard to garbage collectors in Java. While memory management and garbage collection are an important component in Android-based systems which are written in Java [10, 11], our focus is on big data workloads and their performance

on servers. While the focus has shifted to big data only in the recent years, there has been work done on impact of garbage collector on performance and analysis of GCs on multicore [12] and multithreaded environments. These works used real-time Java benchmarks or experiment on client server environments.

Gidra et al. [13] answered questions pertaining to GC performance on many cores, its effect on applications and explored bottlenecks affecting GC scalability. They experimented with Dacapo benchmarks and demonstrated that GCs did not scale as pause time increased with increase in number of cores.

Carpen-Amarie et al. [14] compared G1 GC, CMS GC and parallel GC. In an academic environment, ParallelOld GC is found to be stable, whereas in client-server environment, G1 and CMS GCs are found to have lower pause times.

Lengauer et al. [15] described a Java benchmark study on memory and GC behaviour on Dacapo and SPECjvm2008 benchmarks. Suitability of a GC was estimated based on number of allocations, GC count, GC pauses, GC time, etc. They concluded that G1 GC performed better in most cases with respect to execution time.

HGrgic et al. [16] and Pufek et al. [17] analyzed suitability of G1 GC, parallel and CMS GC on Dacapo benchmarks. HGrgic et al. [16] performed analysis on JDK 9 and stated that CMS and Parallel GCs are equally good. G1 GC is better suited for multithreaded environments but less preferred for single-threaded environments.

Pufek et al. [17] performed a comparative study on JDK 8, 9 and 11. The authors concluded that G1 GC was best or on par with parallel GC while CMS and serial GC did not seem to perform well.

Iqbal et al. [18] performed a comparative analysis on garbage collectors in Sun HotSpot, Oracle JRockit and IBM J9 on different hardware systems. They concluded that Sun HotSpot garbage collector performs better than the Oracle JRockit and IBM J9 garbage collectors.

Bruno et al. [19] explored the existing GCs for big data environments and addressed the problems of scalability in classic garbage collection algorithms. Schemes to reduce GC overhead were also discussed. A taxonomy of the studied algorithms was provided as a conclusion.

Xu et al. [20] performed analysis on parallel, CMS and G1 GCs on four spark applications, exposed GC inefficiencies and proposed strategies for designing big-data-friendly garbage collectors.

In these papers, the benchmarks under consideration have mostly been smaller sized Java benchmarks. The work done on big data benchmarks explores deficiencies in the garbage collector algorithms. Our paper, however, tries to provide insight into suitability for big data benchmarks. In addition to regular metrics like GC execution time and application run time, we also consider object level statistics. Further, the behavior and impact on the garbage collector between a non-big data Java benchmark and a big data benchmark have been compared. We have chosen OpenJDK 12.0.1 since earlier researchers have considered JDK versions 8 up until the experimental version of JDK 12.

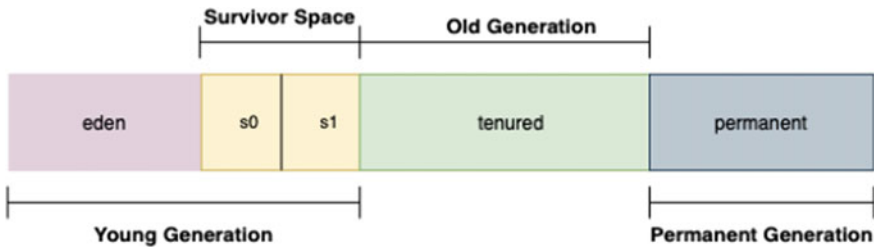


Fig. 1 Java heap

### 3 Overview of Garbage Collection

The objects in Java reside in a section of memory called the heap. As the application runs, the heap created may grow/shrink in size. When the heap becomes full, objects that are no longer used by the application (garbage) are collected. The garbage collection implementation is present in the Java virtual machine (JVM). It involves two basic steps—marking and deletion of unreferenced objects. During the marking phase, all objects are scanned in order to identify the pieces of memory which are in use (referenced). The second step involves deletion of unreferenced objects to free up space. Compaction of remaining referenced objects can be performed after this step for faster allocation of new objects.

In order to optimize the search of finding unreachable objects throughout the heap, heap is split into three generations—young generation, old (tenured) generation and permanent generation (Fig. 1). Young generation is further divided into three parts—Eden, Survivor spaces 0 and 1. New objects are allocated in the young generation. When it fills up, a minor garbage collection takes place which clears all the unreferenced objects. Objects that survive many cycles of GC are promoted to the old generation on reaching a certain age threshold. Major collection occurs when the old generation fills up. Both minor and major collections are “Stop The World” events. This means that all the application threads are halted until the process of garbage collection is complete. Metadata required by JVM is present in the permanent generation. Full garbage collection clears the entire heap, including the permanent generation.

Garbage collection has a significant impact on the application’s performance. Stop The World events, frequency of major collections and the GC execution time can greatly affect the latency/throughput of the application.

### 4 Methodology

This section describes the benchmarks used, hardware setup for running our experiments, measurements carried out and overall approach to the research.

## 4.1 Benchmarks

The rise of big data has subsequently resulted in the need for benchmarking. There are a number of big data benchmark suites available. We have chosen an open-source big data benchmark suite—BigDataBench (version 3.1.5). The workloads are implemented using Spark. “It has six real world, two synthetic datasets and 32 big data workloads, comprising micro and application benchmarks from areas of search engine, social networks, e-commerce, multimedia and bioinformatics” [21]. “It contains several data generation tools like BDGS, which generates synthetic data by scaling the real seed data, while preserving the characteristics of raw data” [21].

Additionally to run experiments on DaCapo benchmarks, we have used the DaCapo benchmark suite (version 9.12). “This benchmark suite is intended as a tool for Java benchmarking and consists of a set of open-source, real-world applications with non-trivial memory loads” [22].

We have performed our experiments on micro-benchmarks from the BigDataBench suite which comprise sort, grep and wordcount workloads. 30 GB of data for each workload was generated. Sort workload sorts the input data. Grep searches for a particular pattern of characters, and wordcount workload counts the number of occurrences of every word in input data. From the DaCapo benchmark suite, lusearch, sunflow, avrora, pmd and xalan workloads were chosen. Lusearch uses lucene to do a text search of keywords over a corpus of data comprising the works of Shakespeare and the King James Bible. Sunflow renders a set of images using ray tracing. Avrora simulates a number of programs run on a grid of AVR microcontrollers. Pmd analyzes a set of Java classes for a range of source code problems, and Xalan transforms XML documents into HTML [23].

The dataset size “Large” was provided as input to these workloads.

## 4.2 Test Environment

The machine used to perform all the experiments was an Intel 32-Core machine (Intel (R) Xeon (R) CPU E5-2620v4 @2.10GHz) with x86\_64 architecture. It consisted of two threads per core, 8 cores per socket, 2 sockets, 2 NUMA nodes, and the CPU speed was 2.1GHZ. The operating system used was Ubuntu 16.04.7 LTS with Linux 4.4.0-174-generic kernel. The Java version used was OpenJDK 12.0.1. BigDataBench version 3.1.5 was used which runs on Spark version 1.3.0 and Hadoop version 1.2.1. The pre-built jar file (dacapo-9.12-MR1-bach.jar) of the open-source DaCapo benchmark suite was downloaded on the machine. Since we wanted to observe the behavior of GCs on a multicore machine which resembles a distributed system, Spark was configured with only a single node (Table 1).



**Table 1** CPU architecture

Architecture	x86_64
CPU op mode (s)	32-bit, 64-bit
Byte order	Little Endian
CPU (s)	32
CPU GHz	1.201
CPU Max GHz	3
CPU Min GHz	12

### 4.3 Measurement

**Big Data Workloads:** As all workloads are implemented on Spark, JVM options to switch garbage collectors, disable GC overhead limit, allow detailed GC logging and redirect garbage collection outputs to a log file were provided in the Spark configuration file. Minimal tuning to ensure good performance of Spark was done prior to execution of workloads.

The source code of our workloads and methodology is present in a repository on GitHub. This link can be provided on request.

**Dacapo Workloads:** Command-line interface was used to run Dacapo workloads. The jar file along with a few JVM options to switch garbage collectors, enable GC logging were provided. Java heap was tuned by setting initial heap size to 1 GB.

The metrics measured from the garbage collection log files are shown in Table 2.

### 4.4 Procedure

Big data workloads were run with 30 GB of input data, and DaCapo benchmarks were run with dataset size “large.” These workloads were run with every garbage collector. GC logs of the same were forwarded to the GCeasy [24] tool for analysis.

**Big Data Workloads:** In order to gauge the effect of data size on the garbage collector, we first performed our experiments on 1 GB of data. This is stored in the Hadoop distributed file system (HDFS). JVM options were specified in the Spark configuration file to change the garbage collectors and obtain the log files. Each workload was then run for three iterations, using the input data. This was repeated for all three garbage collectors. The average values of metrics (from three iterations) were recorded.

We repeated the same process with 15 GB and 30 GB of data for all three workloads. Figure 2a, b illustrates the sensitivity of metrics—GC execution time and number of garbage collections with increasing input data. When comparing 1 GB and 30 GB of data, the GC execution time is 414.38 times higher for sort, 367.08

**Table 2** Metrics to evaluate garbage collector performance

Metric	Description
GC execution time	Total time taken for the execution of GC
Number of collections	Number of collections performed by the GC for one entire run of the workload
Total pause time	Pause time of the GC is defined as the amount of time Stop-The-World events take place. The application threads are halted, while the GC runs
Application run time	Total time the application runs, including the GC time. The times taken into consideration are the (User+sys) times, which gives the CPU time used by the process across multiple threads
Total created bytes	Amount of bytes created by the application. To have the best application performance, it is advised that the least amount of bytes is created
Total promoted bytes	Amount of bytes promoted from the Young Generation to the Old Generation
Reclaimed bytes	Amount of bytes reclaimed (cleared) by the garbage collector to service new allocation requests

times higher for grep and 802.31 times higher for wordcount. Similarly, the number of collections is 54.16 times higher for sort, 350.66 times higher for grep and 1478.33 times higher for wordcount.

When comparing 1 GB, 15 GB and 30 GB sizes of input data (Fig. 2a—GC execution time), the trend is linear for sort and wordcount but nonlinear for grep. In Fig. 2b, GC collections demonstrate an increasing linear trend for all three workloads.

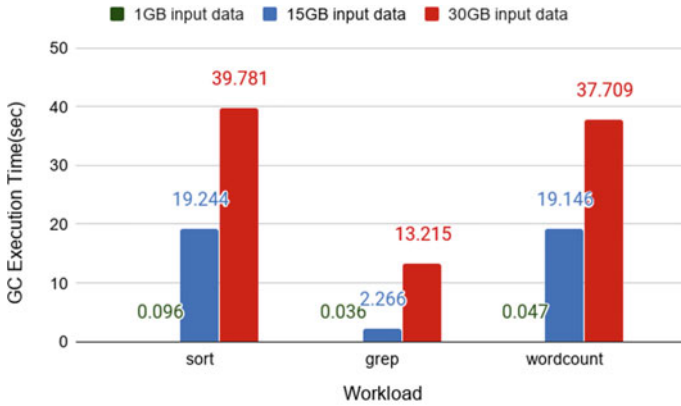
This shows that increasing the data size increases the load on the garbage collector. It also indicates that the GC execution time and the number of garbage collections are a function of the input size.

Following this, the values generated using 30 GB of data were compared among the different garbage collectors, and appropriate conclusions were drawn.

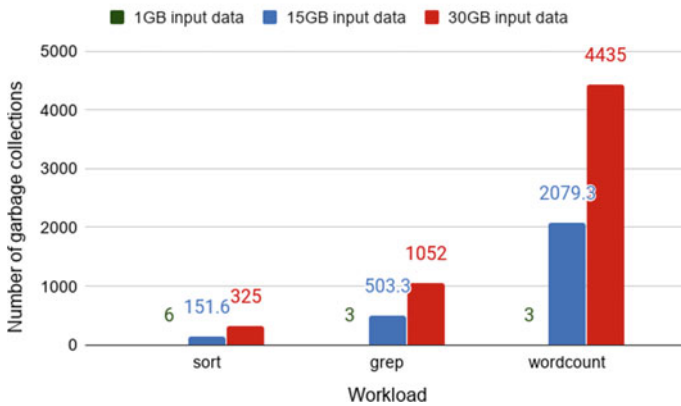
**DaCapo Benchmark Suite:** The five Dacapo benchmarks were run with every garbage collector. Each run had ten iterations, nine of which were warmup iterations. The tenth iteration was considered for the application run time. To compute the time taken by the benchmark to run, the average of run times obtained on execution of the benchmark with every garbage collector was considered.

## 5 Results

In our results, we first see how the Dacapo benchmarks and big data benchmarks vary in terms of how long the benchmarks ran and their memory consumption. Then, we move on to analysis of big data workloads to correlate the GC algorithm and



(a) GC Execution Time



(b) Number of garbage collections

Fig. 2 Performance of G1 GC on workloads with varying input

working of each GC with the workloads. This helps justify why a particular metric is low or high.

Table 3 shows the average of application run times and the created bytes (for G1 GC) for the Dacapo workloads. Table 4 shows the total number of bytes created during an entire run for the big data workloads when configured with G1 GC. We see that the total created bytes for the Dacapo workloads vary between 2 and 90 GB. Big data workloads, however, generate around 280–820 GB of data. We also observe (Table 3) that the application run times for the Dacapo benchmarks are in the range of 0–3 seconds except for avrora which takes around 30s. Big data workloads take between 8 min (grep) and around 53 min (wordcount) as seen in Table 5.

**Table 3** Run times and memory used by Dacapo benchmarks

Workload	Application run time (sec)	Created Bytes (GB)
Avrora	30.691	2
Lusearch	0.442	22.03
Sunflow	2.062	58.59
Pmd	2.082	7.13
Xalan	2.627	87.87

**Table 4** Amount of bytes created by big data workloads

Workload	Created bytes (GB)
Sort	439.37
Grep	281.2
Wordcount	819.34

**Table 5** Comparison of GC metrics for big data workloads

Workload	GC	GC execution time (sec)	No. of collections	Total pause time (sec)	Application runtime (sec)
Sort	G1	39.781	325	36.07	2730.49
	CMS	90.317	738.333	61.86	3123.069
	Parallel	61.76	267.3	61.76	3224.439
Grep	G1	13.215	1052	13.095	518.89
	CMS	27.127	2042.333	25.093	547.661
	Parallel	20.023	1833	20.023	556.631
Wordcount	G1	37.709	4435	32.799	2982.738
	CMS	204.82	6356.333	56.905	3238.987
	Parallel	54.13	6683.3	54.13	2888.811

These results present a high-level view of the immense pressure big data workloads put on the JVM and garbage collectors when compared to the Dacapo benchmarks which have very less run time and created objects.

According to Table 5, for workloads grep and wordcount, the number of garbage collections is the least for G1 GC. This matches with the theory of G1 GC, which states that it performs garbage first collection, so it minimizes the number of garbage collections by choosing regions with high amounts of garbage. We also see that G1 GC gives the least GC execution time and total pause time for grep, sort and wordcount workloads. This is a desirable characteristic. For sort and grep workloads, application run time is the least for G1 GC. However, for wordcount, parallel GC has the least application run time.

**Table 6** Comparison of object level statistics for big data workloads

Workload	GC	Promoted bytes (GB)	Reclaimed bytes (GB)
Sort	G1	155.57	293.8
	CMS	182.757	694.906
	Parallel	156.503	222.027
Grep	G1	0.371	281.163
	CMS	0.712	269.988
	Parallel	1.317	181.273
Wordcount	G1	4.84	814.5
	CMS	23.887	765.087
	Parallel	13.75	247.86

From these results, we observe the following anomalies:

- The number of garbage collections is the least for parallel GC in case of sort workload, whereas it is G1 GC for the other two workloads.
- The application run time for wordcount workload is the least when using parallel GC. This value is nearly on par with G1 GC.

As one would expect the default GC for JDK 12 to work best for these workloads, it is interesting to note that parallel GC seems to be performing nearly on par for the wordcount workload. Considering that application run time is an important metric that users look to minimize, based on this parameter, it appears as though parallel GC performs better than G1 and CMS GCs for the wordcount workload.

However, looking at the other metrics in Table 5, it is seen that GC execution time as well as number of collections is the least for G1 GC when compared with the other two GCs. Since the application run time of G1 GC is nearly on par with parallel GC (3.2% higher) and the other metrics of G1 GC indicate higher efficiency of the garbage collector, G1 GC would still be overall better suited for wordcount workload.

From Table 6, we see that the number of reclaimed bytes is consistently the least for parallel GC for all workloads. This suggests that for these particular big data workloads, parallel GC is the least efficient in terms of clearing away dereferenced objects and reclaiming heap space.

## 6 Conclusions

From these results, it is clear that big data applications have anomalies and need not always perform the best with the default GC. It can also be concluded that the characteristics of a workload itself impact how well a garbage collector can act on it. With this work, we hope to provide insight into how certain GC metrics vary with different big data applications.

While application run time could be an important metric for application users, the intention behind measuring other metrics like number of collections is to provide a

more holistic view of the working of a garbage collector. Since applications can be throughput or latency oriented, these metrics need to be viewed as a combination so that researchers may vary any metric or combination of metrics based on their needs. We hope that this work will prove as a guide for those who want to explore garbage collector suitability for other big data applications as well.

## 7 Future Work

While this paper can be used as reference to estimate suitability, further work can be done on analyzing workload characteristics. This would help in characterizing or grouping different applications in order to be able to predict suitability of a garbage collector.

The scope of our research is limited to the study of micro benchmarks, which use textual data. This could be extended to include workloads that are more intensive and are of different types (graphical analytics/machine learning based) such as Pagerank, connected components, naive Bayes and K-means workloads. The release of JDK 11 and 12 introduced new garbage collectors such as Epsilon GC, Z GC and Shenandoah GC. These garbage collectors could be explored in further studies. Tuning of garbage collectors could also be performed, by providing additional JVM parameters [25]. This way the true potential of GCs could be studied.

**Acknowledgements** The authors wish to thank Dr. Prakash Raghavendra from AMD India Pvt. Ltd. for providing intellectual assistance throughout the study.

## References

1. W. Gao, J. Zhan, L. Wang, C. Luo, D. Zheng, X. Wen, H. Tang, Bigdatabench: a scalable and unified big data and AI benchmark suite. arXiv preprint [arXiv:1802.08254](https://arxiv.org/abs/1802.08254) (2018)
2. Apache Spark. <https://spark.apache.org/docs/1.3.0/>. Accessed 14 July 2020
3. OpenJDK 12. <https://openjdk.java.net/projects/jdk/12/>. Accessed 14 July 2020
4. G1 GC. <https://docs.oracle.com/en/java/javase/12/gctuning/garbage-first-garbage-collector.html>. Accessed 14 July 2020
5. CMS GC. <https://docs.oracle.com/en/java/javase/12/gctuning/concurrent-mark-sweep-cms-collector.html>. Accessed 14 July 2020
6. Parallel GC. <https://docs.oracle.com/en/java/javase/12/gctuning/parallel-collector1.html>. Accessed 14 July 2020
7. L. Gidra, G. Thomas, J. Sopena, M. Shapiro, N. Nguyen, NumaGiC: a garbage collector for big data on big NUMA machines. *ACM SIGARCH Comput. Arch. News* **43**(1), 661–673 (2015)
8. K. Nguyen, L. Fang, G. Xu, B. Demsky, S. Lu, S. Alamian, O. Mutlu, Yak: a high-performance big-data-friendly garbage collector, in *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, pp. 349–365 (2016)
9. S.M. Blackburn, R. Garner, C. Hoffmann, A.M. Khang, K.S. McKinley, R. Bentzur, M. Hirzel, The DaCapo benchmarks: Java benchmarking development and analysis, in *Proceedings of the 21st Annual ACM SIGPLAN Conference on Object-Oriented Programming Systems, Languages, and Applications*, pp. 169–190 (2006, October)

10. Y. He, C. Yang, X.F. Li, Improve google android user experience with regional garbage collection, in *IFIP International Conference on Network and Parallel Computing* (Springer, Berlin, Heidelberg, 2011, October), pp. 350–365
11. T. Gerlitz, I. Kalkov, J.F. Schommer, D. Franke, S. Kowalewski, Non-blocking garbage collection for real-time android, in *Proceedings of the 11th International Workshop on Java Technologies for Real-time and Embedded Systems*, pp. 108–117 (2013, October)
12. L. Gidra, G. Thomas, J. Sopena, M. Shapiro, A study of the scalability of stop-the-world garbage collectors on multicores. *ACM SIGPLAN Notices* **48**(4), 229–240 (2013)
13. L. Gidra, G. Thomas, J. Sopena, M. Shapiro, Assessing the scalability of garbage collectors on many cores, in *Proceedings of the 6th Workshop on Programming Languages and Operating Systems* (ACM, New York, 2011, October), p. 7
14. M. Carpen-Amarie, P. Marlier, P. Felber, G. Thomas, A performance study of Java garbage collectors on multicore architectures, in *Proceedings of the Sixth International Workshop on Programming Models and Applications for Multicores and Manycores*, pp. 20–29 (2015, February)
15. P. Lengauer, V. Bitto, H. Möossenböck, M. Weninger, A comprehensive Java benchmark study on memory and garbage collection behavior of DaCapo, DaCapo Scala, and SPECjvm2008, in *Proceedings of the 8th ACM/SPEC on International Conference on Performance Engineering*, pp. 3–14 (2017, April)
16. H. Grgic, B. Mihaljević, A. Radovan, Comparison of garbage collectors in Java programming language, in *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)* (IEEE, New York, 2018, May), pp. 1539–1544
17. P. Pufek, H. Grgić, B. Mihaljević, Analysis of garbage collection algorithms and memory management in Java, in *2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)* (IEEE, New York, 2019, May), pp. 1677–1682
18. S. Iqbal, M. Khan, I. Memon, A Comparative Study of Garbage Collection Techniques in Java Virtual Machines. *Sindh University Research Journal-SURJ (Science Series)* **44**(4) (2012)
19. R. Bruno, P. Ferreira, A study on garbage collection algorithms for big data environments. *ACM Comput. Surv. (CSUR)* **51**(1), 1–35 (2018)
20. L. Xu, T. Guo, W. Dou, W. Wang, J. Wei, An experimental evaluation of garbage collectors on big data applications, in *The 45th International Conference on Very Large Data Bases (VLDB'19)* (2019, January)
21. BigDataBench User Manual. <http://prof.ict.ac.cn/BigDataBench/wp-content/uploads/2014/12/BigDataBench-User-Manual.pdf>. Accessed 14 July 2020
22. DaCapo benchmarks. <http://dacapobench.org>. Accessed 14 July 2020
23. DaCapo benchmarks description. <http://dacapobench.sourceforge.net/benchmarks.html>. Accessed 14 July 2020
24. Gceasy Tool. <https://gceasy.io>. Accessed 14 July 2020
25. J. Singer, G. Kooor, G. Brown, M. Luján, Garbage collection auto-tuning for java mapreduce on multi-cores. *ACM SIGPLAN Notices* **46**(11), 109–118 (2011)

# **Networked Systems and Security**



# An Innovative and Inventive IoT-Based Navigation Device—An Attempt to Avoid Accidents and Avert Confusion



Chennuru Vineeth, Shriram K. Vasudevan, J. Anudeep, G. Kowshik, and Prashant R. Nair

**Abstract** Many routes being created each day; it is a very tiresome job to remember every route. This is the reason maps are created for making our job easier. Due to an increase in technology and lowering of data rates in many countries, these maps are accessible to most of the people in the daily commute. When we want to go to a new route, we cannot remember the whole route by seeing the route provided by the map. Therefore, we need to check the route at regular intervals. This method of the commute is very much suited for pedestrians because they can hold their mobile phone in hand and can follow the route, and for some type of four-wheeler drivers as they can dock it to the dashboard and can drive the four-wheeler. The problem comes in the case of two-wheeler riders because they cannot hold their phone and drive or cannot dock the phone to their bike as it causes serious distraction from the traffic. So, to solve this situation, we have designed a device that can show the directions of the upcoming turn without using a mobile phone while driving.

**Keywords** Navigation checks · Intel UP<sup>2</sup> board · IoT and smart devices

---

S. K. Vasudevan (✉) · P. R. Nair

K. Ramakrishnan College of Technology, Samayapuram, Tamilnadu, India

e-mail: [shriramkv@gmail.com](mailto:shriramkv@gmail.com)

C. Vineeth

Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, India

J. Anudeep

Department of Electronics and Communication Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, India

G. Kowshik

Department of Electronics and Instrumentation Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, India

# 1 Introduction

These days many people started using two-wheelers instead of four-wheelers in most of the developing countries. This may be due to an increase in traffic in metropolitan cities or due to an increase in the cost of four-wheelers or may be due to an increase in awareness of pollution caused by large vehicles. In any case, the usage of two-wheelers is increasing day by day. With the increase in the usage of two-wheelers, there is a sufficiently similar increase in the number of accidents. It may be due to distraction, drowsiness, or reckless driving, etc.

Among the accidents caused by distraction, one of the main reasons is due to the usage of mobile phones while driving. Mainly, the riders use mobile phones for navigation purposes when going to new or unknown routes. According to the recent reports by India times, it is reported that 2100 people have died and many injured last year on roads mainly due to the usage of phones [1]. The statistics in Fig. 1 indicates the use of phones while driving in different parts of the globe.

Not only general two-wheeler commuters but also the food delivery and cab services face the same problem of delayed delivery or delayed service. Many people who order food online complain that their food is being delivered late. On enquiring the delivery personnel about the delay, the common reason they hear is that they were stuck in a traffic jam because of going to the wrong route or unable to find the route [3, 4]. For avoiding delayed delivery, many delivery personnel risk their life by jumping signals or riding the two-wheeler at over speed. These things pose risk not only to their life but also to others going on the same road. Most of the delay is caused to the two-wheelers due to navigation checks, so to reduce these risks and



Fig. 1 Statistics about the use of phones while driving [2]

confusion; we have developed a device with the help of IoT, which helps the riders in navigation without opening the mobile phone while driving.

## 2 Problem Statement

To design a system, which can eliminate the checking of map details and navigation updates by the drivers by picking their mobile phone out while driving, and to make the process of driving with more concentration oriented to avoid unnecessary accidents.

## 3 Existing Solutions

Here are some of the existing methods for navigation to new or unknown routes by the riders while driving.

One of the widely used technologies for navigation is maps. Many people use Google Maps, Apple Maps, etc. for navigation purposes. These applications have a feature of reading out the next turn and distance of turn through the speaker. Therefore, the two-wheeler riders use their earphones for hearing the navigation directions. This method is not at all recommended because it is against law to use earphones while driving, and it reduces the rider ability to listen to the horns and external traffic sounds while driving.

Some of the two-wheelers have a mounting device for their phone for seeing the upcoming turns and distance of turns. This is a very convenient way for the riders but this is the most dangerous way as this poses a greater risk for accidents. It distracts the drivers not only through hearing but also visually.

Ahire, D. and Patil, H. proposed a smart helmet that displays the route and directions on the helmet's visor. It is built with the help of augmented reality. The major drawback of the system is projecting the light onto the visor can obstruct the view of the driver and the vehicles heading toward him cannot be known, which is a major problem by wearing this helmet. This helmet also has a pair of earphones using this will, even more, slacken the hearing of the driver toward horns or indications [5] (Table 1).

## 4 Architecture

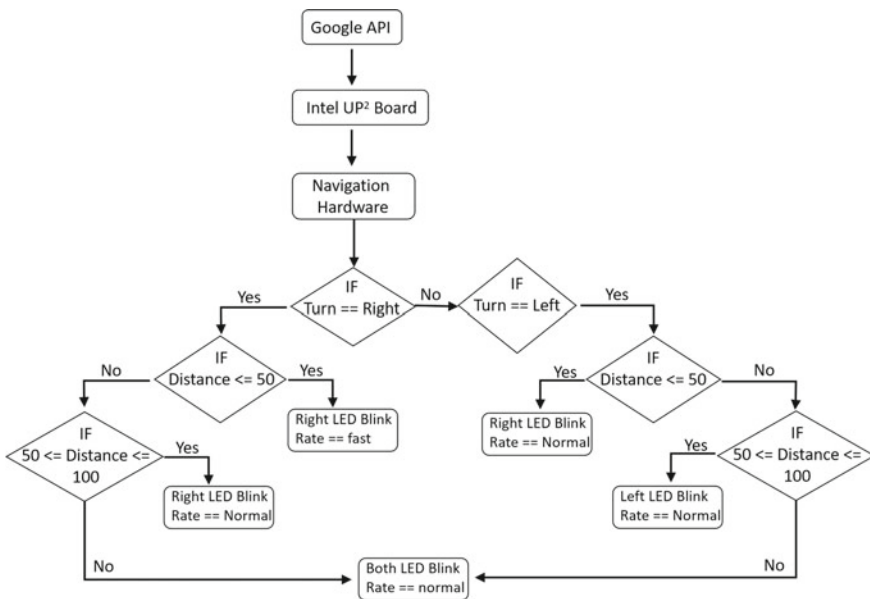
The designed system consists of a pair of small hardware that can be placed on both sides of handles or can be stuck to the hand of the rider. The hardware consists of an LED, microcontroller, and Bluetooth for communication between the microcontroller of the hardware and the main processing unit placed in front of the bike.

**Table 1** Comparison table of the existing systems with the proposed system

	Does not obstruct hearing	Does not obstruct driving view	Mobile applications	Virtual reality	Wireless communication
Earphones	×	✓	✓	×	×
Mounted device	✓	×	✓	×	✓
Smart helmet	×	×	✓	✓	✓
Proposed system	✓	✓	✓	×	✓

The code in the main processing unit is written in python and uses Google APIs for getting the upcoming direction of the turn. We have placed a display unit on top of the Intel UP2 kit for entering the origin and destination.

Figure 2 shows the workflow of the proposed system. The Intel UP2 board is the processing unit in which a python code is written for getting the direction and distance details of upcoming turn from Google. We have utilized Google Maps API for getting the appropriate details. Depending on the distance of turn, the blink rate of LED varies. For prototype purposes, we have used a processing unit like Intel UP2 board but other boards like Raspberry Pi 4 or even the system can be pre-integrated with the two-wheeler itself. A display is provided on top of the Intel UP2 board for entering the source and destination of the route to be traveled (Fig. 3).



**Fig. 2** Workflow of the proposed system

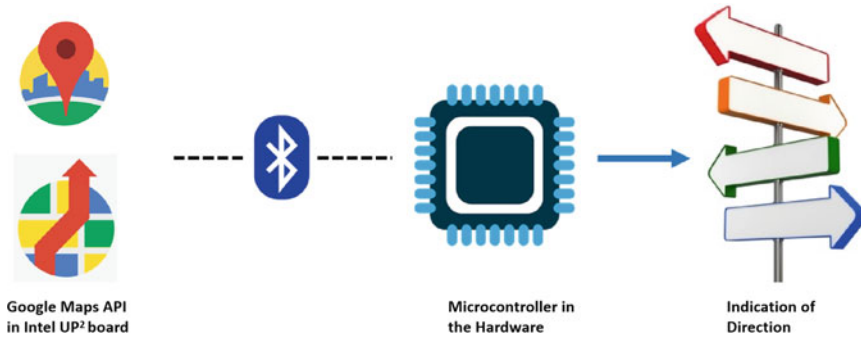


Fig. 3 Architecture of the proposed system

### 4.1 Intel UP2 Board

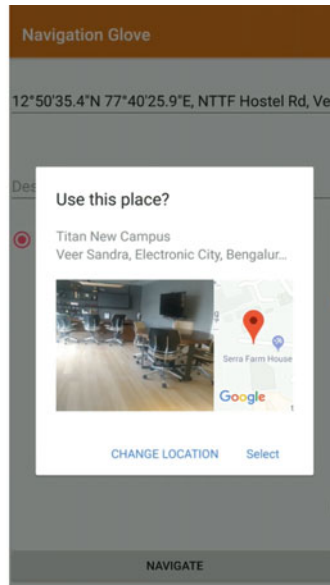
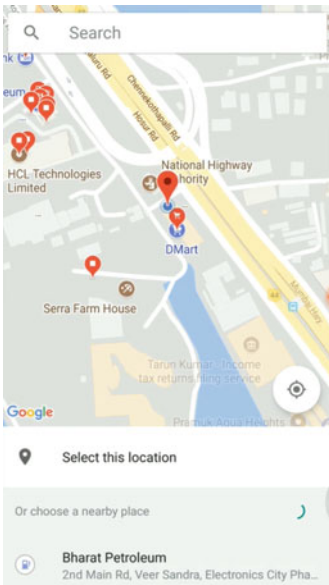
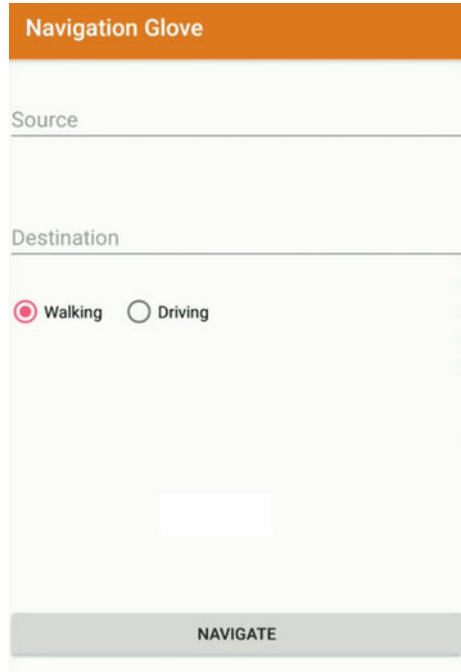
Nowadays, mobile phones are available with most of the people who ride two-wheelers because it has become handy and easy to use. However, to minimize the distraction of users while driving for navigation information, we are using a processing unit known as the Intel UP2 board. The main reason for using an external board is that the user need not use his mobile phone for the navigation purpose. In addition, we cannot have the whole processing in the navigation hardware, as these are small microcontrollers and have very little power supply. Therefore, we have attached a GPS module and display to this board, which must be kept at a comfortable place before the rider. As this is a prototype, we have used the Intel UP2 board or else it can be directly integrated into the vehicle itself. The designed interface of the board is explained below. Figure 4 shows the initial page of the application, in which the user must enter the source and destination of his route. Similar to Google Maps App, he can choose walking or driving. Walking can be chosen when the two-wheeler is a cycle.

On clicking the source or destination text input field, the user will be redirected to the area of the Google Maps around his location. The user can use the red pin for selecting the location or use the nearby places for selecting the source or destination. We have used Google places API and Maps API for better user experience (UI) (Fig. 5).

After entering the source and destination, the user will be redirected to the page, where the user needs to select the glove hardware to which the Intel UP2 board needs to send the data. The page consists of all the available nearby Bluetooth devices. Select the navigation glove (name of the Bluetooth on the hardware). On clicking the Bluetooth name, the Intel UP2 pairs with the hardware for sending the navigation details (Fig. 6).

After the Intel board successfully pairs with the hardware, it starts sending the direction and distance of the upcoming turn details in the form of JSON. It uses the GPS location from the GPS module attached to it and the Google Geolocation

**Fig. 4** Source and destination field in the display



**Fig. 5** Screen for selecting the custom source and destination

**Fig. 6** Display screen for showing and connecting with the available Bluetooth



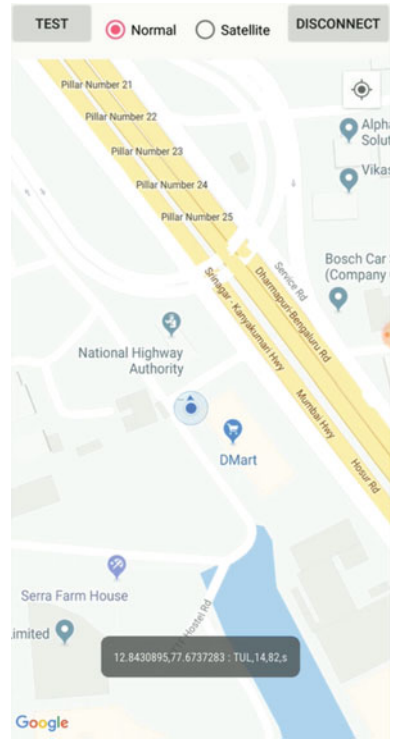
and direction APIs for getting the required information. For debugging purposes, a toast kind of message is shown at the bottom of the screen. At this point, the user has finished the configuration and they can start the journey. The same information shown in the toast will be transferred to the hardware through Bluetooth (Fig. 7).

### 4.2 Internal Working

Immediately after pairing the hardware with the Intel UP2 board, the board gets the user present location and pings the Google Maps API along with the destination location for getting the details of the upcoming turn. The returned data will be in the form of JSON. It consists of a lot of information regarding the direction of a turn to be taken, the distance of the upcoming turn, etc. It uses the link for pinging the Google Maps API for getting the information.

The received JSON data consists of start and end location coordinates, travel mode selected, the distance for the upcoming turn in kilometers and meters, the direction of the turn, time taken to reach the turn, etc., (Figs. 8, 9, and 10).

**Fig. 7** Display screen for debugging purposes



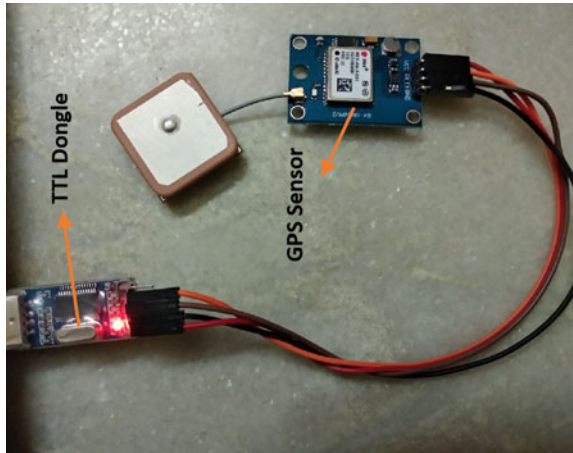
**Fig. 8** Intel UP2 board for getting the direction and distance information



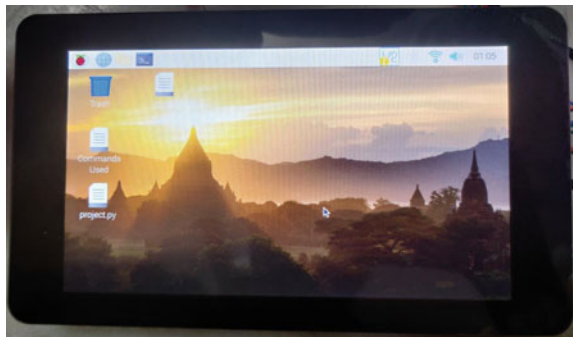
**Intel UP<sup>2</sup> board**



**Fig. 9** GPS sensor attached to Intel UP2 through TTL dongle



**Fig. 10** Display for providing the user Inputs



### 4.3 Hardware for Navigation

The hardware for navigation mainly comprises a microcontroller, Bluetooth module, power regulation circuitry, and LEDs for indication. ATtiny85 is the microcontroller used for processing the data obtained from the application. ATtiny85 is used due to its small size, which can minimize the circuitry, but it can neither process huge data nor accept heavy codes since its flash memory is only 8 KB. HC-05 Bluetooth module is used for receiving data from the board, and another pair of Bluetooth modules is used to communicate between the gloves. For regulating the 9 V voltage across the circuit, we have used LM7805 MOSFET, 1  $\mu$ F ceramic, and 10  $\mu$ F capacitors. The data received by the Bluetooth is sent to the microcontroller on the respective glove, which will parse the data to determine the direction, the distance of turn; depending on this information, the LED blink rate is executed (Figs. 11 and 12).

The below code is used to parse the obtained coded data and extract direction of the turn, distance in which turn is to be taken, etc., information from it (Fig. 13).

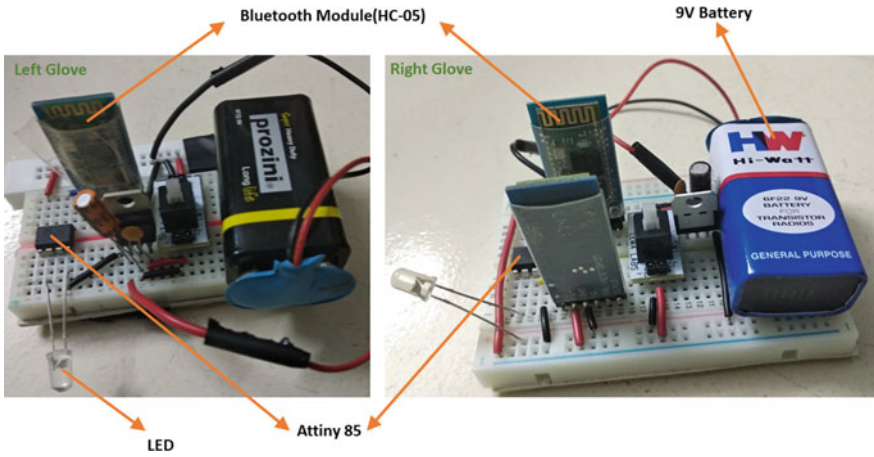
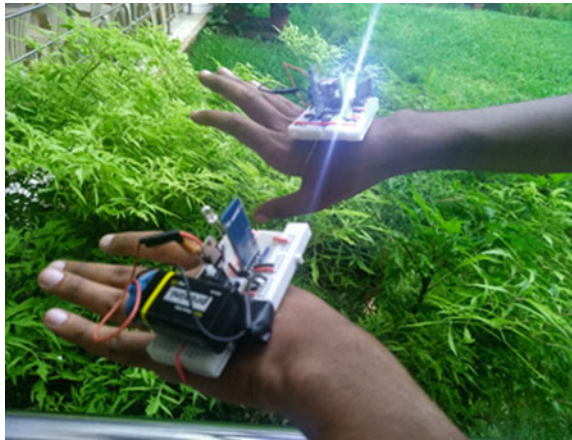


Fig. 11 Experimental setup of the proposed system

Fig. 12 Experimental setup visualized as a wearable



```

for (int i = 0; i
< input.length(); i++) {
  if (input.substring(i, i + 1) == ",") {
    pieces[counter] = (input.substring(lastIndex, i)).trim();
    lastIndex = i + 1;
    counter++;
  }
}

```

In Fig. 11, the experimental setup of our proposed system, which can be attached to the user’s hand, or the handle of the two-wheelers is shown. The visualization of

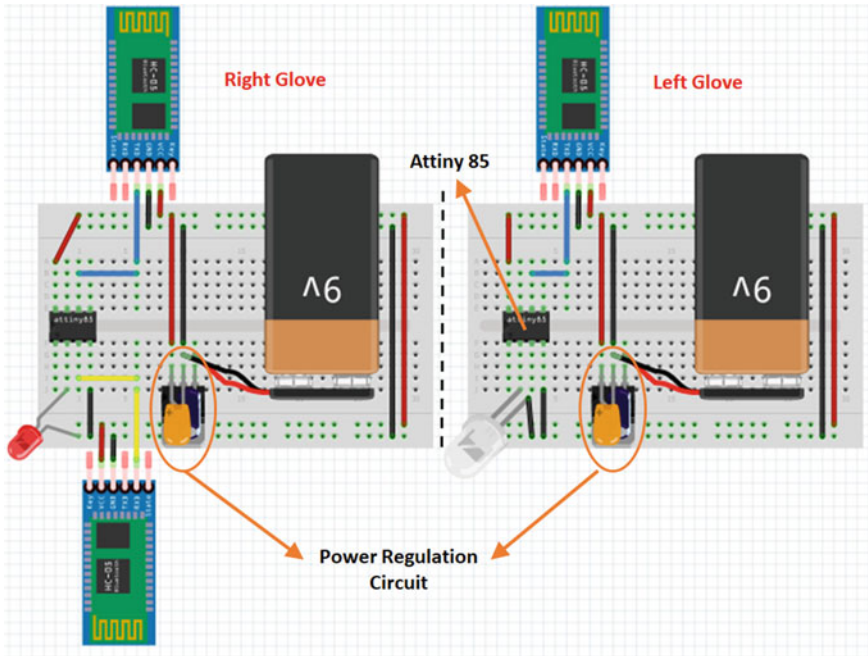


Fig. 13 Connection diagram of the navigation hardware

this experimental setup as wearable can be seen in Fig. 12. The complete connection diagram, a schematic diagram is shown in Fig. 11.

### 5 Component Table

See Table 2.

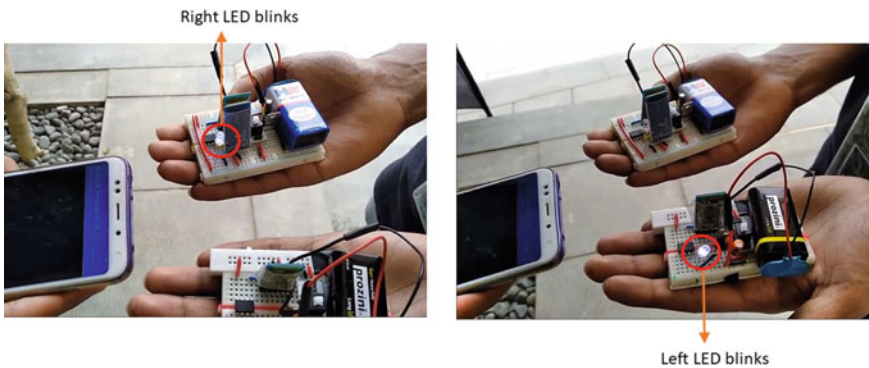
### 6 Results

The complete testing and results are obtained in real time. Figure 14 shows the blinking of right LED and left LED when obtained a similar format JSON code as explained in Sect. 4.2.

Figure 15 shows the blinking of left glove LED at a normal rate indicating a left turn to be taken when there is a left turn indication by the display. Figure 16 shows the simultaneous blinking of both LEDs for the indication of straight movement for the driver. Figure 15 shows the blinking of the right glove LED at a faster rate indicating

**Table 2** Cost of each component and their respective quantity

S. No.	Component name	Quantity	Price
1	Attiny85	2	\$6
2	Bluetooth module (HC-05)	3	\$23
3	LM7805 MOSFET	2	\$1
4	9 V battery	2	\$2
5	GPS module	1	\$10
6	1 $\mu$ F ceramic capacitor	2	\$0.5
7	10 $\mu$ F capacitor	2	\$0.5
8	LED's	4	\$1
9	Switches	2	\$2



**Fig. 14** Demonstration of working of the navigation hardware

a right turn to be taken when there is a right turn indication by the app, where the turn is less than 50 m (Fig. 17).

## 7 Conclusion and Future Enhancements

IoT has been growing in the present day in a very appreciable manner. We have used the same for developing a useful product, which minimizes the accidents and confusion of drivers regarding the navigation while driving [6–10]. Although these days’ mobile applications have become abundant sources for navigation, even then they are not able to overcome the inconvenience in navigation for the drivers while driving. With further research, we can develop a cost effective solution of the same idea by using a mobile application instead of Intel UP2 board or even integrate the same methodology in the two-wheeler itself. The proposed system can be readily implemented in the market, but the following enhancements can improve its market value.

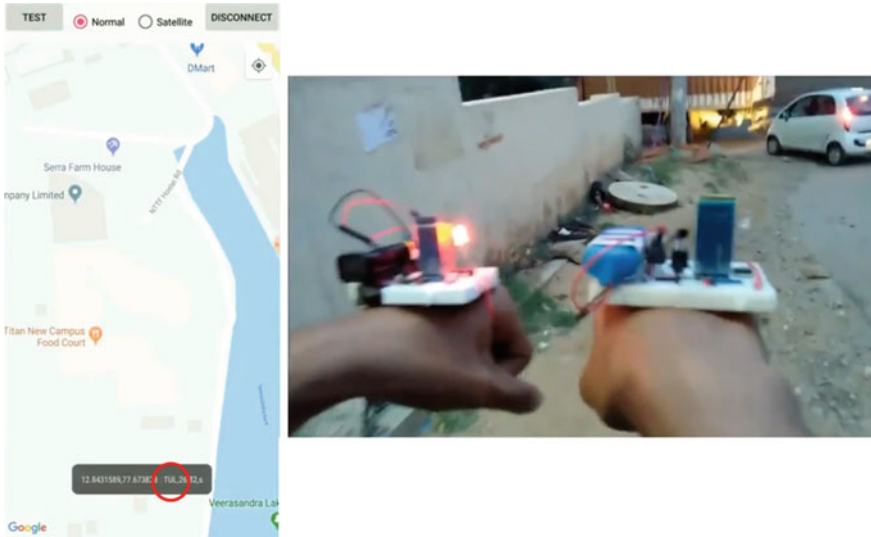


Fig. 15 Indication of left turn by our proposed system

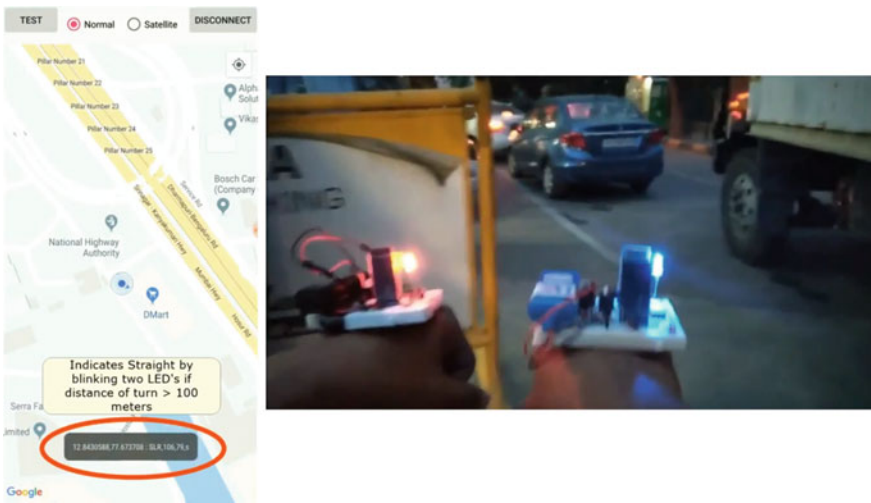
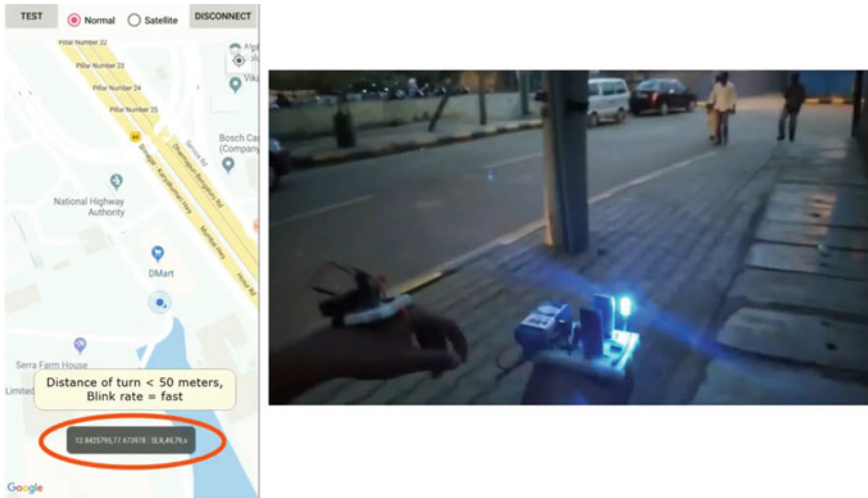


Fig. 16 Indication of straight by our proposed system

- There are many cases where the driver misses LED indication due to improper visibility. Therefore, by introducing the vibration motors, indication can be notified properly even in bad light ambiance conditions.
- In many countries, pavements are provided separate tiles for visually challenged people so they do not feel difficulty while walking straight but they usually face difficulty during turnings, and mostly, they take external help. These turns can be



**Fig. 17** Indication of right turn by our proposed system

indicated by vibration at the turn, and voice assistants can be added in the mobile application for assisting blinding people in selecting various features like source, destination.

- If the hardware is used in the form of glove, then the food delivery and cab drivers personnel's glove can be integrated with NFC to support faster payments.
- For rugged usage of a glove, it must be made waterproof and shockproof which can protect the circuitry from water, the driver from shock during a rainy day.
- Presently, we are representing only some of the Google maneuvers. In the future, a smart user interface is to be developed like OLED, hologram for effective representation of all the available maneuvers.

## References

1. Using Mobile Phones While Driving Resulted In 2100 Deaths Across India Last Year. *Indiatimes*. Updated 07 Sept 2017
2. Weighted percentage of adults aged 18–64 years who reported that they have used mobile phone (CDC Website)
3. Neeraj21891, Zomato—delayed delivery (Consumer Complaints)
4. Delivery Driver Issues (Wasserstrom)
5. D. Ahire, H. Patil, Smart helmet with live map navigation system (2018)
6. K. Velusamy, D. Venkitaramanan, S.K. Vasudevan, P. Periasamy, B. Arumugam, Internet of things in cloud. *J. Eng. Appl. Sci.* **8**(9), 304–313 (2013)
7. R. Sivaraman, S.K. Vasudevan, A. Kannegulla, A.S. Reddy, Sensor based smart traffic regulatory/control system. *Inform. Technol. J.* **12**(9), 1863–1867 (2013)

8. G. Kowshik, J. Anudeep, P.V. Krishna, S.K. Vasudevan, I. Shah, An inventive and innovative system to detect fall of old aged persons—a novel attempt with IoT, sensors and data analytics to prevent the post fall effects. *Int. J. Med. Eng. Inform.* **12**(1), 1–18 (2020)
9. A. Balachandran, M. Siva, V. Parthasarathi, S.K. Parthasarathi, An innovation in the field of street lighting system with cost and energy efficiency. *Indian J. Sci. Technol.* **8**(17) (2015)
10. S. Rohit, S.K. Vasudevan, S. Lokesh, K. Ajeeth, V. Nair, An Intelligent and Cost Effective Footboard Accident Prevention System. *Inform. Technol. J.* **12**(11), 2265–2268 (2013)

# Deploy—Web Hosting Using Docker Container



Minto Sunny, Sen Shaji, Sheen Sabu, Udith Uthaman, and Gemini George

**Abstract** In traditional web hosting, websites/web applications are configured on a bare metal server a virtual private server. For hosting multiple websites, directories are created for each website and a Linux user is created corresponding to each website. This means that a single web server/application server daemon process is responsible for serving all these websites. This is called shared web hosting. This is not a suitable solution if your application handles secret or sensitive data such as credit card numbers and bank account information. If any application handles secret/sensitive data, such applications must be deployed on dedicated servers, this is costly. This paper presents the docker containers technology which is currently being used in many production environments to package their applications in isolated environment. Further, the work elaborates how docker technology has overcome the previous issues which includes building and deploying large applications. The docker container-based deployments on the other hand isolate a website/web application and it's dependencies into self-contained units which we can run anywhere. With docker-based deployment, we can achieve a docker cloud where we can horizontally scale up and scale down the containers dynamically based on the traffic volume. Further, we can run a large monolith application or a micro-service on a docker container.

## 1 Introduction

Containers have unique recognition because of its importance in virtualization of infrastructure. One or more independent machines run virtually on physical hardware via an intermediate layer, containers run the user space on top of the operating system kernel. Docker containers provide the difference between more than one user work space instances.

---

M. Sunny · S. Shaji · S. Sabu (✉) · U. Uthaman · G. George  
Department of Computer Science and Engineering, St. Joseph's College of Engineering  
and Technology, Palai, Kerala, India  
e-mail: [sheensabu2020@cs.sjctpalai.ac.in](mailto:sheensabu2020@cs.sjctpalai.ac.in)

APJ Abdul Kalam Technological University, Thiruvananthapuram, Kerala, India

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2021  
S. M. Thampi et al. (eds.), *Advances in Computing and Network Communications*,  
Lecture Notes in Electrical Engineering 735,  
[https://doi.org/10.1007/978-981-33-6977-1\\_26](https://doi.org/10.1007/978-981-33-6977-1_26)



This paper exclusively discusses one of the containers technologies which is currently being used in many production environments to package their applications in isolated environment. The new docker containers are none other than docker which changed the view of deploying the applications in production environment. Docker is an open-source engine which was introduced by Docker Incorporation in 2013 under apache 2.0 license. The primary goal of the docker is to provide fast and lightweight environment in which to run the developers code as well as the efficient workflow to get that from the Dev environment to test environment and then into production environment [1]. Docker containers are built from application images which are stored and managed in docker hub. Users can also create their own docker registries to store their customized images which are created from a docker file or from an existing container. These flexible functionality features of docker have made it popular with in no time (Fig. 1).

Docker is one of the easiest ways for web hosting. Docker is a software platform that allows to test, build and deploy web applications. The easiest way or the basic way is just setting up a server and an Nginx. This reduces the network traffic and provides high security. By using docker, all the servers remain isolated that increase the security. It is most effective when we host bank in websites because each of the servers remains independent. The web servers are implemented inside docker containers, and hence, each server remains independent and isolated [2]. This too 2q reduces network traffic. Another advantage of this is that when any of the servers are failed, thread is divided into the rest of the servers. For this purpose, we use a load balancer [3].

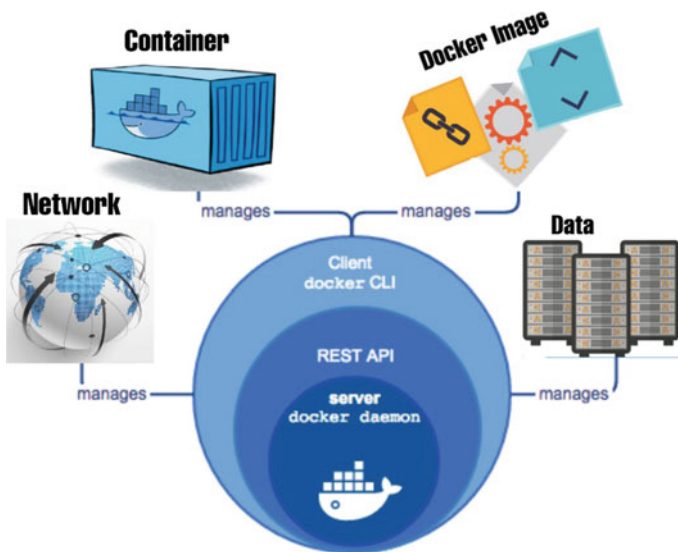


Fig. 1 Architecture

The main objectives of this web hosting using docker container are that it can decrease deployment to seconds. This is because of the fact that it can create a container for every process and even does not boot an OS. Thus, even without worrying about the cost to bring it up again, it would be higher than what is affordable and data can be created as well as destroyed. Docker guarantees that applications that are running on containers are completely segregated and isolated from each other by granting us complete control over traffic flow and management and thus is safer from a security point of view. The way docker handles the problem is one of the main key benefits of it. It gives flexibility to users to take their own configurations, put that into the code, and further deploy it without any problems. However, the requirements of the infrastructure are no longer linked with the environment of the application, as docker can be used in a wide variety of environments [4].

In today's application development environment, these large applications have been divided into small applications which collectively run across a collection of commodity hardware. Containers have become handful in running such applications on the same OS as they share the same kernel and hardware [5]. In this paper, we are discussing new container technology which is docker and we will be presenting how this technology has overcome the previous issues which includes building and deploying large applications. This paper also discusses the security features of docker which provides an additional layer of isolation and security for application services [6].

## 2 Related Works

### 2.1 Web Hosting

Web hosting is a service that permits individuals and companies to publish a web page or website on the Internet. Consequently, a web hosting service provider or a web host is an organization that offers the services and technologies required for the web page or website to be seen on the Internet. Servers are large huge storage spaces, special computers that store websites.

While online users want to view a web page or website, all they have to do is to enter the web page address into their search browser. Their computer will subsequently connect to the server, and the browser will display the web page. Generally, web hosts expect that the client have already got a website name so on host with them. However, the online host can assist the client in purchasing a website name just in case they are doing not have one. The Internet offers numerous web hosting options. There are generally two types of hosting a description of which follows.

- **Dedicated Hosting:** Dedicated servers are deployed by companies to meet the ever-growing computing demand. More storage and more security are wanted by companies. View Original Cloud servers are often utilized in lieu of dedicated servers [7] this may not only decrease the general IT budget of companies but are

going to be more flexible. Dedicated hosting plans are ideal for large organizations or websites with much higher traffic [8]. The clients get full control of the server which allows them to configure it to satisfy their own needs. The dedicated plans also come in managed and unmanaged forms where the hosting center can manage the server for the client in case of any problems. Dedicated managed servers are generally more expensive. The pricing is also influenced by the amount of resources needed such as bandwidth, storage space, and amount of RAM, among other things.

- **Shared Hosting:** Shared hosting may be a quite web hosting in which multiple websites reside on one web server. It is cost-effective and makes the administration easier for websites' owners [9]. However, shared hosting has some performance and security issues. In default shared hosting configuration, all websites' scripts are executed under the online server's user account no matter their owners. Therefore, an Internet site is in a position to access other websites' resources. This security problem arises from lack of proper isolation between different websites hosted on an equivalent web server [2] during this survey, we have examined different methods for handling mentioned security issue [2].

## 2.2 Docker

Docker is an open-source platform that runs applications and makes the tactic easier to develop and distribute. The applications that are inbuilt the docker are packaged with all the supporting dependencies into a typical form called a container. Such containers keep running in an isolated way on top of the operating system's kernel. Docker provides a facility to automate the applications when they are deployed into Containers [10] during a container environment where the applications are virtualized and executed, docker adds up an additional layer of deployment engine on top of it. The way that docker is meant is to give a quick and a lightweight environment where code can be run efficiently, and moreover, it provides an additional facility of the proficient work process to require the code from the computer for testing before production [11].

**Internal Components of Docker:** There are four main internal components of docker, including docker client and server, docker images, docker registries, and docker containers. These components will be explained in details in the following sections.

- **Docker Client and Server** Docker is often explained as a client and server-based application. View Original The entire RESTful (representational state transfer) API and a instruction client binary are shipped by docker. Docker daemon/server and docker client are often run on an equivalent machine or an area docker client is often connected with a foreign server or daemon, which is running on another machine [12].
- **Docker Images** There are two methods to create a picture the primary one is to build a picture by employing a read-only template. The foundation of each image may

be a base image. Operating system images are basically the bottom images, such as Ubuntu 14.04 LTS, or Fedora 20 the pictures of operating system create a container with a capability of complete running OS. Base image also can be created from the scratch. Required applications are often added to the bottom image by modifying it, but it is necessary to create a replacement image. the method of building a replacement image is named “committing a change.” The second method is to make a docker file. The docker file contains a listing of instructions when “docker build” command is run from terminal it follows all the instruction set and builds an image this is often an automatic way of building an image [5].

- Docker Registries Docker images are placed in docker registries. It works correspondingly to source code repositories where images can be pushed or pulled from a single source. There are two types of registries, public and private. Docker hub is a public directory where everyone can request needed images and push their own images without creating an image from the scratch. Images can be distributed to a particular area (public or private) by using docker hub feature.
- Docker Containers Docker image creates a docker container. Containers hold the whole kit required for an application, and therefore, the application can be run in an isolated way. for instance , suppose there is a picture of Ubuntu OS with SQL SERVER, when this image is run with docker run command, a container will be created and SQL SERVER are going to be running on Ubuntu OS.

Harness a company based in Sanfrancisco, the industry’s first continuous delivery—that performs service-as-a-platform designed to provide a simple, safe and secure way for engineering and DevOpsteams to release applications into production. Its headquarters is in San Francisco , CA(USA). Harness, found in 2016, uses machine learning to detect the quality of deployments and automatically roll back failed ones, saving time and it reduces the need for custom scripting and manual oversight. Harness automates the whole continuous-delivery process, uses machine learning to support when deployment fail. Every developer on the earth can enjoy releasing their code quickly and securely. But so far, they’ve had to depend on custom scripts and manual processes [13]. As we have seen in the previous sections, there are many existing solutions but they all have drawbacks at the same time. Existing systems are either costly or they would need an additional tool or requirements. With docker-based deployment, we can achieve a docker cloud where we can horizontally scale up and scale down the containers dynamically based on the traffic volume. We can run a large monolith application or a micro-service on a docker container [14].

### 3 Proposed Work and Approach

This paper exclusively discusses one of the containers technologies which is currently being used in many production environments to package their applications in isolated environment. This newly evolved containers are none aside from docker which has changed the attitude of deploying the applications in production environment. The

primary goal of the docker is to provide fast and lightweight environment in which to run the developers code as well as the efficient work flow to get that from the Dev environment to test environment and then into production environment. Users can also create their own docker registries to store their customized images which are created from a docker file or from an existing container. These flexible functionality features of docker have made it popular with in no time [15].

Web hosting is a service that permits individuals and companies to publish a web page or website on the Internet. Docker is one of the easiest ways for web hosting. Docker is a software platform that allows to test, build, and deploy web application. The easiest way or the basic way is just setting up a server and an Nginx. This reduces the network traffic and provides high security. By using docker, all the servers remain isolated that increase the security [4]. It is most effective when we host banking websites because each of the servers remain independent. The web servers are implemented inside docker containers, and this is why each server remains independent and isolated. This too reduces network traffic. Another advantage of this is that when any of the servers are failed, thread is divided into the rest of the servers. For this purpose, we use a load balancer.

The main objectives of this web hosting using docker container is it can decrease deployment to seconds. It is because of the fact that it can create a container for every process and event does not boot an OS. Hence, even without worrying about the cost to bring it up again, it would be higher than what is affordable, data can be created as well as destroyed. Docker makes sure that applications that are running on containers are completely segregated and isolated from one another , from a security point of view, by granting us complete control over traffic flow and management. The way Docker simplifies the matters is one of the key benefits of it [16]. It gives flexibility to users to take their own configuration, put that into the code, and further deploy it without any problems. However, the wants of the infrastructure are not any longer linked with the environment of the appliance, as docker are often utilized in a good sort of environments.

In today's application development environment, these large applications have been divided into small applications which collectively run across a collection of commodity hardware. Containers have become handful in running these applications on the same OS as they share the same kernel and hardware. The mission of docker is to supply following features.

### ***3.1 Easy and Lightweight Way to Model Reality***

Dockers are so fast such that one can easily containerize their applications within minutes. Users can modify their applications and dockerize their applications within no time. When a change is applied to an application a new container will be created to run these modified applications. Docker container launches instantly. Then the modified applications are packaged into the newly created containers [16].

### ***3.2 Logical Segregation of Duties***

With docker, it has become easy for an organization to segregate the duties between development and operations teams. Development focuses on developing the applications inside the containers while operations team focuses on managing these containers [17]. Docker enhances the consistency by providing the same environments in which developers write the code and operations team deploy the code. This methodology removes the conflicts between Dev and Ops teams by resolving “worked in Dev, failed in Ops” problem.

### ***3.3 Fast and Efficient Application Life Cycle Development***

The downtime within the production environment is often reduced by using docker. They reduce the cycle time between code being written by developers and code being tested, deployed by the operations team into the production environment [2]. With the above features, docker has resolved many challenges like dependency hell, imprecise documentations, tackling code-rot with image versions and barriers to adoption and research [4]. Docker containers also enhance the security features of application in two ways. One is by providing isolation between application and another is providing isolation between application and host system. They also reduce the host surface area to protect both the host and co-located containers by restricting access to the host.

## **4 Implementation Details**

### ***4.1 Website Implementation***

First of all, we'd like to put in the docker engine on of these host machines. We need to install few packages and setup stable repository, finally install the docker engine on the host machine. After installing the docker engine successfully, we deploy our website in that container which can be accessed by the outside world. Here, we will be creating a docker file which consists of all the instructions which are needed to host our website which is running inside our container. Before that we need to build our website and it should be ready to get deployed in our container. Here, we are using ubuntu as our container OS and running the update command inside the container. Figure 2 describes the overall design of the work.

Once our image has been ready to use, we should be able to spin our container to host our website. After creating the directory, we will be using docker run command and few other parameters to spin our container. Once the command has been executed Docker engine will spinning up a container with a random container id. If we want to change the content of our website, we can simply change the code in “index.html”

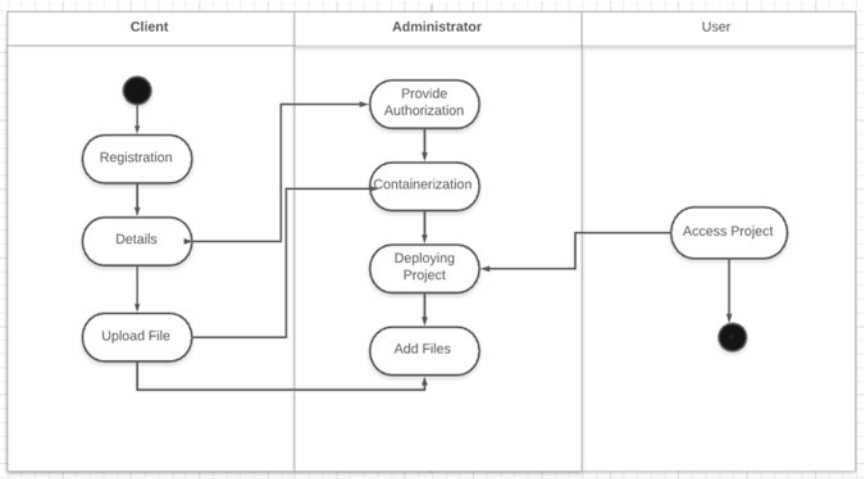


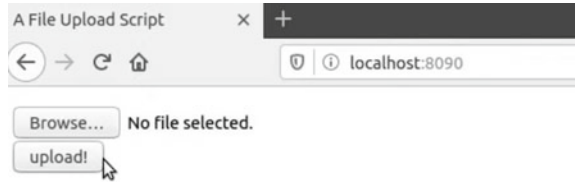
Fig. 2 Overall design

file on our docker host. Since it is mapped to the container our website content will be changed automatically, once we refresh our web page. We can create an image out of our container using docker commit command followed by our container name and desired image name. Once the image has been created, we will be pushing the image to our docker hub repo using docker push. Client needs to login to our website and log in to his account. If the client does not have an account he has to create one using our registration link. After creating an account, the client can upload files and save it. In our main site, his work will be deployed and anyone can access his project and use it. So there is an administrator, a client who owns the project, and a user who uses the client’s project. For example, the client has the program code for a simple calculator: Firstly, he uploads his file into our website in his own account, after successfully uploading the file his work will displayed in the main site as a calculator, and the user can use the client’s project. Simply we provide a space for deployment. When the project file is uploaded, a docker container is automatically created and the file is placed inside the container. The aspect of containerization is gaining a lot of attention specially with the surging popularity of docker engine for simplifying and streamlining containerization. Docker image creates a docker container. Containers hold the entire kit required for an application, and therefore, the application is often run in an isolated way for instance, suppose there’s a picture of Ubuntu OS with SQL SERVER, when this image is run with docker run command, then a container are going to be created and SQL SERVER are going to be running on Ubuntu OS.

The system is divided into three modules:

1. Registration
2. Code uploading
3. Containerization

**Fig. 3** PHP web app for uploading a file



**Registration:** Registration module consists of procedures for client registration where each client must register within the website. A user gains access to a computer system through security measures. The registered user details are going to be but future uses. they are a security measure designed to stop unauthorized access to confidential data. When a login fails (i.e., the username and password combination does not match a user account), the user is disallowed access.

**Code Uploading:** Here, the client is permitted to upload the project. Client side code is the front-end UI code. The static files do not change in the entire application’s life. Static files got to exist somewhere in order that your users can download and run them in their browser on the client side. Server side code deals with all the logic of your application [5]. It should be run on a server (machine), commonly a virtual one like an EC2 instance, much like run it when developing locally.

**Containerization:** We are using docker sdk for python for the containerization work easily.

## 4.2 Web App Implementation Using Docker

Nowadays, many people develop CRUD applications and they need a platform to deploy their application so that they can access it anywhere. Here we are offering an easy to use web GUI-based tool which they can use to easily upload and deploy their code use containers to store web applications. Figure 3 is a php web app for file uploading is running in a docker container.

We will only be supporting PHP and MySQL-based applications Key Benefits Include: Optimize Application Virtualization, Simplify Deployment and Migration, Scale the Apps Smoothly, Improve App Security, Speed up of the Applications. Figure 4 describes a simple calculator web app runs in a docker container.

## 5 Conclusion

From this paper, we can conclude that docker containers have made application life cycle development easy in all the environments such as develop, test, and production.



<b>Your Result</b>	200
<b>Enter your First num</b>	10
<b>Enter your Second num</b>	20
<b>Select Your Choice</b>	+ ▾
<input type="button" value="Show Result"/>	

**Fig. 4** PHP Web app in a docker container

It also evident that docker is often wont reproduce the environments on our local desktops or remote servers and deploy our application with no additional requirement of tools which may consume the resources of the machines. It also evident that docker are often wont to reproduce the environments on our local desktops or remote servers within no time to check and deploy our application with no additional installation of tools which may consume the resources of the machines.

## References

1. 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. <https://doi.org/10.1109/HPDC.2001.945188>
2. M.R.M. Bella, M. Data, W. Yahya, Web server load balancing based on memory utilization using Docker swarm
3. 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)
4. National Center for Biotechnology Information. <http://www.ncbi.nlm.nih.gov>
5. E. N. Preeth, Fr. Jaison Paul Mulerickal, B. Paul, Y. Sastri, Evaluation of Docker containers based on hardware utilization, in *2015 International Conference on Control Communication & Computing India (ICCC)*. <https://doi.org/10.1109/ICCC.2015.7432984>
6. Y. Li, Y. Xia, Auto-scaling web applications in hybrid cloud based on docker, in *2016 5th International Conference on Computer Science and Network Technology (ICCSNT)*. <https://doi.org/10.1109/CCSNT.2016.8070122>
7. F. Abidi, V. Singh, Cloud servers versus dedicated servers. <https://doi.org/10.1109/MITE.2013.6756294>
8. N. Naik, Building a virtual system of systems using docker swarm in multiple clouds, in *2019 IEEE 23rd International Conference on Computer Supported Cooperative Work in Design (CSCWD)*. <https://doi.org/10.1109/SysEng.2016.7753148>
9. C. Kan, DoCloud: an elastic cloud platform for Web applications based on Docker, in *2016 18th International Conference on Advanced Communication Technology (ICACT)*. <https://doi.org/10.1109/ICACT.2016.7423440>

10. P. Dziurzanski, L.S. Indrusiak, Value-based allocation of Docker containers, in *2018 26th Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP)*. <https://doi.org/10.1109/PDP2018.2018.00064>
11. A. Lingayat, R.R. Badre, A.K. Gupta, Performance evaluation for deploying Docker containers on baremetal and virtual machine, in *2018 3rd International Conference on Communication and Electronics Systems (ICES)*. <https://doi.org/10.1109/CESYS.2018.8723998>
12. M. AbdelBaky, J. Diaz-Montes, M. Parashar, *Docker Containers Across Multiple Clouds and Data Centers* (Rutgers Discovery Informatics Institute Piscataway, NJ). <https://doi.org/10.1109/UCC.2015.58>
13. V. Sharma, H.K. Saxena, A.K. Singh, Docker for multi-containers web application, in *2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA)*. <https://doi.org/10.1109/ICIMIA48430.2020.9074925>
14. H.T. Ciptaningtyas, B.J. Santoso, M.F. Razi, Resource elasticity controller for Docker-based web applications, in *2017 11th International Conference on Information & Communication Technology and System*. <https://doi.org/10.1109/ICTS.2017.8265669>
15. A. Ahmed, G. Pierre, Docker container deployment in fog computing infrastructures, in *2018 IEEE International Conference on Edge Computing (EDGE)*. <https://doi.org/10.1109/EDGE.2018.00008>
16. J. Ha, J. Kim, H. Park, J. Lee, H. Jo, H. Kim, J. Jang, A web-based service deployment method to edge devices in smart factory exploiting Docker, in *2017 International Conference on Information and Communication Technology Convergence (ICTC)*. <https://doi.org/10.1109/ICTC.2017.8190760>
17. M.S. Eremija, N.R. Ilić, M. Cvetanović, J. Protić, Z. Radivojević, Identity provider deployment based on container technology, in *2017 25th Telecommunication Forum (TELFOR)*. <https://doi.org/10.1109/TELFOR.2017.8249427>
18. A. Azab, Enabling Docker containers for high-performance and many-task computing, in *2017 IEEE International Conference on Cloud Engineering (IC2E)*. <https://doi.org/10.1109/IC2E.2017.52>
19. K. Brady, S. Moon, T. Nguyen, J. Coffman, Docker container security in cloud computing, in *2020 10th Annual Computing and Communication Workshop and Conference (CCWC)*. <https://doi.org/10.1109/CCWC47524.2020.9031195>
20. M.R.M. Bella, M. Data, W. Yahya, Web server load balancing based on memory utilization using Docker swarm, in *2018 International Conference on Sustainable Information Engineering and Technology (SIET)*. <https://doi.org/10.1109/SIET.2018.8693212>

# Enhancement of VerticalThings DSL with Learnable Features



Sandesh Ghanta, P. V. Surya Chaitanya, Sai Sarath Chandra Ganti, M. P. V. Roshan Patnaik, and G. Gopakumar

**Abstract** Resource-efficient ML for edge and endpoint IoT devices is a field of active research and increasing development. Libraries have been providing support for machine learning enthusiasts to run ML algorithms in the cloud. Executing ML algorithms on nodes is a challenge, as resources are highly constrained. To optimally use these resources, the developer often needs to have complete knowledge of the underlying architecture. VerticalThings is a domain-specific language (DSL) developed for programming ML-based embedded applications. The language offers constructs for key platform functions such as resource management, concurrency, task isolation, and security. This enables static analysis of (a) important safety and security properties, and (b) timing and power considerations. To enhance this DSL further, we developed a DSL named FieryIce which provides intelligent learning of parameters based on sensor data. We would integrate both the DSLs within the IDE developed for VerticalThings. The learnable parameters are learnt at compile time avoiding the use of scarce memory of the embedded systems. This paper shows the capabilities of a domain-specific language (DSL) named FieryIce which is designed to help embedded developers use VerticalThings and develop ML-based embedded applications with ease. The contributions of this article are: (a) A domain-specific language (DSL) named FieryIce and its capabilities to perform machine learning-related tasks; (b) how FieryIce helps students better their understanding of machine learning algorithms.

---

S. Ghanta (✉) · P. V. Surya Chaitanya · S. S. C. Ganti · M. P. V. Roshan Patnaik · G. Gopakumar  
Amrita Vishwa Vidyapeetham, Amritapuri, India  
e-mail: [sghanta05@gmail.com](mailto:sghanta05@gmail.com)

P. V. Surya Chaitanya  
e-mail: [psuryachaitanya@gmail.com](mailto:psuryachaitanya@gmail.com)

S. S. C. Ganti  
e-mail: [saisarathganti@gmail.com](mailto:saisarathganti@gmail.com)

M. P. V. Roshan Patnaik  
e-mail: [mproshan968@gmail.com](mailto:mproshan968@gmail.com)

G. Gopakumar  
e-mail: [gopakumarg@am.amrita.edu](mailto:gopakumarg@am.amrita.edu)

**Keywords** Machine learning · DSL · FieryIce · Linear regression · Logistic regression

## 1 Introduction

During recent years, there has been an increase in the use of IoT devices to automate and better perform many tasks. Highly innovative solutions have been made such as using IoT devices to perform better agronomy [1], monitoring the quality of water [2] and helping patients who need long term personal care [3]. In all such solutions, the use of ML if any is mostly performed in the cloud. There exists several problems with this approach such as cost of bandwidth, lack of security, and lack of network reliability. This encourages solutions which perform ML in the edge device itself. VerticalThings is a DSL which helps perform ML on motes. FieryICE is a supplement to VerticalThings.

### 1.1 Background

**Linear Regression** Linear regression [4] models the relationship between a scalar response (or dependent variable) and one or more explanatory variables (or independent variables). It is often used to solve problems related to prediction, forecasting, and error reduction by fitting a predictive model to an observed dataset of values of the response and explanatory variables.

**Logistic Regression** A logistic model [5] is often used to model the probability of occurrence of a certain class or an event. Such kinds of models are generally used to determine if an event occurs, however, it can be extended to determine if several events occur. It is a general practice to assign Boolean values to the occurrence of each event, such as if an event occurs its output is considered as 1 and if not it is considered as 0. It should be noted that the sum of probabilities of the event occurring and not occurring is 1.

**Gradient Descent** Gradient descent [6] is a first-order iterative optimization algorithm for finding a local minimum of a differentiable function. It is often used to find the local minimum of a convex function. There are several variants of gradient descent like ‘Adam gradient descent,’ ‘batch gradient descent,’ ‘mini-batch gradient descent,’ etc.

**Automatic Differentiation** Automatic differentiation (AD) [7] is a set of techniques to numerically evaluate the derivative of a function specified by a computer program. It takes advantage of the fact that every program, no matter however complicated it is, can be broken down into a sequence of steps which execute arithmetic operations and call elementary functions (like sine, cosine, tangent, and exponential). AD applies

chain rule repeatedly to these operations to compute the derivative of the function. There are two modes of AD, forward mode and reverse mode. In forward mode AD, the independent variable is fixed with respect to which differentiation is performed and computes the derivative of each sub-expression recursively. In reverse mode AD, the dependent variable to be differentiated is fixed and the derivative is computed with respect to each sub-expression recursively. The reverse mode computes gradients in an extremely efficient way, at least theoretically[3]. This is why it is one of the most promising modes of AD [7].

**VerticalThings** VerticalThings [8] is a programming language that provides the functions of a secure microkernel for embedded and IoT applications. Toward this purpose, a type system based on the capability model is proposed, in which resource classes are abstracted as capability classes. VerticalThings enables static analysis of key properties including safety, security, power, and timing considerations. It is used by embedded developers due to its expressive power and high level syntax which helps user write secure and efficient code easily.

**Related Work** Several DSL's which help performing specific ML-related tasks do exist, such as CVXPY [9] a DSL which helps users solve convex optimization problems; however, CVXPY does not let the user choose which type of optimization strategy to use, we address this limitation in FieryICE. There exists another DSL named Jet [10] which is meant for high performance big data processing in embedded devices. However, Jet does not support rule-based machine learning (RBML). RBML is a preferred solution over neural networks for mission critical situations because it is important for the programmer to be able to infer and easily understand why a particular decision was performed, which is simply not possible in neural networks. RBML is increasingly used in embedded devices, and much support does not exist as of now for such use cases, FieryICE is an attempt to address this limitation.

## 1.2 Problem Definition

Embedded developers writing ML code in VerticalThings have to write code to train their model from the scratch, as VerticalThings does not have any supporting ML libraries currently. Writing training code from scratch often becomes repetitive and redundant work which consumes a lot of development time. This limitation forces the developer to shift to a higher level language for training his model separately making it less developer friendly. FieryICE intends to provide intelligent learning of model parameters, especially suited for rule-based machine learning (RBML) during compile time which exempts developer from writing tedious architecture-specific code in the nascent VerticalThings language nor to shift to a high level language for this purpose. FieryICE would be integrated with the IDE of VerticalThings.

## 2 Proposed System

We propose a domain-specific language named FieryICE that supports VerticalThings, for addressing the problems persisting in programming of ML-based embedded applications and also help nascent ML programmers learn and use basic ML algorithms. Users of VerticalThings have to shift to a high level language like Python for training and learning parameters that are used in the embedded applications. The proposed DSL alleviates the pain of users of VerticalThings by providing simple syntax which will learn the parameters. ML algorithms especially their training is computationally intensive and involves a lot of matrix and vector operations. FieryICE has high level syntax which supports nested matrix and vector-based arithmetic operations. The proposed language and syntax make code more readable and expressive. FieryICE helps developers model and run linear and logistic regression tasks with a few lines of code. FieryICE lets developers model equations which can include scalars, matrices, nonlinear trigonometric, and logarithmic functions. FieryICE can be integrated with VerticalThings through the vtIDE, a special IDE designed for VerticalThings. This would help developers run FieryICE code quickly without leaving the vtIDE. These features of FieryICE make VerticalThings more usable and will bring in more competitive advantage and innovation within the domain.

## 3 The Language

### 3.1 Grammar of the Language

The language is simple and intuitive. The following is a glimpse of the grammar.

```
file : learnline inputline matrixdecls* equation* outputline EOF;
```

**Learn Line** The first line of the file will be the learn line which takes the parameters to be learnt as input. Ex: The learnable parameters in a simple regression equation  $y = m * x + c$  where  $x$  is the feature and  $y$  is the expected output as decided by  $m$  and  $c$ .

```
learnline_ : LEARN LPAREN varlist RPAREN
```

Eg: learn(m, c);

**Input Line** This line is used to specify the variables in the model that are considered as input.

```
inputline_ : INPUT LPAREN varlist RPAREN
```

Eg: input(x) // where x is feature data  
input(x,y,z) // where x y z are feature data

**Matrix Declarations** Every variable in the language is by default a matrix of size 1 \* 1 which can be considered as a scalar. The data can also be fed in the form of matrices. The matrices have to be declared with proper dimensions in the next line.

Eg: matrix m[3][3],c[3][3],y[3][3],x[3][3]

**Model of the Algorithm** Following lines will define the model of the algorithm. The model contains one or more equations which the data fits into.

Eg:

```
a = 2 * x + z
b = 4 * a
k = a + ( b * o )
l = b + k
y = (m * l) + c
```

**Output Line** The last line is used to specify the variables that are considered as output in the above equations. The equation corresponding to the output variable is considered as the model which gives the final expected output.

outputline : OUIPUT LPAREN varlist RPAREN

eg: output(y) // expected output y i.e, the result of transformations by the equations given in the model.  
output(y,z)

Listing 1.1 and listing 1.2 contain a few complete code samples of the DSL.

---

**Listing 1.1** FieryICE Code Snippet 1

```
learn(m,c)//m,c are learnable parameters
input(x,z,o) // x,z,o is the training data
a = 2 * x + z
b = 4 * a
k = a + ( b * o )
l = b + k
y = (m * l) + c
output(y)
```

---

---

**Listing 1.2** FieryICE Code Snippet 2

```
learn(m,c)
input(x)
matrix m[3][3],c[3][3],y[3][3],x[3][3]
y = m*x + c
output(y)
```

---

### 3.2 *Compilation of Code*

The FieryICE files are stored with a '.fy' extension. The FieryICE file when compiled generates a Python code which executes the ML algorithm. The Python code learns the learnable parameters specified in the FieryICE file based on the input data, model and the output data provided.

**Input File** The input data is provided in an input file. The data in the input file is given in separate lines for each variable specified as input in the FieryICE code. If the first variable in the input line  $x$  is declared as matrix of dimensions  $m, n$ , i.e.,  $x[m][n]$  in the FieryICE code, the Python code expects a list of  $m \times n$  space separated integers/float/double precision numbers all of same type in the first line of input file. If a variable is not declared as a matrix in the FieryICE code, it is considered as a  $1 \times 1$  matrix and only one number is expected as input by the generated Python code.

**Output File** The expected output is provided in the output file. The output data is given in separate line for each variable specified as output in the FieryICE code. The output file contains data in the same order of output variables specified in FieryICE code. The format is similar to that specified in the input file.

**Execution of Code** The code can be executed with the help of a shell script which performs actions in the following sequence. Compile FieryICE file  $\rightarrow$  Generate Python code  $\rightarrow$  Execute the Python code  $\rightarrow$  Return the learnt parameters.

## 4 *Internal Working*

### 4.1 *Lexer and Parser*

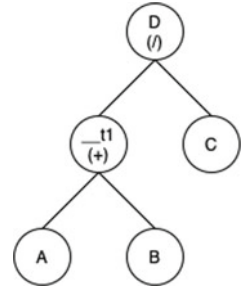
We have used ANTLR [11] specifications to write the lexical tokens and the grammar of FieryICE. ANTLR is a powerful parser generator that generates parsers which were used by us to build parse trees, and the data structure representing how a grammar matches the input. We have also used ANTLR to generate tree walkers based on the grammar and they are used to visit the nodes of those parse trees to generate abstract syntax trees (AST).

### 4.2 *Abstract Syntax Tree*

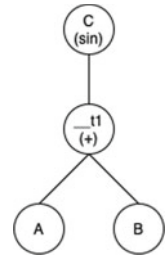
The FieryICE compiler converts the user written FieryICE code to Python3 code. To do the same, we generate an abstract syntax tree (AST) of the DSL code using the ANTLR tree walkers. The AST generator converts the nested arithmetic operations written by the user to multiple binary/unary operations, and links them using tem-



**Fig. 1** AST of simple expression



**Fig. 2** AST of expression involving trigonometric function



porary variables. For example, if the user code is  $d = (a + b)/c$ , the AST generator converts it to the tree shown in Fig. 1.

The root node in the AST represents the output variable of the user code. The leaf nodes of the constructed tree represent the learnable and input variables defined by the user. Every other node is the result of calling an operation on its children. Currently, there are two kinds of operations arithmetic and function calls. An arithmetic operation needs two nodes, while a function call is on a single node. AST for the expression  $c = \sin(a + b)$  is in Fig. 2.

### 4.3 Generation of Python3 User Code

After the AST is generated, we generate the equivalent Python3 code by performing a depth first search (DFS) [12] traversal of the AST and generating the equivalent code for every operation being performed. For the above AST, the generated Python3 code would be

---

```

_t0 = a
_t1 = b
_t2 = a + b
_t3 = c
_t4 = _t2 * _t3
d = _t4

```

---

#### 4.4 Support for Matrix and Vector Operations

FieryICE has a high level syntax which lets programmers to define arithmetic operations on matrices and vectors just like we do for scalars. To provide support for such kind of syntax, internally, we are converting every variable defined by the user to an object of the ‘variable’ class. The ‘variable’ class is used by the FieryICE compiler, standard arithmetic operations like ‘+’, ‘-’, ‘\*’, ‘/’ between objects of variable class have been overridden and the result of such operations is again an object of variable class. A user-defined matrix of dimension  $m \times n$  is converted to an object of the variable class having a 2D list of dimensions  $m \times n$ . A user-defined scalar is converted to an object of the variable class having a 2D list of dimension  $1 \times 1$ . This helps us to generate Python3 code for matrices the same way we generate code for scalars.

#### 4.5 Cost Function Generation

If the programmer defines a variable to be learnable, then we populate the generated variable object with symbolic variables of CASADI. When operations are performed between symbolic variables and a scalar/matrix, the return is also a symbolic variable. Because of this the predicted output which is obtained by running the generated Python3 code would be a symbolic variable. We use this predicted output to form the cost function, cost function is defined as the mean squared error between expected and predicted output. As the predicted output is a symbolic variable, even the cost will be a symbolic variable

$$\text{Cost} = \frac{1}{2m} \sum_{i=1}^m (y^i - \hat{y}^i)^2. \quad (1)$$

### 4.6 Cost Minimization

We use the widely used and efficient gradient descent technique to find values of the learnable variables such that the cost function is minimized. We tested out various variants of gradient descent and have noticed that there is no universal most optimal technique, and the optimal technique to be used depends upon the dataset used, so we have provided an interface which lets the programmer choose the variant of gradient descent he wants to use. To speed up batch, mini-batch, and stochastic gradient descent, we have used backtracking line search algorithm [13] to dynamically update the learning rate for faster convergence.

**Batch Gradient Descent** Batch gradient descent updates the parameters by computing the gradient of the cost function with respect to the learnable variables for the entire training dataset.

$$\theta = \theta - \eta * \nabla_{\theta} J(\theta) . \tag{2}$$

As it has to iterate the entire dataset to update, a single parameter batch gradient descent is very slow.

**Stochastic Gradient Descent** Stochastic gradient descent performs a parameter update for a training example.

$$\theta = \theta - \eta * \nabla_{\theta} J(\theta; x^{(i)}; y^{(i)}) . \tag{3}$$

Due to this, it is considerably faster than batch gradient descent. Since the update happens for each sample, the gradient transitions look noisy.

**Mini-batch Gradient Descent** Mini-Batch gradient descent is a merge of both stochastic and batch gradient descent, instead of iterating over the entire dataset, it iterates over batches of fixed size, and this way it is faster than batch gradient descent. As there is more than a single sample in the batch, the randomness is considerably reduced and it converges smoother than stochastic gradient descent.

$$\theta = \theta - \eta * \nabla_{\theta} J(\theta; x^{(i:i+n)}; y^{(i:i+n)}) . \tag{4}$$

**Adam Gradient Descent** Adam [14] is an optimization algorithm that can be to update the network weights iteratively based on training data. The method computes individual adaptive learning rates for different parameters from estimates of first and second moments of the gradients. Adaptive gradient algorithm (AdaGrad) that maintains a per-parameter learning rate that improves performance on problems with sparse gradients (e.g., natural language and computer vision problems). Root mean square propagation (RMSProp) that also maintains per-parameter learning rates that are adapted based on the average of recent magnitudes of the gradients for the weight (e.g., how quickly it is changing). This means the algorithm does well on online and non-stationary problems (e.g., noisy). The algorithm calculates an exponential moving average of the gradient and the squared gradient, and the parameters beta1 and

beta2 control the decay rates of these moving averages. Empirical results demonstrate that Adam works well in practice and compares favorably to other stochastic optimization methods.

It is also possible for the programmer to modify the generated Python3 code to use a custom cost function instead of the mean square error. This is possible as we are using the CASADI library to calculate the partial derivative of the cost function with respect to the learning parameter.

## 4.7 Analysis

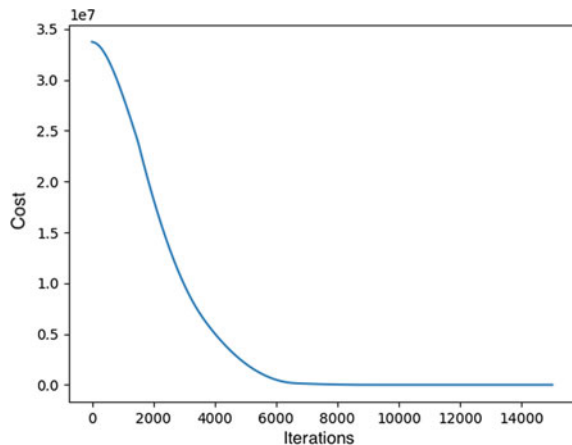
The generated Python3 code also creates plots to help the programmer understand the speed and the working of the algorithm. The following plots will be generated.

**Cost Versus Iteration Plot** The cost versus iteration plot (shown in Fig. 3) helps the programmer understand how the value of the cost function varies over iterations. It helps the programmer visualize whether the cost function is converging as expected or not. The latter occurs if the variant of gradient descent used is not compatible with the dataset provided. This would act as a prompt to the programmer to improve his training data or change the variant of gradient descent used.

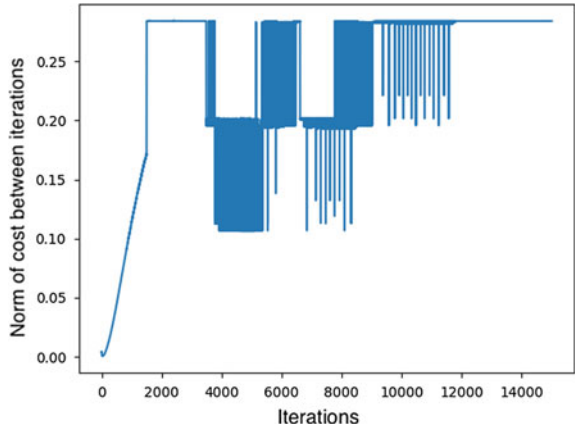
**Norm Versus Iteration Plot** For every learnable parameter defined by the programmer, a norm versus iteration plot (shown in Fig. 4) is generated. Norm of a variable at an iteration is defined as the absolute difference between the value of the variable between the previous iteration and the current iteration. This plot helps the programmer keep track of how much a learnable variable changes over iterations.

**Learnable Variable Value Versus Iteration Plot** For every learnable parameter defined by the programmer, a plot (shown in Figs. 5 and 6) is generated which contains

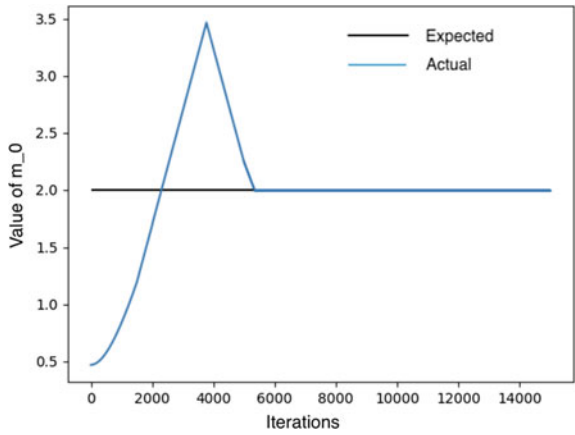
**Fig. 3** Cost function over iterations



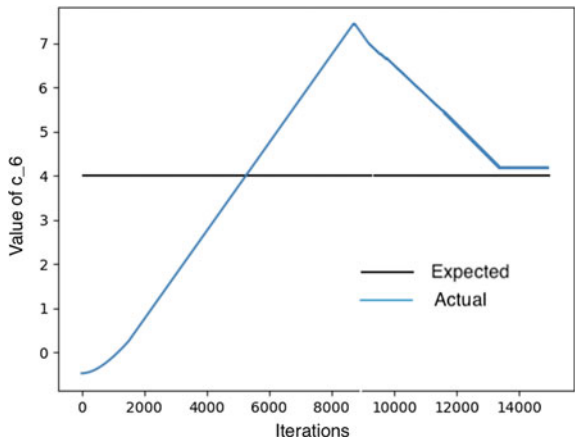
**Fig. 4** Norm of learnt parameter over iterations



**Fig. 5** Plot of expected value of a learnable parameter  $m_0$  versus actual learnt value



**Fig. 6** Expected versus actual value of  $c_6$



the value of the learnable variable over iterations. In the plots, the blue line represents the value of the learnable parameter over iterations and the black horizontal line represents the expected value. In Fig. 5, we can see that the final value learnable variable reaches the expected value, whereas in Fig. 6, it slightly digresses from the expected value, this is because gradient descent minimizes the total error which consists of error of all the individual learnable variables, but not that of a single variable. There could be several minimums of the error function, gradient descent finds one of such minimum.

## 5 Conclusion

Currently, existing popular libraries for ML often have a steep learning curve as the API they expose is not trivial and utilizing such an API often requires the user to have a complete understanding of the API and a fairly non-trivial grasp of the programming language being used to call the API, this causes the necessity for simplistic methods to model ML solutions. FieryICE is one such solution which alleviates the pain of developers. This paper shows the capabilities and inner workings of FieryICE, we have generated various plots through which we show the accuracy and efficiency of FieryICE. This paper also serves as a foundation to those who want to create their own DSL for ML purposes. FieryICE can be used by embedded developers due to the unavailability of powerful and expressive libraries in low-level programming languages, and it can also be used by those who are new to programming to learn and experiment with ML.

## References

1. P. Rekha, V.P. Rangan, M.V. Ramesh, K.V. Nibi, High yield groundnut agronomy: an IoT based precision farming framework, in *IEEE Global Humanitarian Technology Conference (GHTC)*, San Jose, CA 2017, 1–5 (2017). <https://doi.org/10.1109/GHTC.2017.8239287>
2. M.V. Ramesh et al., Water quality monitoring and waste management using IoT, in *2017 IEEE Global Humanitarian Technology Conference (GHTC)*, San Jose, CA, 2017, pp. 1–7. <https://doi.org/10.1109/GHTC.2017.8239311>
3. R. Ani, S. Krishna, N. Anju, M.S. Aslam, O.S. Deepa, IoT based patient monitoring and diagnostic prediction tool using ensemble classifier, in *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Udupi, 2017, pp. 1588–1593. <https://doi.org/10.1109/ICACCI.2017.8126068>
4. P.K. Sen, A rank-invariant method of linear and polynomial regression analysis. I, II, III. *Nederl. Akad. Wetensch. Proc.* **53**, pp. 386–392, 521–525, 1397–1412; P. Kumar, Estimates of the regression coefficient based on Kendall's tau. *J. Am. Stat. Assoc.* **63**(324), 1379–1389 (1968)
5. D.W. Hosmer, S. Lemeshow, *Applied Logistic Regression*, 2nd edn. (Wiley, Hoboken, 2000). ISBN 978-0-471-35632-5
6. C. Lemaréchal, Cauchy and the Gradient Method (PDF). *Doc Math Extra*, pp 251–254 (2012)

7. A.G. Baydin, B.A. Pearlmutter, A.A. Radul, J.M. Siskind, Automatic differentiation in machine learning. *J. Mach. Learn. Res.* **18**, 1–43 (2018)
8. J. Poroor, Work-in-progress: VerticalThings—a language-based microkernel for constrained IoT devicer, in *2018 International Conference on Embedded Software (EMSOFT)*, Turin, 2018, pp. 1–3. <https://doi.org/10.1109/EMSOFT.2018.8537193>
9. S. Diamond, S. Boyd, CVXPY: a python-embedded modeling language for convex optimization. *J. Mach. Learn. Res.* **171**, 2909–2913 (2016)
10. S. Ackermann, V. Jovanovic, T. Rompf, M. Odersky, Jet: an embedded DSL for high performance big data processing, in *International Workshop on End-to-end Management of Big Data (BigData 2012)*
11. T.J. Parr, R.W. Quong, ANTLR: a predicated-LL ( $k$ ) parser generator. *Softw.: Pract. Exper.* **25**, 789–810 (1995). <https://doi.org/10.1002/spe.4380250705>
12. R. Tarjan, Depth-first search and linear graph algorithms, in *12th Annual Symposium on Switching and Automata Theory (swat, East Lansing, MI, USA)*, 1971, pp. 114–121 (1971). <https://doi.org/10.1109/SWAT.1971.10>
13. N. Geoffrey, F. Pedregosa, A. Askari, M. Jaggi, Linearly convergent Frank-Wolfe with backtracking line-search (2020)
14. D.P. Kingma, J. Ba, Adam: a method for stochastic optimization, in *Published as a Conference Paper at the 3rd International Conference for Learning Representations*, San Diego (2015)
15. M. Khamis, W. Gomaa, W. Taha, Domain specific languages for machine learning (2016). <https://doi.org/10.13140/RG.2.1.1416.5361>

# Demand-Based Dynamic Slot Allocation for Effective Superframe Utilization in Wireless Body Area Network



A. Justin Gopinath and B. Nithya

**Abstract** Wireless body area network (WBAN) is an emerging technology for remotely monitoring the critically affected patients regularly, which is a utility platform for a medical pandemic like COVID-19. IEEE 802.15.6 medium access control (MAC) defines the communication standard to pillar the quality requirements of the sensor nodes. Most of the existing works are focused on optimizing the conventional MAC by adopting dynamic scheduled access and efficient contention scheme to utilize the superframe structure. However, utilizing the entire slots based on demand from different priority sensor nodes is a challenging task. To address this issue, an efficient time slot allocation method, namely the demand-based dynamic slot allocation (DDSA) algorithm, is proposed. DDSA computes sensor node priority based on the run-time parameters such as critical index, remaining energy, and delivery demand. The slot assignment is proportional to the priority order, and the critical index factor resolves slot conflict. This guarantees data priority preservation with fair allocation for critical and non-critical medical data. The simulation is carried out using the Castalia-OMNeT++ simulator, and the results are shown that the proposed DDSA algorithm outperforms priority-based MAC and the conventional method in terms of packet reception rate, energy efficiency, and latency.

**Keywords** IEEE 802.15.6 · Scheduled access · Dynamic slot allocation · Superframe utilization · Fair allocation

## 1 Introduction

Nowadays, people are frequently affected by chronic diseases due to their lifestyle and the surrounding environmental conditions. To control or fight against the disease, periodic health monitoring is compulsory. They have advised the frequent hospital

---

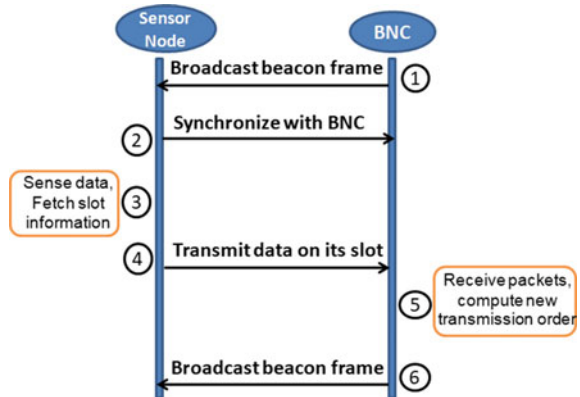
A. Justin Gopinath (✉) · B. Nithya  
Department of Computer Science and Engineering, National Institute of Technology,  
Tiruchirappalli 620015, Tamilnadu, India  
e-mail: [406116005@nitt.edu](mailto:406116005@nitt.edu)

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2021  
S. M. Thampi et al. (eds.), *Advances in Computing and Network Communications*,  
Lecture Notes in Electrical Engineering 735,  
[https://doi.org/10.1007/978-981-33-6977-1\\_28](https://doi.org/10.1007/978-981-33-6977-1_28)

361



**Fig. 1** Communication process between sensor and BNC



visits for check-up and interaction with medical officers. However, it may not be possible for aged or disabled people. In these circumstances, WBAN plays a vital role in monitoring the patients remotely by the doctors. In WBAN, the sensors are placed on the human body to collect the medical information. This patient's critical data is transmitted to the remote server, which is located in the hospital. After analyzing the received information by the doctor, the response is communicated back to the patient based upon the emergency of the patient. To achieve these facilities, the data transmission in WBAN must be prioritized, uninterrupted, and congestion-free [1, 2].

The data flow in WBAN has two modes: Firstly, the sensor nodes that are placed on the human body are connected to the body network coordinator (BNC), which forms a star network (intra-WBAN). Secondly, BNC transmits the received information to the remote server through the Internet (inter-WBAN). In intra-WBAN communication, the critical data to be transmitted to BNC within the stipulated time since sensor nodes collect both critical and non-critical data. In inter-WBAN, each BNC has to ensure the delivery of critical data to the hospital [3].

The communication between the sensor and BNC is illustrated in Fig. 1 in detail. Initially, BNC broadcasts the beacon frame, which includes BAN ID, sync information, transmission schedule, etc., to the sensor nodes. Upon receiving this information, each sensor synchronizes with BNC. According to the schedule time slot, the sensor wakes up and transmits the sensed data to the BNC. Along with the sampled data, sensors send the priority requirement to the BNC. After receiving additional control information, BNC computes a new transmission schedule and broadcasts through the beacon frame [4].

IEEE task force introduced the IEEE 802.15.6 standard for enabling reliable communication in WBAN. IEEE 802.15.6 protocol standard adopts three variations of communication mode, which are beacon frame following superframe structure, superframe without beacon, and non-beacon without superframe mode. Each beacon interval is divided into several slots called superframe. IEEE 802.15.6 protocol adopts contention-based access, slotted Aloha method, or scheduled access methods for

accessing the channel [5]. The conventional method of scheduled access adopts fixed slot allocation, which does not meet the Quality of Service (QoS) requirement of critical medical application. Also, the fixed allocation underutilizes the time slots when there are no data to be transmitted. Therefore, many research works [6, 7] are focused on dynamic slot allocation methods, where priorities of the sensor nodes are taken into consideration while allocating the slots. However, node's priority alone is not sufficient to decide on transmission order, and therefore, a set of works on priority calculation methods are developed based on different real-time information such as remaining energy, delivery ratio, and also the importance of the sensed data. However, there is a research scope in improving the dynamic slot allocation without adding additional overheads.

The challenges of dynamic slot allocation method are (i) priority calculation of the sensor nodes (ii) utilization of entire slots (iii) fair slot allocation and (iv) time slot conflicts. To address the above challenges, this paper proposes an efficient time slot allocation method, namely demand-based dynamic slot allocation (DDSA) algorithm, which preserves the data priority with fair allocation for critical and non-critical medical data.

The contribution of the paper is the following:

1. The proposed DDSA algorithm computes the node's priority with three real-time metrics: critical index, remaining energy, and demand delivery, which exhibit the network status. Based on the computed priority, the proposed algorithm assigns the slots to the sensor nodes. The slot conflicts are resolved by reassessing the critical index value.
2. The proposed DDSA algorithm ensures the fair allocation by dividing the exclusive access to both critical and non-critical data transfer without compromising the QoS of critical applications.
3. The simulation is carried out using Castalia with OMNeT++. The simulation results are compared with the conventional 802.15.6 MAC and priority-based MAC by packet reception rate, energy efficiency, and packet latency.

The remaining of the paper is arranged as follows: Sect. 2 gives the existing works on dynamic slot allocation. The proposed DDSA algorithm is discussed in Sect. 3. The simulation setup and the results are analyzed in Sect. 4. Finally, Sect. 5 concludes the paper.

## 2 Related Works

This section discusses some of the recent amendments on dynamic slot allocation strategies to transmit time critical data to the coordinator using different strategies.

Alam et al. [8] have proposed a throughput and channel aware (TCA) method for scheduled channel access in WBAN applications. In the first phase, TCA selects potential candidate links based on the packet error rate. The second phase assigns the slot concerning node priority and data rate. The coordinator node maintains the status

register to keep track of the requirements of data to be transmitted. The experiment is carried out with different mobility patterns such as sitting, walking, and running to obtain different path loss values and recommends the transceiver with a better operable transmission power level. However, this algorithm assigns the slot to the node if it has data to be transmitted without considering the criticality of the data.

The scheduled-based slot allocation and power control method is proposed by Wang et al. [9] to optimize the energy efficiency of sensor nodes. They modeled the optimization problem using the Markov process and used reinforcement learning to find the optimal strategy in which the slot allocation is performed based on path loss and data quantity. Each slot in the superframe structure is further divided into decision and transmission phases. The coordinator node selects a sensor to access the channel during the decision phase and adjusts its transmission power. The data is transmitted during the transmission phase. This algorithm is attained more than 0.9 fairness index with maximal energy efficiency.

Priority-based time slot allocation (PAT) method is proposed in [10] for medical emergency applications. PAT is a game theory approach that formulates a fitness function of data criticality and energy dissipation factor of each local processing unit (LPU). It uses a hawk-dove game to apply its strategy for the fitness function. The LPU with higher fitness is benefited with higher preferences. This strategy is effectively utilized by the LPU to transmit data for the inter-WBAN framework. The formulation of fitness function has a challenge in collecting health data from the different sensor to prioritize the node criticality.

Saboor et al. [11] have proposed a CSMA/CA-based dynamic slot allocation scheme that uses a non-overlapping back-off method instead of binary exponential backoff (BEB) to reduce the packet collisions. The slot size is computed by the coordinator for each traffic category with the information such as payload, priority, and data rate. The simulated results reveal that this algorithm increases the superframe utilization by 50% than the conventional method. Sangeetha et al. [12] proposed a fuzzy approach for dynamic slot allocation where the fuzzy-controller decision is based on the energy consumption of the node, waiting for packets in buffer and arrival rate.

Sun et al. [13] have proposed a priority-based medium access control scheme (PMAC) with node priority design by considering the importance of data, sampling frequency, remaining energy, and timeout condition for WBAN. It maximizes the overall utility of all nodes by adjusting transmission time and order of transmission. It is achieved by dividing the set of nodes into GOOD and BAD nodes based on data delivery of each node. Also, the algorithm uses overtime factor to increase the priority of node with buffered data and allows to transmit the packet within the next superframe.

From the existing works, it is inferred that the dynamic slot allocation improves the performance instead of fixed slot allocation. However, every proposal applies its priority computation for assigning the slots, and they did not address the fair slot allocation based on the run-time demand for the efficient utilization of the entire superframe. Further, very few approaches have considered the heterogeneous traffic within the sensor node and also the fair allocation slots for all the sensor nodes in

the network. To alleviate these limitations, this paper proposes a novel algorithm called demand-based dynamic slot allocation (DDSA) algorithm, which effectively maximizes the utilization of entire slots by all the active sensor nodes.

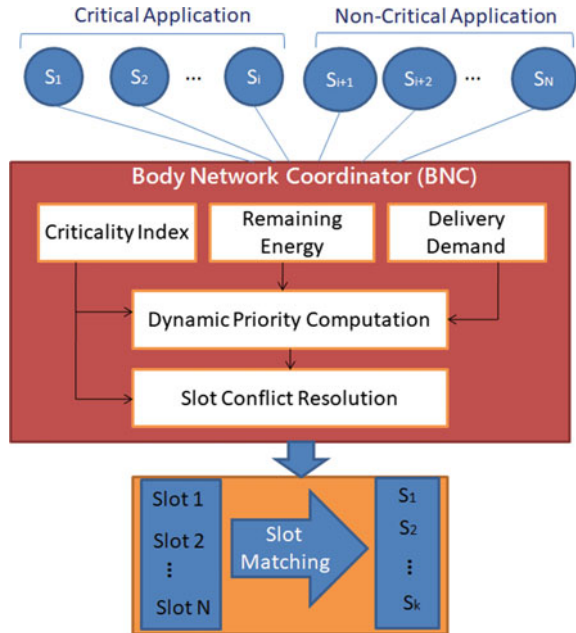
To achieve this, the proposed algorithm divides the set of sensor nodes into a critical and non-critical category based on their sensing data. Then, the coordinator node performs the priority calculation based on real-time dynamic parameters such as remaining energy, demand requirement, and importance of data. The total interval is divided based on the requirement of emergency and non-critical messages. First, the critical messages are served, and then the remaining interval is allocated to non-critical messages. Thus, the data delivery of non-critical applications is ensured through fair allocation and also increases the utilization of the entire superframe. The next section elaborates on the proposed DDSA algorithm.

### 3 Proposed Demand-Based Dynamic Slot Allocation (DDSA) Algorithm

The primary objective of the proposed DDSA algorithm is to dynamically adjust the slot assignment based on the real-time sensor node information and to ensure fair allocation that means all the nodes are getting a fair chance to transmit their data. The fixed allocation does not encounter the network status, and the bandwidth is wasted due to the unavailability of data during the allotted interval. To curtail these issues, the dynamic slot allocation is proposed, which ensures the node priority constraints and manages the complete interval by effectively distributing the slots to all sensor nodes. The proposed DDSA framework is depicted in Fig. 2.

Initially, the BNC divides the set of sensors into two subsets, such as critical application (heart or brain monitoring) and non-critical application (muscle tissue monitoring), which is based on its importance of sensing data. Each sensor is generating different priority data (heterogeneous data). After classifying these applications, BNC in the proposed DDSA algorithm performs three-phase scrutinizing for the dynamic slot allocation. In the first phase, BNC gathers three run-time metrics, which exactly mimic the current network status. First, the critical index metric represents the significant of the current data. The sensor node assigns the different priority values to indicate the criticality of the current data. If the packet has a higher priority, then it is assumed as emergency data by the BNC. The second metric is the remaining energy of the node, which checks whether the node has sufficient energy to transmit the data. Finally, the delivery demand is considered, which reveals the quantity of data to be transmitted. If the sensor node has buffered data, then it can demand more than one slot to BNC. In this way, each metric contributes to exhibiting the current network status to BNC. Unlike the existing works, the proposed DDSA algorithm considers not only the primary metrics such as the critical index and the remaining energy of the nodes, but also, it utilizes delivery demand to estimate the total number of available slots and preserve the priority requirements.

**Fig. 2** Proposed DDSA framework



Based on these received values, the dynamic priority computation phase estimates the new priority for the corresponding sensor node. For each received value, BNC assigns the priority level and accumulates into a single value, which represents the priority of the node. BNC sort the sensor nodes within critical and non-critical subsets based on the obtained values from larger value to smaller values. From the second phase, the same priority may be assigned to more than one sensor node. This conflict is resolved by the third phase of the proposed DDSA algorithm. The slot conflict resolution phase checks conflict among sensors belonging to the same set, and then, it is resolved by the critical index of those nodes. This decision metric demands immediate channel access based on the criticality of the sensed data. For example, if sensors  $S_1$  and  $S_2$  have the same priority value as 0.9, the conflict is resolved by critical index value ranging from 0 to 7. Suppose the critical index value of  $S_1$  and  $S_2$  are 5, 7, respectively. The slot is allotted to  $S_2$ , followed by  $S_1$ . Therefore, BNC performs better slot matching between available slots (N) and the set of sensor nodes ( $S_k$ ). It is observed that not all the nodes may be serviced within the current superframe because only the nodes that have data is assigned the slots. Finally, BNC updates its transmission order and broadcasts to sensor nodes.

After receiving this information, each sensor starts transmitting its data in the allotted slots without any conflicts, thus increasing the effective utilization of superframe structure. As the proposed, DDSA algorithm increases or decreases the total slot length based on the requirement of a critical and non-critical set of sensor nodes; it is guaranteed that the nodes are getting the fair chance to transmit their data. The proposed algorithm is explained in detail in Algorithm 1.

**Algorithm 1** Proposed DDSA Algorithm

---

**Input:** Total number of slots ( $N_{slots}$ ), Critical Index ( $Node_{CI}$ ), Remaining Energy ( $E_{rem}$ ), Delivery Demand ( $Node_{del}$ )

**Output:** Updated Transmission Schedule

**begin**

Step 1: BNC classifies sensor nodes based on the type of sensing data.

Step 2: for each sensor node do  
      $Send(Node_{CI}, E_{rem}, Node_{del})$   
 end for

Step 3: for each beacon interval do  
 Step 3.1: collects the real-time information from all sensors  
 Step 3.2: computes the number of slots required for Critical(C) and Non-critical(NC) sensor nodes  
 $Slot_{req} = Node_{del}^C + N_{del}^{NC}$   
 if ( $Slot_{req} > N_{slots}$ ) then  
 $N_{del}^{NC} = N_{slots} - N_{del}^C$   
 endif  
 Step 3.3: computes new priority based on the received information  
 $Node_{pri} = computePriority(Node_{CI}, E_{rem}, Node_{del})$   
 Sort the nodes as per  $Node_{pri}$  from large to small  
 Step 3.4: slot conflict resolution  
 if (two or more nodes have equal  $Node_{pri}$ ) then  
 $ConflictResolve(Node_{CI})$   
 endif  
 Step 3.5:  $SendBeacon()$  with the updated transmission schedule

Step 4: repeat steps 2 & 3.

**end**

---

In Step 1, BNC classifies the set of sensor nodes into low- and high-priority sensor nodes respect to their nature of sensing data. Each sensor node periodically sends its information such as remaining energy, data priority, and transmission demand to the BNC. Upon having this information, BNC computes the new priority for each sensor node, as shown in Step 3. Further, it computes the total required slot length by the critical and non-critical sensor nodes. Based on this, BNC divides the beacon interval into two phases. The first phase is exclusive access for sensor nodes with high priority and the remaining for sensor nodes with low priority. This ensures that within the superframe structure the non-critical data is also being transmitted. As discussed in the proposed algorithm, the  $computePriority()$  method computes new priority value as it evaluates individual metric values. After finding a new priority, if there is a slot conflict among nodes, then it is adjusted by the critical index factor using the  $ConflictResolve()$  method because the critical data to be transmitted within the stipulated time. A node with a higher critical index value gets the slot first than the other. Finally, BNC shares the updated slot assignment to all sensor nodes by broadcasting the beacon frame. As a result, all the nodes will access the channel only during their allotted slots, thereby granting exclusive access to critical and non-critical data without compromising the priority of data.

## 4 Simulation Results and Discussion

The WBAN environment is simulated in the Castalia network simulator on the top of OMNeT++. The proposed DDSA algorithm is evaluated and compared with the conventional protocol (802.15.6 MAC beacon-enabled mode) and PMAC [13]. The following subsection discusses the simulation setup and simulation results with packet reception rate, energy consumption, and latency.

### 4.1 Simulation Setup

The simulation of the proposed DDSA algorithm is carried out by varying  $N$  sensor nodes as 5, 10, 15, 20, 25 to reflect the scalability of the proposed DDSA algorithm. In each case, one node is a BNC, and remaining are sensing nodes connected to the BNC, as shown in Fig. 3. The simulation classifies a set of sensor nodes as high-priority (urgent, emergency) and low-priority (normal, alert) sensor nodes. The data generation rate of high-priority sensor nodes ranges from 25 to 72 Kbps and for low-priority sensor nodes ranges from 7 to 25 Kbps [14]. Each data packet has different user priority levels from 0 to 7 for measuring the criticality index. The physical and MAC layer parameters are configured as per the standard [13, 15]. The sensor nodes and BNC are placed, as shown in Fig. 3, and the 2.4 GHz band is used for communication. The initial energy of the nodes is 18,720 J, and the transmit power is  $-15$  dBm. The simulation is carried out for 300 s, repeated five times, and the averaged results are depicted in the following subsection.

### 4.2 Simulation Results and Discussion

In this section, the effectiveness of the proposed DDSA algorithm is evaluated with the key performance indicators such as packet reception rate, energy consumption rate, and latency. The results are obtained by varying the number of sensor nodes and superframe length, to realize the performance of proposed DDSA and other related algorithms. The behavior of the proposed DDSA algorithm is further analyzed with respect to high- and low-priority sensor nodes classification.

**Packet Reception Rate (PRR)** PRR is the ratio of the sum of received packets by the BNC to the sum of allocated slots during the superframe. Figure 4a shows the packet reception rate of the proposed DDSA algorithm along with PMAC and conventional method. As the number of sensor nodes increases, the PRR value also increases because the number of data packets is increased. However, the PRR value of 802.15.6 MAC lies between 40 and 55% with varying the number of sensor nodes because the data packets are dropped when it reaches the maximum number of tries. Also, it does not differentiate the different priorities of sensor nodes based on the criticality of

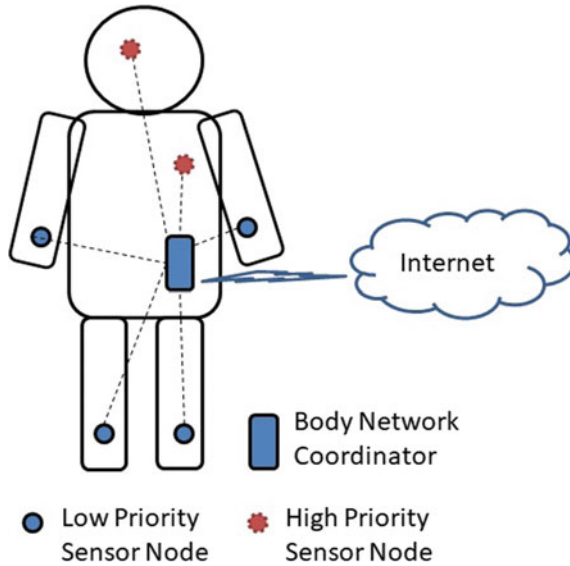


Fig. 3 Simulation setup

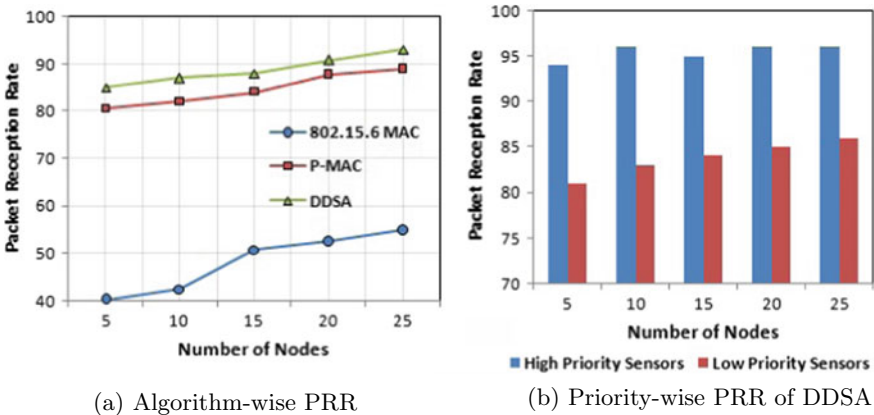


Fig. 4 Packet reception rate

the data. However, the proposed DDSA algorithm maintains the average PRR value above 85%. This is possible due to the effective slot assignment based on the delivery demands of each sensor node. Figure 4b shows the PRR value of the proposed DDSA algorithm obtained by individual nodes. The proposed DDSA algorithm guarantees the data transfer of low priority traffic by balancing the superframe length for high-priority and low-priority traffic based on the demands. Figure 4b is witnessing the fact that the proposed DDSA algorithm performs fair slot allocation for the requested demand.



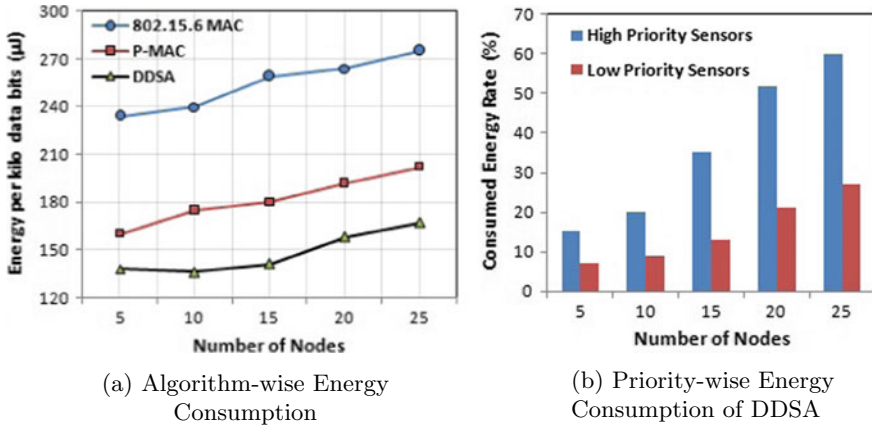
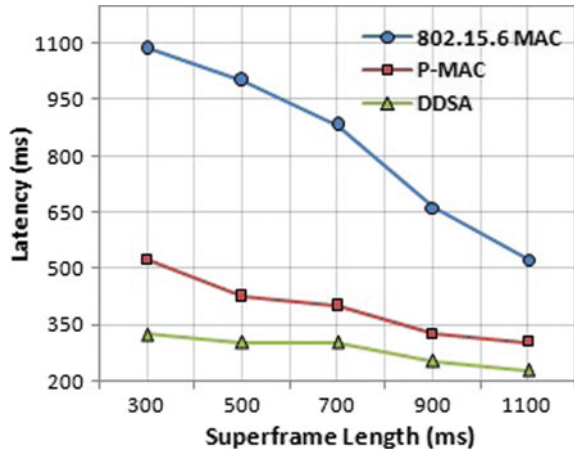


Fig. 5 Energy efficiency

**Energy Efficiency** This section demonstrates the energy consumption of the sensor nodes, measured in terms of rate of energy consumption for the successful transmission of kilobits of data. Figure 5a shows the consumed energy for data transmission by all sensor nodes for all three algorithms. The conventional method carried out scheduled access, followed by the contention access, which leads to the dropping of data due to contention and retransmits the data again. This is a cause for more energy consumption by the 802.15.6 MAC scheme. On the other side, the schedule access-based PMAC and the proposed DDSA algorithm reduces the energy consumption due to dynamic allocation of slots by considering the criticality of data and avoids idle listening of sensor nodes. Moreover, the proposed algorithm further reduces the energy consumption than PMAC by allocating more than one slot based on the run-time demand of the sensors. Figure 5b shows the rate of energy consumption for high- and low-traffic sensor nodes. In all cases, the high-priority sensor nodes consumed more energy than the low-priority sensor nodes because of the higher rate of packet delivery by the high-priority sensor nodes.

**Latency** The latency of the data packet is the time interval between the generation of data frame at the sensor and the arrival of data frame at the BNC. Figure 6 shows the latency of data packets experienced by all sensor nodes during different superframe length. From Fig. 6, it is interpreted that the latency is decreased as the superframe length is increased. This is because the sensor node gets more slots for data transmission. The packet latency of 802.15.6 MAC has a significant increase due to the retransmission of packets in the consecutive superframe. However, the proposed DDSA algorithm reduces the overall latency by 10% than PMAC because DDSA achieves better packet delivery, as shown in Fig. 4a, due to the effective dynamic slot allocation scheme.

Fig. 6 Packet latency



## 5 Conclusion

The proposed DDSA algorithm effectively assigns the time slot based on the node's priority, computed by the criticality index, remaining energy, and delivery demand. To facilitate the fair allocation to critical and non-critical data transmission, the entire superframe length is divided based on the demands of high- and low-priority sensor nodes. The slot conflict is resolved by the criticality of the data, thus reducing packet collisions. The performance of the proposed DDSA algorithm is evaluated by simulation results using the Castalia simulator in terms of packet reception rate, energy consumption, and latency. It is inferred from the simulation results that the proposed DDSA maintains more than 85% delivery rate for the varying number of sensor nodes compared with P-MAC and IEEE 802.15.6. Moreover, the proposed DDSA algorithm effectively minimizes latency due to the effective dynamic slot allocation.

In the future, the performance of the proposed DDSA algorithm can be analyzed with different mobility patterns to adopt different real-time environments, and it can be enhanced for inter-WBAN communication.

## References

1. B. Latré, B. Braem, I. Moerman, C. Blondia, P. Demeester, A survey on wireless body area networks. *Wireless Networks* **17**(1), 1–18 (2011)
2. S.H. Cheng, C.Y. Huang, Coloring-based inter-WBAN scheduling for mobile wireless body area networks. *IEEE Trans. Parallel Distrib. Syst.* **24**(2), 250–259 (2012)
3. L.A. Ferhi, K. Sethom, F. Choubani, Energy efficiency optimization for wireless body area networks under 802.15.6 standard. *Wireless Personal Commun.* **109**(3), 1769–1779 (2019)
4. T. Bai, J. Lin, G. Li, H. Wang, P. Ran, Z. Li, G. Jeon, An optimized protocol for QoS and energy efficiency on wireless body area networks. *Peer-to-Peer Network. Appl.* **12**(2), 326–336 (2019)

5. F. Ullah, A.H. Abdullah, O. Kaiwartya, S. Kumar, M.M. Arshad, medium access control (MAC) for wireless body area network (WBAN): superframe structure, multiple access technique, taxonomy, and challenges. *Hum.-Centric Comput. Inform. Sci.* **7**(1), 34 (2017)
6. A. Alsiddiky, W. Awwad, H. Fouad, A.S. Hassanein, A.M. Soliman, Priority-based data transmission using selective decision modes in wearable sensor based healthcare applications. *Comput. Commun.* (2020)
7. R. Khan, M.M. Alam, T. Paso, J. Haapola, Throughput and channel aware mac scheduling for Smartban standard. *IEEE Access* **7**, 63133–63145 (2019)
8. M.M. Alam, D.B. Arbia, E.B. Hamida, Joint throughput and channel aware (TCA) dynamic scheduling algorithm for emerging wearable applications, in *2016 IEEE Wireless Communications and Networking Conference* (IEEE, New York, 2016, April), pp. 1–6
9. L. Wang, G. Zhang, J. Li, G. Lin, Joint optimization of power control and time slot allocation for wireless body area networks via deep reinforcement learning. *Wireless Networks* pp. 1–10 (2020)
10. S. Misra, S. Sarkar, Priority-based time-slot allocation in wireless body area networks during medical emergency situations: an evolutionary game-theoretic perspective. *IEEE J. Biomed. Health Inform.* **19**(2), 541–548 (2014)
11. A. Saboor, R. Ahmad, W. Ahmed, A.K. Kiani, M.M. Alam, A. Kuusik, Y. Le Moullec, Dynamic slot allocation using non overlapping backoff algorithm in IEEE 802.15. 6 WBAN. *IEEE Sens. J.* (2020)
12. S. Pushpan, B. Velusamy, Fuzzy-based dynamic time slot allocation for wireless body area networks. *Sensors* **19**(9), 2112 (2019)
13. G. Sun, K. Wang, H. Yu, X. Du, M. Guizani, Priority-based medium access control for wireless body area networks with high-performance design. *IEEE Internet Things J.* **6**(3), 5363–5375 (2019)
14. B. Liu, Z. Yan, C.W. Chen, Medium access control for wireless body area networks with QoS provisioning and energy efficient design. *IEEE Trans. Mob. Comput.* **16**(2), 422–434 (2016)
15. S. Sarkar, S. Misra, B. Bandyopadhyay, C. Chakraborty, M.S. Obaidat, Performance analysis of IEEE 802.15. 6 MAC protocol under non-ideal channel conditions and saturated traffic regime. *IEEE Trans. Comput.* **64**(10), 2912–2925 (2015)

# A Survey on Congestion Control Algorithms of Wireless Body Area Network



Vamsikiran Mekathoti and B. Nithya

**Abstract** Nowadays, research on wireless body area network (WBAN) touches its extremity as the need arising more for the present mundane lifestyle of the world. Exclusive demand for WBAN is mainly due to its special properties such as its mobility, tiny size, and network topology, etc. WBAN is a specialized technology designed to monitor a remote patient (or a subject—as WBAN is not limited to human being), and it grabs attention from researchers as it is emergency-aware. Due to the nature of the WBN, the collisions among data packets are inevitable which in turn increases congestion in the network by triggering more number of retransmissions. To eradicate these issues, several congestion control (CC) algorithms are proposed in the literature. This paper surveys some of the recent CC algorithms and stretches a detailed comparative study of these algorithms. This survey reveals the strength and weakness of these algorithms and the future research direction in this research field.

## 1 Introduction

Wireless body area network (WBAN) is a logical sub-thought of wireless sensor network (WSN), designed to operate autonomously to connect various biosensors, located inside, along with or on the body (but not limited to humans) [1]. Nowadays, WBAN grabs the attention of the entire world where major science-developed countries are piled up with this research work and IEEE recognized WBAN with standard number 802.15.6.

WBAN majorly supports healthcare applications, as coming to healthcare, WBAN turns into emergency-aware technology as it must behold a remote patient's or subject's condition. The motivation behind WBAN invention is to support a tiny sensor network. This tiny WBAN can be connected through a gateway and make use

---

V. Mekathoti (✉) · B. Nithya  
NIT, Trichy, Trichy, Tamil Nadu 620015, India  
e-mail: [gcseme@gmail.com](mailto:gcseme@gmail.com)

B. Nithya  
e-mail: [bnithyanitt@gmail.com](mailto:bnithyanitt@gmail.com)

of the Internet to reach the destination server (DS). This DS is possibly a hospital server or a military base server or a sports concert server. On receiving a signal from the subject, DS takes a necessary action such as reply on the same communication to the subject or making an alert through cellular phone or a nearby assistant may be informed to serve the subject.

To support WBAN technology, IEEE 802.15.6 standard defines protocols and layers with some limitations in their design. These limitations are mainly because of the following reasons: (i) The range of WBAN is tiny, (ii) connectivity support is wireless, and (iii) protocol design is subjected to miniature in all the parameters considered. Typically, WBAN layered architecture [2] is nested with physical layer (PHY), medium access control layer (MAC), network layer (NW), transport layer (TR), and application layer (AL). This survey focuses on the congestion control (CC) aspect of WBAN's transport layer. Congestion occurs in the network when a node in a network or a channel carries more data than its bearable capacity. Consequently, it provokes more queuing delays, packets loss, frequent retransmissions, and blocking of new connections. As a result, the overall performance of the WBAN severely degrades, thus leading to reduced quality of service (QoS). Since the healthcare is a dominant application of WBAN and also it is emergency aware, the reasonable QoS must be assured. But, congestion severely affects data transmission in WBAN, and it affects the QoS. To cope with these issues, massive research [12–21] is going on to control the congestion in the network. Still, there is no perfect solution to eliminate or control the congestion. Because, while controlling the congestion, most of the CC algorithms [12–21] neglected certain relevant features of WBAN such as bandwidth, sensor devices with minimal buffer, priority of the nodes, hotspots which affect the human tissues, and many-to-one topology, which make the situation very worsen. For example, on the occurrence of packet loss, retransmission is initiated to send the missing packets. Without knowing the degree of the congestion in the network, these retransmissions put extra load on the network and worsen the QoS of WBAN.

There are numerous surveys [2] and [4–9] are made about the architecture, protocol standards, applications, challenges, and recent trends in WBAN. But, surveys that are specific to CC in WBAN are still penury. This survey gives an ample information pertaining to congestion control in WBAN. This paper further discusses some of the research works made on CC in the recent literature and highlights their strength and weakness. For the better understanding, some of the general terminologies in WBAN are discussed for the better understanding of the proposed survey.

- *WBAN layered architecture*: It is a three-layered architecture. The inner core layer of the architecture forms an intra-body area network (Intra-BAN) with several interconnected bio sensor nodes (BSN). Here, BAN can communicate with outer layers through a coordinator node (CN). Second layer is a combination of one or more BANs, which forms inter-body area network (Inter-BAN) communication. Here, the communication can be made using CNs. These are connected to Internet through a gateway and travels to a specific DS, which can be mentioned as 'Beyond

BAN Communication' (BBC). This appears as the outer layer in the hawk-eye view architecture of WBAN.

- *Bio-Sensor Node (BSN)*: BSN is the most important element of WBAN. As the name indicates, it senses the data from the human body (however, WBAN is not limited to humans) from various parts such as brain, heart, blood, etc. These sensors collect the various data, like blood pressure (BP), electro myograph (EMG), electro cardiogram (ECG), temperature (T), etc.
- *Sink*: The sensed data from BSNs reaches the destination node called sink. Sink node has a buffer to hold data and have the property of taking input from several sensor nodes and forwards the data to the coordinator node.
- *Coordinator Node (CN)*: Coordinator node may be a personal digital assistant (PDA) or a Wi-Fi router or a smartphone. This CN has GUI capability to show the reports or statistics at the patient's end. Also, CN is connected to gateway to forward the information into the Internet to reach DS.
- *Gateway*: The data that are accumulated at the CN must be available to the external world through a gateway and reaches the Internet. Here, the portion of the Internet is to carry the vital information of a patient to destination server (DS).
- *Destination Server (DS)*: DS is digital device with the sufficient storage capability, a display monitor to show the incoming data, and a predefined program or software to analyze packets coming from a remote patient. DS may be a hospital server, a caretaker's smart device to watch his patient's condition, or possibly a cloud server to store the history of patient's data.
- *Buffer*: It is a temporary memory to store the information in the form of packets. In the context of WBAN, every node will be availed with some storage capacity. Generally, a sink node needs to have more buffer size than a BSN buffer size.
- *Communication Channel*: WBAN uses either radio frequency (RF) or non-RF-based communication techniques. Human body communication (HBC) is a non-RF technique, where the communication channel is the human body. In HBC, a BSN senses the data and gives it to an electrode, electrode again inputs into the human body, then the human body outputs the necessary information to another specific electrode to reach another BSN.
- *Temperature or hotspot*: Elements such as nodes or communication channel of a BAN gets heated when a node is retransmitting duplicate packets due to packet losses. Retransmitting duplicate packets along with current packets, need more amount of energy to be utilized in a limited time, may cause hotspots in the network.
- *Priority*: Priority in WBAN is classified into two categories, one is the priority of a node and the priority of sensed data. However, these two categories are based on the data a BSN is sensing. Priority varies with a patient to a patient. For example, a heart patient needs more attention towards ECG data or heartbeat, in this case, data (or ECG data) being sensed, attains the highest priority than other data, such as temperature (T), electromyography (EMG), etc.

Rest of the paper is organized as follows: Section 2 presents the existing surveys related to WBAN and recent CC algorithms, also it discusses the survey gap in the

design of CC in WBAN. Section 3 discusses the existing congestion control algorithms from the literature. Section 4 portrays tabular representation of comparative study of CC algorithms. Finally, the paper is concluded in Sect. 5.

## 2 Related Work

There are ample surveys available on WBAN in general. A survey given in [4], presents recent trends in WBAN. This survey displays a diversified trend ongoing in WBAN such as flexible antenna, dual-band printed antenna, MAC protocols, multi-hop protocols, energy constraint network, Zig-bee technology, PHY and MAC layers, AODV routing protocol, ultrasound sensor, transport layer, human body communication, etc. Also study proposed in [4], focuses on one of the challenges faced by WBAN, i.e., congestion. It highlights various existing congestion control strategies of transport layer protocols, over heterogeneous communication systems to ensure the quality of service of WBAN. It investigates dual band printed dipole antenna for 2.4/5.2 GHz for effective transmission of data, MAC protocols, priority of node data, multi-hop WBAN construction, i.e., clustered topology setup, mobility support, transmission efficiency improvement, whereas existing schemes work on 1-hop-based star network, and this survey also discusses various WBAN challenges such as energy optimization, security, etc.

A survey proposed in [5] displays the security challenges in WBAN as it is a favorite domain for the attackers. Because of tiny architecture and limited buffer capacity, algorithms made for WBAN are dumbfounded in security issues. This paper discusses various challenges such as energy, security, mobility, QoS, cooperation between nodes, but it mainly focuses on one of the security threats, i.e., denial of service (DoS). This survey proposes a futuristic solution, i.e., node cooperation to avoid DoS attack.

Survey presented in [6] discusses the communication technologies currently addressing in the literature for healthcare monitoring (HCM) using WBAN. This survey addresses the critical challenges existing in present HCM system. It also focus on energy issues as the WBAN sensor devices are power constrained and will be drained fast, because the present communications systems available such as ZigBee, Bluetooth, and 6LoWPAN consume more energy. [6] Investigates low power wide area network (LPWAN) communication systems such as Sigfox narrowband technology, long range (LoRa), narrowband Internet of Things (NB-IoT), long term evolution (4G), and category M1 (LTE-M). These LPWAN communication systems focuses on the WBAN QoS attributes and consumes low energy for the reliable communication in WBAN.

Reliability and quality of service challenges of WBAN are surveyed in [7]. This survey says the reliability, and QoS is based on network infrastructure model and effective link management. In present scenario, WBAN facing challenges in data privacy, transmission channels, energy efficiency, specific absorption, and faults. In security dimension, data privacy as one of the QoS parameter, facing challenges due

to WBAN sensor devices, and the tiny BAN architecture can be easily tampered by the attackers as it cannot facilitate firewalls due to lower buffer capacities. Also [7] focuses on the issues like low bandwidth transmission channel, energy consumption challenges like rapid battery draining due to number of retransmissions, and the WBAN node relays will produce heat as the energy dissipated in relays can be observed by body tissues.

Survey [8] gives a broader idea of WBAN algorithms, also it slightly indexes security issues, power consumptions, hotspot due to radiation, and general basic issues of WBAN standard. It mentions the importance of various algorithms supporting in various aspects, such as ultra wide band (UWB), ISM bands inheriting from IEEE 802.15.4, MAC layer importance in channel allocation, MAC super frame structure, Bluetooth, topology proposed by IEEE TaskGroup6 (TG6), etc. This survey still elaborates the understanding of UWB and ISM bands. UWB is a communication technology service for the WBAN and has the advantage of high rate of transmission but fades out its advantage in energy utilization. ISM bands works at 2.45 GHz comprising of 16 channels with the bit rate of 250 kbps.

A survey proposed in [2] motivates to survey on CC in WBAN. It gives basic knowledge over topology, security challenges, and communication technologies related to WBAN, such as industrial scientific medical (ISM) band with frequency 2.4 GHz, ZigBee, and Bluetooth. It mainly shows a comparative study specific to congestion control in WBAN of different transport layer algorithms. [2] Presents features, architecture, and some standard algorithms such as congestion detection and avoidance (CODA), event to sink reliable transport (ESRT), flush transport protocol (FTP), etc. Comparison table presents the aspects such as congestion detection, congestion notification type, direction of flow (Sink to sensor nodes—downstream, Sensor to sink node—Upstream), congestion avoidance type, end-to-end or hop-by-hop transmission, whether additive increase and multiplicative decrease (AIMD) used in the various transport layer algorithms to control the congestion in WBAN. Apart from [2], this paper proposes more parametric information used by the various existing algorithms and advantages, disadvantages, method of approach to eradicate the congestion in WBAN from the literature of CC in WBAN.

Survey presented in [9] discusses the parameters that are severely affecting the congestion in the network. This survey presents the several policies and algorithms from the literature for avoiding, diagnosing, and controlling the congestion. Study proposed in [9] acknowledges that the resources used in this network are limited in storage, bandwidth, and energy. Also, this survey says, the major parameters that are reason for the congestion to take place in sensor network are node buffer overflow, collision in transmission channel, abnormal transmission rate, packet collision, and several nodes transmitting data from many nodes to the sink node.

From the above discussion, it is concluded that controlling the congestion is one of the most important ways to enhance QoS of WBAN. Considering this vital point, this paper identifies the some of the recent CC algorithms [12–21] which are not discussed in the above-mentioned surveys. In the next subsection, the methodology adopted in these algorithms is discussed along with their detailed analysis under



different metrics. This comparative study gives a quick glimpse about the recent CC algorithms and identifies the research gap in this research area.

### 3 Congestion Control (CC) Algorithms

This section portrays the recent CC algorithms proposed in the literature. The analysis made at the end of this section reveals the achieved enhancements and limitations of these algorithms.

*Network status aware congestion control (NSACC)* algorithm proposed in [12], sniffs the status on the network whether it can compensate more packets or it exhausted. A fuzzy-controller estimates the severity of the congestion and regulates the packet sending rate. Rate Regulation (RR) module in the NSACC algorithm regulates the transmission rate to better utilize the bandwidth available, enhances the throughput, and reduces the number of retransmissions.

*Congestion Control Scheme Based on Fuzzy Logic (CCSBF)* [13] exploits the type-2 fuzzy logic controller (T2FLC) system, which evaluates the severity of the congestion in the network. The fuzzy inputs, node rate, and buffer capacity are compared with their predefined threshold to obtain the fuzzy output in terms of congestion severity. Rate adjustment unit (RAU) in this scheme controls the congestion in the networks. RAU adjusts the output rate by scheduling heterogeneous traffic based on the priority. In RAU, a child node sends implicit congestion notification (ICN) to its parent node, likewise hop-by-hop to sink node to notify the congestion degree. Moreover, it controls the output rate of each node to mitigate congestion. It has been shown [13] that it attains better network quality of service (QoS), throughput, packet loss ratio, and optimized energy compared to prioritization-based congestion control protocol [10] and congestion control protocol for prioritized heterogeneous traffic [11]. However, hop by hop spreading of ICN with congestion degree to the sink may put extra load on the network as it needs to visit all intermediate nodes along the route.

*A fuzzy priority-based congestion control (FPBCC)* scheme proposed in [14] is based on the priority of the biosensor data, and it is similar to one of the previous works in this domain, i.e., random early detection (RED) and active queue management (AQM). This scheme [14] tunes the transmission rate of the sender by identifying the traffic load parameter (TLP) using the two-input–single-output fuzzy logic system. It is compared with instantaneous queue size (IQS) and average queue size (AQS) with the predefined minimum and maximum thresholds. Exponential weighted priority-based rate control (EWPBRC) scheme estimates the new transmission rate of the child node in the next iteration, and the transmission rate of each child node is regulated by EWPBRC and TLP. It is simulated in OPNeT and MATLAB with varying number of time slots. FPBCC [14] scheme obtains optimum TLP and regulated send rate, thereby achieves better performance in terms of diminished delay time, packet loss

probability, optimized energy, and end-to-delay. But FPBCC is evaluated only with stationary patients.

**An adaptive rate control for congestion avoidance (ARCCA)** in [15] proposes a cross-layer optimization scheme to optimize the congestion level. ARCCA controls the congestion dynamically at each node by identifying the congestion risk degree (CRD) using a valuation function with a buffer occupancy metric. Sensor node adapts a nature, i.e., itself it identifies its transmitting traffic and evaluates CRD, and it adjusts its transmission rate. It is compared with no rate control mechanism and shown that it attains better results, notably higher throughput, optimum link utilization, energy-efficient, and decreased packet drop rate. The sensor nodes contain a limited buffer to store or program, but ARCCA needs a program to evaluate the CRD using valuation function, which is far from reality. But enhancing the buffer capacity, registers, and a tiny processor becomes extreme evolutions to WBAN's growth.

**Congestion avoidance and mitigation protocol (CAMP)** [16] redirects packets from congested nodes to their neighbors, which can enqueue the redirected packets with their queues. The congestion is identified at the packet level by the number of packet flows and number of packet retransmissions carried out for each packet. As the CAMP focuses on implanted biosensors, it concentrates to reduce the temperature by mitigating the congestion. CAMP achieves better results in terms of temperature, throughput, number of retransmissions, and network lifetime than healthcare aware optimized congestion avoidance and control (HAOCA) algorithm. Of course, sharing the congested node traffic information to its first level neighbors may suffer with congestion symptoms after some time-intervals, because this node has the responsibility to transmit its own data along with congested nodes data. Then, this first level neighbor node may follow the same scheme to share its congested data to second level neighbors. This chain of congested neighbor nodes affects the entire network performance as it further triggers more congestion.

**Priority-Based Congestion Control and Bandwidth Normalization (PBCCBN)** proposed in [17] utilize the bandwidth effectively to avoid the congestion in prior and to achieve the maximum throughput by using priority of the data and size of the data. PBC CBN technique do not have congestion detection phase, where it directly focuses on the effective utilization of bandwidth to prevent congestion before its occurrences. This technique initially calculates the fitness value using a fitness function to sniff the condition of the bandwidth and then places the high priority data along with low priority data simultaneously on the transmission channel, so that it utilizes the bandwidth effectively. It attains better throughput and energy and eradicates dead nodes and delay transmission in the network when compared to traditional MAC protocols. Suppose if the packets delivered from the sensors flood in the network, then it leads to the collision in the communication channel. This can be resolved either by putting an output regulator for sensors or there must be a part of free bandwidth to utilize in case of flooding.

**Priority-based congestion control (PBCC)** in [18] proposed for WBAN, detects the congestion, based on the packet loss ratio in the network. PBCC calculates a

congestion detection (CD) value which is equated with the queue length (L) of a node to regulate the send rate of that particular node. Transmission rate at the node is regulated with the idea of the additive increase and multiplicative decrease (AIMD). It applies a quick start at the beginning for the high prioritized node data and slows down the transmission rate when it reaches congestion state. Every node will be prioritized with the data that is being generated, and each node has a node index value which indicates the priority. Priority values are defined as the high priority when it holds a lower natural number and vice-versa. As the name indicates, PBCC serves best to the critical or prioritized data to be transmitted faster than low priority data. PBCC attains better results like higher throughput and lower packet drop ratio when compared to the conventional TCP protocol. In-fact, the low prioritized nodes may suffer starvation for not being allowed to transmit their data as the high prioritized nodes keep on transmitting the data. This scenario does not suit, when a patient needs emergency service.

**Priority-based Congestion Control Protocol (PCCP)** [19] utilizes network resource management to make the network congestion-free. This protocol mainly serves WSN, also it can be hired to WBAN as the conceptual architecture of both the standards is similar. Rate adjustment scheme in PCCP prevents the upstream congestion hop-by-hop up to sink node by assessing congestion index (CI) which is a ratio of average packet service time and average packet inter-arrival time. PCCP creates a priority table of the sensor data and broadcasts hop-by-hop to all the nodes in the network. When CI is more, a node sniffs the data with more priority from the priority table and schedules the high priority data, otherwise PCCP neglects the priority of the nodes and boosts up the transmission rate of the entire network, thereby, PCCP attains maximum link utilization, throughput, and minimizes loss and delay in the network. In case of low congestion, PCCP hikes the scheduling and data rate of all traffic sources without paying attention to its priority index. In the case of high congestion, it decreases the sending rate of all traffic sources.

**Rate control scheme (RCS)** [20] is an energy efficient and emergency aware scheme that upholds an average transmission rate in the entire network by using rate control factor (RCF). The hub calculates the RCF, on occurrence of an emergency event in the network, sends RCF hop-by-hop to all the remaining nodes, and enables rate control bit (RCB) to the nodes, to regularize the sending rate. Here, emergency event indicates the risk level of the network. RCS grabs a better QoS, energy efficient, and attains enhanced throughput in the network, also achieves higher link utilization compared to traditional algorithms. However, regularizing the traffic in the entire network on occurrence of emergency event may degrade average throughput and effective network bandwidth utilization.

**Event-to-sink reliable transport (ESRT)** algorithm in [21] sets a congestion notification bit (CNB) in the packet, which are transmitted in the network towards the sink, upon buffer overflows. Sink detects the congestion by identifying the packet with CNB and broadcasts a control signal to the entire network to reach all the source nodes for intimating the congestion occurrence and to control it with the local buffer

availability. ESRT achieves optimized energy utilization and protects the reliability of the network. However, sending a control signal to entire network further puts extra load on networks which in turn deteriorates network performance and lifetime too.

**Congestion detection and avoidance (CODA)** [22] detects the congestion using buffer overflow rate by taking input parameters as buffer queue length, channel load, and report time. CODA has open-loop and closed-loop rate control strategies which provides transient and persistent solution for congestion. On detecting the congestion at the node, CODA back pressures the implicit congestion notification hop-by-hop to reach the sink to control the congestion, also a closed-loop multi-source regulation proposed in CODA to keep entire network congestion free. CODA follows AIMD to control the congestion.

This framework is designed for CSMA-based sensor networks. It is implemented using ns-2 and testbed based on Tiny-OS running on Berkeley Rene2 motes. CODA guarantees average energy tax and reliability in transport mechanism. However, backpressure may again deteriorate the energy utilization.

## 4 Summary and Comparative Study of CC Algorithms

This section gives a detailed comparison of aforementioned CC algorithms under various metrics. It is inferred that these algorithms have well-defined methodology to propose a concrete solution to eradicate CC, however, the process of controlling congestion may affect the other important features of WBAN, or else, it may not be focusing on certain limitations of WBAN, e.g., in most of the cases a patient is moving, but an algorithm [14] proposed in literature is bounded to stationary patients, etc.

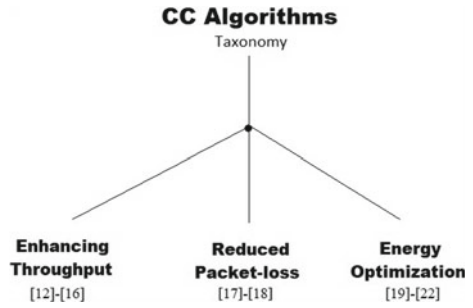
**Taxonomy** of the CC algorithms is based on outcomes yielded from each algorithm proposed in the literature survey, however, all these works consider congestion control as the prime concept. Research works from [1–4] are for enhancing the throughput of the network. Works proposed in [5, 6] reduces the packet losses in the network, whereas the works from [7–10] concentrates on the energy optimization of the network for green communication in WBAN. This taxonomy given in Fig. 1

This section portrays the comparison of algorithms with two Tables 1 and 2. The parameters pertained to methodology proposed in the CC algorithms are summarized in Table 1. The implementation specific details are focused in Table 2.

## 5 Conclusion

WBAN becomes a preceding technology among present research technologies, as the world's lifestyle for the safety, comfort, connectedness, and security of human being enlarged. WBAN serves a remote patient with limited available resources, and

**Fig. 1** Taxonomy of CC algorithms



there is a mandate of providing the best QoS. As the congestion is a severe threat to tamper the QoS, existing algorithms from the literature focus on the congestion control to clutch the QoS. These algorithms first identify the congestion using various parameters such as buffer capacity, bandwidth, data rate, and threshold values, etc., then notifies the network about the severity of the congestion. Finally, these algorithms implement their specific congestion control mechanism in the congestion affected area. Though these algorithms attempted to curtail the congestion in the network, the futuristic algorithms need to be designed using network specific dynamic parameters and analyzed with mobility dynamics, data traffic, and network density.

**Table 1** Comparative study of CC algorithms with their methodology

S. No.	Algorithm	Methodology	Inputs to CC	Outputs from CC	Limitations
1	NSACC [12]	Transmission rate is adjusted based on the input parameters using fuzzy-logic controller	Buffer occupancy level, BSN priority and packet arrival rate	Congestion severity index	Rate regulation is limited to fewer number of bio sensor nodes
2	CCSBF [13]	Rate is adjusted based on the priority of the heterogeneous data from BSNs	Buffer size, node rate and incoming packet rate	Data-rate (output is compared with threshold values)	Spreading ICN may put extra load on the network
3	FPBCC [3, 14]	Traffic load parameter is adjusted to optimize the transmission rate and send rate adjusted for the parent node with a fuzzy logical controller (FLC)	Queue length, change in queue length and priority	Maximum drop probability, Congestion indicator	Concentrated only for stationary patients
4	ARCCA [15]	Rate control done dynamically at each node. Type of data packet is located in the header part of the packet. Then, a weighted fair queueing (WFQ) scheduler is used to route data to MAC	Buffer occupancy and Node rate	Congestion risk degree and valuation function	Sensor nodes are limited in buffer size, to store and forward the data or to program something to evaluate

(continued)

**Table 1** (continued)

S. No.	Algorithm	Methodology	Inputs to CC	Outputs from CC	Limitations
5	CAMP [16]	Constant monitoring of queue level to identify traffic load parameter through fuzzy logic system and sending ICN to previous nodes	Queue occupancy level, priority of the data and packet sending rate	fuzzy logic traffic load parameter)	Sharing congested data to neighbor node may further leads to put more load on the network, as each node have responsible to send its own data
6	PBCCBN [17]	Allocate bandwidth for high priority and low priority data and apply bandwidth utilization	Priority of the sensor node and size of the packet	Fitness value	Flooding may occur due to rise in sensors data outcome
7	PBCC [18]	Adjusts the send rate of the sender node based on number of packet losses	Queue length and bandwidth	Packet loss parameter	Low-priority data sensors may suffer starvation
8	PCCP [19]	Priority-based Rate Adjustment	Packet interarrival time and packet service time at the MAC layer	Congestion degree	TCP slow-start with AIMD increases data rate exponentially, leads to unpredictable congestion in network
9	RCSCC [20]	Sends RCF to all normal nodes in order to decrease the rate of the normal traffic and to keep almost the same average rate in the whole WBAN	Total number of nodes and no. of emergency nodes	Rate Control Factor	Maintaining same average traffic rate during all the emergency events, may degrades the utilization of resources
10	ESRT [21]	Sink sends control messages to the source nodes to control congestion	Reliability indicator bit (RIB) for ACK/NACK	Local buffer level monitoring	Broadcasting back to source node may deteriorates network capacity

(continued)

**Table 1** (continued)

S. No.	Algorithm	Methodology	Inputs to CC	Outputs from CC	Limitations
11	CODA [22]	CODA controls the traffic rate using additive increase multiplicative decrease (AIMD) Method followed is: (i) receiver-based congestion detection; (ii) open-loop hop by hop backpressure; and (iii) closed-loop multi-source regulation	Buffer overflow rate	Queue length, Channel load and report rate	As it is necessary to prevent congestion using open-loop hop-by-hop backpressure, but in-turn it may put overburden on the network. Applying both, open-loop and closed-loop techniques in a single system is not desirable

**Table 2** A comparative study of simulation environment of CC algorithms

S. No.	Algorithm	Communication and type	Topology	Simulator	Achieved simulation results
1	NSACC [12]	Hop-by-hop and implicit	Start topology	MATLAB	Enhanced throughput and network lifetime, reduced re-transmissions
2	CCSBF [13]	Hop-by-hop and implicit	Single-path tree topology	OPNET and MATLAB	Network throughput, PLR, end-to-end delay and energy performance
3	FPBCC [3, 14]	Hop-by-hop and implicit	Star topology	OPNET and MATLAB	Packet loss, end-to-end delay and energy
4	ARCCA [15]	-NA- and implicit	Simple topology	NS2	Higher throughput, link utilization, decreased drop rate and energy efficiency

(continued)



**Table 2** (continued)

S. No.	Algorithm	Communication and type	Topology	Simulator	Achieved simulation results
5	CAMP [16]	Hop-by-hop and implicit	Tree topology	NS2	Energy efficient, network lifetime and throughput
6	PBCCBN [17]	Hop-by-hop and -NA-	-NA-	MATLAB	Delay, no. of dead nodes, throughput
7	PBCC [18]	End-to-end and piggybacking	Star topology	MATLAB	Delay, fewer packet losses
8	PCCP [19]	Hop-by-hop and piggybacks the packet scheduling	Star topology	MATLAB	PLR, low energy, lower packet delay, throughput
9	RCSCC [20]	Hop-by-hop and -NA-	Star topology	Castalia with OMNeT++	Energy waste index versus energy index, emergency packet loss versus energy, no. of packets versus latency
10	ESRT [21]	Event-to-sink and -NA-	Dynamic topology	NS2	Power consumption is less and no congestion
11	CODA [22]	Hop-by-hop and implicit	Testbed topology	Testbed based on Berkeley motes	Avg energy tax and reliability in transport mechanism

## References

1. <https://standards.ieee.org/standard/802.15.6-2012.html>
2. S. Gambhir, V. Tickoo, M. Kathuria, Priority based congestion control in WBAN, in *Eighth International Conference on Contemporary Computing (IC3)*. (IEEE, 2015), pp. 428–433
3. S.A. Salehi, M. Razzaque, I. Tomeo-Reyes, N. Hussain, IEEE 802.15. 6 standard in wireless body area networks from a healthcare point of view, in *22nd Asia-Pacific Conference on Communications (APCC)*. (IEEE, 2016), pp. 523–528
4. S.N. Shah, R.H. Jhaveri, Recent research on wireless body area networks: a survey, *Int. J. Comput. Appl.* **975**, 8887–8894 (2016)
5. M. Asam, A. Ajaz, Challenges in wireless body area network. *Proc. Int. J. Adv. Comput. Sci. Appl.* **10**(11) (2019)
6. D.D. Olatinwo, A. Abu-Mahfouz, G. Hancke, A Survey on LPWAN Technologies in WBAN for Remote Health-Care Monitoring. *Sensors* **19**(23), 5268–5294 (2019)
7. K.G. Mkongwa, Q. Liu, C. Zhang, F.A. Siddiqui, Reliability and quality of service issues in wireless body area networks: a survey. *IJSPS* **7**(1), 26–31 (2019)

8. I Pandey, H.S. Dutta, J.S. Banerjee, WBAN: a smart approach to next generation e-healthcare system, in *3rd International Conference on Computing Methodologies and Communication (ICCMC)*. (IEEE, 2019), pp. 344–349
9. A. Bohloulzadeh, M. Rajaei, A survey on congestion control in wireless sensor networks. *Int. J. Wir. Inf. Netw.* 1–20 (2020)
10. M.H. Yaghmaee, N.F. Bahalgardi, D. Adjeroh, A prioritization based congestion control protocol for healthcare monitoring application in wireless sensor networks. *Wir. Personal Commun.* **72**(4), 2605–2631 (2013)
11. M.M. Monowar, M.O. Rahman, A.-S.K. Pathan, C.S. Hong, Congestion control protocol for wireless sensor networks handling prioritized heterogeneous traffic, in *Proceedings of the 5th Annual International Conference on Mobile:2288* (2008).
12. M.V. Kiran, B. Nithya, NSACC: network status aware congestion control algorithm for wireless body area network. *Procedia Comput. Sci.* **171**, 42–51 (2020)
13. S. Ghanavati, J. Abawaji, D. Izadi, A congestion control scheme based on fuzzy logic in wireless body area networks, in *IEEE 14th International Symposium on Network Computing and Applications* (2015), pp 235–242
14. F. Pasandideh, A.A. Rezaee, A fuzzy priority based congestion control scheme in wireless body area networks. *Int. J. Wir. Mobile Comput.* **14**(1),1–15
15. Y.-M. Baek, B.-H. Lee, J. Li, Q. Shu, J.-H. Han, K.-J. Han, An adaptive rate control for congestion avoidance in wireless body area networks, in *International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery* (IEEE, 2009), pp. 1–4
16. M. Anwar, A.H. Abdullah, R.R. Saedudin, F. Masud, F. Ullah, CAMP: Congestion avoidance and mitigation protocol for wireless body area networks. *Int. J. Integr. Eng.* **10**(6) (2018)
17. H.S. RavinderKaur, Priority based congestion control and bandwidth normalisation in WBAN. *Int. J. Electron. Commun. Technol (IJECT)* **2**,1–5 (2017)
18. S. Gambhir, V. Tickoo, M. Kathuria, Priority based congestion control in WBAN, in *Eighth International Conference on Contemporary Computing (IC3)* (IEEE, 2015), pp. 428–433
19. D. Patil, S.N. Dhage, Priority-based congestion control protocol (PCCP) for controlling upstream congestion in wireless sensor network, in *International Conference on Communication, Information Computing Technology (ICCICT)* (IEEE, 2012), pp 1–6
20. R. Jaramillo, A. Quintero, S. Chamberland, Rate control scheme for congestion control in wireless body area networks, in *IEEE 12th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)* (2016), pp. 1–6
21. Y. Sankarasubramaniam, Ö.B. Akan, I.F. Akyildiz, Esrt: Event-to-sink reliable transport in wireless sensor networks, in *Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking Computing* (2003), pp. 177–188
22. C.-Y. Wan, S.B. Eisenman, A.T. Campbell, CODA: Congestion detection and avoidance in sensor networks, in *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems* (2003), pp. 266–279

# Applications of RSSI Preprocessing in Multi-Domain Wireless Networks: A Survey



Tapesh Sarsodia, Uma Rathore Bhatt, and Raksha Upadhyay

**Abstract** Today's age of communication has been looking for technologies and techniques to support high data rate applications with required quality of services. Advanced communication network architectures like Internet of things (IoT), fifth generation (5G), and long term evolution (LTE) with supporting high end transmission and reception processes have evolved to meet present requirements. It is also observed that to further enhance network performance, incorporation of received signal strength indicator (RSSI)/channel state information (CSI)-based preprocessing techniques have been exhibiting substantial impact. Physical layer key generation in wireless networks, localization of nodes in wireless networks, signal identification, human activity recognition, etc., are few such applications, using RSSI/CSI preprocessing for their performance improvement in multi-domain wireless networks. Hence, this paper describes above-mentioned applications using different preprocessing techniques of RSSI, which is not investigated comprehensively in literature so far. Therefore, the purpose of this paper is to reveal the impact of RSSI preprocessing techniques in system performance enhancement as per the need of application. As an outcome, we find the possibility of applying other preprocessing techniques in existing and upcoming applications in future to achieve desired system performance.

**Keywords** Human activity recognition · Key generation · Localization · Signal preprocessing · Wireless networks

---

T. Sarsodia · U. R. Bhatt (✉) · R. Upadhyay  
Institute of Engineering and Technology, Devi Ahilya University, Madhya Pradesh, Indore  
452017, India  
e-mail: [uvrathore@gmail.com](mailto:uvrathore@gmail.com)

T. Sarsodia  
e-mail: [tapeshs162@gmail.com](mailto:tapeshs162@gmail.com)

R. Upadhyay  
e-mail: [raksha\\_upadhyay@yahoo.co.in](mailto:raksha_upadhyay@yahoo.co.in)

## 1 Introduction

Current applications of wireless communication systems are expecting very high data rates, excellent performance with efficient use of available resources. This led to the advanced development of various aspects addressing security, low power usage, superior network architectures design, advanced system design techniques, etc., in the past few decades [1]. Apart from all these aspects, recently preprocessing of received signal strength indicator (RSSI) or channel state information (CSI) values in an application specific manner has led to additional improvement in system performance and is not addressed in the literature comprehensively [2]. RSSI values are gathered from beacons, exchanged periodically between communicating nodes. On the other hand, CSI is impulse response of the channel. Before actual communication, first RSSI/CSI values are processed in application specific manner, to get some additional insight of system; hence, it is called as preprocessing. There are various emerging application areas, in which preprocessing of RSSI/CSI enhances system performance. Physical layer key generation, localization of nodes in power constraint networks, channel identification, human activity recognition, etc., are emerging application areas in which RSSI/CSI processing is being used [3–5].

Generally, data preprocessing includes data cleaning, its integration, transformation, reduction, and discretization. Data cleaning includes filling of missing values, smoothening of noisy data, and resolving the inconsistencies. Different signal processing techniques, basic statistical approaches, etc., are generally used for data cleaning of RSSI/CSI. These techniques may result in significant additional improvement in system performance of the current communication system. For example, bit error rate performance of communication systems including long term evolution (LTE), 5G [6, 7], and bit disagreement rate (BDR) of physical layer key generation [8] system for upcoming communication networks, power efficiency in different power constraint communication networks like Internet of things (IoT), wireless sensor networks (WSNs) [9], etc., may be improved. Various data processing techniques available are data integration, data transformation, etc. [10] Data integration considers the incorporation of data from all other sources available in the network to improve the performance of the network in terms of packet delay, packet dropping probability, channel throughput, etc. Smart farming [11] and voice over data are the example applications of data integration [12]. Data transformation of RSSI/CSI uses different techniques in time and frequency domain to improve the quality of transmission and reduce the complexity of data for further processing. Data transformation may use various transformation techniques like discrete cosine transform (DCT), wavelet transform (WT) [13], etc., which further helps to reduce the network complexity and to improve the network quality of service (QoS). Data reduction is one of the important aspects in any communication system. There is a variety of data reduction techniques like principal component analysis (PCA), individual component analysis (ICA), low variance filter, random forest, etc., are available [14]. Incorporation of these techniques results in a surprising reduction in power requirement and the complexity of systems. Since power efficiency [15] is one of the

crucial factors for power constraint modern networks, data reduction techniques are also very useful. Similarly, data discretization has also been a very important process that produces data into small desired intervals for further processing. This helps the network to analyze different aspects of information collected from various sources. Discretization provides diversity to the network performance by generating different dimensions to the raw data available for an enhanced study of the network. It is concluded that prior to actual data transmission, RSSI/CSI preprocessing may be used for system performance enhancement. In this paper, only RSSI has been considered for further discussion. Therefore, motivation behind this paper is to discuss different multi-domain application areas comprehensively using preprocessing of RSSI. The aim is to reveal substantial improvement in system performance by the inclusion of these techniques in different emerging applications and examine the future aspect of research in this direction. We have not found any research paper focusing on RSSI preprocessing on different applications of wireless communication. The rest of the paper is organized as follows: In Sect. 2, we discuss briefly about RSSI. After that we discuss different applications or fields where RSSI preprocessing played an important role along with some future works. All this were discussed successively in Sects. 3, 4, 5, 6, and 7. Finally, Sect. 8 concludes the paper followed with references.

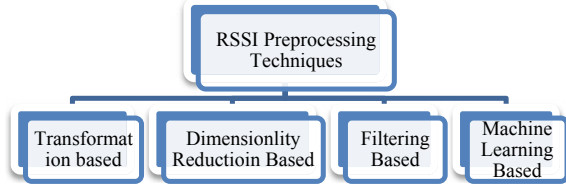
## 2 RSSI Preprocessing Techniques

In wireless networks, prior to actual data transfer beacons are exchanged between two communicating nodes. Beacon consists of RSSI which depicts the strength of signal received at a particular node and is link quality indicator [16]. RSSI depends on several factors like distance, channel quality, mobility, etc., which directly or indirectly affects any network performance. Hence, characterizing RSSI and its processing plays important role in improvement of network performance.

RSSI is characterized on the basis of dynamic range, accuracy, linearity, and averaging period. Dynamic range relates to minimum and maximum energy that receiver can measure while accuracy indicates average error occurred in each received RSSI. Deviation of RSSI with standard straight line graph shows linearity, and averaging period indicates the averaged value of received RSSI over a particular time period [17]. Processing of RSSI results in improved and accurate characterization, which will further utilized for particular application. The different techniques found for processing of RSSI as shown in Fig. 1.

RSSI data may be analyzed in time domain or frequency domain depending on application. Various transformation techniques like DCT, discrete wavelet transform (DWT), and discrete Fourier transform (DFT), etc., are used for such purpose. RSSI plays a key role in improving the performance of any wireless network by reducing its complexities in terms of dimension reduction of data received, which in turn also reduces energy requirements of the network. Sometimes we need to reduce data dimensions for reducing complexity and energy consumption of the network. For achieving such reduction in data, we use dimensionality techniques

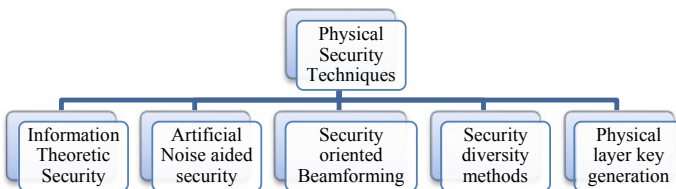
**Fig. 1** Possible classification of RSSI-based techniques



like PCA, ICA, etc. Received RSSI over wireless channel suffers from inherent impairments like channel noise, fading, effects of mobility, etc. These impairments may be reduced using filtering approaches. Various de-noising and smoothening filtering approaches are implemented on RSSI. In past few years, machine learning (ML)-based approaches are also being used as RSSI processing techniques for more accurate outcomes in different applications. ML-based techniques are support vector machine (SVM), artificial neural networks (ANN), decision tree (DT), etc. Along with ML techniques, deep learning and neural network-based techniques like convolutional neural networks (CNN) were also attracting the researchers for designing of an efficient RSSI processing-based wireless network [18, 19]. Subsequent sections are briefly discussing various application areas in which RSSI processing plays key role in system performance. For all these applications, a general roadmap would be to extract RSSI first, preprocessed it, using any of the techniques described in this section. Selection of technique was done as per the need of application.

### 3 Physical Layer Secure Key Generation (PLKG)

Typically advanced wireless networks like LTE, Wi-Fi, Bluetooth, and 5G support high speed data over the network with secured exchange of data. Security over wireless channel has been a prominent area for researchers. Traditionally, security is achieved using cryptography. Traditional cryptographic techniques include symmetric and asymmetric encryption which may not suitable for current power efficient requirements due to high computational burden. Physical layer security is an alternative approach, which has attracted the attention of researchers [20]. Various physical layer security techniques are shown in Fig. 2. Information theoretic



**Fig. 2** Different physical layer security techniques [21]

security system uses Shannon theory for providing secrecy over channel. Artificial noise aided security scheme generates additional noise in the network which adversely affects eavesdropper only and not to legitimate user. Security-oriented beamforming techniques allow source node to transmit its data to destination node successfully by creating destructive interference at attackers end making his signal too weak to reconstruct. Security diversity methods work same as artificial noise aided security methods but with improved power usage in the network. This method considers antenna-based diversity techniques and many other related methods. In physical layer key generation security technique, the data at both ends are encoded with same array of bits namely key which is generated at both legitimate users ends for secured transmission. PLKG utilizes parameters like RSSI, CSI for performing secured transmission over wireless channel [21].

In PLKG, beacons of transmitter and receiver are exchanged and processed to generate same keys at both ends. This process eliminates the sharing of keys hence additional security is provided. Generated keys must be sufficiently random with desired value of key generation rate (KGR), key disagreement rate (KDR), and key error rate (KER). KGR shows amount of bits generated unit time per measurement. High value of KGR is preferable for key generation. KDR shows number of bits in which pre-key bits differs at both ends. Normally, this value was considered low. KER defined as the ratio of number of failed key groups to the number of total key groups. PLKG includes four main steps for successful key generation are channel probing and RF preprocessing, quantization, information reconciliation, and privacy amplification and key generation. All these steps are briefly summarized in Fig. 3 [22].

For PLKG, RSSI preprocessing is used to improve the KDR and KGR while maintaining sufficient randomness. In recent literature, various techniques of preprocessing are used to enhance the performance of PLKG and are listed in Table 1.

From the above-discussed findings, it is observed that RSSI preprocessing plays an important role for generating secure key between two legitimate users present in a wireless channel. Authors suggested different algorithms for generating secure key at both ends by using different preprocessing techniques for signals like PCA, DWT, DCT, MWA, etc. This survey shows that there is a lot of scope in applying

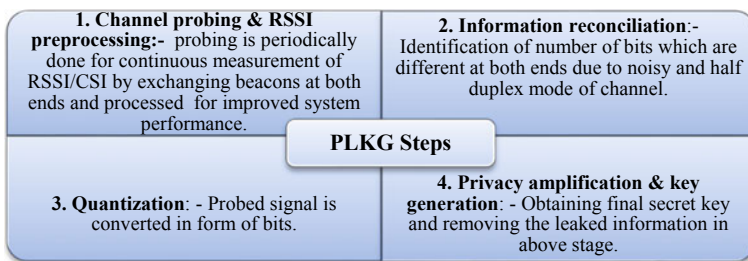


Fig. 3 Different PLKG steps [22]

**Table 1** RSSI preprocessing in key generation

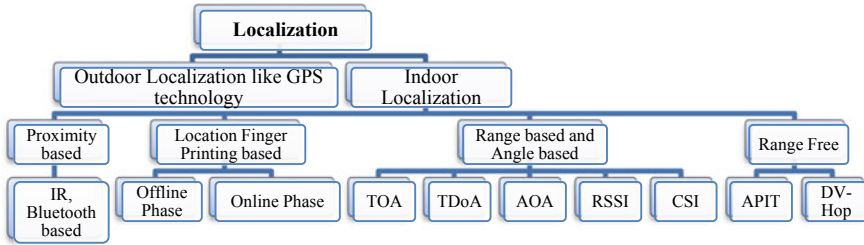
Author name and year	Brief description	Outcomes
Zhan et. al. (2017) [23]	RSSI is passed through DWT compressor and multi-level quantizer, which is followed by two universal hash functions	Improved BDR as compare to existing algorithms
Li et. al. (2018) [24]	Comparison of different approaches like DCT, DWT, and PCA with common eigen vector on RSSI for PLKG system	PCA outperforms in terms of high KGR, low KER, and improved randomness
Soni et. al. (2019) [25]	Improved secret key generation scheme for IoT networks which process RSSI data by moving window averaging (MWA) approach followed by Llyod- Max quantizer	Reduces BDR appreciably at low signal to noise ratio (SNR)
Soni et. al. (2019) [26]	Different quantization schemes like linear, Lloyd, Ambekar, and common Llyod are compared for PLKG taking input as RSSI	Llyod Max quantizer performs with improved randomness and low BDR
Soni et. al. (2019) [27]	Authors suggested a MWA-based filtering approach on RSSI using Ljung Box Q test and Allan Variance	Low BDR with improved randomness in their key
Lin et. al. (2020) [28]	Authors propose to use wavelet shrinkage-based approach followed by a quantizer with guard band security on RSS signals in PLKG system	Algorithm outperforms on the basis of BGR and BDR significantly

preprocessing techniques to PLKG system. For example, different dimensionality reduction techniques like high correlation filters, uniform manifold approximation, etc., may be used in the future to achieve desired characteristics of key generation. Similarly, various filtering techniques and non-linear processing techniques are also used for future purpose.

## 4 Localization of Nodes in Wireless Networks

Localization in wireless networks plays a key role in enhancing the network performance. Localization is the process of locating any object/node as per its geographical and physical coordinates. It also helps to predict exact location of different mobile nodes in a wireless environment which in turn helps to fix the number of nodes in the network. Different wireless networks Wi-Fi, Bluetooth, Zigbee, and infrared, etc., use localization for efficient deployment of their nodes. This may result in add-on





**Fig. 4** Various localization schemes [30, 31]

in network performance in terms of cost, power consumption, utilization of network resources, etc. Localization [29] concepts are being used in wide range of applications like wireless container positioning system in shipyards using tags, healthcare systems like telemedicine, wireless construction material management, for road safety and mining safety, etc.

There are various techniques used for localization in wireless network which are shown in Fig. 4. Outdoor localization mainly depends on GPS for collecting random information of all sensing nodes [30]. GPS provides accurate results, but it suffers from complex processing and hence practically unviable for denser wireless networks. Moreover for indoor wireless networks localization, other alternatives are more attractive. Generally, indoor systems need beacon or anchor nodes for identification. Usually, beacon data generated by anchor nodes used to identify the exact location of nodes. Anchor nodes are the nodes which have their exact location determined by GPS installed at them or manual positioning and rest of the nodes depict their location via anchor nodes. From Fig. 4, it is clear that indoor localization considers proximity-based localization as in Bluetooth or infrared (IR) enabled devices, range or angle-based localization which considers angle of arrival (AoA), time of arrival (TOA), time difference of arrival (TDoA), etc., and lastly range free or distance-based localization which includes DV hop count-based, approximate point-in-triangulation test (APIT), etc. [31]

Each method have its own advantages but RSSI-based technique is considered to be most vital because of its robustness and cost saving approach, and it can support existing infrastructure of network also. In this approach, localization and positioning are done with the help of RSSI-based distance estimation or measurement [32]. Localization using RSSI processing has been extensively used in wireless networks, which are summarized in Table 2.

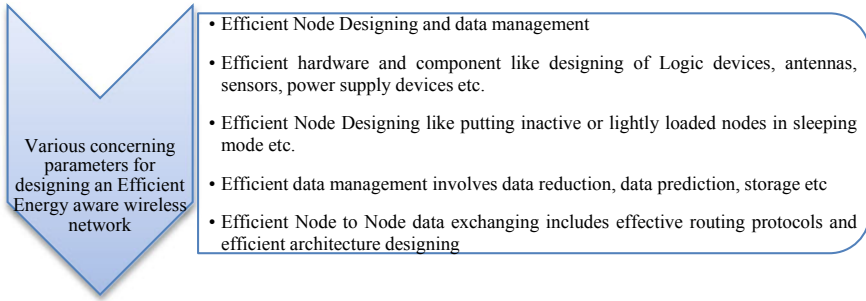
From the above discussion, it is clear that preprocessing of RSSI received from various access points available in indoor or outdoor environment proves itself a major tool for identifying mobile nodes in a compact indoor environment and in denser outdoor environment. Furthermore, decision trees, regression analysis, Bayesian networks, and genetic algorithms can also be part of processing methods of RSSI signals for better localization with low positioning error rate. Hence, it is clear that RSSI-based localization provides sufficient space for future research.

**Table 2** RSSI preprocessing in localization

Author name and year	Brief description	Outcomes
Abusara et. al. (2016) [33]	mFOS algorithm considers RSSI of those APs in network which are more informative than other, and hence others were rejected. This reduces overall network dimensionality	More accurate APs localization along with improved accuracy and reduced complexity
Hou et.al (2017) [34]	Author uses RSSI data and preprocesses it to denoise using two filtering techniques viz. smoothing filter-based and wavelet transform-based	Reduced localization error with greater stability
Roy et. al. (2019) [35]	RSSI data collected from various grids was preprocessed to denoise, and then different grids were formed with minimum error rate followed by machine learning (ML) algorithm for further location identification	Proposed work improves localization accuracy up to 96.62% with low localization error rate of 4.54%
Wu et. al. (2019) [36]	The algorithm work in two phases: one is off line phase in which data collected and processed using PCA and SVM technique and another is online phase in which different ML functions were formed	Improved localization error rate with high accuracy
Liu et. al. (2019) [37]	The algorithm has two phases: - First one is dynamic de-noising of RSSI data and after that iterative clustering is done. Second one is feature enhancement, in which they jagged fluctuations	Reduced positioning error up to greater extent and may lie between 28 and 33%
Anagnostopoulos and Kalousis (2019) [38]	Proposed some possible machine learning-based RSSI data sets solutions which uses fingerprint-based localization method in outdoor positioning for a Sigfox like urban area networks	Reduced mean localization error up to 298 m and median error to 109 m

## 5 Energy Efficient Wireless Networks

Energy efficiency plays an important role in designing any data networks. Various algorithms are being used to make different communication networks like Bluetooth network, WSN, Wi-Fi, low power wireless public area network (LoWPAN), etc., as energy efficient [39]. In general, for designing an energy aware wireless network, different aspects need to be considered, as depicted in Fig. 5. For energy efficient designing, one should consider optimum selection of different hardware



**Fig. 5** Designing parameters for energy efficient wireless networks [40]

components, energy aware node designing including software algorithms, effective routing protocols, efficient architecture design, etc. [40] Apart from long history of all these traditional constituents to make energy aware network, recently RSSI processing-based techniques is used for the same purpose.

Table 3 summarizes different research works considering RSSI processing for energy efficient wireless networks.

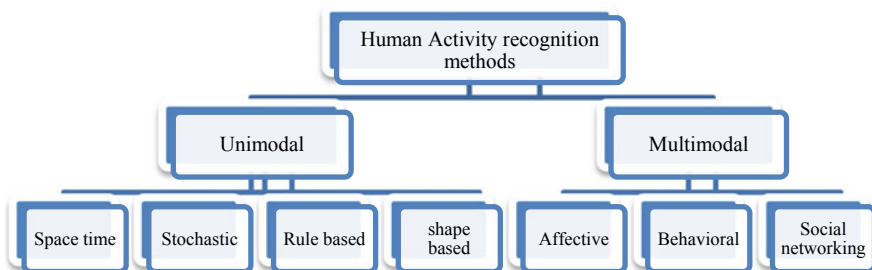
**Table 3** RSSI preprocessing in different energy efficient networks

Author name and year	Brief description	Outcomes
Soni et. al. (2019) [41]	Authors combine PCA and MWA algorithms and apply on RSSI in suggested power range for IoT network and in presence of color noise. PCA reduces the dimension of RSSI and results in power saving	Low BDR and improved randomness with energy aware future IoT systems
Margelis (2017) [42]	DCT for preprocessing of RSSI, which is followed by quantization and Slepian-Wolf coding for information reconciliation stage	Efficient secret keys at a faster rate with low energy entropy
Xu (2009) [43]	Author uses multiple agents to make energy efficient network. RSSI data processed properly in order to fulfill its integrity before transmitting it to destination nodes	Reduce transmission error along with energy saving
Nisha and Basha (2020) [44]	The algorithm preprocesses residual energy and RSSI for making energy aware clusters. After that triangular fuzzy membership functions are used for choosing cluster heads	Reduced energy consumption in the network

It is concluded that preprocessing of received signal strength (RSS) is utilized in many works, which helps in designing an energy efficient network. Furthermore, the designing of energy aware networks can be supported by RSSI processing based on ML by making the network self-sustained. By using various ML models, the nodes themselves study their energy status by maintaining an energy aware table for each node in the network, which helps them to decide whether they have to remain in active mode or in sleep mode. Similarly, different ANN and dimensionality reduction techniques were also used as RSSI processing techniques for reduction of power usage in different networks in multiple ways.

## 6 Human Activity Recognition (HAR) Systems

In past few decades, rapid development is noted in the field of human activity detection and its related methodologies. These human activity-based techniques become one of the important fields for researchers in past few years mainly because of generation of various new human neural and mental diseases. These diseases force observers to continuously monitor human activities inside or outside their homes. Human activity monitoring also found applications in military surveillance, telemedicine networks, etc. [45] Human activities are mainly classified as macro activities and minor activities. Macro activities include sitting on chair, standing position, lying on a bed, etc., whereas micro activities relates to hand gestures, facial expressions, peeling a fruit, etc. Human activity-based systems are classified on the basis of gestures, atomic actions, human to object or human to human interactions, behaviors, and events [46]. Commonly used human activity recognition methods are shown in Fig. 6. Unimodal consider RSSI data with unique modality like gestures image only. Unimodal also classifies itself on the basis of evaluating raw RSSI data like spatiotemporal features, trajectories, statistical model, rule-based, motion of body part, etc. Multimodal includes RSSI data of all different physical gestures. It is also categorized based on type of RSSI data collected like expressive communication, facial expressions, social interlinking with other human, etc. Interestingly for HAR



**Fig. 6** Human activity recognition methods [46]

**Table 4** RSSI preprocessing in HAR

Author name and year	Brief description	Outcomes
Sigg et. al. (2014) [47]	Author investigated various aspects of RSSI-based data collected on mobile handsets for proper gesture recognition	Achieve high accuracy in recognizing up to 11 gestures
Mukherjee et. al. (2018) [48]	Author creates an indoor WSN network and detects four different motions by a human as per signal received. At last they compare radar-based system data with RSSI data	Improved accuracy in case of RSSI-based WSN network
Booranawong et. al. (2019) [49]	Author proposed an effective filtering approach which automatically reduces RSSI variations at each subsequent level for reducing the complexities in the RSSI processing, and preprocessed RSSI data helps to identify human motion effectively	Improved accuracy with reduced computational complexities
Su et. al. (2020) [50]	Author proposes an effective wearable mosaic antenna for efficient human activity recognition. RSSI collected from antenna preprocessed with machine learning approach which performs better than other available wearable antennas	Achieved an accuracy of 91.9% for RF-based recognition systems

also, RSSI processing is proving itself a major technique, and a few reported works are given in Table 4.

It is clearly evident from literature that RSSI preprocessing performs an important role in improving the accuracy of a network deployed and reducing the overall network complexity for HAR. In future, various RSSI processing techniques like ML-based, dimensionality reduction-based, transformation-based, etc., can be used individually or combined for getting improved performance for HAR.

## 7 Miscellaneous Applications

This section summarizes few more application areas of RSSI processing apart from key generation, localization, energy efficient network, and HAR. RSSI preprocessing might be helpful in various other applications like signal recognition [51] and channel estimation in indoor environment by employing various ML techniques along with neural network algorithms. Various algorithms for proper handoff sessions are also

proposed, which are based on RSSI processing, in order to provide high QoS to mobile users [52]. RSSI may be used as controlling element for automatic gain control systems, in various trans-receivers [53]. Wireless channel identification (Line-of-sight (LOS)/non-line-of-sight (NLOS)) [54], position, and power adjustment of hardware components also done based on RSSI, received from various sensors. Hybrid beamforming was proving itself as one of the promising technology for providing high data rate to massive multiple-input multiple-output (MIMO) systems. Author in [55] proposed a novel RSSI-based unsupervised deep learning method to design an effective precoder analog codebook which uses partial CSI feedback. Their system outperforms in terms of reduced signaling overhead and improved spectral efficiency. Other applications are also being developed where RSSI preprocessing has been playing a key role to improve network performance substantially. It seems that RSSI processing may be beneficial for upcoming multi-domain applications apart from discussed in this section.

## 8 Conclusion

In this paper, we made a detailed discussion on various applications of preprocessing of RSSI in real-world applications. Preprocessing of RSSI plays an important role in different applications which decides the overall system performance. First we discussed PLKG in wireless networks using RSSI, adopting different preprocessing techniques like MWA, DCT, DWT, various dimensionality reduction techniques, etc. There is vast majority of such techniques, such as Wiener filtering, nonlinear processing of RSSI, etc., are still available which have not been applied so far. We expect that incorporating these possible techniques will add new dimensions in PLKG. Similarly, localization process may be further improved by using optimizing techniques like genetic algorithm, particle swarm, etc., over RSSI. Power constraint networks like IoT, WSN may use techniques which are less complex and require less computation power for processing of RSSI. There are varieties of dimensionality reduction techniques other than PCA which can be possible alternatives to existing ones. We found HAR as another application area, in which RSSI processing is further need to be explored. This paper also summarizes various preprocessing techniques applied over different applications of wireless networks. The paper concludes that lot of future scope of RSSI preprocessing techniques is available in different applications to achieve desired network performance and this review may be useful for researchers.

**Acknowledgements** We would like to acknowledge IET, DAVV, Research Center, Indore, India. This paper can be used as part of a Ph.D. thesis in the future for the first author.

## References

1. C. Wang et al., Cellular architecture and key technologies for 5G wireless communication networks. *IEEE Commun. Mag.* **52**(2), 122–130 (2014)
2. D. Konings, Device-free localization systems utilizing wireless RSSI: a comparative practical investigation. *IEEE Sens. J.* **19**(7), 2747–2757 (2019)
3. Y. Shiu et al., Physical layer security in wireless networks: a tutorial. *IEEE Wirel. Commun.* **18**(2), 66–74 (2011)
4. T.L. Marzetta, B.M. Hochwald, Fast transfer of channel state information in wireless systems. *IEEE Trans. Signal Process.* **54**(4), 1268–1278 (2006)
5. X. Ding, S. Dong, Improving positioning algorithm based on RSSI. *Wireless Pers Commun* **110**, 1947–1961 (2020)
6. D. Li, Y. Lei, H. Zhang, A novel outdoor positioning technique using LTE network fingerprints. *J. Sens.* **20**, 169 (2020)
7. J.A. Santana, E. Macías, Á. Suárez et al., Adaptive estimation of WiFi RSSI and its impact over advanced wireless services. *Mobile Netw. Appl.* **22**, 1100–1112 (2017)
8. B. Han, S. Peng, C. Wu, X. Wang, B. Wang, LoRa-based physical layer key generation for secure V2V/V2I communications. *Sensors* **20**, 682 (2020)
9. B.S. Meena, S. Deb, K. Hemachandran, Impact of heterogeneous IoT devices for indoor localization using RSSI, in *Intelligent Computing in Engineering. Advances in Intelligent Systems and Computing* ed by V. Solanki, M. Hoang, Z. Lu, P. Pattnaik, 1125 (Springer, Singapore 2020), pp. 187–198
10. S. Alasadi, W. Bhaya, Review of data preprocessing techniques in data mining. *J. Eng. Appl. Sci.* **12**(16), 4102–4107 (2017)
11. J. Bauer, N. Aschenbruck, Towards a Low-cost RSSI-based crop monitoring. *ACM Trans. Internet Things*, **1**(4), 26
12. P. Koutsakis, M. Paterakis, Highly efficient voice—data integration over medium and high capacity wireless TDMA channels. *Wireless Netw.* **7**, 43–54 (2001)
13. R. Upadhyay et al., A study on principal component analysis over wireless channel. *J. Telecommun. Electron. Comput. Eng.* **11**(4), 5–9 (2019)
14. P.E. Lopez-de-Teruel et al., Using dimensionality reduction techniques for refining passive indoor positioning systems based on radio fingerprinting. *Sensors* **17**(4), 871 (2017)
15. Muladi et al., Adaptive power management for self-powered IoT on smart shoes, in *AIP Conference Proceedings. American Institute of Physics* 2228 (2020), 030019.
16. A. Guidara, et al., Impacts of temperature and humidity variations on RSSI in indoor wireless sensor networks, in *22nd International Conference on Knowledge-Based and Intelligent Information & Engineering Systems in Elsevier Procedia Computer Science*, 126 (2018), 1072–1081
17. S. Farahani, Location estimation methods. in *ZigBee Wireless Networks and Transceivers (Chapter 7)* (2008), 225–246.
18. H. Ahmadi, R. Bouallegue, Exploiting machine learning strategies and RSSI for localization in wireless sensor networks: a survey, in *13th International Wireless Communications and Mobile Computing Conference (IWCMC)*, (Valencia, 2017), pp 1150–1154
19. C. Hsieh, J. Chen, B. Nien, Deep learning-based indoor localization using received signal strength and channel state information. *IEEE Access* **7**, 33256–33267 (2019)
20. S. Li, Q. Du, A review of physical layer security techniques for internet of things: challenges and solutions. *J. Entropy* **20**, 730 (2018)
21. Y. Zou, J. Zhu, X. Wang, L. Hanjo, A survey on wireless security: technical challenges. *Recent advances and future trends. Proc IEEE* **104**(9):1727–1765
22. J. Zhang et al., Key generation from wireless channels: a review. *IEEE Access.* **4**, 614–626 (2016)
23. F. Zhan, N. Yao, On the using of Discrete wavelet transform for physical layer key generation. *J. Adhoc Netw.* **64**, 22–31 (2017)

24. G. Li et al., High-agreement uncorrelated secret key generation based on principal component analysis preprocessing. *IEEE Trans. Commun.* **66**(7), 3022–3303 (2018)
25. A. Soni, R. Upadhyay, A. Kumar, Wireless physical layer key generation with improved bit disagreement of the internet of things using moving window averaging. *J. Phys. Commun.* **33**, 249–258 (2019)
26. A. Soni, R. Upadhyay, A. Kumar, RSS based phy layer key generation in wireless communication, in *Proceedings of Recent Advances in Interdisciplinary Trends in Engineering & Applications (RAITEA)*
27. A. Soni, R. Upadhyay, A. Kumar, Performance improvement of wireless secret key generation with colored noise for IoT. *Int. J. Commun. Syst.* **32** (2019)
28. R. Lin, L. Xu, H. Fang, et al., Efficient physical layer key generation technique in wireless communications. *J. Wireless Com. Netw.* **13** (2020)
29. R.T. Reza, V.M. Srivastava, Effect of GSM frequency band on received signal strength and distance estimation from cell tower. in *10th International Conference on Developments in eSystems Engineering (DeSE)* (Paris, 2017), pp. 151–154
30. J. Kuriakose S. Joshi R. Vikram Raju A. Kilaru A, A review on localization in wireless sensor networks. in *Advances in Signal Processing and Intelligent Recognition Systems, Advances in Intelligent Systems and Computing*, 264 (2014)
31. G. Deak, K. Curran, J. Condell, A survey of active and passive indoor localisation systems. *Comput. Commun.* **35**, 1939–1954 (2012)
32. R. Niu, A. Vempaty, P.K. Varshney, Received-signal-strength-based localization in wireless sensor networks. *Proc. IEEE* **106**(7), 1166–1182 (2018)
33. A. Abusara, M.S. Hassan, M.H. Ismail, RSS fingerprints dimensionality reduction in WLAN-based indoor positioning. in *Wireless Telecommunications Symposium (WTS)* (London, 2016), pp. 1–6
34. X. Hou, T. Arslan, J. GU, Indoor localization for Bluetooth low energy using wavelet and smoothing filter. in *International Conference on Localization and GNSS (ICL-GNSS)*, (Nottingham, 2017), pp. 1–6
35. P. Roy, M. Kundu, C. Chowdhury, Indoor Localization using Stable Set of Wireless Access Points Subject to Varying Granularity Levels, in *International Conference on Wireless Communications Signal Processing and Networking (WiSPNET)*. (Chennai, India, 2019), pp. 491–496
36. K. Wu, M. Yang, C. Ma, J. Yan, CSI-based wireless localization and activity recognition using support vector machine, in *IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC)* (Dalian, China, 2019), pp. 1–5
37. W. Liu, Q. Cheng, Z. Deng, X. Fu, X. Zheng, C-Map: hyper-resolution adaptive preprocessing system for CSI amplitude-based fingerprint localization. *IEEE Access* **7**, 135063–135075
38. G.G. Anagnostopoulos, A. Kalousis, A reproducible analysis of RSSI fingerprinting for outdoor localization using sigfox: preprocessing and hyperparameter tuning. in *International Conference on Indoor Positioning and Indoor Navigation (IPIN)* (Pisa, Italy, 2019), pp. 1–8
39. G. Anastasi et al., Energy conservation in wireless sensor networks: a survey. *J. Ad Hoc Netw.* **7**(3), 537–568 (2008)
40. J. Ogbekor, et al., *Energy Efficient Design Techniques in Next-Generation Wireless Communication Networks: Emerging Trends and Future Directions* (2020), 1-19
41. A. Soni, R. Upadhyay, A. Kumar, AvDR—based wireless secure key generation with colored noise for IoT. *Fluct. Noise Lett. World Sci.* **19**(2), 1–18
42. G. Margelis, et al., Physical layer secret-key generation with discreet cosine transform for the Internet of Things. in *IEEE International Conference on Communications (ICC)* (Paris, 2017), pp. 1–6
43. M. Xu, Research and design of data preprocessing of wireless sensor networks based on Multi-Agents. in *IEEE International Conference on Network Infrastructure and Digital Content* (Beijing, 2009), pp. 50–53
44. U.N. Nisha, A.M. Basha, Triangular fuzzy-based spectral clustering for energy-efficient routing in wireless sensor network. *J. Supercomput.* **76**, 4302–4327 (2020)



45. C. Jobanputra et al., Human activity recognition: a survey. *Procedia Comput. Sci.* **155**, 698–703 (2019)
46. V. Michalis, et al., A review of human activity recognition methods. *J. Frontiers Robot.* **AI.2** 28 (2015)
47. S. Sigg, U. Blanke, G. Tröster, The telepathic phone: Frictionless activity recognition from WiFi-RSSI, in *IEEE International Conference on Pervasive Computing and Communications (PerCom)* (Budapest, 2014), pp 148–155
48. M. Mukherjee, A.B. Bhattacharya, RSSI based indoor human activity recognition system. *J. Techno. Int. J. Health Eng. Manage. Sci.* **2**(5), 185–190 (2018)
49. A. Booranawong, N. Jindapetch, H. Saito, Adaptive filtering methods for RSSI signals in a device-free human detection and tracking system. *IEEE Syst. J.* **13**(3), 2998–3009 (2019)
50. W. Su, Wearable antennas for cross-body communication and human activity recognition. *IEEE Access* **8**, 58575–58584 (2020)
51. Y.T Wang, et al., Wireless signal identification in 230 MHz band based on interference cleaning and convolutional neural network. in *Proceedings of the 9th International Conference on Communication and Network Security (ICCNS 2019)* (Association for Computing Machinery, New York, NY, USA, 2019), pp. 133–136.
52. F. Kaleem, et al., A fuzzy preprocessing module for optimizing the access network selection in wireless networks. *J Adv. Fuzzy Syst.* Hindawi Publishing Corporation 1687–7101 (2013)
53. H.K. Boyapati, et al., Implementation of RSSI indexed look up table based AGC for improved dynamic range of DSSS based wireless RF transceivers. in *2nd International Conference on Next Generation Computing Technologies (NGCT)* (Dehradun, 2016), pp. 373–377
54. F. Carpi et al., RSSI-based Methods for LOS/NLOS Channel Identification in Indoor Scenarios, in *16th International Symposium on Wireless Communication Systems (ISWCS)*. (Oulu, Finland, 2019), pp. 171–175
55. H. Hojatian, et al., Unsupervised deep learning for massive MIMO hybrid beamforming. *J. Electr. Eng. and Syst. Sci. Signal Process.* arXiv.org, eess.2.

# Exploring IoT-Enabled Multi-Hazard Warning System for Disaster-Prone Areas



Vishal Menon, R. Arjun Rathya, Abhiram Prasad, Athira Gopinath, N. B. Sai Shibu, and G. Gayathri

**Abstract** Natural disaster in India has become a great challenge in the recent years. Each year the rates have been escalating affecting both the social as well as economic progress of the country. India's topographic/climatic and socio-economic features make the country most vulnerable to the devastating effects of such calamities. Hence, it is the need of the hour to come with a system capable of long term as well as quick prediction of disaster. This can be useful for early preparedness and developing well-planned mitigation/relief system which can reduce the effects of such disaster and can be also useful in channelizing the finding in a right way during calamities. The proposed system consists of modules for prediction of: Weather pattern, flood, earthquake, landslide, fire and gas leakage. The sensor node deployed at various disaster-prone areas transmits sensor data to a local aggregator that pre-process the data and relays to remote monitoring servers. The remote monitoring platform has algorithms for prediction of disaster as well as suggestions for quick response. Hence, the possibility of disaster can be predicted prior to the onset of these calamities.

**Keywords** Internet of things · Disaster management · Multi hazard warning · Simulation study

---

V. Menon · R. Arjun Rathya · A. Prasad  
Department of Computer Science & Engineering (CSE), Amrita Vishwa Vidyapeetham,  
Amritapuri, India

A. Gopinath · N. B. Sai Shibu (✉)  
Center for Wireless Networks & Applications (WNA), Amrita Vishwa Vidyapeetham,  
Amritapuri, India  
e-mail: [saishibunb@am.amrita.edu](mailto:saishibunb@am.amrita.edu)

G. Gayathri  
Department of Mechanical Engineering (ME), Amrita Vishwa Vidyapeetham, Amritapuri, India

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2021  
S. M. Thampi et al. (eds.), *Advances in Computing and Network Communications*,  
Lecture Notes in Electrical Engineering 735,  
[https://doi.org/10.1007/978-981-33-6977-1\\_31](https://doi.org/10.1007/978-981-33-6977-1_31)

405

# 1 Introduction

In recent years, India has become one of the countries, which are most likely to be hit by disaster and has made it one of the most vulnerable countries, this predicament has now become a common trend. Its unique climatic condition due to its geographical and topographical features makes India highly vulnerable to cyclones, avalanches, flood, lightning, earthquake and many other disasters. Its socio-economic vulnerability also makes it one of the most effected countries during such catastrophe. According to National Disaster Management Authority (NDMA), almost 40 million hectares of Indian commonwealth are exposed to devastating flood (which is 12% of total land), 68% of the nation's land is endangered by droughts, avalanches and landslide, along 5700 km of the 7516 coastline is prone to tsunamis and cyclones, and 58.6% of total landmass is prone to effects of earthquake [1]. Amid to its susceptible conditions, India, is placed among the top disaster-prone countries. According to the 2019 Global Risk Index annual report, India was the 14th most susceptible country in world. The report also stated that almost 2736 deaths in 2017 owing to disasters. Further, economic losses in India due to these disasters accounted for almost 13,789 million dollars, which is the 4th highest in the world. So, now the question stands whether to come up with a solution to cut out these losses or consider it as the plight of our nation. One way to reduce the economic impact of these frequent disasters is by developing a well-defined disaster prediction and mitigation system.

Lately, there has been an upshot in the field of IoT, and more and more systems are being connected via IoT. IoT has given a major contribution in the development of various fields of agriculture, water management, education, etc. In this project, we are trying to develop a disaster preparedness system using the applications of IoT. Our IoT-based system collects data with the help of interconnected WI-FI modules which consists of multiple sensors which are used to detect a disaster. This system would alert the person about the disaster and help the person to save his life. With the help of multiple Wi-Fi modules and sensors, any disaster can be alerted, and the data of various sensors can be stored in various cloud servers which can be analysed for predicting the forthcoming calamities.

The paper is organized as follows, and Sect. 2 briefly describes the state of the art and other similar works performed at various institutions. Section 3 explains the proposed system and the design. Section 4 describes the system architecture development and implementation with the required sensor modules and explains how the proposed system handle multiple disasters simultaneously. Section 5 provides details on workflow the mobile app used for the system. The paper is concluded in Sect. 6.

## 2 Related Work

This section summarizes few similar works performed by other researchers in this field. Technologies such as crowdsourcing, IoT, AI, ML are widely used for developing system that help in predicting or detecting natural calamities. There few system that are capable of informing the public about the natural disaster and help saving their life. Ramesh Guntha, Sethuraman. N. Rao and Avinash Shivdas provide a case study [2] on crowdsourced technology deployed during the Kerala 2018–2019 flood. Here, they have initially studied the various crowdsourced application in previous crises and have tried to develop an app called Amrita-Kripa which addresses the downsides of the previous crowdsource models, but the developed model had its own challenges due to duplicacy of data and misuses of the app during the supply face, we can tackle these problems by developing algorithms to check the authenticity of the requests.

Deekshit et al. [3] have developed a smart geophone sensor network which collects data from the deployed site and detects the possibility of seismic actions. The system consists of: Arduino, geophone sensor, Wi-fi/Bluetooth modules. In the proposed design, the deployment site is divided into various regions, and based on the regions, the sensors are fed with algorithms to detect different frequencies. Algorithms for removing noise and selecting relevant data were also designed in this paper. Tieman et al. [4] deal with the designing of a micro-electromechanical system-based accelerometer which can measure accelerations in three axis. Here, they are using the combination of two sensors: ADXL150 and ADXL250 and a microcontroller. While ADXL150 senses change in acceleration over one axis, the other sensor checks for the other two axes. The data is collected and handled with the help of a microcontroller. Here, the accelerometer described in the paper is a low-cost model built with commercial off the shelf components that allows onboard data addition and helps in connecting multiple interfaces.

Kusriyanto and Putra [5] have developed a weather station using the following: Arduino Mega 2560, FC-37 rain sensor, BMP-180 air pressure sensor, temperature and humidity sensor and a WI-FI module. The system uses weather prediction algorithms which are based on the barometric pressure. The accuracy of the sensor as well as the algorithms was compared to the data from PCE-THB 40 module, and the results were quite satisfactory. The paper Utz et al. [6] elaborate the working process of an accelerometer of MEMS based with CMOS of ultra-low noise which are integrated with a data-IC that contains a bulk micro machined sensing capacitor of high accuracy/precision which is cost efficient and focuses mainly on the high accuracy of measurement and low noise. The main parts of this construction are capacitive sensing element and Analog ROIC. The output noise is lesser than  $1 \mu * g / \sqrt{Hz}$ , so the accelerometer can be used for measuring seismic waves with high-precision.

Ardiansyah et al. in [7] use a fuzzy logic method to detect rainfall patterns and rainfall levels. The fuzzy logic is based on computing the degree of truth instead of denoting the output as HIGH(1) or LOW(0) value which is used in most of the current computer programs. The FL helps to solve the problems and gives the best decision

considering all the data. The system proposed in this paper consists of Arduino, temperature and rain sensors for measurement of temperature, humidity and prediction of weather. The five possible predictions shown in this paper are: Cloudy (cloudy weather), Sunny (dry weather), Drizzle (Chance of rain), Wet (raining) and Heavy rain. The paper also states the effectiveness of the fuzzy logic method in providing data for intensity of rainfall. Kunnath and Ramesh presents [8], a deployed wireless sensor network that is capable of detecting natural disasters like landslides. The authors present the paper with a deployed model that uses a nested wireless geophone network which is capable of sensing any kind of ground movement and follows a corresponding signal processing algorithm that helps in producing appropriate warnings. They emphasize the advantage of using a three axis geophone at each layer after their pilot model which they deployed in a colony in Munnar, which is well known for its landslides in Kerala. Based on the ideas gathered from these paper, we propose an IoT-enabled multi-hazard warning system in the paper, and the proposed system is described in the following section.

### 3 System Architecture and Design

The proposed system design as shown in Fig. 1 consists of various modules arranged in a hierarchical manner. This proves to be useful as the sensors are distributed thought the city, hence these modules can be divided for each zone/ward in a city, and this makes both analysis and relief planning easier. The proposed diagram also contains the information of handling and analysing sensor data. The various modules in the proposed system are explained below.

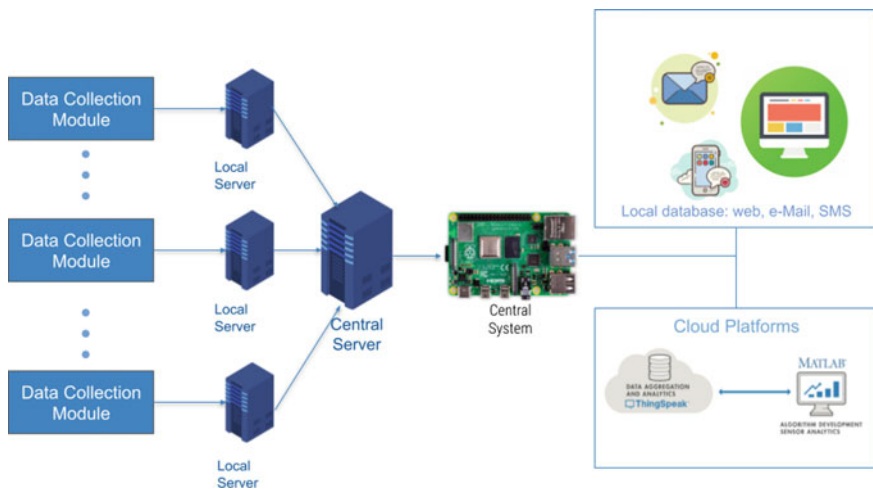


Fig. 1 Block diagram

### 3.1 Data Collection Module

Data captured from the sensors that are deployed in the field are collected by the data collection module. This module comprises point of use (PoU), echo and push modules as shown in Fig. 2.

**Point of Use Module** These modules are setup at the points of data collection. These modules are deployed with various sensors (depending on the disaster to be analysed: can be accelerometer (earthquake), ultrasonic (water level), etc.), and an Arduino board (with Bluetooth) is used for data collection. The sensors used in this module depend on the disaster to be predicted and can be divided into

- Weather analysis
- Flood detection module
- Earthquake detection
- Landslide detection
- Fire detection module
- Gas leakage detection.

**Echo Module** The PoU modules may be out of Wi-Fi range, and hence, a lot of useful data can be lost. As a solution for this, we have designed the echo modules, and these modules consist of Arduino board(with Bluetooth)/ alternatives. These modules act as a hop in the network. A number of such hops are deployed depending on the distance between PoU modules and region within the Wi-Fi range/network.

**Push Module** These act as receivers and also push the data from the PoU's to servers. This module consists of Arduino board (with Bluetooth+Wi-Fi)/ alternatives. The communication configuration of the PoU, echo and push modules is shown in Fig. 3.

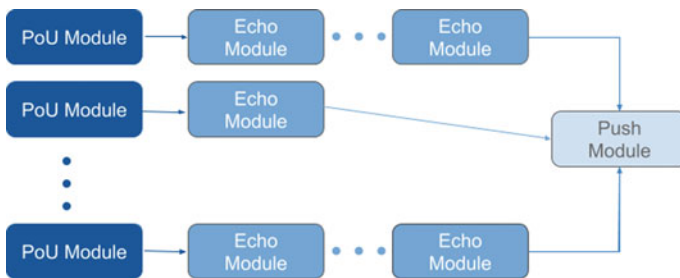
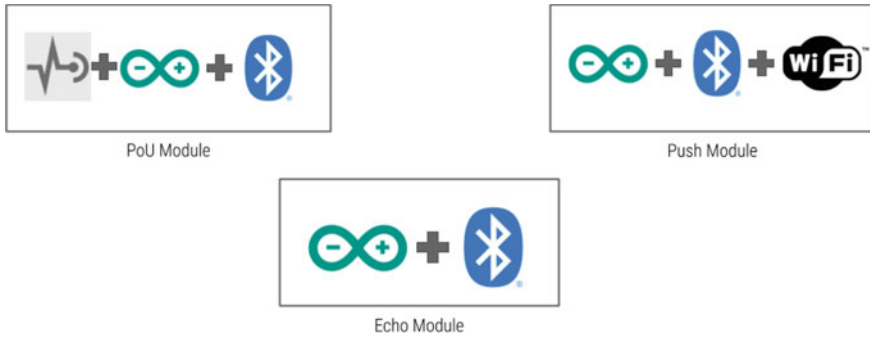


Fig. 2 Data collection module containing PoU, echo and push modules as sub-modules



**Fig. 3** Basic framework of PoU, echo and push modules

### 3.2 Data Server

The servers are arranged in an hierarchical way. The data from various push modules are sent to a local server, and similarly, there are many local servers divided throughout the city. The data from all these local servers are then uploaded to a central server.

### 3.3 Central System

The central system must be capable of processing a high amount of data, so for this purpose, we use single board computers like: Raspberry Pi, etc. A set of response algorithms are set in the central system which would alert the user in case of emergency. It should also follow the following tasks:

- Send quick response E-mail/SMS alert when any of the sensor data crosses the specified threshold values. (Quick response)
- Load different reports on a daily/monthly/yearly basis. (Quick response)
- Upload data to respective tables for visualization/prediction of forthcoming events. (long term)
- Upload data into the cloud. (long term)

Note: Here, in the central system, we are using Raspberry Pi because of its efficiency to stream and process larger data quickly when compared to other MCU's and single board computers.

### 3.4 Data Storage and Analysis

For this purpose, we are using both cloud and local database: Data collected from the sensors deployed in the area are stored in both local database and cloud storage. Data collected from the sensors will be used for data analysis and machine learning-based models in future applications.

1. Local Database: This database is used for quick reference and response actions. For this purpose, we have proposed an app. The user can get a feed on the sensor data risk status on a daily/monthly/yearly basis. The quick response algorithms are designed in such a way that it will send e-Mail/SMS to all the residents of the locality when the sensors data crosses the predefined threshold values.
2. Cloud Database: This part of the system is used for data visualization and prediction. For this purpose, we are using the ThingSpeak API. Here, we can store the sensor data on daily/monthly/yearly basis and can use MATLAB designed codes, and these codes used ML/CNN/classification algorithms for disaster prediction and analysis. The algorithms are designed to categorize the current situation on various ranges:
  - Green (Safe)
  - Yellow (possibility—the need for planning)
  - Orange (Onset of disaster/mild effects—well-developed relief plan ready to be deployed)
  - Red (Devastating effects—executing relief plans).

This module also helps in visualizing the data in various graphs and chart format. The data visualization/predictions from ThinSpeak are also incorporated in the mobile app which makes the access of information easier and available at a single platform rather than switching sites each time to see the ThingSpeak output.

## 4 Algorithm Development and Implementation

Once the system architecture is defined, the algorithms were developed and implemented in a simulation environment to validate the results. Algorithms were developed for weather prediction, flood estimation, earthquake, landslide, fire and gas leakage detection. The steps involved in algorithm development and implementation are described in this section. The common element in all the PoU modules is the Arduino board with Bluetooth, but here we are using an alternative board ESP32, which consists of an inbuilt Wi-Fi/Bluetooth module.



## 4.1 Weather Station Module

This module is designed for weather prediction on both daily as well as long-term basis. The system consists of an Arduino board (with Bluetooth)/alternatives and various sensors for weather prediction.

1. ESP32 board (alternative for Arduino + Bluetooth)
2. Rain Sensor (FC-37): The FC-37 sensor is used for sensing rainfall and also measuring the intensity of rain. The sensor consists of two parts: electronic and collector board. The basic idea implemented here is the varying resistance of the collector board depending on the amount of water on the surface of the board. When the surface is wet, the resistance value of the collector increases, hence the output voltage falls, and when the surface is dry, the resistance is low, and the output voltage is high. Hence, the intensity of the rain can be measured.
3. Air pressure BMP180: These are designed to measure the atmospheric and barometric pressure. Air has weight and owing to its weight, pressure is felt whenever there is air. The BMP180 can measure this pressure and can give a digital output for the measured value. The BMP180 also consists a good temperature sensor, which comes in handy as the temperature affects the pressure. Hence, temperature effects have to be compensated during pressure reading.
4. DHT22: The DHT22 is used for measuring temperature and humidity. The sensor consists of an inbuilt NTC used to measure temperature and an 8-bit microcontroller used to output the temperature and humidity values as serial data. The sensor is easy to interface with microcontrollers because of its factory calibration. The sensor can measure humidity from 0 to 100% with an accuracy of  $\pm 1\%$  and temperature from  $-40\text{ }^{\circ}\text{C}$  with an accuracy of  $1\text{ }^{\circ}\text{C}$ .
5. Algorithm: Two types of algorithms have to be designed as shown in Fig. 4:
  - (a) Quick prediction: This algorithm is used for predicting the forecast based on current data and for the respective day. The quick prediction modules are based on Zambretti algorithm. Zambretti algorithm uses values like absolute values of pressure, pressure trend, season, and wind direction (Note: wind direction and season have a small impact on output). Based on these values Zambretti algorithm calculates a forecast number 'Z'. Further, the forecast number 'Z' can be used to predict the weather forecast using the forecast table. The details of the calculations and forecast tables are provided in [9].
  - (b) Long-term prediction: Prediction of weather just for a particular day cannot be useful for disaster preparedness, and for this, we can use various ML/CNN/ classification algorithms using MATLAB (ThingSpeak) which can predict the chances of storm/cyclone/flood/forest fire, etc. The data from these modules can be used for the prediction of various disasters after considering data from other modules also.

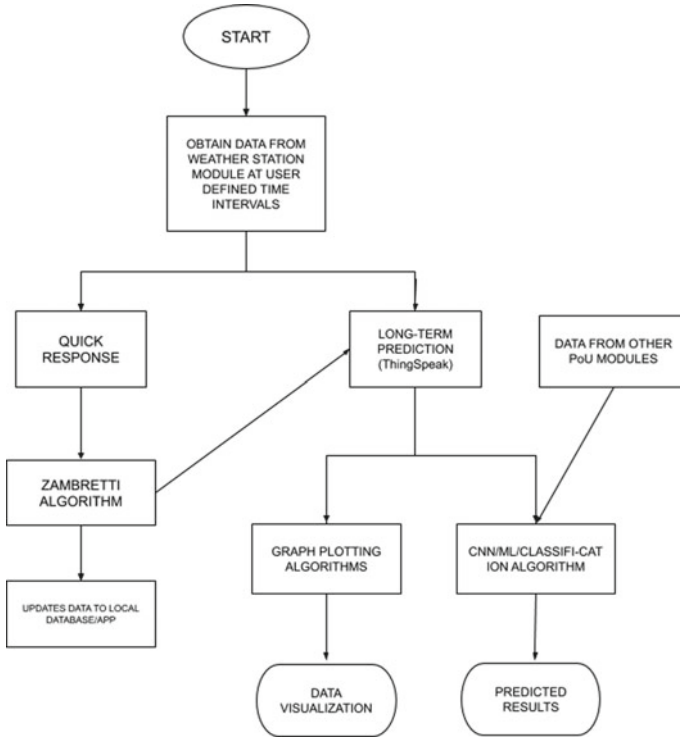


Fig. 4 Algorithm for weather prediction

### 4.2 Flood Detection/Prediction Module

This module is used for flood detection and prediction. The module uses a water level sensors (ultrasonic sensor) and microcontroller for detection, and a combination of this module and weather station module or other modules are used for prediction.

1. ESP32 board (alternative for Arduino + Bluetooth)
2. Ultrasonic sensor: Any ultrasonic sensor according to the manoeuvre can be used. The selection must be based on the range of the sensor. The maximum depth of the point of deployment must not exceed the range of the sensor.
3. Algorithm: Here also, the algorithm can be categorized into two as shown in Figs. 5 and 6:
  - (a) Quick detection: These algorithms are set for quick detection of a flood. Three-level threshold system can be used here. Based on the threshold values, the state of the crises can be divided into red, yellow, orange or green. The values might change from one point of deployment to another. The threshold can be set by the authorities.

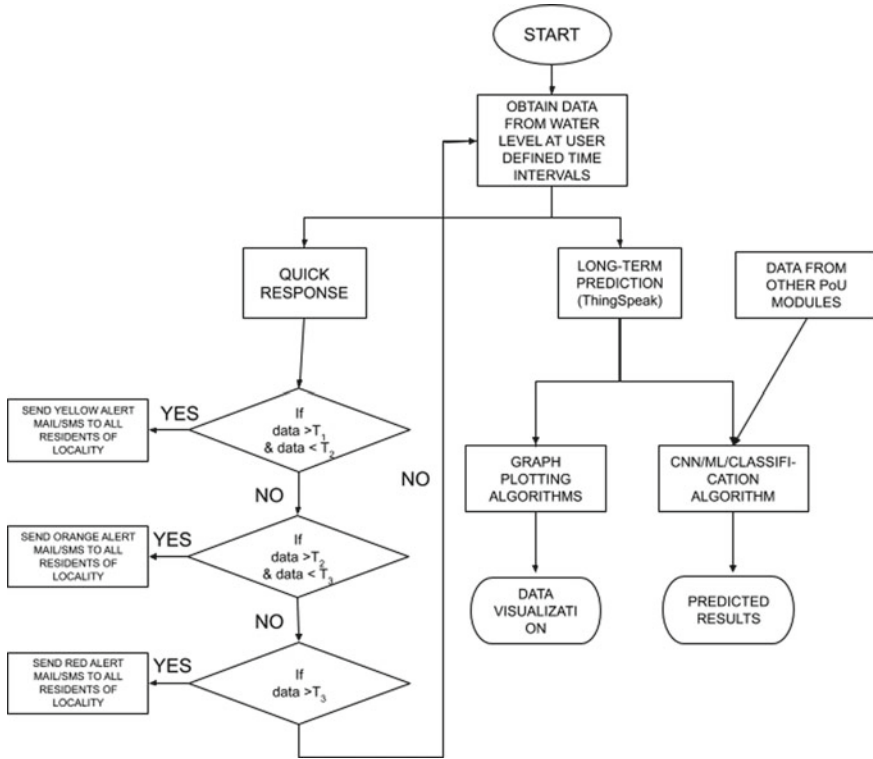


Fig. 5 Algorithm for flood detection/prediction, here in quick response branch  $T_1$ ,  $T_2$ ,  $T_3$  are the first, second and third thresholds

- (b) Long-term prediction: These algorithms work on cumulative data such as on the rain pattern data and water level over a period of time which is stored in the cloud platforms (ThingSpeak). Using various CNN/ML/classification algorithms designed using assessed data from recent events are used to predict the possibility of a flood. The algorithm will also classify the intensity of the disaster which can be further useful for deciding the level of preparedness needed. We will get the degree of preparedness shown in Sect. 3.4 as the output.

### 4.3 Earthquake Sensor Module

This module is used to detect vibrations of earthquake using a geophone sensor scattered at short intervals.

1. ESP32 board (alternative for Arduino + Bluetooth)

2. Geophone Sensor: A type of sensor which senses the movement of the ground and converts the movement into voltage. The sensor also sends the results to the access points, and the values will be recorded. The seismic waves measured from the base lines are further used for analysing the structure of earth.
3. Algorithm: Here, algorithm is classified into two categories as shown in Fig. 6:
  - (a) Quick Detection: When an earthquake takes place, the occurrence of S-waves (it is the second wave you feel during an earthquake and only moves through solid parts and stopped by liquids and gases) and the waves produced by the surface which causes more damage than the P-waves (compressional waves which travels through any kind of material like solid, liquid or gas) is detected using a geophone sensor. So, an early warning or alert regarding the earthquake can be given. The speed of P-wave is 5.6 and S-wave is 3.2 km/s. Here, for 7.51 km distance of each sensor from the secured area, one second of response time is added. Therefore, to provide more resolution for response time, we must assign more sensors at smaller intervals in an area. A low-cost geophone can detect sensitivity larger than 25 V/m/s and frequency more than 4.5 Hz. The algorithm for the earthquake is taken as we compare all the values of the waves which is then determined whether the wave is greater or lesser than the threshold value (TV), and if the value is greater, an alert message will be sent, and if the value is lesser, no message will be sent to the access point. The equation given below is the relation between response time and radial distance of the sensor used [10]:

$$t_{\text{response}} \times 7.51_{\text{Km/s}} = d_{\text{radial}} \quad (1)$$

where:

$$t_{\text{response}} = \text{Allowed Response Time in Seconds}$$

$$d_{\text{radial}} = \text{Radial Distance of Seismic Sensor from the Protected Area}$$

- (b) Long-term prediction: These algorithms work by analysing the data from the modules over a long period of time like over a month or year using CNN/ML/classification models which are designed using assessed data from previous events, and the algorithm provides prediction of the intensity of the catastrophe which can be useful for preparedness.

#### ***4.4 Landslide Sensor Module***

For a landslide detection, the smart geophone system is deployed in different regions to detect different frequencies at different time periods. Here, we have divided the landslide prone area into three layers namely top, middle and bottom. The top layer has the highest frequency of waves, so the geophones of high-frequency are assigned

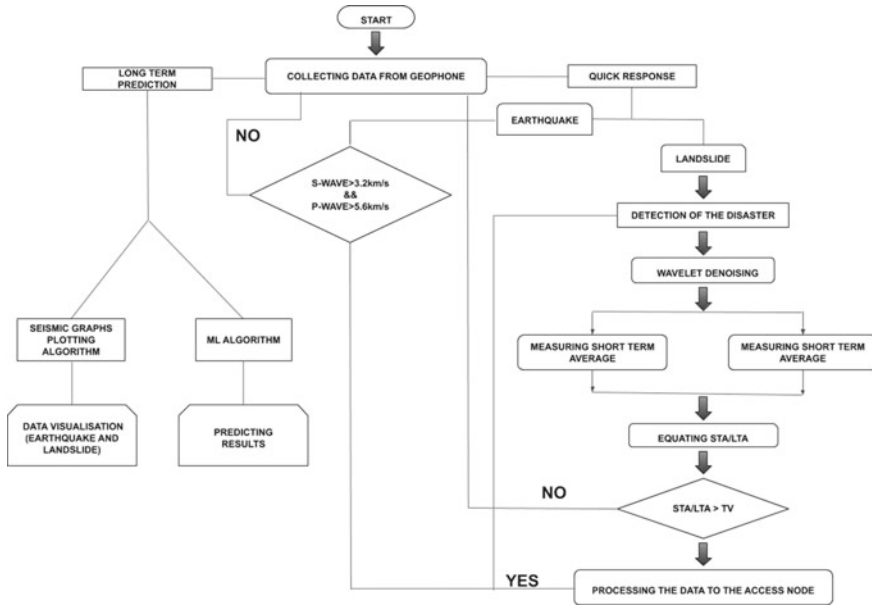


Fig. 6 Algorithm for earthquake and landslide

at the top layer, and the geophones of low-frequency are placed at the bottom area as they have the lowest frequency of waves. The frequency obtained from the leaf nodes of the sensor is delivered to the data centre, the data are analysed by the data centre, and according to the analysis of the data, a warning is sent to the corresponding user.

1. ESP32 board (alternative for Arduino + Bluetooth)
2. Geophone Sensor: A type of sensor which senses the movement of the ground and converts the movement into voltage. The sensor also sends the results to the access points, and the values will be recorded. The seismic waves measured from the base lines are further used for analysing the structure of earth.
3. Algorithm : Here, algorithm is classified into two categories Fig. 6:

(a) Quick Detection: The algorithm for detection of landslides includes wavelet denoising part, short-term average part (STA), long-term average part (LTA) and comparison part. The wavelet denoising technique abstracts desultory noise from the data accumulated by the geophone. Short-term average part (STA), long-term average part (LTA) of the signal is utilized to calculate the STA/LTA. When the value of STA/LTA is more preponderant than the threshold value, a digitalized signal of perpetual events from the geophone is sent to the data centre, and if the STA/LTA ratio is less than the threshold value, the digitalized signal from geophone is not sent to the data centre. And a landslide is recognized when all the geophones at an area has registered

an identical event or otherwise the noise is caused by a footstep or a moving object like a vehicle as mentioned in [11].

- (b) Long-term prediction: As all the above sensors, this algorithm also has the CNN/ML/classification models which are designed using the data of previous events, a continuous analysis is done for several months or years to get an accurate prediction of the catastrophe which gives an early warning, and the authorities will be able to provide all sorts of preparedness.

## 4.5 Fire and Gas Detection Module

The module consists of a combination of MQ2 gas sensor and flame sensor as its core along with an Arduino board (with Bluetooth)/alternatives. A fan is included in the module as an actuator. With the help of DC generators, these sensors are supplied with initial power supply to operate and are connected to the MCU. Here, we are using an alternative board ESP32, which consists of inbuilt WI-FI/Bluetooth modules.

1. ESP32 board (alternative for Arduino + Bluetooth)
2. MQ2 sensor is a gas as well as smoke detector. It has a concentration scope of 200 to 10,000 ppm which improves its ability to sense smoke and gases like LPG, hydrogen, carbon monoxide, alcohol, propane and even methane which prove to be highly inflammable gases.
3. The flame sensor senses the presence of flame or fires. With its trademark in quick responses due to an approximate of 600 detection angle, adjustable sensitivity and most impressively high photosensitivity, the module works best in detecting fires or any kind of light sources.
4. Algorithm: designed in two types as in Fig. 7:
  - (a) Quick prediction Algorithm: When a gas or smoke is detected by the MQ2 sensor, the read analog outs from the sensor moves to the MCU and results in HIGH MCU outputs depending on the threshold values.  $R_s$  is being resistance of sensor in gas concentration, and  $R_0$  is being resistance of sensor in fresh air,

$$R_s = [(V_{in} \times R_L) \div R_{out}] - R_L \quad (2)$$

In clean air

$$R_s \div R_0 = 9.8 \quad (3)$$

The ratio of resistance of sensor in gas concentration to that of fresh air is cross checked with the graph provided in the datasheet illustrating  $R_s/R_0$  against concentration (in ppm) to find the concentration of the detected gas [12]. This helps in detecting gas leakages, puts up the chances for fire spreads and even helps in predicting poisonous gas formations in industries when the composition of gas varies with different exhausts. The alert messages from

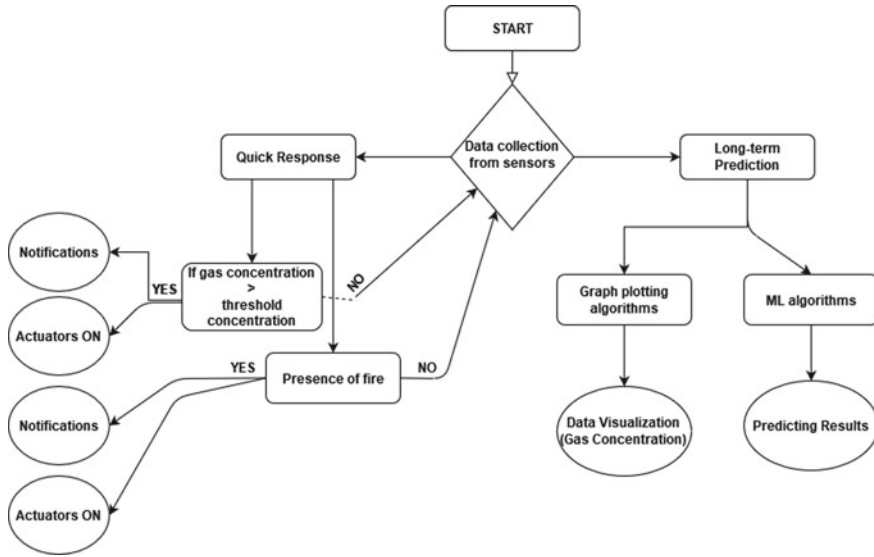


Fig. 7 Algorithm for fire and gas detection

the ESP32 module with the help of inbuilt GSM module are set through the cloud and make the crowd aware of the situation. Actuators like exhaust fans will be HIGH depending on the reaction time captured by the sensors, and it behaves accordingly and stops when there is no gas or smoke within the detected range. Similarly in the case for fire detection modules, the MCU puts up the actuators and alerts based on HIGH or LOW output values from the flame sensor which are the inputs for the MCU.

- (b) Long-term prediction algorithm: In case of disaster preparedness, quick prediction will not be suited for handling the situations. However, the regular updates of the sensed values from the modules to the ThingSpeak help in maintaining an appropriate vision about the forthcoming disaster on the basis of an acceptable analysed and visualized prediction with the help of machine learning algorithms. For example, gas leaks could be predicted on the basis of daily emission rates, and chances of fire spreads could be estimated on the basis of various gas concentrations in the atmosphere and temperature details from the weather station module.

### 4.6 Monitoring Multiple Disasters Simultaneously

In the section above, we have explained the working of the system considering separate modules for various disasters. Each of these modules can also interact with

each other sharing and receiving data, but these functions are limited to only few configurations. Most of the disaster modules work independently, but for estimations of few disasters, data from two or more modules might be required. One such example is flood detection: for flood detection, the water level data alone cannot be enough for future predictions, so we have to take both water level and weather data in account for future predictions, in such situations, the data from both the modules are used in the ML predictors, so in such cases, sharing happens. The model and idea given here is purely theoretical, and real-time issues with such configuration might be in-cognizant.

## 5 Mobile Application Workflow and Design

When the user initially joins the app, it requests the user to add an E-mail for logging in, and this E-mail can be used during an emergency situation to send the alert message. Once the initial set-up is done, the app uses GPS to obtain the user's current location. The visualization and data provided to the user will be based on the location. The app consists of the following module:

1. Disaster Modules (Home): This module consists of further modules for weather, flood, landslide, earthquake, fire and gas poisoning. Within each module, visualization and predictions are available.
2. Timeline and analysis of recent events: Here, a report on the recent catastrophic event, the effected rates and relief details are given. This module also features news feed related to disasters.
3. Map: It contains a detailed map marked with effected regions and safe regions around our locality when a disaster occurs.
4. A detailed Government guidelines for mitigation and preparedness to be taken in case of an emergency.

Once the user selects any one of these disaster modules in the home section, the user enters a window containing the following sub-modules:

- Prediction: If selected, the app shows a MATLAB-based prediction done in ThingSpeak. Here, the predicted probability rate, mortality/economic effects of the forthcoming possible disaster and possible damage analysis are given.
- Visualization: If selected, the app shows a MATLAB-based visualization done in ThingSpeak based on the data collected from PoU's (sensor modules).
- Map: A detailed map of the locality/city marked red, green, yellow or orange depending on the severity of the region.

A detailed workflow of the app is given in Figs. 8 and 9.





Fig. 8 Workflow of the app

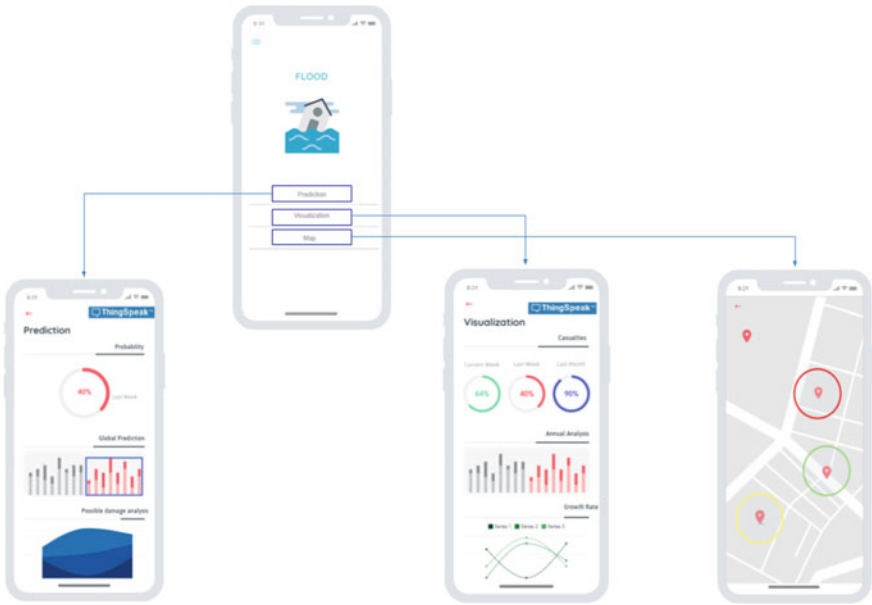


Fig. 9 Workflow individual modules

## 6 Conclusion and Future Work

In this paper, we proposed IoT-enabled multi-hazard warning system to help in predicting and detecting various natural hazards like landslides, earthquake, flood and gas leakage. The proposed system architecture was discussed and implemented using the ESP32 system on chip. The system was simulated to detect quick responses and could also predict long-term hazards. This system can be useful as it can be used to analyse the forthcoming calamities proving the citizens as well as the authorities time for developing a well-constructed disaster mitigation plan for fighting the situation. Early predictions can also be useful for minimizing expenditure and also helps the authorities to channelize the funding in a more effective manner.

The proposed system is limited to only prediction phase/identification of the disaster, and the system can be upgraded by adding relief planning and distribution phase using various crowdsourced technologies which makes it a pre-eminent system for managing the crises during hazardous events. Sampling of useful data from unwanted data is still a challenge as effective algorithms for identification of useful data are not available. Proper identification can be useful for reducing both the power consumption as well as the memory requirement of the PoU (sensor modules) and can also make the prediction algorithms more effective and fast as it reduces data duplicity. The proposed system also lacks real-time experimentation [amid to the pandemic situation], and hence, there are many drawbacks which are oblivious. Hence, there is still room for improvement.

**Acknowledgements** We express our deep gratitude to our beloved Chancellor and world-renowned humanitarian leader Shri. (Dr) Mata Amritanandamayi Devi (AMMA), for inspiration and motivation. We would like to thank the staff and faculty members of the department for providing immense support and suggestions to improve this paper.

## References

1. P.A.G. Yashobanta Parida, India is not prepared for natural disasters. <https://www.thehindubusinessline.com/opinion/india-is-not-prepared-for-natural-disasters/article30463153.ece#> (2020)
2. A.S. Ramesh Guntha, S.N. Rao, Lessons learned from deploying crowdsourced technology for disaster relief during Kerala floods, in *Procedia Computer Science 171 (2020) 2410–2419, Third International Conference on Computing and Network Communications (CoCoNet'19)*. <https://doi.org/10.1016/j.procs.2020.04.313>
3. V.N. Deekshit, M.V. Ramesh, P.K. Indukala, G.J. Nair, Smart geophone sensor network for effective detection of landslide induced geophone signals, in *2016 International Conference on Communication and Signal Processing (ICCSP)* (IEEE, New York, 2016), pp. 1565–1569
4. J. Tieman, J. Schmalzel, R. Krchnavek, Design of a mems-based, 3-axis accelerometer smart sensor, in *2nd ISA/IEEE Sensors for Industry Conference*, pp. 19–23 (2002)
5. M. Kusriyanto, A.A. Putra, Weather station design using iot platform based on Arduino mega, in *International Symposium on Electronics and Smart Devices (ISESD)*, vol. 2018, pp. 1–4 (2018)

6. A. Utz, C. Walk, A. Stanitzki, M. Mokhtari, M. Kraft, R. Kokozinski, A high precision mems based capacitive accelerometer for seismic measurements. *IEEE SENSORS* **2017**, 1–3 (2017)
7. A.Y. Ardiansyah, R. Sarno, O. Giandi, Rain detection system for estimate weather level using Mamdani fuzzy inference system, in *International Conference on Information and Communications Technology (ICOIACT)*, vol. 2018, pp. 848–854 (2018)
8. A.T. Kunnath, M.V. Ramesh, Integrating geophone network to real-time wireless sensor network system for landslide detection, in *First International Conference on Sensor Device Technologies and Applications*, vol. 2010, pp. 167–171 (2010)
9. T.M. Mengazi, C.R. Rada, Weather forecast based on pressure, temperature and humidity only (for implementation in arduino). <https://earthscience.stackexchange.com/questions/16366/weather-forecast-based-on-pressure-temperature-and-humidity-only-for-implement> (1968)
10. J. Santos, A.N. Catapang, E.D. Reyta, Understanding the fundamentals of earthquake signal sensing networks, in *2019, Analog Dialouge by Analog Devices*, vol. 53, pp. 1–11 (2019)
11. V.N. Deekshit, M.V. Ramesh, P.K. Indukala, G.J. Nair, Smart geophone sensor network for effective detection of landslide induced geophone signals, in *International Conference on Communication and Signal Processing, April 6–8, 2016* (IEEE, New York, 2016), pp. 1565–1569
12. Pololu, Mq-2 semiconductor sensor for combustible gas. <https://www.pololu.com/file/0J309/MQ2.pdf>

# IOT Based Smart and Secure Surveillance System Using Video Summarization



M. Surya Priya, D. Diana Josephine, and P. Abinaya

**Abstract** Nowadays, security is important for every commercial property to prevent robberies and thefts and to ensure secure safe business operations. In CCTV (Closed-circuit television) systems, the data is non-intelligently recorded which produces huge volumes. It makes it difficult to search for the desired content from the big data. It is found from the literature that limited work is done in the field of a secure surveillance system using real-time videos. Therefore, there is a need for video summarization, classification (action recognition), and encryption. This paper aims to make decisions about abnormal events like suspicious activity detection in surveillance applications incorporating the above-said techniques. This IoT (Internet of Things) based smart secure surveillance system allows for reduced storage of unwanted data and helps to protect the confidential data to be sent to the user by cryptographic methods.

**Keywords** Image encryption · Feature extraction · Cryptography and video summarization

## 1 Introduction

Usage of smart phones and other IoT assisted devices are increasing rapidly and these results in expanding the collection rate of images exponentially. IoT buzz is everywhere and it refers to connection of people and things at any time, in any place, with anyone and anything, using any network and any service [1]. Thus, efficient mechanisms for managing, searching, retrieving and indexing of image big data repositories are needed. In addition to that, digital images play an important role in different areas of interest such as commercial, military, and medical applications. Recently, the problem of secure dissemination of secret information over the internet is growing fast. Hence security of a specific data has become a major concern in recent years for which researchers have presented numerous techniques. Security systems help individuals to feel safe. Secure surveillance system is needed in order to prevent

---

M. Surya Priya · D. Diana Josephine (✉) · P. Abinaya  
Department of ECE, CIT, Coimbatore, India  
e-mail: [dianajosephine@cit.edu.in](mailto:dianajosephine@cit.edu.in)

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2021  
S. M. Thampi et al. (eds.), *Advances in Computing and Network Communications*,  
Lecture Notes in Electrical Engineering 735,  
[https://doi.org/10.1007/978-981-33-6977-1\\_32](https://doi.org/10.1007/978-981-33-6977-1_32)

423

robberies, theft and also to ensure secure and safe business operations. The intelligent surveillance system eventually minimizes the required storage space and makes the system cost-effective. Cryptography is one of the solutions for information security and is considered as one essential aspect for secure communication over the public network Internet.

In this paper, video summarization method is used to generate a short summary of the content of a longer video document by selecting the most informative frames and transmitting the informative frames via IoT. With the real time data, dataset has been created. Then feature extraction has to be done to find the occultation difference of image between normal and abnormal image. While transmitting an abnormal image to the user, an encryption algorithm is used so that an attacker cannot collect any useful information and followed by that decryption is done by the user.

The proposed method provides a complete end-end framework and reduces the storage and transmission cost. The encrypted frame is sent to the user via Gmail and he/she can decrypt it with the help of secret keys.

The rest of the paper is organised as follows: Sect. 2 comprises of context and references related to smart surveillance system. Section 3 gives the proposed system framework. Section 4 describes the most important system model of the proposed system and the obtained results. Section 5 discusses various observations during the course of the work and Sect. 5—The conclusion and future work.

## 2 Related Works

Many research works have been done under video summarization techniques and a few is discussed under this section. This literature study has been done to explore the different techniques/algorithms employed under different stages, their benefits and to find the better one that exactly suit the need.

*“Secure Surveillance Framework for IoT systems using Probabilistic Image Encryption”* [2]. This paper proposes a secure surveillance framework for IoT systems by intelligent integration of video summarization and image encryption. An efficient video summarization method is used to extract the informative frames. When an event is detected from key-frames, an alert is sent to the concerned authority autonomously. A new fast probabilistic and lightweight encryption algorithm is proposed and used for the encryption of key frame. This algorithm reduces the bandwidth, storage, transmission cost and the time required for analysts to browse large volumes of surveillance data and make decisions about abnormal events such as suspicious activity detection and fire detection in surveillance applications with better security.

*“Depreciating Motivation and Empirical Security Analysis of Chaos-Based Image and Video Encryption”* [3]. This paper aims to show the two main motivations for preferring chaos-based image encryption over classical strong cryptographic encryption, regarding computational effort and security benefits. Several statistical tests were done and it is shown that the chaos-based encryption algorithm has better

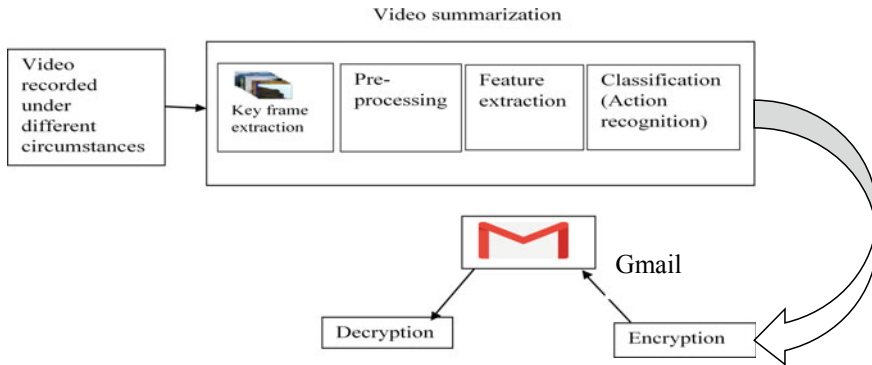
computational effort and security benefits than classical algorithm. Chaos-based ciphers require less computing resources and reduce the complexity of work than classical. But running time of chaos-based image encryption algorithm is not faster than classical strong cryptographic encryption algorithm.

*“Multi-View Surveillance Video Summarization via Joint Embedding and Sparse Optimization”* [4]. This paper proposes a Multi-view surveillance video summarization using joint embedding and sparse optimization technique. The objective function is to capture the multi view correlations via an embedding, which helps in extracting a diverse set of representatives and use a norm to model the scarcity while selecting representative shots for the summary. Therefore, to join and optimize both of the objectives, and to solve non-smooth and non-convex objective an efficient alternating algorithm based on half-quadratic minimization is proposed with convergence analysis. A key advantage of the proposed approach with respect to the state-of-the-art is that it can summarize multi-view videos without assuming any prior correspondences or alignment between them. Experiments on several multi-view datasets were demonstrated and it out performs the state-of-the-art methods. But it takes more processing time to generate good quality video summary.

*“Action Recognition in Video Sequences using Deep Bi-Directional LSTM With CNN Features”* [5]. This paper proposes an action recognition method in video sequencing using convolution neural network (CNN) and deep bidirectional LSTM (DB-LSTM) network. In this, deep features are extracted from every sixth frame of the videos, which helps to reduce the redundancy and complexity. The sequential information among frame features are learnt using DB-LSTM network, where multiple layers are stacked together in both forward pass and backward pass of DB-LSTM to increase its depth. This method is capable of learning long term sequences and can process lengthy videos by analyzing features for a certain time. Action recognition using this method is better comparing to state-of-the-art methods. It also captures all the tiny changes effectively.

### 3 Proposed Framework

A sample video is recorded and it is converted into key frames and stored in JPEG format. The extracted key frames will undergo preprocessing state in which the key frames are resized, gray scaled and sharpened [6]. The sharpened image is filtered using median filter to remove noise effectively. For feature extraction Blob Analysis algorithm is used. Blob analysis is image processing’s most basic method for analyzing the shape features of an object such as the presence, number, area, position, length, and direction of lumps. After feature extraction, classification of the frames is done. For classification, nearest neighborhood classifier is used. It is a supervised machine learning model that uses classification algorithms for two-group classification problems. By using nearest neighborhood classifier, the abnormal frame is detected from the recorded video. For safety and security purpose, the detected abnormal frame is encrypted using AES (Advanced Encryption Standard) algorithm.

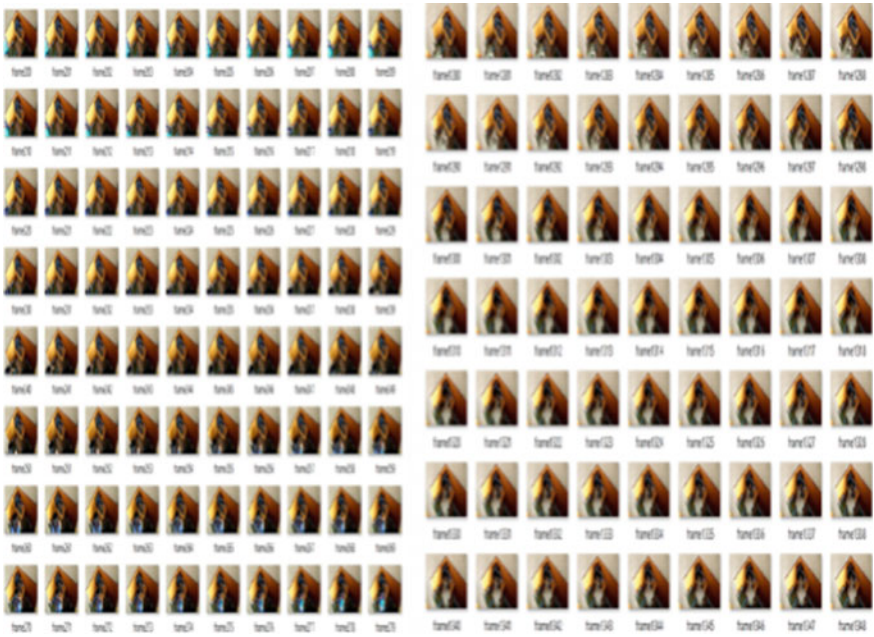


**Fig. 1** Block diagram of the proposed model of IoT based smart and secure surveillance system

The encrypted frame is send to the concerned user via Gmail. The user will decrypt the frame and proceed with further actions See Fig. 1 for the block diagram of the proposed model that clearly depicts the above said workflow.

## 4 System Model and Results

Video is recorded under different circumstances say classroom, office, ATM cell, home, etc., using mobile camera or with a digital camera. This recorded video is used as an input for further processing. For our work, video is recorded in a classroom environment for duration of 22 s using a mobile camera. As detection of abnormal data is the main focus, the video is taken such that it has an abnormal content in it. The proposed framework includes key frame extraction, pre-processing, feature extraction and classification which all fall under a single technique called the video summarization [2, 4, 7]. Video summarization and action recognition plays an important role in secure surveillance system which helps in efficient storage, quick browsing and retrieval of large collection of video data without losing important aspects. The recorded video contains a total of 645 frames. Firstly, the frames are extracted from the recorded video (see Fig. 2). The dimensions, resolutions and bit depth of the frame are  $584 \times 322$ , 96 dpi and 24 respectively. Refer Tables 1 and 2 for the recorded video and extracted frame metrics. Next, the frames are taken for pre-processing. Pre-processing is a process which improves the image by suppressing unwanted distortions and enhancing key image features for further processing. It includes image resizing, filtering, segmentation and detection of edges [4, 5, 7, 8]. Uncertainties such as random image noise, partial volume effects and intensity non uniformity artifact (INU) are introduced into the image, due to the manual recording of the video. This results in smooth and slowly varying change in image pixel values and lead to information loss, SNR reduction and degradation in edge and finer details of image. To overcome the above effects, median filter is used. In a continuous image,



**Fig. 2** Extracted frames

**Table 1** Recorded video properties

Video size	00.00.26
No. of frames extracted from the videos	645
Bits per pixel	24
Frame rate	24.4681 frames/s
Frame height	322
Video format	RGB24
Frame width	584

**Table 2** Extracted frame properties

Dimensions	584 × 322
Width	584 pixels
Height	322 pixels
Horizontal resolution	96 dpi
Vertical resolution	96 dpi
Bit depth	24
No. of training frames	150



**Fig. 3** Detection of edge from a sample frame



a sharp intensity transition between neighboring pixels is considered as an edge. Edge corresponds to fast change in gray level and thus, considered as high frequency information. Thus edge detection is the process of separation of high frequency information is. For the converted frames, edges are detected for more accurate identification (see Fig. 3).

Next is the feature extraction process. A feature is defined as the “interest” part of an image. The desirable property for a feature detector is repeatability; i.e. whether or not the same feature will be detected in different images of the same scene and trained.

Step edges, lines and junctions usually convey the most relevant information of an image; hence it is important to detect them in a reliable way. In our model the image features are extracted using Blob analysis method and then abnormal key frame is detected by comparing it with collected database (see Fig. 5) using nearest neighborhood classifier [9]. The Blob analysis computes statistics for connected regions in a binary image before which the image has to be resized, sharpened and filtered. The algorithm includes blob extraction and blob classification. The blob extraction is to separate blobs or object in a binary image. In blob classification, the necessary image is extracted and the rest will be omitted. This method refers to analyzing binary images by first extracting the blobs, then representing them compactly and finally classifying the type of each blob. This also uses Foreground detection using Gaussian mixture models. The “ForegroundDetector” system object compares a color or grayscale video frame to a background model to determine whether individual pixels are part of the background or the foreground. It then computes a foreground mask. By using background subtraction, foreground objects in an image taken from a stationary camera is detected. The “strel object” represents a flat morphological structuring element, which is an essential part of morphological dilation and erosion operations, is used to highlight the extracted feature (see Fig. 4).

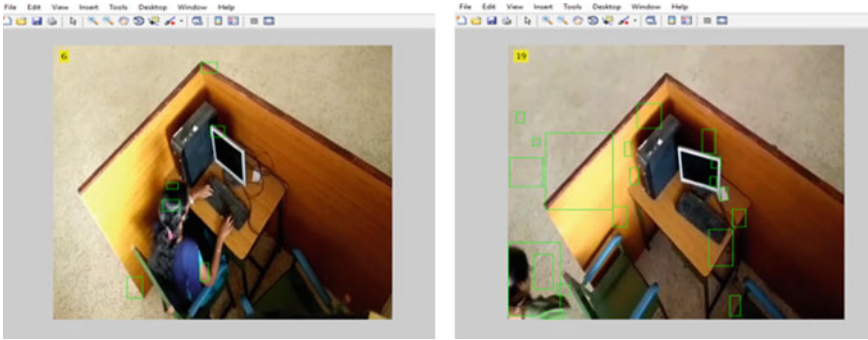


Fig. 4 Sample feature extracted images



Fig. 5 Sample collected database

The extracted key features are represented by square/rectangle box in green. Feature extraction from two sample frames are shown in Fig. 4. The accuracy of results in abnormal detection is highly dependent on the versatility of the database.

The created database shown in Fig. 5 consists of different types of normal and abnormal activities taken under different circumstances and at different time. Nearly 1500 real time images are captured and trained.

Classification refers to comparing database, which contains predefined patterns with the detected object to classify into proper category. Image classification analyzes the numerical properties of various image features and organizes data into categories. Classification algorithms typically employ two phases of processing which

includes *training* and *testing*. In classification, the abnormal image is separated by comparing each and every frame with the collected database. For classification, K-nearest neighbor (K-nn) classifier is used [7]. It is one of the introductory supervised classifier, which every data science learner should be aware of. The simple version of the K-nearest neighbor classifier algorithms is to predict the target label by finding the nearest neighbor class. The closest class will be identified using the distance measures like Euclidean distance. The K-nn pseudo code employed in the system model is explained below

Let  $(X_i, C_i)$  where  $i = 1, 2, \dots, n$  be data points.  $X_i$  denotes feature values and  $C_i$  denotes labels for  $X_i$  for each  $i$ . Assuming the number of classes as ‘ $c$ ’  $C_i \in \{1, 2, 3, \dots, c\}$  for all values of  $i$ . Let  $x$  be a point for which label is not known, and we would like to find the label class using k-nearest neighbor algorithms.

1. Calculate “ $d(x, x_i)$ ”  $i = 1, 2, \dots, n$ ; where  $d$  denotes the Euclidean distance between the points.
2. Arrange the calculated  $n$  Euclidean distances in non-decreasing order.
3. Let  $k$  be a +ve integer, take the first  $k$  distances from this sorted list.
4. Find those  $k$ -points corresponding to these  $k$ -distances.
5. Let  $k_i$  denotes the number of points belonging to the  $i$ th class among  $k$  points i.e.  $k \geq 0$
6. If  $k_i > k_j \forall i \neq j$  then put  $x$  in class  $i$ .

Using cross-validation technique, K-nn algorithm with different values of  $K$  is tested. Cross-validation is a statistical technique which involves partitioning the data into subsets, training the data on a subset and use the other subset to evaluate the model’s performance with better accuracy. It depends on individual cases, at times best process is to run through each possible value of  $k$  and test the result. From the 645 extracted frames our model could correctly detect/classify the abnormal frame (see Fig. 6). Only a single abnormal frame is extracted which will be then sent to the user to know about the exact condition prevailing in the respective environment.

The last stage is the cryptography. In cryptography, encryption is done which refers to coding the message or information in such a way that only authorized parties can access it and those who are not authorized cannot. The intended abnormal image (see Fig. 6) is encrypted using an encryption algorithm—a cipher—generating ciphers that can be read only if decrypted. In our model, Advanced Encryption Standard (AES) algorithm is used. The encrypted image (see Fig. 7) is then sent to the intended user via Gmail. It can be seen from Fig. 7 that no information could be got from it and it is believed to withstand any type of Brute attacks. To enhance user’s integrity credentials like user name and password are included to unlock the abnormal frame contents (see Fig. 8). This method is more simple and good in security when compared with other encryption algorithms. The AES algorithm is used in some applications that require fast processing such as smart cards, cellular phones and image-video encryption. The same is used to decrypt the image (see Fig. 9). However, a central consideration for any cryptographic system is its susceptibility to possible attacks against the encryption algorithm such as statistical attack, differential attack, and various brute attacks. The following are the stops involved in encryption-email-decryption.



Fig. 6 Classified abnormal image

Fig. 7 Encrypted frame

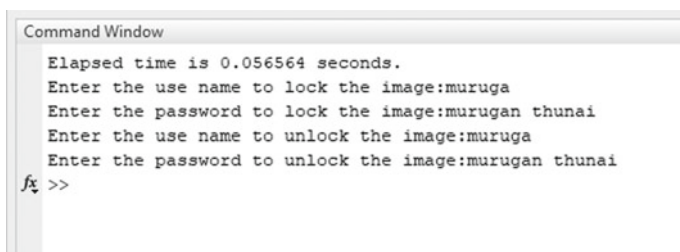
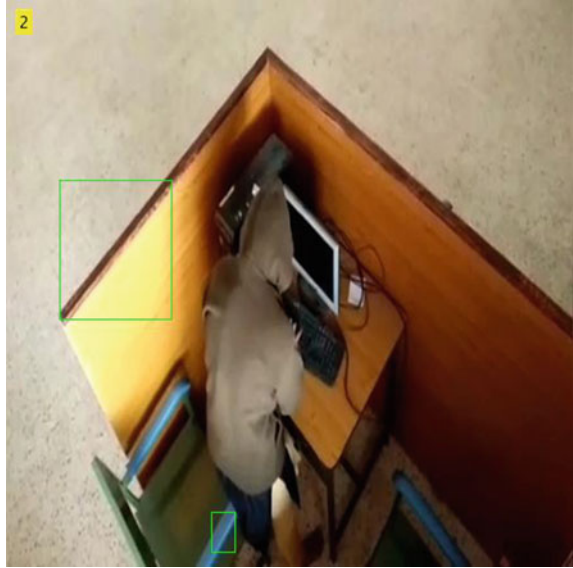


Fig. 8 User verification

**Fig. 9** Decrypted abnormal frame



1. Encrypt the abnormal frame using 'ImageEncryptionGui'. This GUI does the Image Encryption of any RGB, Gray image of different formats.
2. Assign the sender's email address (an actual, real e-mail address).
3. Get the domain to check whether emails can be directly sent from the recording device or not (since some domains have firewall restrictions)
4. Get the email of the intended recipient and setup email and SMTP server address.
5. Send the message/information via email to recipient with the file attached (the abnormal encrypted frame) (see Fig. 10).
6. The recipient can unlock the message/information after the verification process and takes necessary action based on the severe of the abnormality.
7. Alert sender that the message, with attached file, has been sent (see Fig. 11).

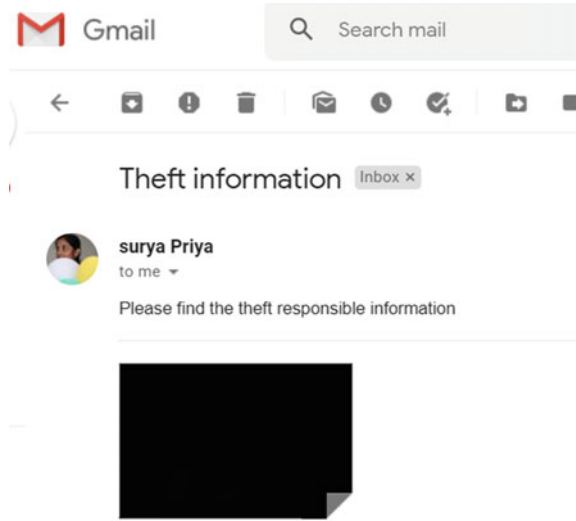
## 5 Observations

Simulation is carried out in Matlab for detecting the abnormal activities in a classroom environment. Refer Table 3 for the technical specifications of the captured video and their parameter settings.

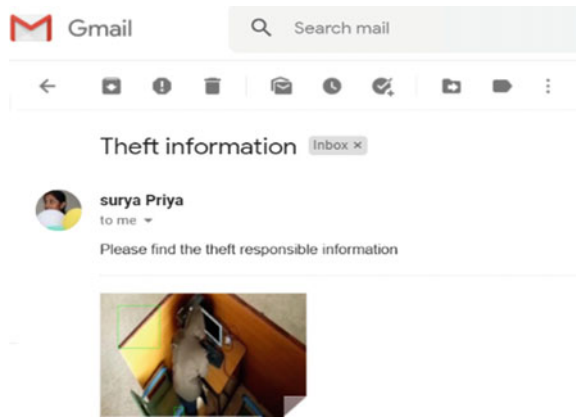
It is seen that the simulated model works fine for the recorded video. The frame extraction and edged detections were perfect. This is because the recorded video is of very short duration and we found it difficult to handle huge sized volumes.

We had difficulties with classification using K-nn algorithm in fixing k, calculating n and Euclidean distance. The execution for this particular section is quite high and it is in the order of 15 s. But encryption is done faster as it deals with a single

**Fig. 10** Mailed ciphered image



**Fig. 11** Mailed decrypted image



**Table 3** Technical specifications of the captured video

Parameters	Settings
Recorded video properties	Rear and front video size: 16:9 FHD 1920 × 1080
Recorded audio properties	Bit rate: 129 Kbps Channels: 2 (stereo) Audio sample rate: 44.100 KHz
Distance between the camera and the circumstance	5, 6 m
Time required to send e-mail	20 s (depends on Internet speed and Web site load) for 50 Mbps LAN

**Table 4** Comparison with the existing system

Reference paper no	Existing method	Proposed method
2	It reduces the bandwidth, storage and transmission cost and also provides better security. But in this work the encrypted frame is not sent to the user	The proposed method also reduces the storage and transmission cost. The encrypted frame is sent to the user via Gmail and he/she can decrypt it with the help of secret keys
3	Chaos-Based Image Encryption is used. The running time of the process is not faster. A comparison between chaos-based image encryption method and classical cryptographic encryption method is made	No algorithm comparison is made. The elapsed time of our model is 0.056564 min which is three times smaller compared to the existing
4	There is a problem in developing an end-end efficient framework	Our model has a complete end-end framework
1	It took approximately 1.12 s for processing a 1-s video clip	Our model took approximately 0.75 s for processing a 1-s video clip

abnormal frame. E-mail transfer was very challenging as it deals with SMTP setup and there were firewall restriction too. User credentials verification was just right and unintended users were prevented from access. Overall the model works well, as proper algorithm and techniques were in use and at exact place.

## 6 Conclusion and Future Work

This paper has proposed a IoT based smart and secure surveillance system using video summarization. Due to recent advances in IoT-assisted networks for surveillance in industrial environments, a significant amount of redundant video data is generated. Its transmission, analysis, and management are difficult and challenging, requiring image prioritization. In this work, an efficient video summarization method is first used to extract the informative frames from the recorded video data which can be used for abnormal event detection. For classification, the K-nearest neighbor rule is quite simple, but computationally very intensive. Since the extracted key frames are important for further analysis, their privacy and security is of paramount importance during transmission. Therefore, we employed advanced encryption standard (AES) for the encryption of key frames prior to transmission, considering the memory and processing requirements of constrained devices. Our algorithm is secure and simple because an attacker cannot collect any useful information about a key frame from its corresponding ciphered image. The experimental results verify the efficiency, security, and robustness of our algorithm compared to other image encryption methods. The current work mainly focuses to detect the abnormal activity from the recorded

videos. Further research can be conducted to implement this method in real time and further to improve the security measures in other specific areas.

## References

1. A. Ghasempour, Internet of things in smart grid: architecture, applications, services, key technologies, and challenges. *Invent J* **4**(1), 1–12 (2019)
2. K. Muhammad, R. Hamza, J. Ahmad, J. Lloret, H. Wang, S.W. Baik, Secure surveillance framework for IoT systems using probabilistic image encryption. *IEEE Trans. Ind. Inform.* 1551–3203 (2017)
3. M. Preishuber, T. Hutter, S. Katzenbeisser, A. Uhl, Depreciating motivation and empirical security analysis of chaos-based image and video encryption. *IEEE Trans. Inform. Forensics Sec* **13**, 1556–6013 (2018)
4. R. Panda, A.K.R. Chowdhury, Multi-view surveillance video summarization via joint embedding and sparse optimization. *IEEE Trans. Multimed.* 1879–2188 (2017)
5. A. Ullah, J. Ahmad, K. Muhammad, M. Sajjad, S.W. Baik, Action recognition in video sequences using deep bi-directional LSTM with CNN features. *IEEE Trans. Visual Surv. Biometr.* **06**, 2169–3536 (2018)
6. K. Muhammad, M. Sajjad, M.Y. Lee, S.W. Baik, Efficient visual attention driven framework for key frames extraction from hysteroscopy videos. *Elsevier Biomed. Image Process. Control* 1746–8094 (2016)
7. A.S. Murugan, K.S. Devi, A. Sivaranjani, P. Srinivasan, A study on various methods used for video summarization and moving object detection for video surveillance applications. *Springer Multimed. Tools Appl.* 23273–23290 (2018)
8. S. Giraddi, S. Gadwal, J. Pujari, Abnormality detection in retinal images using haar wavelet and first order features, in *2nd International Conference on Applied and Theoretical Computing and Communication Technology*, pp. 657–661 (2016)
9. A. Niranjil Kumar, C. Sureshkumar, Abnormal crowd detection and tracking in surveillance video sequences. *Int. J. Adv. Res. Comput. Commun. Eng.* 7935–7939 (2014)



# An Efficient and Innovative IoT-Based Intelligent Real-Time Staff Assessment Wearable



J. Anudeep, Shriram K. Vasudevan, G. Kowshik, Chennuru Vineeth, and Prashant R. Nair

**Abstract** The traditional method for assessing the performance of the workers in industrial workspaces is by measuring their inertial movements. Presently in industries at the workspaces, every 20–30 workers will have an in-charge to monitor their work. Although workers are being assessed by those in-charges, there are still many mishaps happening in the industrial workspaces. Workers with poor work performance are earning the wages that are same as that of a worker with good performance. These expenditures may seem to be minimal in number, but recent statistics reveal that this affects the company's total productivity in a disastrous way. Some studies state that on an average, companies are losing \$3,156 on the workers due to their idleness (Duffy J, Productivity report|bridging research and practice on personal productivity). Forbes magazines revealed that 31% of the workers are roughly wasting 1 h of time per day at their work apart from their allotted leisure times (Wasting time at work: the epidemic continues—the fobs report). Many companies in the UK with industrial workspaces claim that they lost 15.4 billion dollars annually only due to worker illness and their maintenance (Number of workplace injury and work-related ill health cases|Page 8—HSE, UK report). However, unfortunately, besides having distinct inertial measurement unit (IMU) systems in place, companies are facing many difficulties in identifying and estimating the worker performances and the health of the worker. Therefore, we have developed a system for overcoming

---

C. Vineeth · P. R. Nair

Department of Computer Science and Engineering, Amrita School of Engineering, Coimbatore, Amrita Vishwa Vidyapeetham, Coimbatore, India

S. K. Vasudevan (✉)

K. Ramakrishnan College of Technology, Samayapuram, Trichy, India

e-mail: [shriramkv@gmail.com](mailto:shriramkv@gmail.com)

J. Anudeep

Department of Electronics and Communication Engineering, Amrita School of Engineering, Coimbatore, Amrita Vishwa Vidyapeetham, Coimbatore, India

G. Kowshik

Department of Electronics and Instrumentation Engineering, Amrita School of Engineering, Coimbatore, Amrita Vishwa Vidyapeetham, Coimbatore, India

the difficulties faced by using these IMUs using the power of IoT and Android application.

**Keywords** Wearable IMUs · Worker health · Performance analysis · Wage equity · Mishaps at workspaces

## 1 Introduction

Most of the companies with industrial workspaces will have a major requirement of systems that can monitor their workers efficiently and make an analysis of their performances. A few years ago when the IMU systems were not so developed, the work hours of the workers were calculated using paper punch systems, which was a poor and non-reliable method. In many companies, workers suddenly fall sick without prior intimation to the management, which causes sudden productivity shortage for the industry. We need a much better system that will not only assess the performance of the worker but also can predict when the worker might fall sick as the health of their workmen is the prior factor for the industry's development. In order to make this job easier and for it to be done in a smarter way, we propose a new system, which has been developed using wearable technology and IoT.

Why are we depending on wearable technology? To answer this question, let us first know what the different types of mishaps that are happening at the workspaces are.

1. Whether the workers are working all the time?
2. Are they working in the workspace (near machinery)?
3. Are all the workers doing the work or if they are getting proxies by bluffing the admin?
4. Is the worker all fit or facing any afflictions regarding health?

Monitoring each worker in all the above-mentioned aspects is a highly difficult task in real-time scenarios. In order to know whether the worker is working near the machinery or not, it can only be known through visionary inputs like cameras, etc. It is also a very difficult task for the admin to monitor each worker at the same time. Therefore, by considering all the above-mentioned problems, a wearable band is developed which is equipped with sensors to monitor health and do a performance analysis of workers in real time with minimal errors.

## 2 Problem Statement

To design and build an IoT-based IMU as a wearable for worker assessment while also monitoring the worker's mental and physical health. Also, to provide a comprehensive report and analysis to the supervisor through an interactive Android application.

### 3 Existing Solutions

These are some of the existing inertial measurement systems, which are used to assess the performance of the workers at the workspaces of industries. Some of the existing methods are cited below.

The system proposed by Nicolas Vignaisa, Markus makes use of sensors planted on different parts of the body that tracks the different body movements, and the system uses a visual feedback system using a head-mounted camera. Besides the efficiency of the system about reporting the performance of the worker, the computation time for the whole process is very high and complex. The other major drawback of this system is that it makes use of a various number of sensors all over the body, which may be irksome to the worker and would not motivate the worker to wear the set-up all day during work. As the number of sensors per worker increases, the cost of the system for measuring the performance of one worker also increases drastically which makes the system unsuitable for the market or company to be implemented on large scale [4].

Paul Lukowicz, Jamie A. Ward have proposed a system to assess the worker performance that makes use of just accelerometers and microphones placed at different places on the body. The accelerometer is used for detecting the movements of the body, and all the audio signals acquired from these microphones are correlated in order to assess the environment of the person. Microphones at different places on the body can modulate a very high level of noise due to the environment. Privacy of the worker about his conversation is lost. It will become a hectic task for the admin to actually know where the worker is working by assessing these microphones. However, whereas our system can eliminate these types of problems to admin by having different types of sensors integrated at one place with an Android app [5].

### 4 Proposed Architecture

The proposed system consists of different sensors embedded on a wearable band, which uploads the data to cloud for remote access, by the Android application. The custom-made Android application gets the sensor data from the cloud and uses a machine learning algorithm for assessing the performance of the worker, predicting when the worker might fall sick in the near future. In the Android application, the admin can see the live data as well as the complete health and performance analysis of the worker. Figure 1 shows the workflow of the proposed system, and Fig. 2 shows the complete architecture.

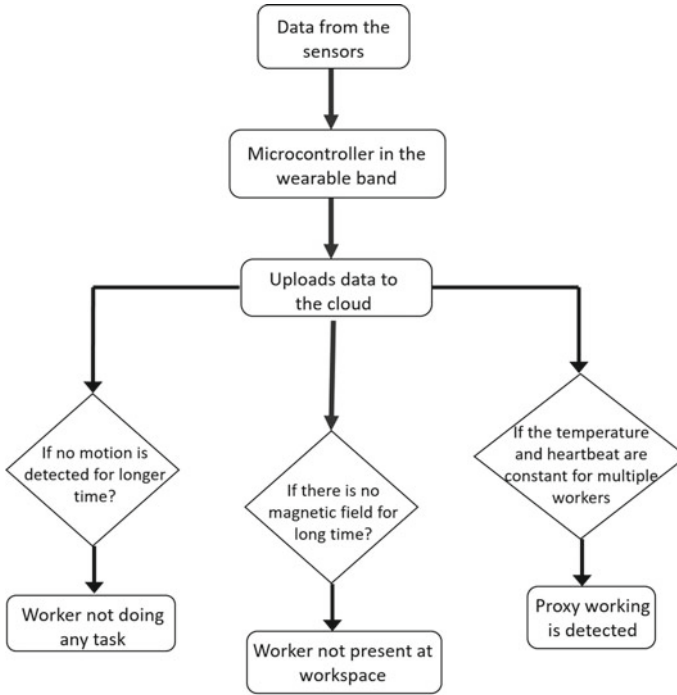


Fig. 1 Workflow of the proposed system

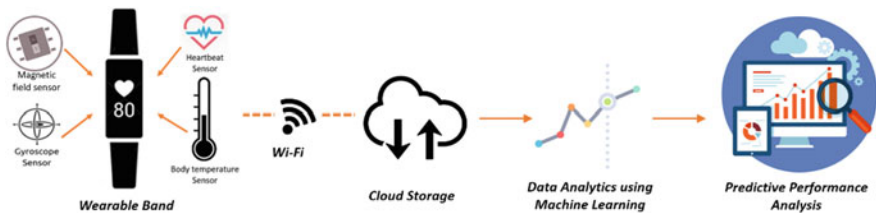


Fig. 2 Proposed architecture of the product

### 4.1 Wearable Band

A lightweight wearable band is equipped with sensors like gyro, accelerometer, magnetic field sensor, heart rate, temperature sensor. The gyro, accelerometer sensors are used to calculate the movement of the worker. Magnetic field sensor calculates the amount of magnetic field the worker is exposed to. Pulse rate sensor calculates the heart rate of the worker while doing various kinds of work and the stress level of the worker. The temperature sensor is used to monitor the body temperature of the worker. All the data from these sensors is sent to the microcontroller present inside

the band where the processing of the parameters is done. All this data is sent to the cloud for training a machine learning model, which can predict the performance of the worker. All the processed data will be visualized through an Android application for easy assessment of the admin.

### 4.1.1 Gyro, Accelerometer Sensor

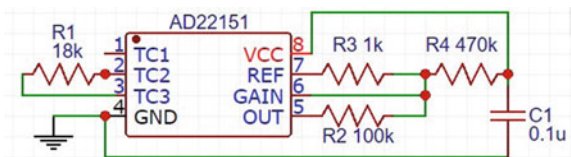
The accelerometer (GY-61) sensor and gyro (MPU-6050) sensor are attached to the proposed wearable in order to monitor the movements of the worker. These sensors measure the rate of change of angle of the hand, thereby detecting the motion of the worker. These sensors calculate the distance moved by the worker, which act as a pedometer and give us a view of whether the worker is moving or sitting idle at one place without doing the work. However, one cannot rely on a single parameter to conclude that the worker is moving his hands just to do the work, so we need some more useful parameters to assess the performance of the worker. In order to achieve this, we have integrated some other sensors explained in later sections.

### 4.1.2 Magnetic Field Sensor

Magnetic field (AD22151) sensor inside the band measures the intensity of the magnetic field around the worker which is used to know whether the worker is near the machinery or not. The sensor used is built based on the miniaturized electromechanical system (MEMS) technology. Due to the miniaturized fabrication of inductor coils and other requisites, it is very easy to integrate the sensor inside the wearable, and it does not occupy much space inside the wearable band.

Figure 3 shows the connected circuit diagram of the magnetic field sensor with required resistors and capacitors. For making the magnetic field sensor (AD22151) work, resistors R1, R2, R3, R4 and capacitor C1 should be connected as shown in Fig. 3. The 5 V DC is connected to eighth pin, and GND is connected to the fourth pin of the sensor. The output of the sensor is taken from the fifth pin. By varying R3, R4 resistor values, the sensitivity of the sensor can be varied.

**Fig. 3** Connection diagram of magnetic field sensor



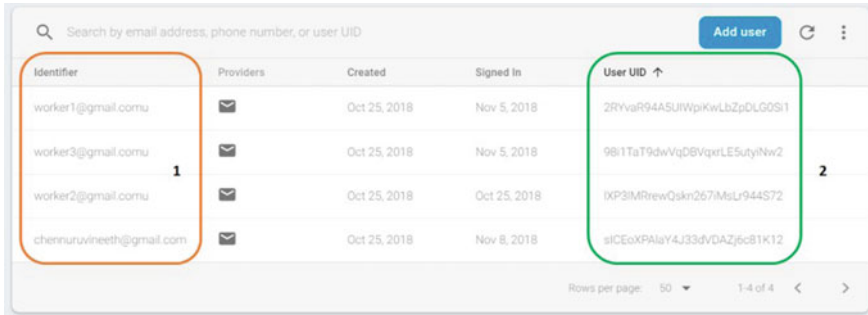


Fig. 4 Firebase authentication service with admin login

### 4.1.3 Heart Rate Sensor

Heart rate sensor is used for monitoring the heartbeat rate of the worker. These sensors work on the principle called photo plethysmography, which is a typical process of counting the signal pulses that are reflected from the body when exposed to the light from an LED. These sensor measurements not only serve as a parameter to monitor the health of the workers but also help the admin in detecting the work proxies by the workers, which are explained in Sect. 4.5.3. The data from this sensor will be constantly fed to a machine learning algorithm in order to predict the health afflictions.

### 4.1.4 Temperature Sensor

Heart rate sensor alone cannot help us to monitor the health or detect the proxies by workers. So in order to acquire accurate data regarding the health condition of the worker, we have used a LM35 temperature sensor to get the body temperature of the worker.

## 4.2 *Firestore*

These days cloud services are holding great importance in fields that require global access to data and render admirable results with higher speeds. For company prototypes, which cannot have their own servers, or clouds, there are some free cloud service providers with adaptable interfaces. Some of them are Firebase, Adafruit, Digital Oceans, Thingspeak, Ubidots, etc. For the prototype purpose, we have also used a free cloud service, and when our system is adapted by industries, they can adopt their own servers or clouds. For the Android applications, Firebase is the best-chosen cloud service, which renders efficient results with higher speed and does not

comprise on complex communication protocols. We have used two of its services namely authentication and database storage, which are explained in detail in the coming sections.

### 4.2.1 Authentication

In general, every industry assigns a certain number of workers to one admin. We have designed an Android app which is explained in Sect. 4.4 where the admin needs to login with username and password to see assigned worker performance analysis. For authentication of admin in the Android app, we have used Firebase authentication service. In the proposed solution, we have created four admins with four different email IDs as shown in Fig. 8, and they have respective passwords, which are not shown in Fig. 4. We cannot see the password in the authentication service page, but can change it, if we forget it with the help of user UID, which is shown in Fig. 4.

### 4.2.2 Database Storage

For evaluating the worker performance in the Android app, we require different sensor data, which are mentioned in Sect. 4.1. These sensor data is uploaded to the Firebase cloud using NodeMCU whose complete process is explained in Sect. 4.3.

Figure 5 (Orange Oval) shows the available databases namely worker-6ef67, user worker. The sensor data is uploaded in worker-6ef67. For accessing or uploading

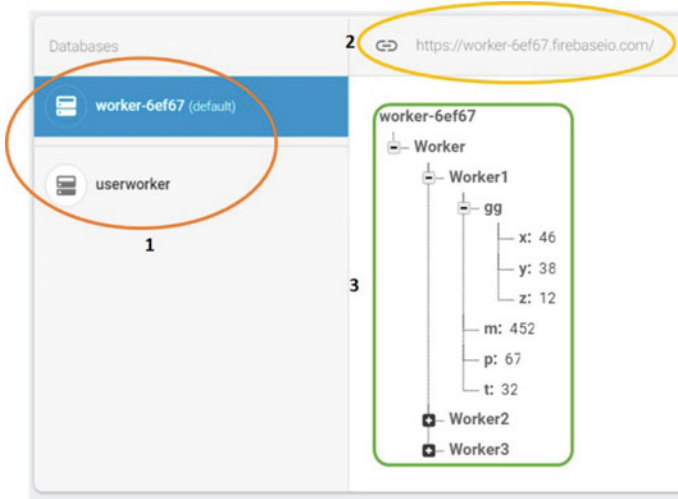


Fig. 5 Database with worker band sensor data

data to the database, the below link is used which is also shown in Fig. 5 (Yellow Oval).

The data is stored in the database in a format similar to JSON format; it has branches under which each parameter is stored. Figure 9 shows the branches under which worker band sensor data is stored. The parent branch is worker, under which, we have three workers namely Worker1, Worker2 and Worker3. In Fig. 9, we have shown the expanded data of only Worker1. It has the following sub-branches.

- gg Gyro Angle Data.
- m Magnetic Intensity exposed by the Worker.
- p Pulse Rate of the Worker.
- t Body Temperature of the Worker.

Further, gyro angle data (gg) has three more branches as gyro, accelerometer get us x, y, z coordinates. To store each angular data separately, three more sub-branches namely x, y, z under gg have been created which is shown in Fig. 5 (Green Square).

### 4.3 NodeMCU (ESP8266)

The cheapest way of accessing or uploading data to the cloud is by using the Generic ESP8266 module, but it has only two GPIO pins which make it very difficult to upload multiple sensors' data. NodeMCU is an open source design board which has a similar type of WiFi module (ESP-12F) inbuilt on it but with more number of GPIO pins. As the sensors to be interfaced require more than two GPIO pins, we have used NodeMCU for uploading data to the Firebase cloud. These modules are low power consuming, light in weight and work on popular protocols like MQTT for data transmission. The process of uploading data to the branches in the database involves multiple steps.

Uploading data to Firebase cloud using NodeMCU involves multiple steps. Initially, NodeMCU must be connected to your network which is entered in Lines 3, 4. Next connection between Firebase must be established for that Firebase hostname which is mentioned in Sect. 4.2.2 must be entered in Line 5. Next, enter Firebase database secrets which can be found at Project Setting > SERVICE ACCOUNTS in Line 6 which is similar to a password for accessing the Firebase account by the NodeMCU for uploading data. In the void loop() function, various data obtained from the sensors using the code mentioned in Sect. 4.1 are uploaded using the setInt to their respective branch.

### 4.4 Android Application

Visualization of worker performance, health analysis can be done both in Web application and in mobile application. In our proposed solution, we have built a mobile



application for visualization of worker performance analysis using Android Studio Platform. In future for the development of the product, we can also build Web application for both Windows and IOS. Integrating all the computation process in the band itself increases the cost as powerful microcontrollers have to be used, increases the size of the band, and it also draws a huge amount of power. In order to avoid these disadvantages, we have included all the computation process in the Android app itself, and the worker assessment band is used for uploading data alone. The interface of the Android application is explained below.

Figure 6 shows the login page of the Android application where both admin and user can log in with their respective credentials. The left side of Fig. 6 shows admin login credentials, and right side shows Worker 3 login credentials being entered in the app. If the admin logs in to the app, he will be redirected to Fig. 7 to access different worker performance and health analysis. If the worker logs in to the app, he does not have the provision to access other worker performance analysis, but he can access his own performance and health analysis by directly getting redirected to Fig. 9.

When the admin logs in with correct credentials, he/she will get redirected to Fig. 7. Here, they can select workers whose performance analysis is to be evaluated. If he selects any worker among the displayed list, he will be redirected to Fig. 9. He also has the provision to add workers under him by clicking the add user button at

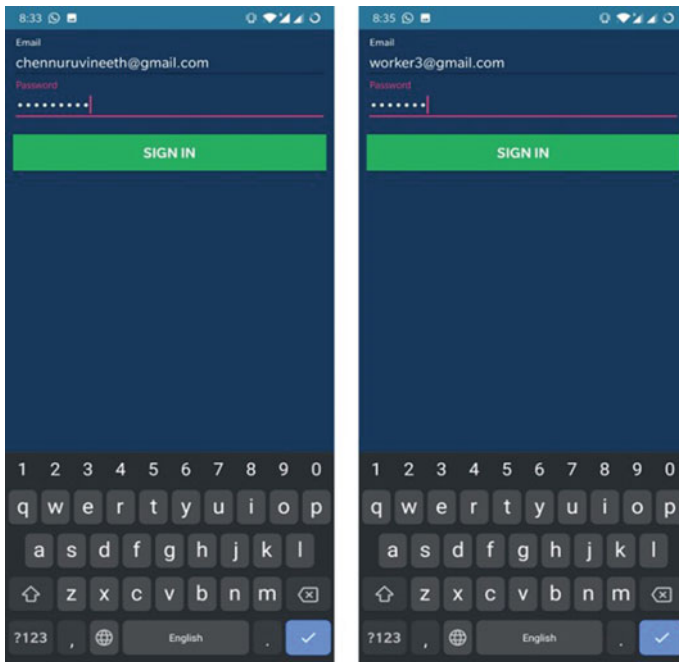
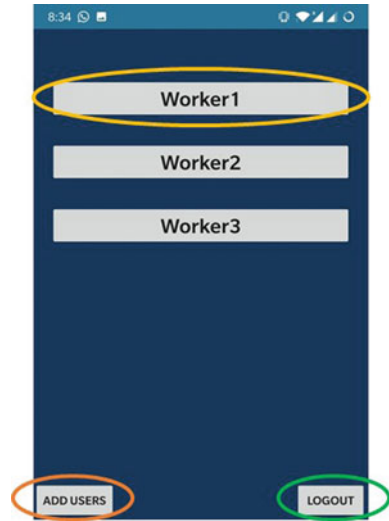


Fig. 6 Login page of Android application

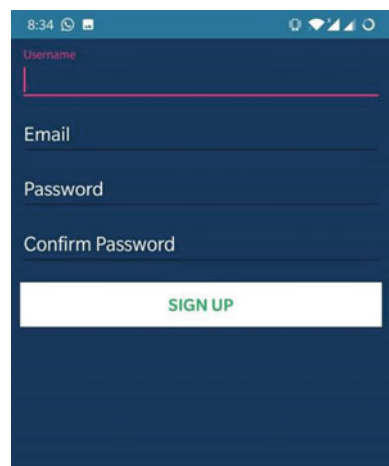
**Fig. 7** Worker selection page for the admin



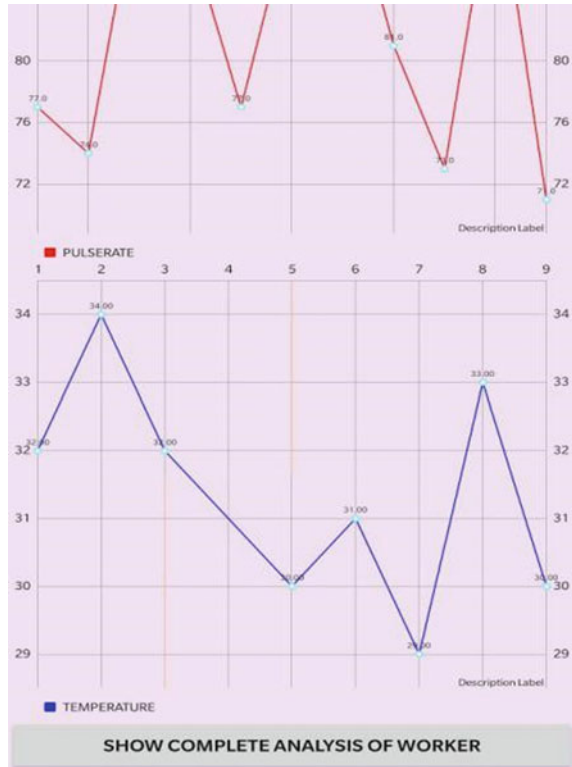
bottom left of Fig. 7. By clicking logout button at bottom right of the page, he will be going to Fig. 6 page.

If the admin clicks the add user button, he will be going to Fig. 8 where he can enter the worker credentials and add the worker under his monitoring. He has to give worker username, email ID, password for signing up a worker. The worker sensor band data continuously is uploaded to Firebase cloud. For admins or workers to have a detailed analysis of their parameters at any point of time, we have provided a page, which is shown in Fig. 9 where they can have a live graphical visualization of sensor data inside worker band. For knowing the overall performance analysis for

**Fig. 8** Signup page



**Fig. 9** Live graphical visualization



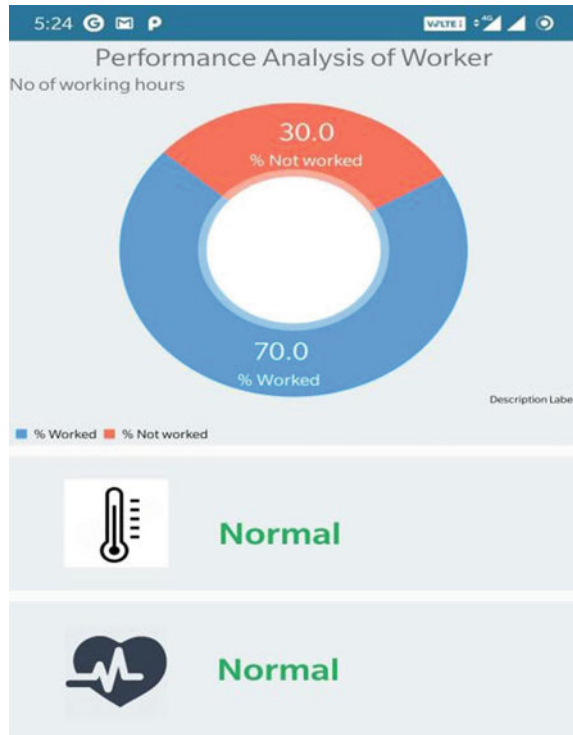
the complete day or particular period of time the user can click the button shown at bottom of Fig. 9, then he will be redirected to Fig. 10.

By clicking the “SHOW COMPLETE ANALYSIS OF WORKER” button shown in Fig. 9, he will be going to Fig. 10 where he can know his total working time in the mentioned working hours near the machinery, his overall heart rate analysis: whether he is in critical, stressed or normal state, etc., and his overall body temperature analysis.

### 4.5 Solution for the Mishaps in Workspaces

Now by knowing the working of each sensor, we can answer the questions mentioned in Sect. 1 of this paper. The way in which problems faced by using the existing available IMU’s following traditional methods for worker assessment are overcome is explained below.

**Fig. 10** Overall performance, health analysis



#### 4.5.1 Whether the Workers are Working All the Time?

To track the number of working hours of the workers, industries employ a person like workspace in-charges or biometric attendance systems are most commonly used. Our proposed system makes use of 6-axis gyro and accelerometer sensors, which can calculate the distance, moved by the workers accurately, the time worker is actively moving in the workspace without sitting idle. This can greatly reduce the need for extra men for monitoring which reduces the net expenditure of the companies.

#### 4.5.2 Are They Working in the Workspace (Near Machinery)?

Most of the workers leave their workspace and go around to some other place. In these scenarios, although the worker will be moving, he is not working with the work which he is assigned to, with available IMU's, this problem is not solved as it gives wrong analysis for the admin even though the worker is not working near the machinery. In order to overcome these type of problems, we use a magnetic field sensor which calculates the intensity of the magnetic field that the worker is exposed to which is generated by the huge machinery present at the workspaces. By getting

the magnetic intensity data, we can easily comment if the worker is working near the machinery or not.

### **4.5.3 Getting proxies by wearing multiple bands by one worker?**

It is a difficult task for the admins to find out whether all the workers are doing the work or whether they are bluffing their admins by giving their bands to the other workers at the workspace and escaping from work. To solve this problem, our algorithm keeps track of the body temperature and heartbeat rate values. If the band is given to another worker, then the body temperature and heartbeat values of both workers remain constant for more time, which enables us to detect the proxies working at the workspaces.

### **4.5.4 Is the Worker All Right or Facing any Afflictions Regarding Health?**

Our system not only monitors the performance of the worker but also takes care of the worker's health condition. By using body temperature, heartbeat rate values, it will instruct the user to take rest for some time as he is stressed or working for a long time. If his health conditions are abnormal, it directly intimates the admin about it, thereby instructing him to take the worker to the hospital. Our Android app logs the complete health data of the worker for predicting when the worker might fall sick in future and he may take leave.

## **5 Experimental Set-up**

The proposed system is experimentally tested, and the representation shown in Fig. 11 shows the connection diagram of the proposed system.

The gyro, LM35, magnetic intensity, heartbeat sensor are connected to NodeMCU for uploading to Firebase cloud. The MPU6050 is a sensor that is integrated with both MEMS accelerometer and gyroscope sensors. In order to read data simultaneously from these sensors, respective I2C address is accessed, and the data is transmitted over the I2C bus which is a two-wire communication line that is mostly used for large-scale sensor integration. In Fig. 11, magnetic field intensity sensor connections with all the biasing resistors and capacitors are not shown as it makes the connection diagram clumsy and not understandable. So, required resistors and capacitors connections to magnetic intensity sensor have to be connected as shown in Fig. 5 in order to make it work.

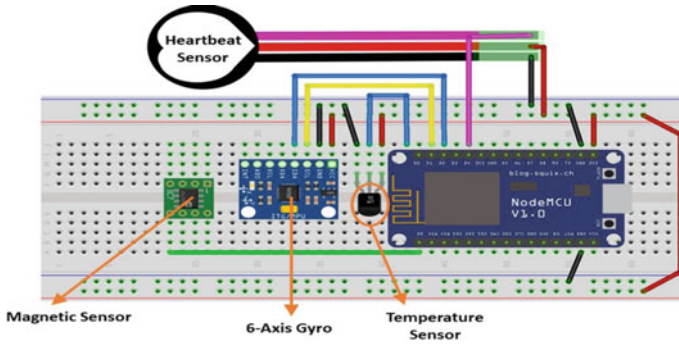


Fig. 11 Connection diagram of the proposed system

## 6 Results

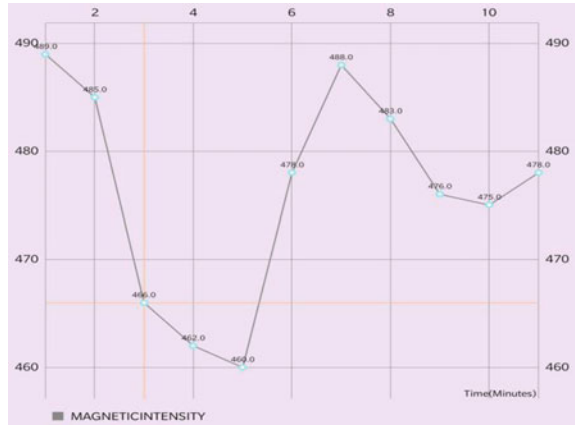
The graphs shown in the following figures are obtained when the wearable is tested in real-time environment and depicts the data transmission of various parameters like gyro angles, magnetic field intensity and heartbeat rate and temperature data with time. Figure 12 shows the live worker gyro data plot against time which represents the movement of the worker, but this data alone cannot say whether the worker is working near machinery or not. With the help of gyro data, we can find the distance moved by the worker, and the time worker is active. By combining this gyro data along with magnetic intensity data, we can find the number of working hours by the worker in the industry.

Figure 13 shows the magnetic intensity the worker is exposed to against time which tells us whether the worker is working near the machinery or not. Our algorithm takes both gyro angles movement data and magnetic intensity data together for determining the number of working hours by the worker and shows the analysis as in Fig. 14 shows

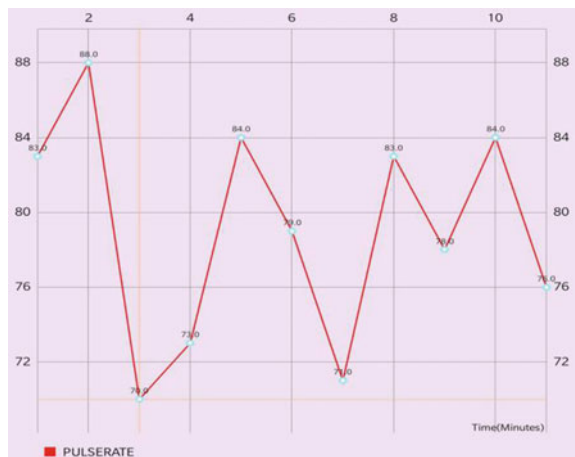
Fig. 12 Gyro angle data versus time plot



**Fig. 13** Worker exposed magnetic field versus time plot

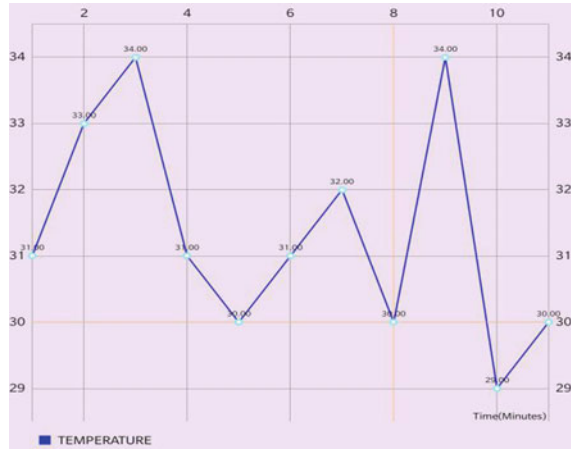


**Fig. 14** Heart rate data versus time plot



the heartbeat rate of the worker for every minute, while he is wearing the band inside the industry. Heartbeat data helps us to determine whether the worker is feeling stressed or tired. If his heartbeat is not normal, we indicate him to take rest or if it is serious, we suggest the admin take the worker to the nearby hospital. Figure 15 shows the body temperature data of worker plotted against time. Heartbeat along with body temperature does not only solve the problem of proxy detection stated in Sect. 4.5.3 but also gives us more accurate health analysis of the worker.

**Fig. 15** Body temperature data versus time plot



## 7 Conclusion and Future Scope

The way IoT and sensors have been growing is something very appreciable, and we have exploited the same to build a product to solve one of the very important, regularly faced day-to-day problem with workspaces in industries [6–8]. The proposed system is very frugal and affordable so that companies can easily implement our system on their workers, which also makes real-time performance analysis of the workers easy for the admins. Our system is viable to work in all kinds of extreme conditions, and it is designed in such a way that the workers or the admins need not have prior knowledge on the technical functionality of the system and gets accustomed to its usage with ease. In order to improve the scalability of the proposed product, the below mentioned points are some of the future enhancements and improvements that can be made.

- Our app can be made to predict the percentage of motivation of worker towards his work by taking different parameters like number of leaves taken by worker, period of leaves, his daily workspace entry and exit time, his daily working time in the mentioned working hours, etc. So, the admins can know the workers who are interested in the work, and they can be encouraged.
- The accuracy of predicting the performance of the worker can be greatly boosted by using machine learning and deep learning models using CNN architectures.

## References

1. J. Duffy, Productivity report|bridging research and practice on personal productivity
2. Wasting Time At Work: The Epidemic Continues—The Fobs Report
3. Number of workplace injury and work-related ill health cases|Page 8—HSE, UK report



4. N. Vignais, M. Miezal, G. Bleser, K. Mura, D. Gorecky, F. Marin, Innovative system for real-time ergonomic feedback in industrial manufacturing. *Appl. Ergon.* **44**(4), 566–574 (2013)
5. P. Lukowicz, J.A. Ward, H. Junker, M. Stäger, G. Tröster, A. Atrash, T. Starner, Recognizing workshop activity using body worn microphones and accelerometers, in *International Conference on Pervasive Computing* (Springer, Berlin, Heidelberg, 2004), pp. 18–32
6. K. Velusamy, D. Venkitaramanan, S.K. Vasudevan, P. Periasamy, B. Arumugam, Internet of things in cloud. *J. Eng. Appl. Sci.* **8**(9), 304–313 (2013)
7. R. Sivaraman, S.K. Vasudevan, A. Kannegulla, A.S. Reddy, Sensor based smart traffic regulatory/control system. *Inform. Technol. J.* **12**(9), 1863–1867 (2013)
8. E. Aravind, S.K. Vasudevan, Smart meter based on real time pricing. *Proc. Technol.* **21**, 120–124 (2015)

# Performance Evaluation of WebRTC for Peer-to-Peer Communication



Kiran Jadhav, D. G. Narayan, and Mohammed Moin Mulla

**Abstract** In this era of the Internet and developing technology, there are numerous ways of interacting with each other and a plenty of services available to make it possible. These include social media platforms, email, VoIP, messaging applications, etc. One of the important aspects here would be real-time communication (RTC), which means interacting with people all around the world as if they were face-to-face. The advancement in RTC has led to the development of a new innovative technology called Web real-time communication (WebRTC), which enables an easy streaming of audio and video content over the Web. This powerful tool currently revolutionizing Web communication has introduced RTC capabilities into browsers as well as mobile applications. The study of this WebRTC technology and its implementation has been carried out in this paper. The WebRTC standards specified protocols, signaling techniques, and WebRTC communication flow between the peers are discussed. WebRTC is supported on two major browsers Google Chrome and Mozilla Firefox, and experiments are conducted on devices running on these browsers with different configurations. Performance has been measured in terms of peer connection establishment, peer communication, user data transfer, and video streaming parameters.

**Keywords** WebRTC · RTC · Peer-to-peer · Signaling · ICE · STUN · TURN · SDP · SRTP · DTLS

---

K. Jadhav · D. G. Narayan · M. M. Mulla (✉)  
School of Computer Science and Engineering, KLE Technological University, Hubli, Karnataka,  
India

e-mail: [moin.mulla@kletech.ac.in](mailto:moin.mulla@kletech.ac.in)

K. Jadhav

e-mail: [kiranjadhav226@gmail.com](mailto:kiranjadhav226@gmail.com)

D. G. Narayan

e-mail: [narayan\\_dg@kletech.ac.in](mailto:narayan_dg@kletech.ac.in)

# 1 Introduction

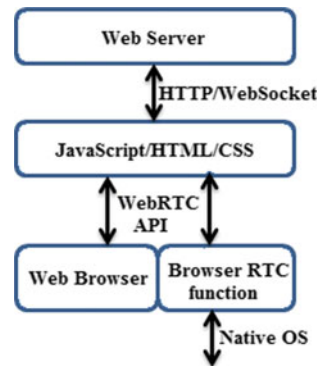
The Internet today connects every nook and corner of the world and has made everything reachable from anywhere. And a means of sharing data or information and interacting with people all around us as if they were face-to-face through the Web connection has become obligatory these days. This type of communication is called real-time communication (RTC) [1]. Being able to communicate from anywhere anytime has been a driving force in the growth of RTC, leading to a systematic use of network and application resources. Combining this RTC technology with the existing information and resources was tough, expensive, and burdensome before. Now, the Web is going through a change that allows the Web browsers to flood data packets directly without the help of any intermediate servers. This new peer-to-peer communication is set up upon a new standard set of APIs, by the Web real-time communication. In May 2011, Ericson implemented WebRTC for the very first time [2].

WebRTC [3] is an open framework providing browsers with the ability to communicate in real time. To put it simply, WebRTC allows us to build real-time applications for the browsers. WebRTC provides all the features of audio, video, and data communication without requiring users to install any supplementary plug-in or software apart from the browser. This is how WebRTC is different from some popular social applications such as Skype and FaceTime, and it is supported by major Web browsers like Google Chrome, Opera, and Mozilla Firefox.

A WebRTC Web application customarily is written as a combination of HTML and JavaScript which communicates with Web browsers via the WebRTC API as shown in Fig. 1, which permits it to relevantly utilize and manage the real-time browser activity. The WebRTC API should deliver a vast set of tasks such as connection management, selection, media control, encoding/decoding, firewall, and NAT traversal.

WebRTC [4] implements three APIs. The API `getUserMedia` enables our application to access user media devices. It handles activities on the media stream such as exhibiting the stream's content, recording or sending it from one peer to another after prompting the user for permission to utilize audio and video input devices. The

Fig. 1 Real-time communication in browser



API `RTCPeerConnection` comes into play once we have a user's local media stream. An `RTCPeerConnection` object is created in order to send this stream using (secure real-time transport protocol (SRTP)). This way the media reaches straight to the other browser, and also it is encrypted in the transit. This API handles signal processing to block out the deafening sound from audio/video data and provides noise cancellation, does compression/decompression of audio and video, handles codec, manages bandwidth, and enables security by encrypting the data. The API `RTCDataChannel` helps to transmit any kind of arbitrary data. The possibilities of this include online video games, chat, and any application that requires the exchange of information in real time. This API uses stream control transmission protocol (SCTP), which connects two clients without the need of any intermediate servers.

The remainder of this paper is organized as follows: After the introduction, Sect. 2 discusses the related works of WebRTC. The major elements of WebRTC and how they operate and design this technology with a sequence diagram are given in Sect. 3. Section 4 discusses and analyzes the results, taking different scenarios into account. Section 5 gives the conclusion of the paper with a view for future work.

## 2 Related Work

In paper [2], the authors discuss the uses of WebRTC such as multichannel communication, screen sharing, video chat, video conferencing, file sharing, real-time marketing, social networking, and financial and health services. Listing the points like WebRTC which is open-sourced or free platform and device-independent, secure, advanced, adaptive to network conditions, and interoperable with VoIP and video, authors talk about the benefits of this technology. They explain the overall working architecture of WebRTC, which is based on client-server model semantics. The paper further lists the steps to build a WebRTC application using NodeJs and also lists the tools to implement the same on mobile. Lastly, a few limitations of WebRTC are mentioned. In paper [4], examination of the performance of the video codecs H.264 and VP9, and investigation of the impact of wired and wireless networks on WebRTC are done. Since congestion control is not supported by UDP, WebRTC uses a customized congestion control algorithm called the Google congestion control (GCC) algorithm that alters itself to the varying states of network operation. Using the latest Web browsers with different types of use cases, the evaluation of its performance has been done. The paper's key contributions include learning the effects of various pseudo-network states on the new implementations of WebRTC, comparing its execution results on separate mobile devices.

Authors of paper [5] introduce a standard for measuring WebRTC peer communication establishment and peer connection performance called `WebRTCBench`, which is publicly available under GPL license. Identifying performance bottlenecks of different WebRTC implementations across a domain of platforms (operating systems) and devices have been discussed. Authors of paper [6] have tested and evaluated the WebRTC applications on different browsers and operating platforms for 3G, 4G,

and local networks. Packet loss is less than 1% for the above 3G, 4G networks, and packet loss is more than 1% for DSL connection. Round trip time is less than 100 ms which means the communication is better. In paper [7], the working of a WebRTC project and the major components of WebRTC APIs are described. The `MediaStream` interface presents a stream of media content, and `MediaStreamTrack` represents the type of media captured from the input source. Further, the transport mechanism, ICE with STUN, and TURN are explained in detail with reference to the `PeerConnection` API, and the use cases of `DataChannel` API are given. The implementation is done here using the `WebSocket` mechanism of signaling and the server/client application is constructed using `NodeJs`. The `WebSocket` server here handles three kinds of control messages (`initialize`, `getUserMedia`, `RTCPeerConnection`, `RTCDataChannel`), two kinds of media information messages (`SDP offer` and `answer`), and one network information message (`ICE candidate`).

In paper [8], the authors assess the performance of numerous network topologies (multiparty: fullmesh and mixer) implementing WebRTC, considering the congestion control techniques used and deployed lately. Receive-side real-time congestion control (RRTCC) is the algorithm employed here. Varying throughput, delay, and effects of fluctuating proportions of cross-traffic on both RTP and TCP are used to evaluate the performance, which suggests that RRTCC yields good results but starves when compared with TCP. Based on experimental observations, it is concluded that RRTCC works fine with low delay networks and can withstand short-term modifications that might comprise of delay or queuing. Article [9] considers applications where network services and Web browsers are treated as the elements of a control loop for which at least soft real-time work is needed. The problem of providing RT transmission has come into existence because of the advancement of network protocols. The article mainly compares their properties in a number of network configurations and examines if they are appropriate as an infrastructure of the control system. Two control loops namely simple PID (proportional–integral–derivative) loop and a multidimensional DMC (dynamic matrix control) are presented on a Web platform as use case studies. `WebSocket` and `WebRTC` communication schemes are described and compared. Finally, keeping aside communication delays, it is shown that these techniques can be utilized as units of the control assembly and also building blocks of a total control loop: controller, human–machine interface (HMI).

The paper [10] covers a study on whether `WebRTC` data channels can be used in Web applications dictating high performance in different simulated network conditions for relaying of arbitrary data. Performance is still not ideal as calibrating the `SCTP` window size has an outcome with finer performance on high latency links but low throughput. There are a few complexities with large window sizes, such as packet loss which may lead to prolonged delays in transporting received packets. The authors suggest that more investigation has to be done not only in computing throughput and latency but also in requirements for CPU and battery on mobile devices. In paper [11], authors work and concentrate on `WebRTC` over LTE testbed based on NS-3 framework. LTE is the current technology for mobile communication systems as standardized by the 3GPP. It comprises LTE customized modules, an ad-hoc server realized with `NodeJs`, and two mobile clients which have `HTML5`

browsers to aid WebRTC audio/video calls. Numerous multimedia WebRTC flows are analyzed, an empirical cumulative distribution function of the user throughput, jitter, and packet loss is depicted, and performance analysis is carried out in various conditions. Paper [12] compares WebRTC servers on virtual machines and Docker containers. The authors have used Kurento media server and see the virtualization type that fits a WebRTC application. They have done a multimedia test on Docker containers and KVM machines which show that the latter have overhead in their performance which can be expensive. Thus, Docker containers perform better than VMs [13].

### 3 Web Real-Time Communication

This section describes the overview and the design details of WebRTC.

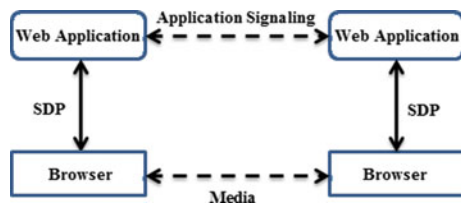
#### 3.1 Overview of WebRTC

The purpose of the design of WebRTC is to define how to supervise the media plane, and the signaling plane is left to the application layer. Signaling is like a manual handshake between two parties, and signaling messages are exchanged by HTTP or WebSocket or any standard signaling protocol like SIP.

WebRTC holds forth the client-server interpretation by presenting a p2p communication model connecting browsers as shown in Fig. 2. This approach is diagrammatically summarized by the JavaScript session establishment protocol (JSEP). WebRTC leverages multiple standards and protocols which include signaling, ICE, STUN/TURN servers, SDP, DTLS, SRTP, SCTP, and UDP/TCP.

Here, different protocols are used for media and data transport in WebRTC as shown in Fig. 3. WebRTC transmits audio, video, and data between browsers over the user datagram protocol (UDP). UDP is apparently faster than TCP as there are no retransmissions, congestion control, or acknowledgements. Low latency and high throughput are of high importance (i.e., speed over reliability) and that is why UDP is preferred over TCP. The RTP control protocol observes transmission statistics information linked with data streams, and thus, SRTP transports the media data

Fig. 2 JSEP



**Fig. 3** WebRTC protocol stack

<b>PeerConnection</b>	<b>DataChannel</b>
<b>SRTP</b>	<b>SCTP</b>
<b>Session (DTLS)</b>	
<b>ICE, STUN, TURN</b>	
<b>Transport (UDP)</b>	
<b>Network (IP)</b>	

together with RTCP. For SRTP key and association management, datagram transport layer security (DTLS) is used. It establishes the keys, helping in encrypting all the information included in the transferring and rendering of media stream data by SRTP. Besides that SRTP also adds sequence numbers, timestamps, and unique stream IDs. SCTP is a connection-oriented and message-oriented protocol that provides reliable transport, in-sequence delivery of packets and rate-adaptive congestion control. It can deal with multiple simultaneous streams, path MTU discovery, and fragmentation.

The two WebRTC endpoints that want to communicate directly perform an offer/answer exchange of SDP messages. Most devices are behind one or more NATs, proxies, firewalls, and anti-virus software that blocks certain ports and protocols. To overcome these problems, WebRTC API can use the interactive connectivity establishment (ICE) which potentially finds out the most efficient option to connect the peers. ICE first tries to make a connection using the host address; if that fails, using a STUN server, ICE obtains an external address, and if that fails, traffic is routed via a TURN relay server. STUN (Session Traversal of UDP through Network Address Translators) servers reside on the public Internet; they allow entities behind NAT to discover the public IP address bindings allocated. This aids WebRTC peers to get an address for themselves that can be accessed publicly which they can pass to other peers via signaling, so that a direct link can be laid out. TURN (Traversal using Relays of UDP or TCP through Network Address Translators) servers can be used in cases where symmetric NATs are used as a fall back. TURN allows exchanging data packets between peers using the relay, but not signaling data. The peers request the server to relay packets to and from other peers using server’s relayed transport address.

When streaming video or initiating a call, there is a need to transfer media details, its format, the transport addresses (peer’s IP address and port), and other meta-data required to trace media objects between the participating peers. All this data is reserved and traded using session description protocol (SDP). When a user starts a WebRTC call, the description created is called an SDP offer, containing all the data about the caller. The recipient then responds with an SDP answer, containing all the data of the receiver. Both devices thus share the information needed for media data exchange which is managed by ICE. Each peer has a local description and a remote description describing both ends of a call.

### 3.2 WebRTC Design

Our implementation of a WebRTC client server model with PubNub signaling solution between them is shown in this section. Video and audio data is not streamed over the PubNub network. It is just a messaging service with low latency. Here, UUID helps to uniquely identify the user or device that connects to PubNub and username instantiates PubNub using your own publish and subscribe keys. Once signaling has occurred, video/audio/data is surged directly between clients using WebRTC's Peer Connection API.

The sequence communication flow in WebRTC starts when the initiator peer contacts the signaling server and queries it to create a signaling channel. This peer then accesses the user media via `getUserMedia` after the permission is granted explicitly. The joiner peer then connects to that same server and joins the channel using a session ID usually referred to as room/channel ID. When the joiner has access to the local user media, a message notification is forwarded to the first peer via the signaling server. The first peer, i.e., the initiator, then instantiates an `RTCPeerConnection` object to create a `PeerConnection`, attaches the local stream, creates an SDP offer, and transports it to the second peer, i.e., the joiner. Once the SDP offer is received, the second peer follows the same steps performed by the first peer of creating a `PeerConnection`, attaching the local stream and creating an SDP answer which is transported back to the first peer. Here, we use PubNub service for signaling to interchange network information using ICE protocol. Once the SDP answer is collected by the initiator peer, the part of negotiation is over.

The same flow sequence is diagrammatically shown in Fig. 4. Same steps are replicated at both the peers including gathering of ICE information where host, server reflexive, and relayed candidates are collected and prioritized, and then default candidates are chosen after eliminating redundant candidates. Thereafter, adding of ICE candidates at each peer, sorting of local and remote ICE candidate pairs, performing checks on each pair takes place. STUN binding request/response is performed at both ends. Both peers can now move to peer-to-peer communication with a data channel to send and receive messages head on.

## 4 Results and Discussion

This section provides the results and its analysis from the WebRTC experiment.

Experiments are conducted on two browsers (Google Chrome, Mozilla Firefox) running on different configurations given in Table 1. Performance analysis of WebRTC over LTE is done.

Figure 5 shows the graph of time required for peer connection initialization which includes time to instantiate an `RTCPeerConnection` object for a new peer connection and time to capture user media via `getUserMedia` except the time taken to wait for user's permission. It also includes time consumed to play the accessed local media



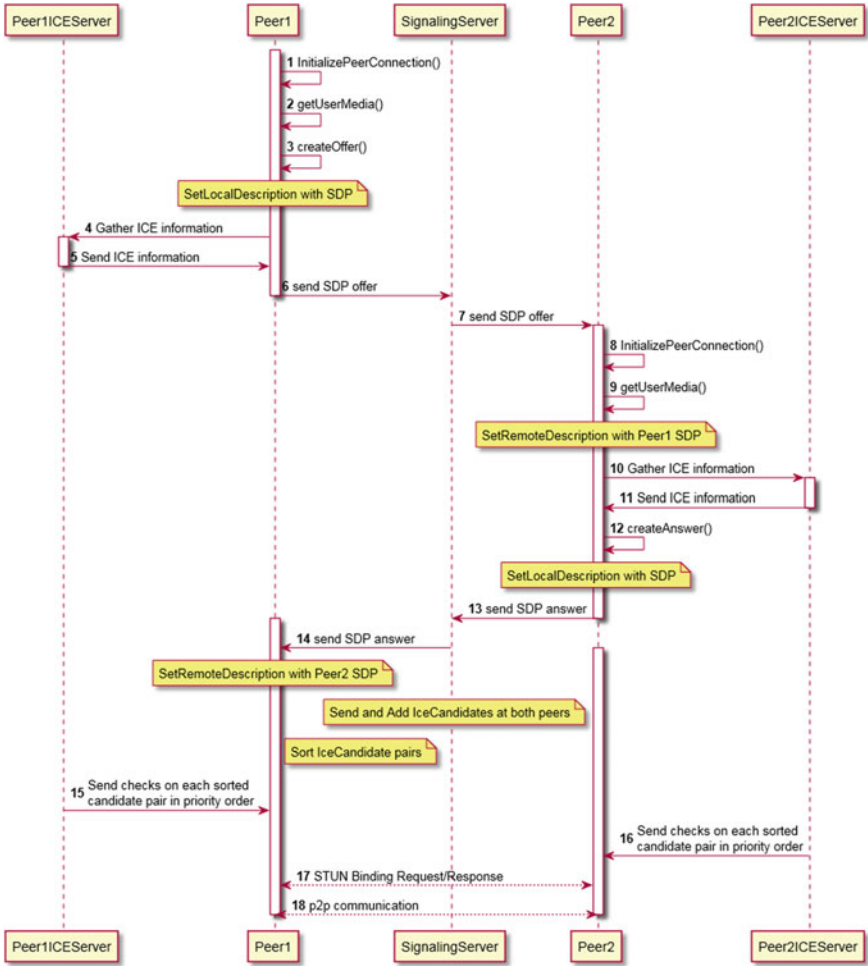
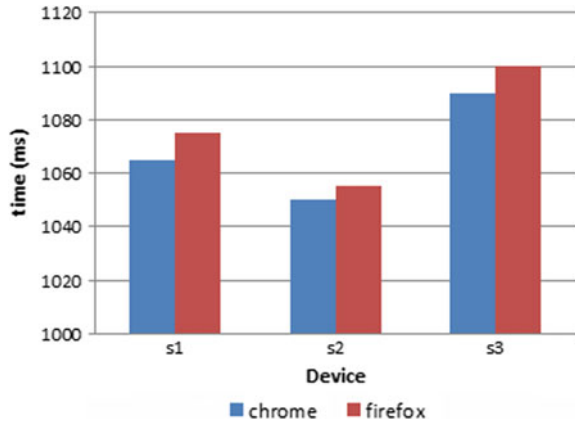


Fig. 4 ICE communication and WebRTC flow sequence diagram

Table 1 Specification of test devices

Device sample	OS	Processor	Browser
S1	Windows 7	Intel® core™ i5	Chrome 83 Firefox 77
S2	Windows 8.1	Intel® core™ i7	Chrome 83 Firefox 77
S3	Android 9	Qualcomm SDM450 Octa core	Chrome 83 Firefox 77

**Fig. 5** Time for peer connection initialization



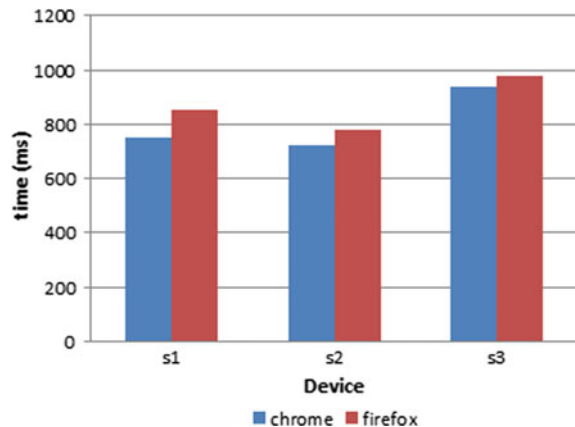
after the request and time required to set up a signaling channel via any standard signaling method.

Time for peer communication as shown in Fig. 6, includes makeCall time to set local session description and create SDP offer and the time spent for finding and exchanging ICE candidates between the interacting peers. It also includes makeAnswer time to set local and remote session descriptions and create an SDP answer, signaling time consumed which helps in relaying of offer/answer information messages and signals over the network. Lastly, the time to create and open a data channel and time needed to play remote media streams are available.

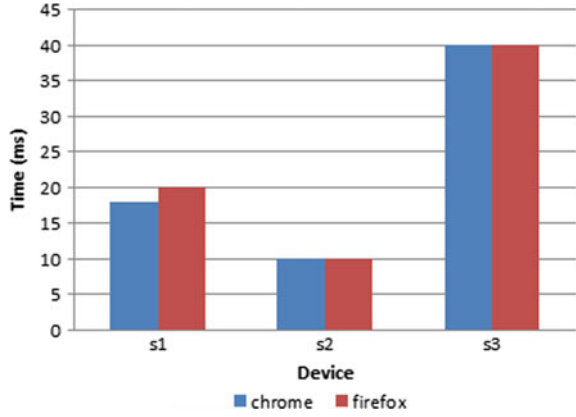
Figures 7 and 8 show the graphs of RTT for a user data message to be transferred between two prime browsers supporting WebRTC, namely Chrome and Firefox on the same machine (device) and over LAN, respectively (Figs. 9 and 10).

Reference videos with resolution 640 \* 360 and 1280 \* 720 of duration 10 s and 20 s, respectively, were used as inputs. Fig. 11 represents average encoded video

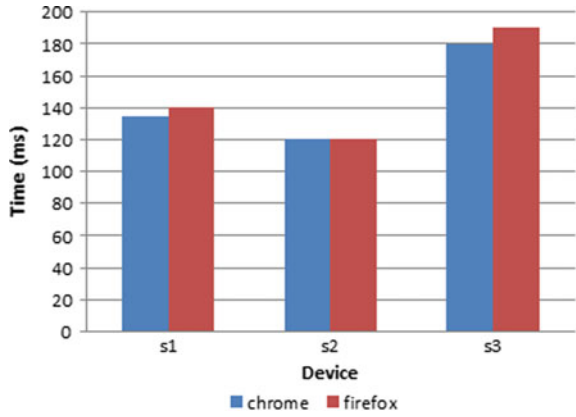
**Fig. 6** Time for peer communication



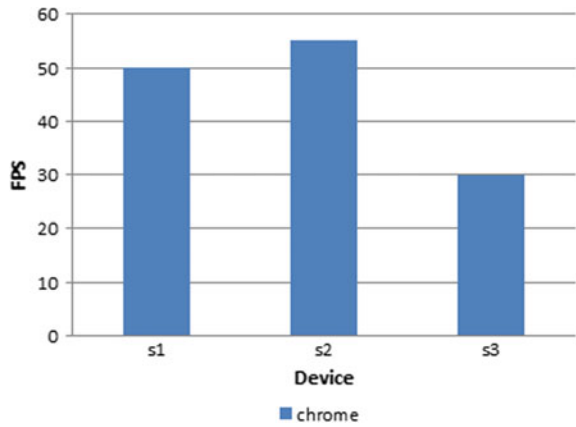
**Fig. 7** RTT to transfer data message on same device



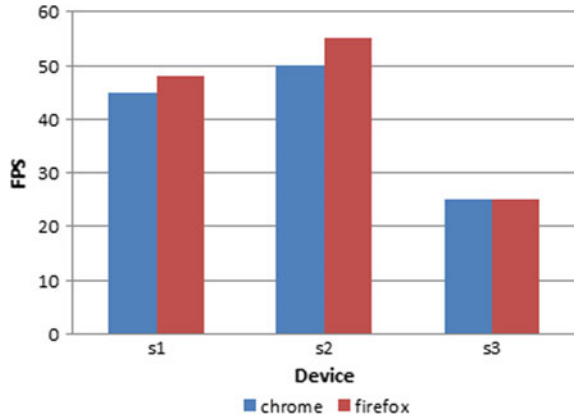
**Fig. 8** RTT to transfer data message over LAN



**Fig. 9** Average encode frame rate



**Fig. 10** Average decode frame rate



frame rate, i.e., FPS (frames per second), whereas Fig. 12 represents average decoded video frame rate for both videos.

From the experiments, it is seen that the time delay is more for Firefox than Chrome during the peer connection initialization and peer communication. When we stream video in Google Chrome, resolution is fixed and the frame rate is low, while streaming video in Mozilla Firefox the video is scaled down and the frame rate is high or the frame rate is conserved. Latency can be observed in mobile devices when data transfer time via DataChannel is considered. It is also observed that video streams in mobile devices do not provide high or better quality streams. Another observation here is that Firefox transfers files with arbitrary large sizes, but Chrome splits the large files into smaller chunks in the beginning and then transfers them one by one.

## 5 Conclusion and Future Work

A detailed study of the technology WebRTC has been done. As evident from our experiments, WebRTC performs comparatively better in Chrome browser than in Firefox given the various parameters like peer communication, rate of data transfer in accordance with time, latency, frame rate and data resolution in video streaming, etc. For mobile devices, WebRTC restricts the data rates and resolutions. Thus, it leads to decreased quality of streaming of videos. WebRTC provides a transparent and easy real-time communication between end users. Thus, WebRTC has a great ability, and in the near future, it may achieve a greater demand in the communication network market.

As future work, we plan to conduct experiments to detect how CPU capacity can affect the video quality for different types of networks.

## References

1. C. Jennings, T. Hardie, M. Westerlund, Real-time communications for the web. *IEEE Commun. Mag.* **51**(4), 20–26 (2013)
2. A. Hussain, P. Kumar, Sharma, *A Framework for Real Time Communication on Web using with WebRTC*, vol. 7, Issue V, May 2019. ISSN: 2321-9653
3. A.B. Johnston, D.C. Burnett, *WebRTC: APIs and RTCWEB Protocols of the HTML5 Real-Time Web*, Digital Codex LLC (2012)
4. B. Jansen, T. Goodwin, V. Gupta, F. Kuipers, G. Zussman, *Performance Evaluation of WebRTC Based Video Conferencing IFIP WG 7.3 Performance* (2017)
5. S. Taheri, L. Aghababaie Beni, A.V. Veidenbaum, A. Nicolau, *A Benchmark for Performance Assessment of WebRTC Implementations*, *CECS*, April 20, 2015
6. E. Alperly Tarim, H. Cumhur Tekin, Performance evaluation of WebRTC-based online consultation platform. *Turkish J. Electric. Eng. Comput. Sci.* (2020). <https://doi.org/10.3906/elk-1903-44>
7. B. Sredojev, D. Samardzija, D. Posarac, WebRTC technology overview and signaling solution design and implementation. *MIPRO* **2015**, 25–29 (2015)
8. V. Singh, A.A. Lozano, J. Ott, *Performance Analysis of Receive-Side Real-Time Congestion Control for WebRTC* (2015)
9. T. Karla, J. Tarnawski, Soft real-time communication with websocket and webRTC protocols performance analysis for web-based control loops (IEEE, 2019). 978-1-7281-0933-6/19/
10. R. Eskola, J.K. Nurminen, Performance evaluation of WebRTC data channels, in *20th IEEE Symposium on Computers and Communication (ISCC)* (2015)
11. G. Carullo, M. Tambasco, M.D. Mauro, M. Longo, A performance evaluation of WebRTC over LTE, in *12th Annual Conference on Wireless On-Demand Network Systems and Services (WONS)* (2016)
12. C.C. Spoiala, A. Calinciuc, C.O. Turcu, C. Filote, Performance comparison of a WebRTC server on Docker versus virtual machine, in *13th International Conference on Development and Application systems*, May 2016
13. A.M. Potdar, N.D.G., Shivaraj Kengond, M.M. Mulla, Performance evaluation of docker container and virtual machine. *Proc. Comput. Sci.* **171**, 1419–1428 (2020). ISSN 1877-0509. <https://doi.org/10.1016/j.procs.2020.04.152>

# Scalable Blockchain Framework for a Food Supply Chain



Manjula K. Pawar, Prakashgoud Patil, P. S. Hiremath, Vaibhav S. Hegde, Shyamsundar Agarwal, and P. B. Naveenkumar

**Abstract** Of late, in a food supply chain (FSC) management, many incidents related to the mislabeling and mishandling of food items are found to occur frequently, which often leaves the customers with a question of how safe and reliable is the food they buy and consume. Since the information regarding the tracking of food items is distributed across different locations widely, and the data is vulnerable to being recorded wrongly, the reliability of tracking of the food items through FSC is suspected. Further, an FSC has various stakeholders that interact and transact continuously. Thus, the scalability issue also plays a vital role in making FSC more efficient. The present-day traceability solutions lack efficiency in terms of scalability and reliability. The scalability can be categorized as throughput, cost, capacity, and response time. In this paper, the proposed method for the traceability solution of a food supply chain (FSC) is based on Blockchain, which plays a vital role in providing transparency and integrity along with other salient features like decentralization, immutability, and verifiability. The proposed FSC is made scalable in terms of throughput and cost using state channeling off-chain algorithm for Blockchain implementation.

---

M. K. Pawar (✉) · P. Patil · P. S. Hiremath · V. S. Hegde · S. Agarwal · P. B. Naveenkumar  
KLE Technological University, Hubballi, Karnataka, India  
e-mail: [manjulap@kletech.ac.in](mailto:manjulap@kletech.ac.in)

P. Patil  
e-mail: [prakashpatil@kletech.ac.in](mailto:prakashpatil@kletech.ac.in)

P. S. Hiremath  
e-mail: [pshiremath@kletech.ac.in](mailto:pshiremath@kletech.ac.in)

V. S. Hegde  
e-mail: [vaibhavhegde15@gmail.com](mailto:vaibhavhegde15@gmail.com)

S. Agarwal  
e-mail: [Agarwalshyamsundar0@gmail.com](mailto:Agarwalshyamsundar0@gmail.com)

P. B. Naveenkumar  
e-mail: [naveenkumar.9482@gmail.com](mailto:naveenkumar.9482@gmail.com)

**Keywords** Food supply chain (FSC) · Blockchain · Scalability · State channeling · Throughput · Cost

## 1 Introduction

A food supply chain (FSC) is the business network of people who work together to move raw materials into finished goods and eventually to the end-user. The supply chain is directly or indirectly responsible for fulfilling the consumer's needs in this time and age where food products are shipped throughout the world, which comprises physical movement of goods through complex food supply chains. There have been incidents, e.g., the meat had been mislabeled, products being poorly handled, that reflects the inefficiency of the FSC [1, 2]. The consumer's interest in how the food is procured, how it is maintained, and how it is handled is of major importance to the industry for the efficiency of its FSC. The variability of information is not straightforward due to the disparate repositories and data aggregation complexity, but the information has been spread over multiple silos. And even if somehow all the information is kept as a record until the manufacturing stage, it is difficult to keep its track after the manufacturing stage. Multiple raw materials are used to create the same product; after this, there arises a possibility of creeping data errors that may be due to mechanical or human failure [3]. Some companies provide a centralized source of information, making the product's process more tedious and fraught with delays.

Food supply chain management focuses on the chain of food manufacturing as an integral process-extending from the primary production via processing and trade until it reaches the consumers. Blockchain technology created the backbone of a new type of internet by allowing digital information to be distributed but not copied. A food supply chain(FSC) that is built on a Blockchain plays a vital role in providing transparency, security [4, 5], and integrity and also provides for salient features like decentralization, immutability, and auditability. However, a Blockchain suffers from the issue of scalability. The scalability of a Blockchain can be addressed in terms of throughput, capacity, cost, and response time [6, 7]. The main concern with the usage of a Blockchain is the smaller number of transactions that can be processed per second, which is referred to as throughput. Blockchain's prominent users are Bitcoin, which can handle 7 transactions per second (TPS), and Ethereum, which can handle 20 transactions per second (TPS). Non-Blockchain application such as visa manages 1700 transactions per second (TPS). Hence, it is necessary to improve Blockchain's scalability in terms of cost or throughput [8, 9]. In this paper, state channeling [10] algorithm is proposed to be used for implementing Blockchain technology solution for the FSC application to improve the scalability by improving throughput and reducing the cost required for processing transactions in Blockchain. The state channeling [10] is the method where the transactions are processed outside the main Blockchain. It works based on the 'looking up' state by considering 'multisig' contract that is controlled by a set of participants on the

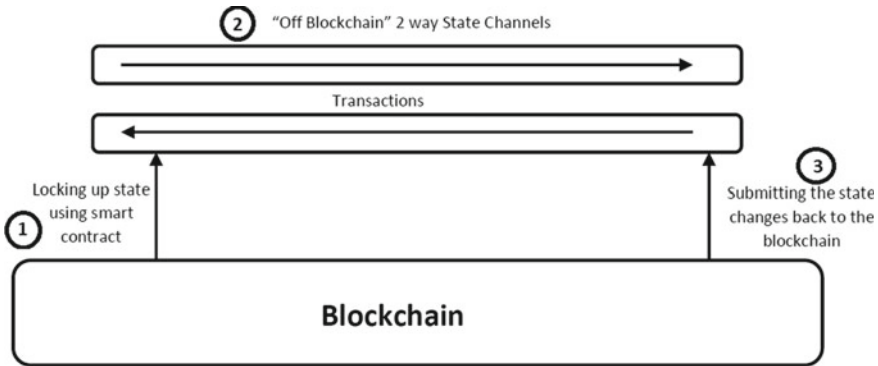


Fig. 1 State channeling

network. The ‘looked up’ form can be regarded as a state deposit, which can be an amount of ether in Ethereum, or it can be an ERC20 token, or it can be an ENS domain name.

Once the state deposit is locked, participants can use off-chain for their messages to exchange and sign valid transactions of Ethereum without deploying them to the chain. These transactions could be on the main chain anytime. The steps of the procedure to be carried out are shown in Fig. 1.

- (a) Part of the Blockchain system state is locked by participants, either by multisignature or by some smart contract, and they can make the updates by agreeing with each other.
- (b) Participants can update by constructing or signing transactions so that the state can be moved to Blockchain but instead are merely held onto for now. It overrides the previous updates.
- (c) Finally, participants submit the state back to the Blockchain. This closes the state channel and unlocks the Blockchain for the next state.

## 2 Related Work

In [11], an architecture is proposed wherein the access control information is requested by a node called management hub on behalf of IoT devices. All the operations allowed by the access control system are defined in a single, smart contract. Further, management hubs can be used to connect numerous constrained networks to the Blockchain at the same time. There is a need for high-performance computational systems as management nodes because multiple IoT devices will have to be connected to the management hub. Also, it will store all the required data to connect with Blockchain.

In [12], an architecture called DeepLinQ is proposed. It is a multilayer architecture to improve flexibility, accountability, and scalability through granular access



control and smart contracts to support privacy-preserving distributed data sharing. The solution has two Blockchain layers, where the base layer preserves CP (Consistency and Partition), and the branch layer ensures AP (Accessibility and Partition) or AC (Accessibility and Consistency). The key properties and design of DeepLinQ are illustrated by using a healthcare data sharing example. The multiple layer design satisfies the POET(Privacy, Ownership, Efficiency, Transparency) properties. The challenge faced herein is that there is no benchmark to evaluate the performance and trade-offs between protocols.

In [13], a solution for Agri-Food supply chain management (AgriBlockIoT) is proposed, which is a fully decentralized, Blockchain-based traceability system. The solution has been implemented on Hyperledger Sawtooth and Ethereum, and it is able to integrate various constrained devices. Hyperledger Sawtooth has better performance than Ethereum in terms of CPU, network usage, and latency. The consensus algorithm of Ethereum takes a toll on the processor, and this may be a barrier for computation power-limited devices, such as edge gateways and IoT devices. Despite this, Ethereum is found to be more competitive as compared to Hyperledger.

In [14], the counterfeiting problem faced at the end of the supply chain is studied, wherein counterfeiters can copy the electronic product code (EPC) of the RFID tag attached to the product. To overcome this problem, a product ownership management system (POMS) is proposed for the post supply chain (i.e., customer, second-hand shop, new customer, etc.). The idea of ‘proof of possession of balance’ from Bitcoin is used to propose proof of possession of the product. With the help of this system, the customer can reject the genuine product if the seller does not have proof of the possession. The main drawback of the system, however, is that the system cannot have both ‘transparency’ and ‘anonymity’ properties.

In [15], various problems are discussed using the current Blockchain, namely Bitcoin and Ethereum. Further, the techniques of sharding, super-quadratic sharding, lighting protocol, DPoS are discussed. It draws a comparison between them by accessing their ability to overcome scalability limits. To overcome scalability, it is suggested to create a new node called Spector node that only takes care of any malicious activity. It is observed that sharding is by far the best scalability solution as it encapsulates all the Blockchain’s basic functionalities. The addition of the Inspector node will make this model better by increasing its security.

In [16], three-level, sharded, permissioned Blockchain architecture, and a consortium framework is used to trace the supply chain (SC), wherein the main objective is to make Blockchain scalable and not to give full access to the consumers or non-SC participants. If trade data is public, then it can be exploited. Hence, an access control list is used to provide for all SC participants, non-SC participants, validators, and governance bodies. The security analysis shows how the proposed solution is prone to a broad range of client and network-based attacks. The query time for product ledger is acceptable. The proposed solution applies to any supply chain industry. For validation, a lottery-based method instead of voting, based on random selection, is used.

In [17], a simple Blockchain for tracing the wine supply chain is used with encryption for privacy. A shared key is used for confidentiality, which is pre-distributed to

relevant entities, and the confidential data is encrypted with the pre-distributed key. The entity generates one public and private key pair and shares the public key with all other participants for the block's authenticity. The solution is implemented on multi-chain. As a result, anyone can view the origin, production, and purchase history of the individual product if it is made public. Still, any customer can verify and authenticate purchased wine by its product id. Once the system receives the id, it traces the id back to all supply chain entities and displays it to the customer.

In [18], RFID and Blockchain technology are combined in the proposed solution. RFID implements procuring, logistics, and sharing in production and warehousing and sales linking. The Blockchain is used for providing transparency and authenticity and consists of mandatory food safety and quality supervision inspection code. The proposed solution incurs a high cost to replace existing systems with a completely new one. The number of transactions is limited to the Blockchain.

### 3 Implementation

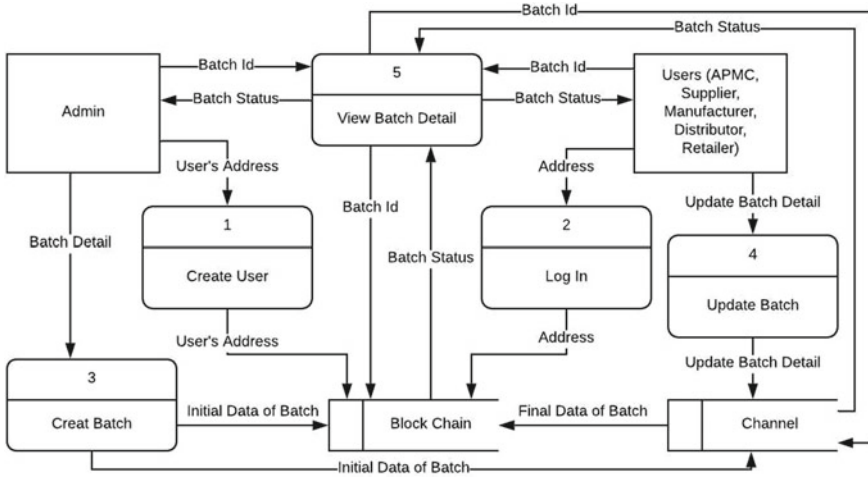
The problem is solved by a private Blockchain network using geth, to which multiple nodes can connect and access the complete FSC on the Blockchain. The admin oversees all the processes, including Create Users and Batches; he can also view the batches' current status. The application has the following users APMC, Supplier, Manufacturer, Distributor, and Retailer. Each user can check the details of the product and update their respective information. The user update is linear, i.e., and the next update can happen after the previous user has completed the update.

The main problem with FSC based on Blockchain is that it is not scalable. It is slow to process a large number of transactions and to resolve this issue, a version of state channeling is used. In state-channeling [10], users interact with each other outside the Blockchain (Off-chain), which dramatically minimizes the 'on-chain' operations.

Whenever the admin creates a new batch, the batch Id and complete data w.r.t to the batch are stored in the channel (i.e., server in our case) and the same details stored in the Blockchain. All the users interact with the server and update their information [19–21] (i.e., from APMC to Retailer). After the last user(i.e., Retailer) finishes his updating, the details stored on the server are recorded on the Blockchain.

There are five modules in the application, as depicted in Fig. 2, which are explained.

**Create User:** Admin has the privilege to create new users by adding their account address along with their respective role and personal details. When a new user is created, a transaction is submitted to the Blockchain, and details of the respective user stored. Here the account address is the address provided by the Ethereum (so a prerequisite for this is that all users must have an Ethereum account).



**Fig. 2** Data flow diagram of the system

**Log In:** The user can log in using his account credentials; the login module verifies the account details and fetches the role according to the account and provides access to the user according to his respective role.

**Create Batch:** Only the admin can create a batch by adding batch details that include necessary information like the farmer, name of the product, and the quantity produced. Whenever a new batch is created, a new batch id is created keccak256 hashing technique is used to create the new batch id considering the admin address and timestamp. (Hashing is used to ensure that the batch ids are unique for each new batch.) For each new batch, a transaction is sent to the Blockchain and to the channel to store the details.

**Update Batch:** The users, according to their respective roles, update the batch details. The updated data is sent to a channel with its batch ID, and these details will be saved on the channel and not on the Blockchain resulting in less interaction with the Blockchain when the last user, i.e., Retailer, updates the data, all the data stored on the channel w.r.t the current batch is sent as a transaction to the Blockchain. This results in only a total of 2 transactions that are sent to the Blockchain per batch, which are shown in Fig. 3, the initial data of the batch in the Fig. 3a and the final data of the batch in the Fig. 3b.

**View Batch:** The admin and the users have the privilege that he can view the batch details anytime, even if the batch is not in the final state. In such conditions, data is retrieved from the channel.

**Scalability:** As stated above, a version of the state channel is used for implementation. The architecture is shown in Fig. 4. A state channel is a second-layer Blockchain scaling method, i.e., a different framework created on the top of the main Blockchain

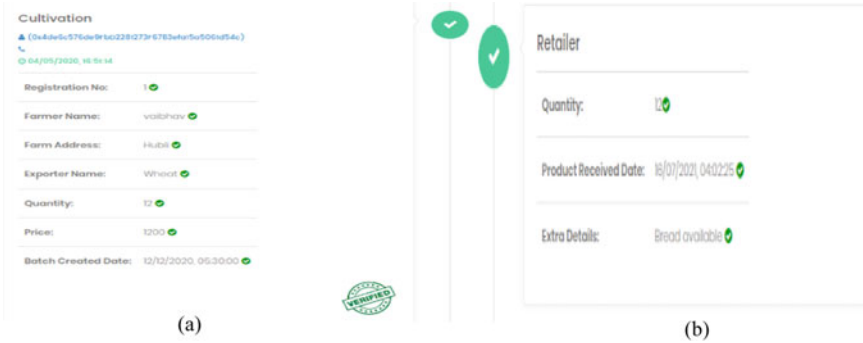


Fig. 3 a Initial data of batch and b final data of batch

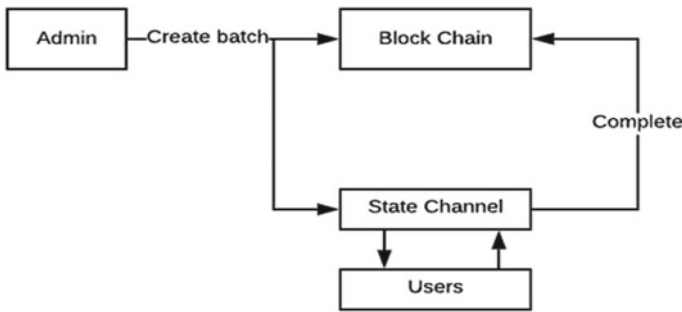


Fig. 4 Architecture diagram

to make it scalable. With this, participants can communicate with each other outside the main Blockchain. This results in a decrease in the number of transactions on the main chain, which reduces the burden on Blockchain and hence improves the throughput. And after processing transactions on the outside channel, the final state is sent to the Blockchain. The implementation uses the PHP server as a channel and storing the data in a JSON file. When the batch’s last entry is sent to the channel, it will send the final data to the Blockchain for maintaining the immutable ledger. The server will be run like a daemon process and will only be interacted by dapp. The data that is sent to the server will be in the form of (sender, batch\_Id, detail) where the sender will be the one who is updating the value, batch\_Id will be the id of the batch whose data is updated, and detail is the updated data.

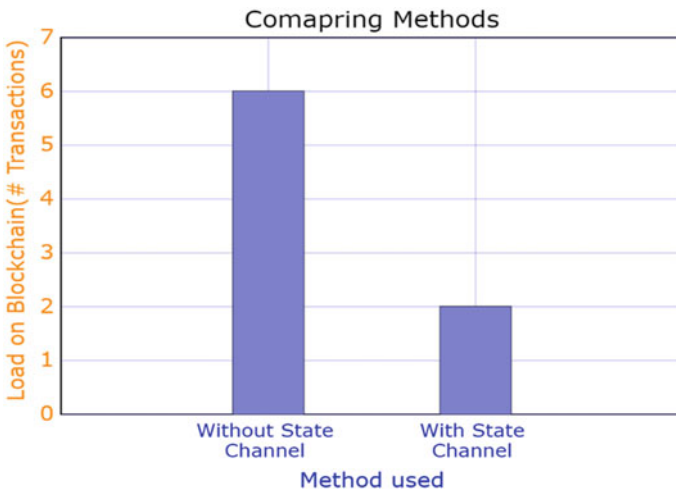
### 4 Results and Discussion

The proposed solution for a scalable FSC comprises the creation of a Food Supply Chain(FSC) system in which admin and the user update the details of different

batches and view the batch’s status and details. The state channeling [10] technique is used to scale the FSC on the Blockchain, where the transactions are moved to a server. The server is considered as a channel where the transaction is carried outside the main Blockchain, and the summary of these transactions is relayed to the main Blockchain.

For the APMC application, before applying the state channeling method, the total number of transactions that were recorded to the Blockchain was 6, namely Batch creation transactions, APMC Details Update Transaction, Supplier Details Update Transaction, Manufacturer Details Update Transaction, Distributor Details Update Transaction, and Retailer Details Update Transactions, whereas after applying state channeling technique, the number of transactions is 2, namely Batch creation transaction and Final Update Transaction (Done after all updates are finished on a given batch). As the update by the users is moved off the Blockchain and onto the server, it decreases the load on the Blockchain, since there are only 2 interactions with the Blockchain, and hence improves the throughput. This is shown in Fig. 5.

Figure 6 represents the comparison between scalable and non-scalable dapp (Decentralized Application) with respect to the gas cost used and the number of batches created. Every transaction is associated with gas cost that has to be executed on Blockchain using Ethereum platform [22] (Gas is the measuring unit for ether



**Fig. 5** Comparison of load on Blockchain to complete one batch’s processing with and without state channel. Example: consider it takes ‘x’ unit of the load to successfully add the transaction to a block. Now consider we have to finish the complete lifecycle (from batch creation to Retailer) for 1000 products. In the case of the approach without state channel, we will be submitting 6000 transactions to the Blockchain, so the total load is 6000x, whereas in the case of the system with state channel, we only offer 2000 transactions to the Blockchain, and the load is 2000x. This shows that the load to process a complete life cycle product is 3times faster in case of the approach with state channel

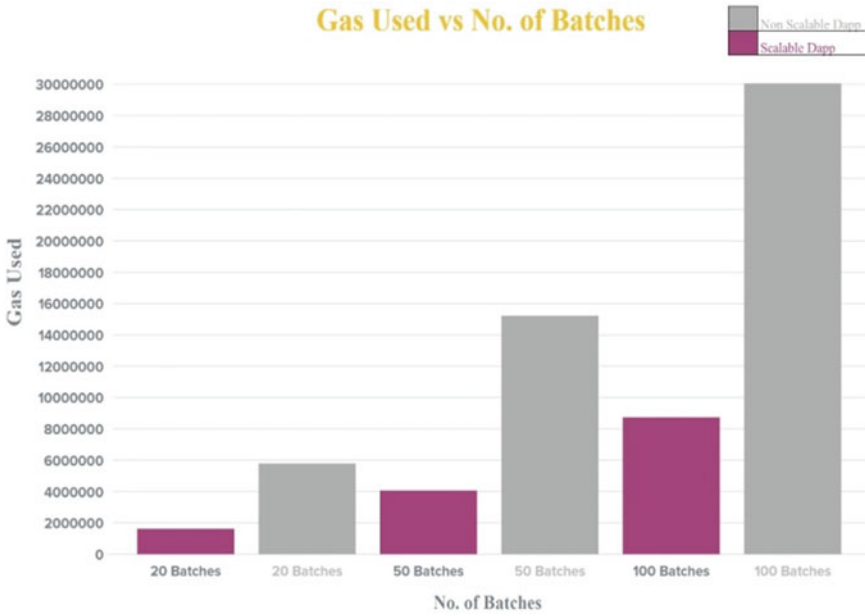


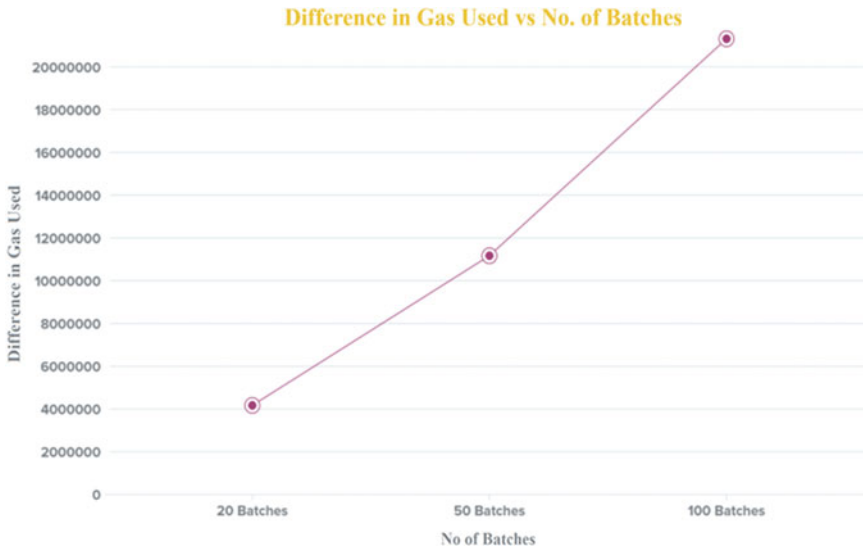
Fig. 6 Gas used versus number of batches created

required for a particular transaction on Ethereum, which is nothing but the cost associated with the transaction). One batch refers to all transactions of a single product from the beginning as in the initial stage till the final stage, i.e., till completion of the sale. Metamask is used to obtain the gas value of each transaction. The gas cost is less for Scalable Dapp than non-scalable dapp. Hence, from Figs.5 and 6, it is observed that as load on Blockchain is reduced, the gas cost also gets reduced. Thus, the scalability is improved, since the lower is the cost, higher is the scalability [6]. Also, it is noticed that, with the increase in the number of batches, the difference between gas used is increasing.

Also, from Fig. 7, it is observed that the difference in gas used between scalable and non-scalable dapp increases with the number of batches significantly.

## 5 Conclusion

In this paper, a traceability solution of a food supply chain (FSC) based on Blockchain is proposed. It is designed to overcome the drawbacks of the existing models, by moving the FSC onto a Blockchain. It can assure the product’s traceability from the harvest date until the product reaches the consumer. The data that is stored is immutable as it is stored on the Blockchain. Most of the existing Blockchain-based models suffered from scalability issue. In the proposed solution, scalability is



**Fig. 7** Difference in gas used versus number of batches created

increased by using state channeling [10]. As the load gets reduced on the Blockchain, the cost gets reduced; in this way, scalability can be enhanced using state channeling algorithm.

The proposed method is more generic in nature, in the sense that it can be applied to any domain with multiple stakeholders, and there is a scope for taking the transactions to be pushed off the chain. It is demonstrated by application to the Food Supply Chain, which has various stakeholders that interact and transact continuously.

Moving the transactions out of the chain and onto a private server raises a security problem, as the server is not as secure as Blockchain. This can be overcome by using signing and encryption methods while interacting with the server. Instead of a server being used as a channel that decreases security, other methods can be explored to use as a channel that does not compromise the system’s security. This aspect will be considered in future work.

**Acknowledgements** The authors are grateful to the reviewers for their helpful comments and suggestions, which enhanced the quality of the paper considerably.

## References

1. Feng Tian, “A supply chain traceability system for food safety based on HACCP, Blockchain & Internet of things,” 2017 International Conference on Service Systems and Service Management, Dalian, 2017, pp. 1–6.
2. Z. Li, IoT-based tracking and tracing platform for prepackaged food supply chain (2017)

3. J. Premanandh, Horsemeat scandal—a wake-up call for regulatory authorities. *Food Control* **34**(2), 568–569 (2013)
4. K.B. Virupakshar, M. Asundi, K. Channal, P. Shettar, S. Patil, D.G. Narayan, Distributed denial of service (DDoS) attacks detection system for openstack-based private cloud. *Proc. Comput. Sci.* **167**, 2297–2307 (2020). ISSN 1877-0509, <https://doi.org/10.1016/j.procs.2020.03.282>
5. P.S. Hiremath, Detection of DDoS attacks in software defined networks, in *2018 3rd International Conference on Computational Systems and Information Technology for Sustainable Solutions (CSITSS)* (Bengaluru, India, 2018), pp. 265–270. <https://doi.org/10.1109/CSITSS.2018.8768551>
6. S. Kim, Y. Kwon, S. Cho, A survey of scalability solutions on blockchain, in *2018 International Conference on Information and Communication Technology Convergence (ICTC)* (Jeju, 2018), pp. 1204–1207
7. M.K. Pawar, P. Patil, P.S. Hiremath (2021) A study on blockchain scalability, in *ICT Systems and Sustainability. Advances in Intelligent Systems and Computing*, vol. 1270, ed. by M. Tuba, S. Akashe, A. Joshi (Springer, Singapore). [https://doi.org/10.1007/978-981-15-8289-9\\_29](https://doi.org/10.1007/978-981-15-8289-9_29)
8. P.M. Dhulavvagol, V.H. Bhajantri, S.G. Totad, Performance analysis of distributed processing system using shard selection techniques on elasticsearch. *Proc. Comput. Sci.* (2019, 2020); ScienceDirect, pp. 1626–1635 (2020), V01-167 (2020).<https://doi.org/10.1016/j.procs.2020.03.303>
9. P.M. Dhulavvagol, V.H. Bhajantri, S.G. Totad, Blockchain ethereum clients performance analysis considering E-voting application. *Proc. Comput. Sci.* (2019, 2020); ScienceDirect, pp. 1626–1635 (2020), V01-167 (2020).<https://doi.org/10.1016/j.procs.2020.03.303>
10. S. Dziembowski, S. Faust, K. Hostáková, *General State Channel Networks* (2018), pp. 949–966. <https://doi.org/10.1145/3243734.3243856>
11. O. Novo, Blockchain meets IoT: an architecture for scalable access management in IoT. *IEEE IoT J.* **5**(2), 1184–1195 (2018)
12. E.Y. Chang et al., DeepLinQ: distributed multi-layer ledgers for privacy-preserving data sharing, in *2018 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR)* (Taichung, Taiwan, 2018), pp. 173–178
13. M.P. Caro, M.S. Ali, M. Vecchio, R. Giaffreda, Blockchain-based traceability in agri-food supply chain management: a practical implementation, in *2018 IoT Vertical and Topical Summit on Agriculture—Tuscany (IOT Tuscany)* (Tuscany, 2018), pp. 1–4
14. K. Toyoda, P.T. Mathiopoulous, I. Sasase, T. Ohtsuki, A novel blockchain-based product ownership management system (POMS) for anti-counterfeits in the post supply chain. *IEEE Access* **5**, 17465–17477 (2017)
15. A. Chauhan, O.P. Malviya, M. Verma, T.S. Mor, Blockchain and scalability, in *2018 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C)* (Lisbon, 2018), pp. 122–128
16. S. Malik, S.S. Kanhere, R. Jurdak, Product chain: scalable blockchain framework to support provenance in supply chains, in *2018 IEEE 17th International Symposium on Network Computing and Applications (NCA)* (Cambridge, MA, 2018), pp. 1–10
17. K. Biswas, V. Muthukkumarasamy, W.L. Tan, Blockchain-based wine supply chain traceability system, in *Future Technologies Conference* (2017)
18. F. Tian, An agri-food supply chain traceability system for China based on RFID & blockchain technology, in *2016 13th International Conference on Service Systems and Service Management (ICSSSM)* (Kunming, 2016), pp. 1–6
19. K.M.M. Rajashekharaiah & Pawar, Manjula & Patil, Mahesh & Kulenavar, Nagaratna & Joshi, Gopalkrishna (2016) Design thinking framework to enhance object oriented design and problem analysis skill in Java programming laboratory: an experience, pp. 200–205. <https://doi.org/10.1109/MITE.2016.048>
20. V. Bhajantri, C. Sujatha, Y. Shilpa, M. Pawar, An experiential learning in web technology course, in *2016 International Conference on Learning and Teaching in Computing and Engineering (LaTICE)* (Mumbai, India, 2016), pp. 125–129. <https://doi.org/10.1109/LaTICE.2016.20>



21. V.S. Handur, P.D. Kalwad, N. Yaligar, V.G. Garagad, M.K. Pawar, An activity based learning: C programming, in *2015 IEEE 3rd International Conference on MOOCs, Innovation and Technology in Education (MITE)* (Amritsar, India, 2015), pp. 310–314. <https://doi.org/10.1109/MITE.2015.7375336>
22. E. Solaiman, T. Wike, I. Sfyraakis, Implementation and evaluation of smart contracts using a hybrid on-and off-blockchain architecture. *Concurr. Comput. Pract. Exp.* e5811 (2020)

# Maximizing Lifetime of Mobile Ad-Hoc Networks with Optimal Cooperative Routing



K. C. Kullayappa Naik, Ch. Balaswamy, and Patil Ramana Reddy

**Abstract** Research in MANET is a challenging task because topology changes frequently and results in link breakages due to node mobility and fast over tiredness of node energy due to limited battery capacity. Therefore, the topology, node mobility, and energy are main important factors that have an impact over the performance of a routing protocol and decreases the overall lifetime of the network. In order to enhance the lifetime of the network, a cooperative communication scheme have been proposed in this paper. Cooperative communication requires cooperative table, relay table, and cooperative neighbor table to store the topological information and implement cooperative transmission among the nodes thereby improving the robustness against the node mobility. Cooperative communication uses multi-hop transmission between the source and destination nodes in order to save energy and thus enhancing the lifetime of the network using minimum energy consumption selection decode and forward (MESDF) routing protocol. The proposed scheme chooses the best relays with minimum energy consumption in a cooperative and distributed manner and considers the link break probability and energy harvesting techniques, to determine the optimal route across a cooperative network. Simulation results clearly shows that the robustness of proposed method increases against the node mobility and saves 21% of node energy in a selected route which in turn increases the lifetime of the network when compared to the existing cooperative and non-cooperative routing methods.

**Keywords** MANET · Cooperative communication · Energy efficiency

---

K. C. Kullayappa Naik (✉) · P. R. Reddy  
ECE Department, JNTUA, Anantapur, India  
e-mail: [kcknaik@gmail.com](mailto:kcknaik@gmail.com)

P. R. Reddy  
e-mail: [prjntu@gmail.com](mailto:prjntu@gmail.com)

Ch. Balaswamy  
ECE Department, GEC, Gudlavalleru, India  
e-mail: [ch.balaswamy7@gmail.com](mailto:ch.balaswamy7@gmail.com)

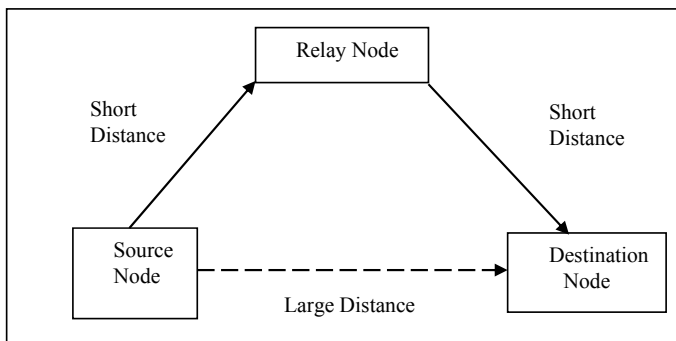
# 1 Introduction

Mobile ad-hoc network is most widely used infrastructure less network and plays an important role in many applications like military communications, emergency systems, conferences, and hotels. The deployment of ad-hoc wireless network [1] is very easy because no-cables, no-configuration, and no-maintenance are required, and hence, it has several benefits such as low cost, short time, reconfigurability, and performing the operation immediately. Moreover, MANET has several disadvantages like limited transmission range, regular link breaks due to mobility of nodes, and fast overtiredness of energy. In order to address the above difficulties, we propose a cooperative communication scheme in MANET which improves the system capacity, network connectivity, reliability, and energy efficiency and decreases interference.

Cooperative communication is a very important technique for modern wireless communication systems. A node in a network can acts as a relay node and is cooperating with source and destination node, and it tries to decode an entire input message and forward it to next hop. The relay node is significantly enhancing the reliability of communication among the nodes in a selected route. Cooperative communication allows multi-hop transmission between the sending and receiving nodes in order to save energy and thus enhancing the lifetime of the network using minimum energy consumption selection decode and forward (MESDF) routing protocol.

In this research paper, we propose a distributed and cooperative routing protocol called as minimum energy consumption selection decode and forward (MESDF) routing protocol which selects the best relay with minimum energy consumption during transmission of data packets from source to destination through an optimal cooperative routing path. The best relays are identified in a cooperative and distributed manner to minimize the energy consumption route while guaranteeing the desired QoS. Furthermore, the results of the proposed scheme are compared with the existing cooperative scheme called constructive relay-based cooperative routing (CRCPR) [2] and non-cooperative scheme.

An example of one-hop cooperative wireless link is shown in Fig. 1. Each node in



**Fig. 1** One-hop cooperative wireless link between source and destination

a cooperative network performs two important roles during transmission which are called as source node and relay node. Here, the main attractive feature of cooperative communication is a relay transmission. A cooperative link (CL) between the source and destination nodes may be classified as two different transmission channels. They are represented by dashed and solid lines. Dashed line between the source and destination nodes represent the direct channel, while the indirect channel is nothing but relay channel is denoted by solid line between the nodes through the relay node. In order to support the cooperative communication mechanism between the source and destination nodes, we need to assign two orthogonal time slots. In the first time slot, source node broadcasts the data packets to all other nodes in the network, and during the second time slot, data is forwarded from relay node to destination node. The relay node in this case is to decode the data which is received from source node and forwards it to the destination node. Therefore, receiving the multiple copies of similar data packets from different channels with the help of a destination node. In cooperative communication, the degree of diversity can be obtained, and therefore, it brings significant enhancement of reception reliability and measures the performance of cooperative transmission.

The remaining of this paper is organized as follows. Section 2 describes about the related work, and MESDF routing protocol was explained in Sect. 3. In Sect. 4, we show the simulation results and discussions about the proposed scheme, and finally, the paper concludes in Sect. 5.

## 2 Related Work

Fast exhaustion of node energy due to limited battery capacity leads to limit the lifetime of MANETs. The current research studies focusing on energy harvesting ability in MANET are a significant interest in long tenure. Bai et al. [2] propose a constructive relay-based cooperative routing (CRCPR) scheme to enhance the robustness of mobility issues and consider energy consumption method to improve the throughput and prolonged the network lifetime.

Sharma et al. [3] proposed a method to enhance the robustness against node mobility and reduce the energy consumption using a multipath routing scheme. The authors try to find the routes with minimum hops, less energy consumption, and appropriate traffic load balancing in a combined way.

Sheng et al. [4] described about the power efficient routing in cooperative networks for minimizing the transmission power for cooperative link, but the selection of relay nodes based on number of neighboring nodes and remaining battery energy is overlooked.

The nodes with EH ability in the network tried to find route with minimum transmission cost with energy count and compared the results with Jakobsen et al. [5]. The final route in the network is selected based on the shortest energy distance.

### 3 Minimum Energy Consumption Selection Decode and Forward Routing

It is a table driven with on-demand cooperative routing protocol. Table-driven means cooperative topology is constructed in advance for all the source-to-destination pairs, and on-demand means route is constructed only when required to forward the data. The main use of relays in the network is to transmit information between source and destination node and is a very effective technique to increase energy efficiency. Because, the distance between source and relay node is very shorter related to distance between source and destination nodes, which means possible to decrease the transmission energy on both sides of the relay nodes.

#### 3.1 Energy Consumption Reduction Technique

The energy consumption ratio (ECR) [6] is a standardized energy metric and is well-defined as ratio of maximum power to maximum data rate and consequently measures the consumed energy per bit of transported information. It can be expressed as

$$\text{ECR} = \frac{\text{Maximum Power}}{\text{Maximum Data Rate}} = \frac{\text{Joules}}{\text{Bit}} \quad (1)$$

The MIMO and relay node [7] are the two main techniques used to save the energy and increase the performance of a network in MANET. In this research paper, we consider the relay node technique for cooperative communication among the nodes in selected route to enhance the energy efficiency thereby prolonged lifetime.

#### 3.2 Multiple-Input Multiple-Output

Recently, a new class of communication has been introduced in cooperative communication which allows single antenna device to take the benefit of multiple-input-multiple-output systems. It designates a set of techniques [8], [9] to reduce energy consumption and expand the throughput thereby increasing the energy efficiency between the sender and receiver nodes.

#### 3.3 Relay Node

The main application of relay node between sender and receiver nodes permits to improve the performance and energy savings. Relaying generally splits longer routes

into shorter route segments thereby decreasing total route damage due to nonlinear relationship of path loss and path distance. Replacing longer paths and associated losses becomes an advantage of cooperative communication with shorter and robust radio links.

The MESDF routing identifies the best relay based on the number of neighboring nodes and remaining battery energy of the nodes which realizes the minimum energy consumption in a selected route. For direct transmission between the source and destination nodes, the corresponding mutual information is given and proposed [4] by

$$I_D = \log\left(1 + \rho |a_{s,d}|^2\right) \tag{2}$$

The transmission power-to-noise power ratio is defined as  $\rho = E_b/N_o$  where  $E_b$  represents the transmission energy per bit and  $N_o$  is the white noise,  $a_{s,d}$  indicates the wireless link between source and destination nodes. The outage probability for direct transmission is given by

$$P_D^{out} = d_{s,d}^k \left(\frac{2^R - 1}{\rho}\right) \tag{3}$$

where  $R$  represents the desired data rate in bit/s/Hz and  $d$  is the distance between source and destination nodes. For cooperative transmission, the distance among the source, relay, and destination nodes is given by the following equations.

During first time slot, source node in the network broadcasts the CREQ packet to rest of the nodes in the network and estimates the distance between source and relay through the received signal strength. Then, destination node receives the information  $y_d = \frac{h_{s,d}}{d_{s,d}^{k/2}}$  from source node, where  $x_s$  denotes information transmitted by source node,  $h_{s,d}$  is the channel,  $d_{s,d}^{k/2}$  represents the distance between source and destination nodes,  $k$  is the path loss exponent, and  $n_d$  represents the white noise. Similarly, during second time slot, destination node broadcasts another CREQ packet with information, estimates the distance between relay and destination node, and receives the information via relay node which is

$$y_d = \begin{cases} \frac{h_{s,d}}{d_{s,d}^{k/2}} x_s + n_d, & \text{if } \left| \frac{h_{s,d}}{d_{s,d}^{k/2}} \right|^2 < q(\rho_s) \\ \frac{h_{r,d}}{d_{r,d}^{k/2}} x_r + n_d, & \text{if } \left| \frac{h_{s,r}}{d_{s,r}^{k/2}} \right|^2 \geq q(\rho_s) \end{cases} \tag{4}$$

where  $q(\rho_s) = (2^{2R} - 1)/\rho_s$  can be derived from direct transmission.

In the proposed scheme, the relay is randomly selected, and hence, mutual information can be written as

$$I_C = \begin{cases} \frac{1}{2} \log(1 + 2\rho_s |a_{s,d}|^2), & |a_{s,r}|^2 < q(\rho_s) \\ \frac{1}{2} \log(1 + \rho_s |a_{s,d}|^2 + \rho_r |a_{r,d}|^2), & |a_{s,r}|^2 > q(\rho_s) \end{cases} \quad (5)$$

Therefore, the outage probability for MESDF routing is given by

$$P_C^{\text{out}} = Pr[I_C < R]$$

$$P_C^{\text{out}} = \frac{1}{2} d_{s,d}^k \left( d_{s,r}^k + \frac{\rho_s}{\rho_r} d_{r,d}^k \right) \frac{(2^{2R} - 1)^2}{\rho_s^2} \quad (6)$$

where  $\rho_s$  and  $\rho_r$  indicate the ratio of transmission power to noise power for source and relay nodes, and therefore,  $I_C < R$  means increasing the performance of cooperative network. The proposed scheme always attains greater energy performance when compared to CRCPR protocol.

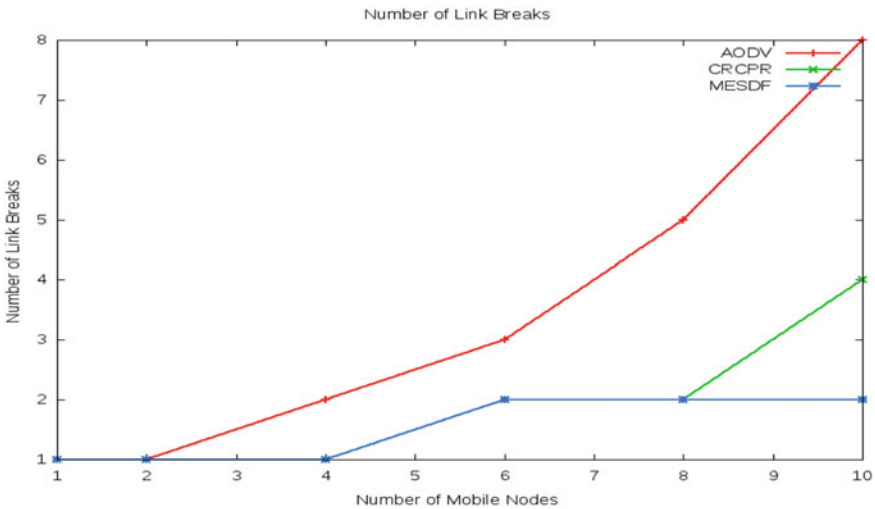
## 4 Simulation Results

In order to investigate the performance of MESDF routing protocol, it needs to use network simulator. Here, we considered the energy harvester [10] for evaluating the performance of MESDF routing protocol. In general, AODV is most widely adopted, and its operation is very simple to understand because AODV has no exact structure to avoid link breakdowns. So, frequent link break [11] will rise quickly when the number of mobile nodes in the network increases. Furthermore, CRCPR is selected as other baseline and uses cooperative table, cooperative neighbor table, and relay table to store the topological information and implement cooperative transmission among the nodes thereby improving the robustness against the node mobility. The performance metrics are examined by varying the number of mobile nodes and energy-restricted nodes using Network Simulator [12–14]. The following are the important parameters required to simulate the cooperative network and are given in Table 1.

1. **Number of Link Failures:** The simulation results shown in Fig. 2 show the frequent link breaks of the three protocols used in a scenario with 50 nodes. For AODV, it has no specific scheme to avoid link breaks. So, the link break frequency will automatically increase when number of mobile nodes increases. In CRCPR, the link break frequency will decrease via the cooperative and relay table when increases the mobile nodes up to certain limit. But in the proposed method that is in MESDF, the number of mobile nodes increases with higher value, and the frequency of link breaks is much lower than CRCPR protocol.
2. **End-to-End Delay:** As we observe that in Fig. 3, when number of mobile nodes involved in a scenario is with 50 nodes, the end-to-end delay of all the three protocols will vary significantly. The end-to-end delay of AODV is higher because there is no specific scheme for avoiding link breaks. More specifically,

**Table 1** Parameters required for simulating the routing protocols

S. no.	Parameter	Assigned value
1	Network area	1000 m <sup>2</sup>
2	Mobility model	Random walk
3	Number of nodes	50
4	Node speed	10 m/s
5	Simulation time	250 s
6	Routing protocol	AODV, CRCPR, and MESDF
7	Data rate	1024 Kbps
8	Packet size	512 bytes
9	Wi-Fi channel	Yans Wi-Fi
10	Initial energy	0.1 J
11	Energy model	Wi-Fi radio energy model



**Fig. 2** Number of link breaks versus number of mobile nodes in a network

due to link break reduction, the end-to-end delay of CRCPR and MESDF is more stable when compared with AODV if increasing the mobile nodes and provides the better performance.

- Throughput:** As we can see that in Fig. 4, throughput of AODV decreases with increasing the mobile nodes in a network. But, the performance of CRCPR and MESDF is more stable and better than AODV because it can utilize the cooperative topology to improve the robustness against the node mobility. Furthermore, the final route selection criteria of CRCPR and MESDF will avoid a node with high link break probability. So, a more stable route will be selected than the shortest path and improves the network throughput.



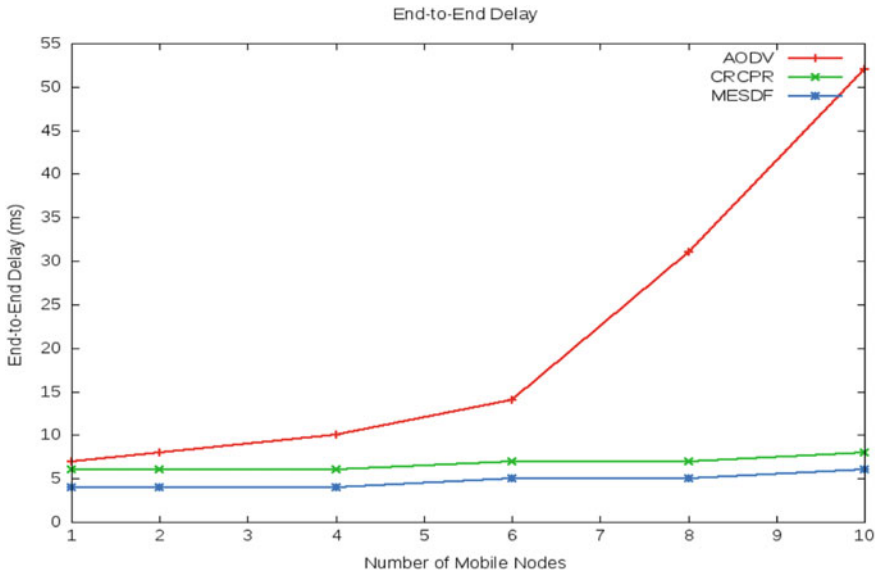


Fig. 3 End-to-end delay versus number of mobile nodes in a network

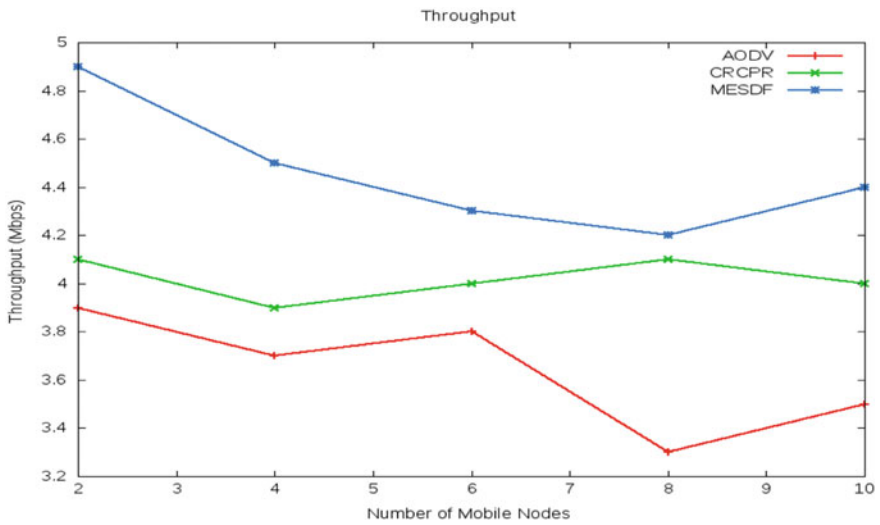


Fig. 4 Throughput versus number of mobile nodes in a network

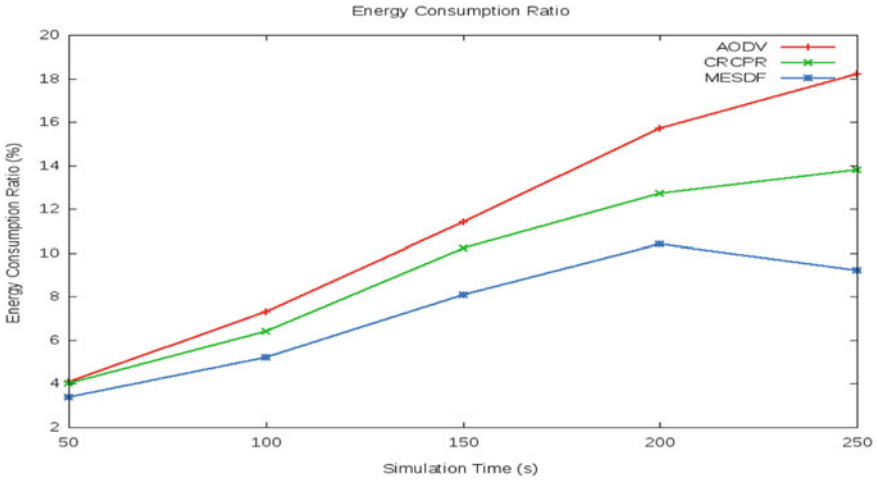
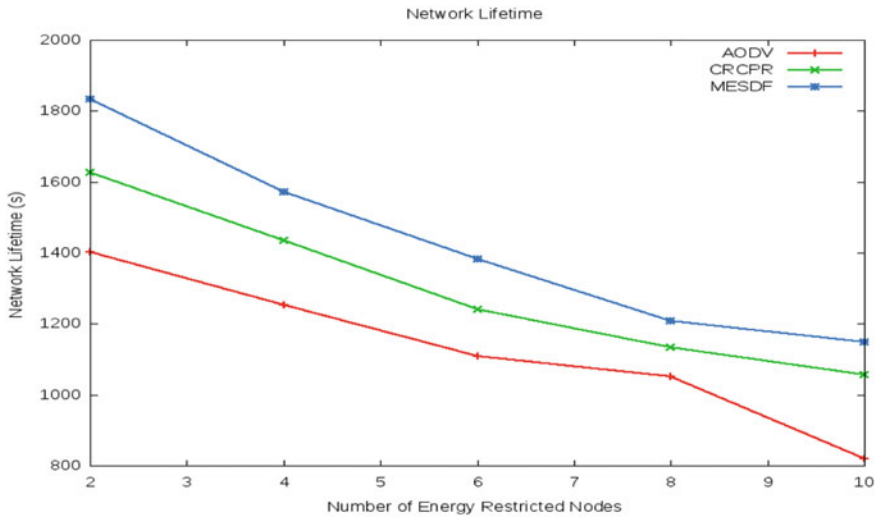


Fig. 5 Energy consumption ratio versus simulation time

4. **Energy Consumption:** The simulation result in Fig. 5 shows that the energy consumption rate of the three protocols is used in a scenario with 50 nodes. For AODV, it has no specific scheme to avoid link breaks. So, the link break frequency will automatically increase when the number of mobile nodes increases thereby consuming more energy when compared to other protocols. In CRCPR, the link break frequency will decrease due to cooperative communication via the cooperative and relay table, and consumption of energy is somewhat less than AODV protocol. But in the proposed method that is in MESDF, protocol consumes less energy for a selected route when compared to existing protocols.
5. **Network Lifetime:** It is nothing but the duration of the network until the first or last or any node along the route knowledges energy drain out. The overall network lifetime of three protocols is shown in Fig. 6. In order to compare the lifetime of the three protocols, we deactivate the energy harvesting (EH) ability and change the role of mobile nodes to energy restricted (ER) node, which assigns the lower energy than the normal nodes. With increasing the ER nodes, the overall network lifetime of both AODV and CRCPR reduces. When the number of ER nodes is lower, the performance of MESDF is better and remains stable due to its route selection criteria and energy harvesting which in turn increases the lifetime of the network.

## 5 Conclusion

The cooperative communication plays an important role in improving system capacity and energy efficiency of a MANET using relay nodes. The use of relay node is to transmit the information between sender and receiver in an effective manner



**Fig. 6** Network lifetime versus no. of energy restricted nodes in a network

to increase the energy efficiency because the distance between the sender and relay is very shorter compared to distance between sender and receiver, which means reduction in transmission energy on both sides is possible. Final selected route in MESDF routing is used for minimizing the energy consumption during transmission of data between the sender and receiver through the relays. The proposed scheme increases the energy efficiency and thereby enhancing the lifetime of the network when compared to existing scheme. The simulation result shows that MESDF routing attains 21% of energy saving in a selected route when compared to existing cooperative and non-cooperative routing methods. Therefore, the proposed scheme gives better performance and prolonged the lifetime.

## References

1. C.E. Perkins, *Ad hoc Networking* (Addison-Wesley, 2001)
2. J. Bai, Y. Sun, Towards constructive relay-based cooperative routing in MANETs. *IEEE Syst. J.* **12**(2) (2018)
3. B.P. Sharma, S. Chugh, V. Jain, Energy efficient load balancing approach to improve AOMDV routing in MANET, in *Proceedings 4th International Conference on Communication System Networking and Technology* (2014), pp. 187–192
4. Z. Sheng, K. Leung, Distributed and power efficient routing in wireless cooperative networks, in *IEEE International Conference on Communications*, (2009), pp. 1–5
5. M. Koefoed, J. Madsen, M.R. Hausen, DEHAR: a distributed energy harvesting aware routing algorithm for ad-hoc multihop wireless sensor networks, in *IEEE International Conference* (2010)

6. S. Kandari, M.K. Pandey, Evaluation of energy consumption by nodes of MANET, in *National Conference on Recent Advances in Electronics & Computer Engineering*, February 13–15, 2015 (IIT Roorkee, India)
7. R. Madan, N.B. Mehta, A.F. Molisch, J. Zhang, Energy-efficient decentralized cooperative routing in wireless networks. *IEEE Trans. Auto. Control* (3), 512–527 (2009)
8. Y. Wang et al., Towards energy-efficient cooperative routing algorithms in wireless networks, in *IEEE Consumer Communication Networks. Conference* (2013), pp. 79–84
9. M. Elhawary, Z.J. Haas, Energy-efficient protocol for cooperative networks. *IEEE/ACM Trans. Netw.* **19**(2), 561–574 (2011)
10. K.K. Win, X. Wu, S. Dasgupta, W.J. Wen, R. Kumar, S.K. Panda, Efficient solar energy harvester for wireless sensor nodes, in *IEEE International Conference on Communication Systems* (2010), pp. 289–294
11. L. Gong, Y. Bai, M. Chen, D. Qian, Link availability prediction in ad hoc networks, in *IEEE International Conference on Parallel Distribution Systems* (2008)
12. A.S. Ibrahim, K.J. Ray Liu, Distributed energy-efficient cooperative routing in wireless networks. *IEEE Trans. Wirel. Commun.* **7**(10) (2008)
13. Network Simulator NS3. Available <https://www.nanasm.org>
14. Network Simulator NS3. Available <https://nsnam.org/documentation/>

# On-Demand Multi-mobile Charging Scheduling Scheme for Wireless Rechargeable Sensor Networks



Charan Ramtej Kodi, Debjit Das, and Shashi Shekar

**Abstract** The sensor nodes in the network senses and processes the data, and each sensor usually has different task burdens due to the environmental change, which results in dynamic change of the energy consumption rate at different nodes. To provide real-time on-demand charging to these sensors is a real challenge. Based on the certain threshold, each sensor node requests for charging, and these charging requests are taken in to the matrix and processed accordingly based on the selection rate by mobile charging vehicle (MCV). This paper deals with wireless charging in sensor networks and explores efficient policies to perform simultaneous multi-mobile charging power transfer through a mobile charging vehicle. The proposed solution, called on-demand multi-mobile charging scheduling scheme (MMCSS), features selection rate (SR) based on which selection of the next charging request node is selected efficiently by considering important parameters. After selecting the node based on the SR, it is checked whether the charging is possible or not based on the next charging node possible (NNCP). Then, the shortest path is given from the MCV to selected node using Dijkstra algorithm. Various MCV charging conditions are discussed below.

**Keywords** Selection rate · Energy consumption rate · On-demand charging · Dijkstra algorithm

---

C. R. Kodi (✉) · D. Das · S. Shekar  
Department of Computer Science and Engineering, Indian Institute of Technology  
(Indian School of Mines), Dhanbad, Jharkhand 826004, India  
e-mail: [charan.17kt000139@cse.iitism.ac.in](mailto:charan.17kt000139@cse.iitism.ac.in)

D. Das  
e-mail: [Debjit.17kt000133@cse.iitism.ac.in](mailto:Debjit.17kt000133@cse.iitism.ac.in)

S. Shekar  
e-mail: [Shasi.17kt000149@cse.iitism.ac.in](mailto:Shasi.17kt000149@cse.iitism.ac.in)

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2021  
S. M. Thampi et al. (eds.), *Advances in Computing and Network Communications*,  
Lecture Notes in Electrical Engineering 735,  
[https://doi.org/10.1007/978-981-33-6977-1\\_37](https://doi.org/10.1007/978-981-33-6977-1_37)

## 1 Introduction

Wireless sensor network has assumed a significant job in numerous monitoring applications and reconnaissance applications including ecological sensing, target following, basic well-being observing, military observation, disaster warning, clinical framework and so on. The sensors are fueled by batteries, which leads to the vitality constraints and continuous battery substitution, and it obstructs the huge scope arrangement of wireless sensor networks (WSNs). There are such a large number of vitality mindful methodologies created in the previous decade to reduce sensor's vitality utilization's and to adjust vitality utilization's among sensors; however, the WSN's lifetime stays as a primary exhibition bottleneck in their organizations since wireless information transmission devours considerable sensor vitality.

In the WSN sensors, they have a limited energy, and due to continuous operations, sensors run out of energy, and each sensor in the network has different energy consumptions due to their different environmental locations. In order to solve this problem, we have on-demand charging, which states that when a sensor node reaches a particular threshold, it sends a charging request to the mobile charging vehicle (MCV). MCV receives multiple charging requests, but selecting the charging requests plays an important role in our network. For this, we have given MMCSS algorithm which deals with this problem under certain parameters.

As mentioned [1], increasing the service time of devices becomes crucial, and the current studies planned a range of strategies to prolong time period of nodes and alleviate sensor node energy restriction downside. In these studies, adjusting dynamically the time length of a node to remain active during an information assortment period, i.e., duty cycle, it is an economical strategy to avoid wasting energy and increase the time period of network. G. J. Han et al. propose a formula to boost the mobile charger's potency underneath considering the connection of energy consumption of mobile chargers and their energy transfer to the nodes in wireless device networks. When we increase the sensor's service time, the energy deteriorates and becomes crucial, so the existing studies have given so many methods to increase the network's lifetime and solved the sensor node's energy problem.

## 2 Related Work

In [2], a mobile charger charges the sensor nodes using wireless modes to maximize the sum of sensor's lifetime with minimal charger traveling distance. This approach also applies the partial recharging strategy to minimize the sensor node's fail rate. This approach proposes two different algorithms for sensor lifetime, maximization problem and algorithm for maximum sensor lifetime with minimum traveling cost for balancing energy and traveling distance. This algorithm efficiently maximizes the sensor network lifetime with an optimized traveling distance. On the other hand, the fixed amount of energy recharge will lead to the downside of this approach. Mainly,

the less energy recharge may increase the MC tour, and the longer charging rate will maximize the dying node's rate. So, the performance of this approach depends on the energy charging rate. However, finding this rate is also a difficult task, and it increases the complexity as well.

In [3], multiple wireless charging vehicles (WCVs) are used to charge the sensors in the network. For this purpose, a scheme named Game Theoretical Collaborative Charging Scheduling (GTCCS) is designed. This scheme uses Nash equilibrium to find the best charging targets for each WCV collaboratively. It converts WCVs into players of the game for higher profit, thereby increasing energy efficiency and reducing the number of dead nodes.

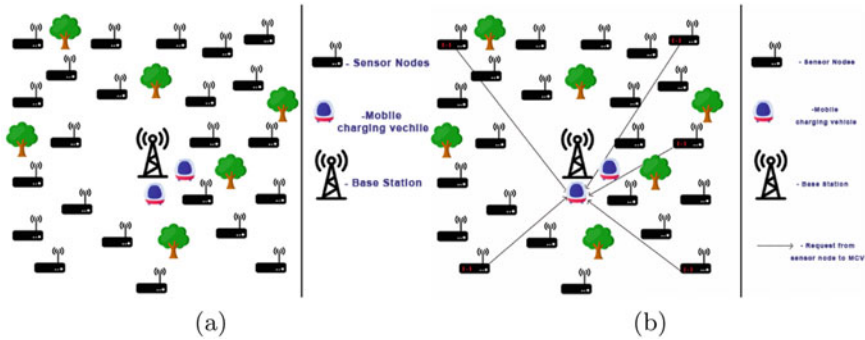
In [1], the sensors are battery-controlled, and their vitality is constrained [4–7]. So, as to take care of the vitality recharging issue in wireless rechargeable sensor networks, numerous scientists have done a great deal of works, which are separated into two kinds which are periodical charging and on-demand charging. A significant number of the current investigations follows periodical charging scheme, in which the mobile charger intermittently crosses the sensors in the sensor network and supplies vitality to the sensors.

### 3 Proposed Work

In this section, we introduce the network components, network model and relevant assumptions. We have shown the problem formulation and the proposed work in detail, and we compared MMCSS algorithm with the existing algorithms RCSS and [8]. We have also provided the illustration of the work for better understanding of the solution. Furthermore, we listed the limitations which identified from the proposed method.

#### 3.1 Network Model

Figure 1 illustrates the network model in this paper. In this paper, the sensor nodes were randomly distributed, and the base station(BS) is equipped with multiple mobile charging vehicle (MCV). Here, the MCVs are responsible to achieve timely energy supplementation. This network model follows on-demand charging scheduling scheme; when one sensor node reaches a particular threshold, it sends the charging request to MCV as shown in Fig. 1b. In this algorithm, we consider two states for a MCV: first, the MCV that is in action state and second, MCV that is in idle state. We assume that no two MCVs can come in to same state either action state or idle state, because in order to send charging requests, sensor nodes should have only one state (MCV in action state). When MCV that is in action falls under certain threshold, then it calls another MCV that is in idle state at the base station. Various MCV charging conditions were discussed in Sect. 3.9.



**Fig. 1** Illustration of figure. **a** Sensors were randomly distributed, two MCVs at the base station **b** Sensors were randomly distributed, two MCVs at the base station, and when each sensor reaches a threshold, it sends a charging request to MCV

We proposed a multiple mobile charging scheduling scheme (MMCSS), where nodes send their charging requests to the MCV, and then, these charging requests are selected based on the selection criteria. Even MCV has limited energy, and when their energy falls down to certain threshold, they will be changed based on their energy criteria continuously until sensors get charged.

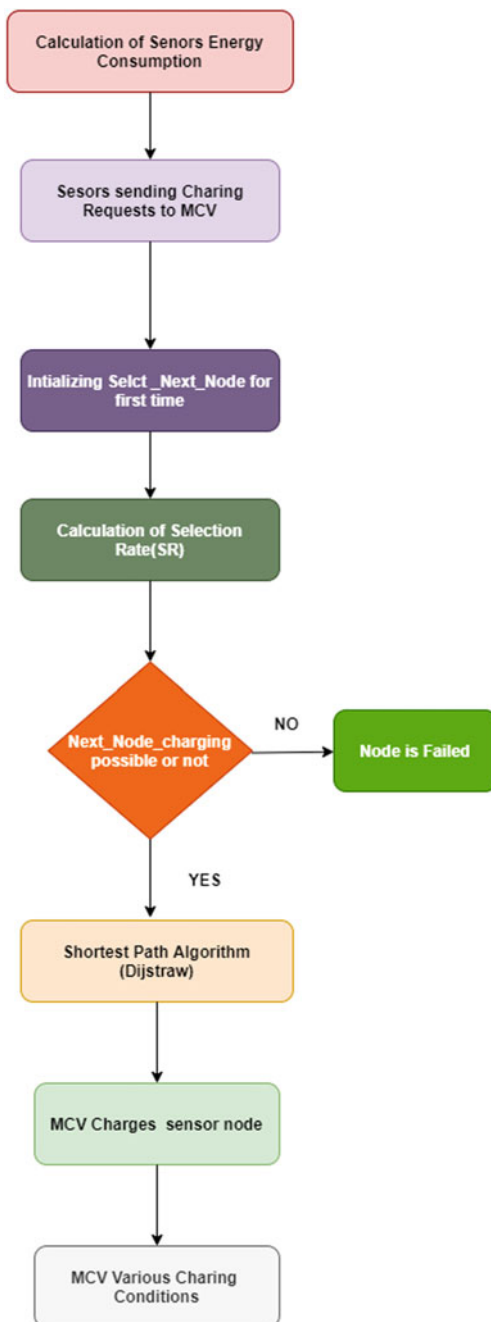
### 3.2 Problem Statement

In the wireless sensor networks (WSNs), due to different environmental locations, the sensor node’s energy consumption changes with time [9–11]. In this paper, we have given a multi-mobile charging scheduling scheme (MMCSS) to charge the requested charging nodes that mainly aims at the following problems, Given mobile charging vehicle (MCV) receiving multiple requests from sensor nodes, this paper details on:

1. How to effectively select the nodes, i.e., based on their selection rate to prevent exhaustion of nodes.
2. How effectively two MCVs charge the nodes one at a time and the difference from the existing papers is that for the next node selection algorithm, we are not considering the parameters based on the network.
3. We are solving the problem in the existing approach, which kills both MCV and node if single MCV does not have the sufficient energy to charge the node and with the increase in simulation time, the number of failed nodes increases in the existing algorithm (Fig. 2).



**Fig. 2** MMCSS work flow diagram



### 3.3 Proposed Algorithm

In this paper, we assume that each node has certain features like node discharge rate, residual energy, node’s distance from charging station, ‘x’ coordinate and ‘y’ coordinate. When each sensor node falls under threshold, it sends charging request to the MCV, and these charging requests get added to the charging matrix. In the charging matrix, the rows are sensors and the columns are features. It selects the next charging node based on selection criteria as discussed below, and MCV also has certain threshold. Managing both MCV and sensor nodes is real challenge.

In this algorithm first, we will check the node for every minute to discharge its energy as depicted in Fig. 3. Here, we are calculating it with the help of Eq. 1. If check for this node is less than or equal to zero, then the node gets failed, and we will delete that request from the charging matrix. If the check for this node is between 0 (or greater than zero) and 225, then we will add the request to the charging matrix. If the request is added for the first time in order to initialize the select next node, we will call the select next node, but after first time to determine the next charging sensor, we will take the selection rate into consideration from the next round and calculate it as described in Fig. 4. And if the check for this node lies between 225 and 500, then the node gets discharged according to its charging rate. If the sensor energy reaches 494J, then only it gets updated into the matrix because it is a tedious task to update the matrix for every time.

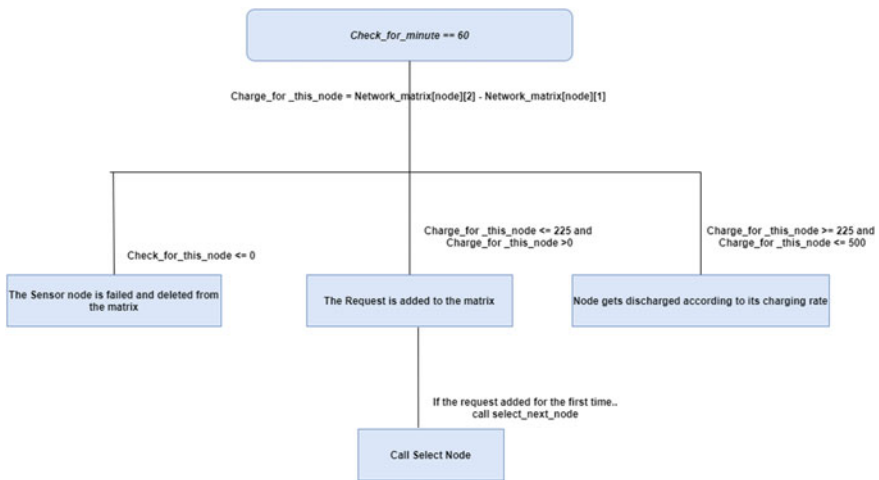


Fig. 3 Sensor nodes and MCVs at various discharges with conditions



Fig. 4 Selection rate flow diagram

$$\begin{aligned}
 & \text{Checkfor node}(X) = \text{matrix}[\text{node}][2] - \text{matrix}[\text{node}][1] \\
 & x = M_{iR} - M_{iE} \\
 & 2 \text{ ----- Residual energy} \\
 & 1 \text{ ----- Node discharge rate}
 \end{aligned}
 \tag{1}$$

Here, in Eq. 1,  $M_{iR}$  is  $i$ th sensor and  $R$ th Column,  $M_{iE}$  is  $i$ th sensor and  $E$ th Column.

### 3.4 Selection Rate (SR)

To select the next charging node, MMCSS calculates its next charging node based on the selection rate criteria (SR). The selection rate criteria are based upon the energy consumption of nodes and distance from requesting node to the current charging node (MCV). It calculates the selection rate for each and every request that is in charging matrix and calculates their normalized distance and normalized energy consumption with the help of Eqs. 2 and 3.

In this paper, we assume that each node has some features: discharge rate, residual energy, node’s distance from charging station, ‘x’ coordinate, ‘y’ coordinate. SR calculation is done in the following section.

$$x = \max_{i=1}^N(M_{id}) \tag{2}$$

$Y = \frac{M_{id}}{x} * 10$  In Eq. 2 Mid is ith sensor and dth distance from requesting node to the current charging node.

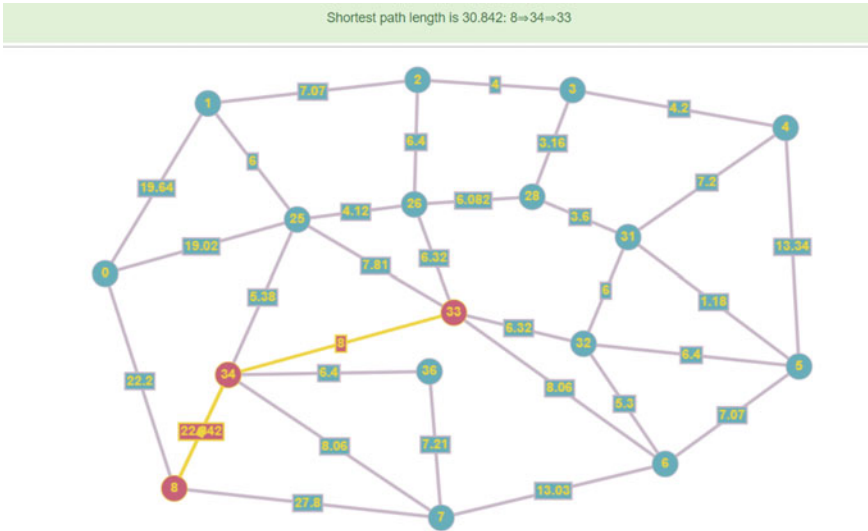
$$u = \max_{i=1}^N(M_{iE}) \tag{3}$$

$V = \frac{M_{iE}}{u} * 10$  In Eq. 3 MiE is ith sensor and E is the energy consumption of each node

### 3.5 SR Calculation

For calculating selection rate(SR), we have taken 20 sensor nodes that were randomly distributed as shown in Fig. 5. When the sensor nodes reach to particular threshold as shown in Table 1, sensor nodes send their charging request to MCV. These charging requests are taken into the charging matrix. Based on this matrix, the SR is calculated. The charging matrix consists of requested charging nodes and their features. By using Eqs. 2 and 3, we found SR for each and every sensor node that is in the given charging matrix. Here, in the charging matrix, the node’s distance from the charging station is calculated dynamically in the algorithm.

4	5.65	200	-4	4
5	11.7	170	-4	11
7	5.38	225	-5	2
2	4.47	175	-2	4
11	5.65	300	4	4
10	11.3	125	8	9
9	8.5	170	-8	3
12	7.81	125	6	-5
11	10.19	110	2	-10
10	2.82	90	2	-2



**Fig. 5** Shortest path from MCV to selected node for charging

**Table 1** Parameters that were taken in this paper [1]

Parameter	Values
No. of sensor nodes	50–90
Operation voltage	2.6 V
Nodes energy capability	500 J
MCV energy capacity	50,000 J
Sensors charging request threshold	225 J
Rate of charging	4 w/s
Moving discharge	12 J/m
Speed of mobile charging vehicle (MCV)	4 m/s

- 0 sensor id
- 1 Energy consumption
- 2 Residual energy
- 3 Node’s distance from charging station
- 4 x coordinate
- 5 y coordinate

$$S = \max \left( \frac{\mathcal{E}}{D} \right) \tag{4}$$

where  $S$  is the selection rate,  $\mathcal{E}$  is denoted as normalized energy consumption rate, and  $\mathcal{D}$  is the normalized distance.

- From the above equations, we get the next charging node as S10

### 3.6 Selection Rate Conditions (SR)

For each and every sensor node that is in the charging matrix, we calculate the selection rate using normalized distance and normalized energy consumption. If the selection rates of two nodes are equal in the charging matrix, then select the node with minimum residual energy. Still if both residual energy and selection rate are equal, then we should select the node with the less distance to the MCV. Here, we selected the node with the less distance to MCV, because if we select the node with long distance, then instead of mobile charging vehicle (MCV) going there, both the MCV and node get failed; instead, we can select the node with less distance to MCV, and then, we can call another MCV that is at the base station. Each and every step of selection rate calculation is given in the selection rate flowchart in Fig. 4 (Tables 2, 3 and 4).

### 3.7 Selection of the Next Charge Possible

When the node is selected based on the selection rate as shown in the flowchart in Fig. 4, we also need to check whether the next node charging possible or not based on the following Eqs. 5 and 6. From Eq. 5, we calculate distance between requesting charging node and current charging node, and then, we find charge needed for the

**Table 2** From Eq. 2 calculation of normalized energy consumption for each and every sensor node

Node ID	1	2	3	4	5	6	7	8	9	10
NEC	3.3	4.1	5.8	1.6	9.1	8.3	7.5	10	9.1	8.3

**Table 3** From Eq. 3 calculation of normalized distance for each and every sensor node

Node ID	1	2	3	4	5	6	7	8	9	10
ND	4.8	10	4.5	3.8	4.8	9.6	7.2	6.6	8.7	2.4

**Table 4** From Eq. 4 calculation of selection rate (SR) for each and every sensor node

Node ID	1	2	3	4	5	6	7	8	9	10
SR	0.68	0.41	1.28	0.42	1.89	0.86	1.04	1.51	1.04	3.4

next charge in Eq. 6. If charge to left is greater than the current MCV threshold, then only MCV charges the node; otherwise, it returns false.

$$\begin{aligned}
 d &= \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \\
 x_1 &= \text{Requesting Charging node}[4] \\
 x_2 &= \text{Current charging node}[4] \\
 y_1 &= \text{Requesting Charging node}[5] \\
 y_2 &= \text{Current charging node}[5]
 \end{aligned}
 \tag{5}$$

$$\text{charge needed for next charge} = (500 - \text{requesting node}[2]) + (\text{distance} * 10)
 \tag{6}$$

**calculate charge needed for the next charge:**

```

charge to left = self.mcv current charge - charge needed for next charge
if charge to left > self.mcv threshold:
return True
else:
return False

```

### 3.8 Shortest Path Algorithm for MMCSS

We used Dijkstra, as a shortest path for MMCSS algorithm. When the next charging node is selected based on the selection rate and after the next charge possible, the MCV moves in the shortest path to the selected charging node by using Dijkstra algorithm as shown in Fig. 5.

---

```

{function Dijkstra(Graph, source):}
create vertex set Q
for each vertex v in Graph:
    dist[v] <- INFINITY
    prev[v] <- UNDEFINED
    add v to Q
dist[source] <- 0
while Q is not empty:
    u <- vertex in Q with min dist[u]
    remove u from Q
    for each neighbor v of u: // only v that are still in Q
        alt <- dist[u] + length(u, v)
        if alt < dist[v]:
            dist[v] <- alt
            prev[v] <- u

return dist[], prev[]

```

---

### 3.9 Conditions for Calling MCV

There were two MCVs in this paper, they are the MCV in action and the MCV in the idle state. We assume that no two MVCs can come into action as either one should be in action or another should be in idle state. Even MCV has certain threshold, these are the conditions of the MCV if it falls under threshold.

**Case 0:** When MCV does not go under threshold after the initialization of the next charging node, it goes to selected sensor node, and then, based on the selection rate (SR), it travels to another sensor node as shown in Fig. 6.

**Case 1:** When the current MCV reaches certain threshold and there is only one priority critical request in its matrix. Then, it will transfer(sends) the request matrix to idle MCV that is at the charging station, and it will come to its charging station. Then, idle MCV becomes the current MCV as shown in Fig. 7.

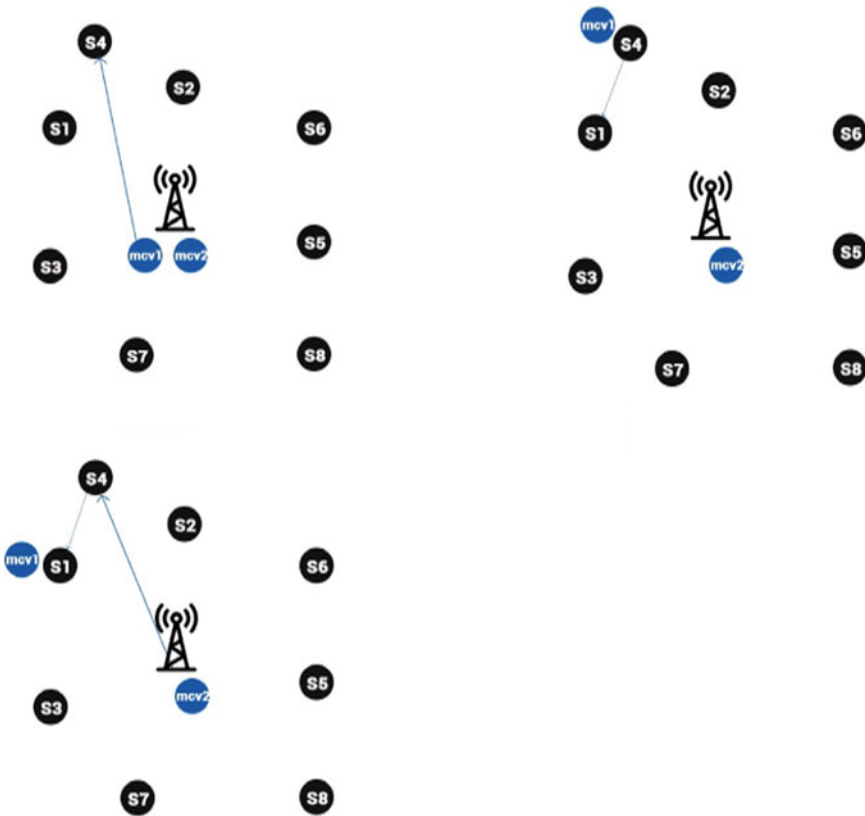


Fig. 6 When MCV does not go under threshold



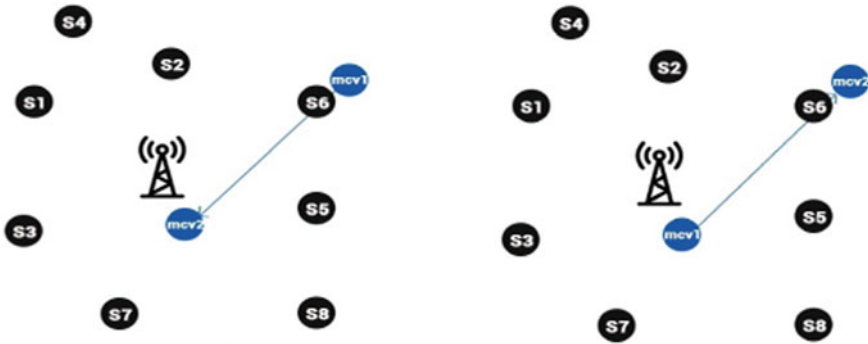


Fig. 7 When MCV goes under threshold and there is only one priority critical request in the matrix

**Case 2:** When the current MCV reaches a particular threshold and there are two priority critical requests in its matrix. It will reach the sensor that is with least distance and then sends the updated matrix to idle MCV and reaches the charging station. Then, the idle MCV that is at the charging station will come in to action before the current MCV reaches to charging station as shown in Fig. 8. But the current MCV becomes idle by not taking any requests from the sensor nodes. When the current MCV transfers the updated matrix to idle MCV followed by this, it also sends the new current MCV’s ID to all the sensor nodes.

### 3.10 Special Case

**Semi-Idle MCV:** When the current MCV transfers the updated matrix to idle MCV followed by this, it also sends the new current MCV’s ID to all the sensor nodes so that all sensor nodes send their charging requests to the new MCV that is going to come in to action. Then, every sensor node should acknowledge to the current MCV when they receive new current MCV’s ID. Suppose if the current MCV node does not get the acknowledge from some sensor nodes, then MCV will go in to semi-idle state.

**Semi-Idle State:** When MCV is said to be in semi-idle state, then it will go to charging station, but still, it receives the request blindly from the nodes and forwards the charging request to the current MCV that is in action.

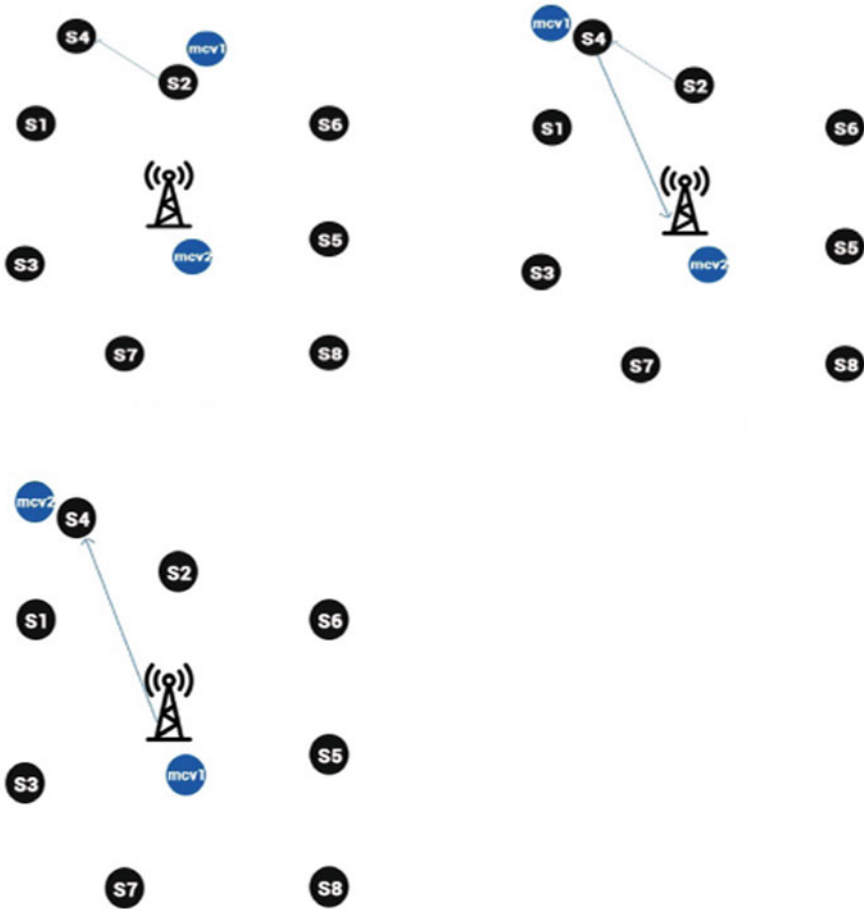
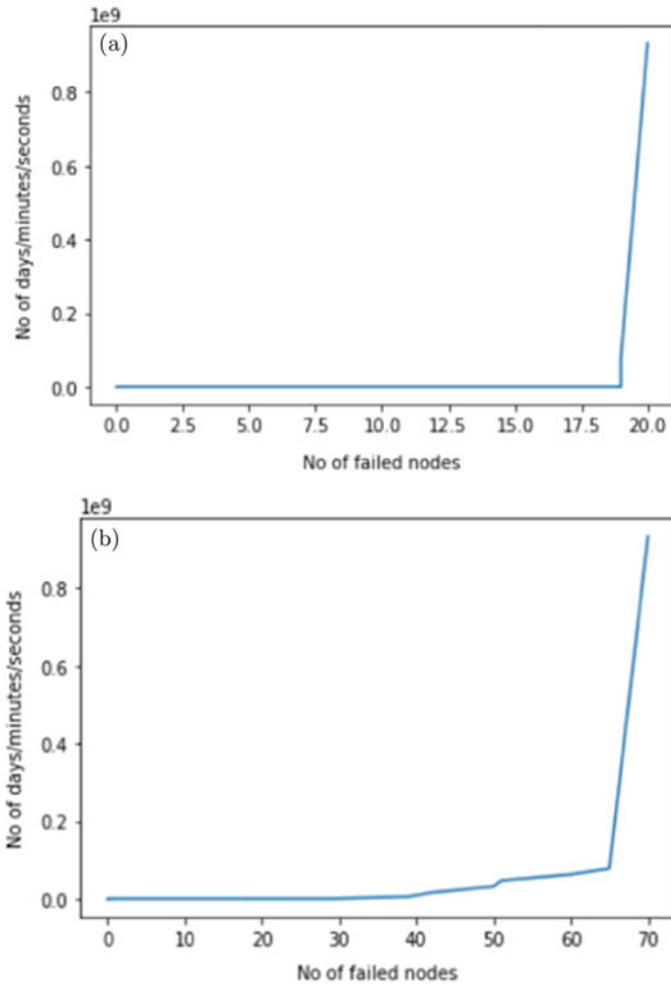


Fig. 8 When MCV goes under threshold and there is two priority critical request in the matrix

## 4 Results and Discussion

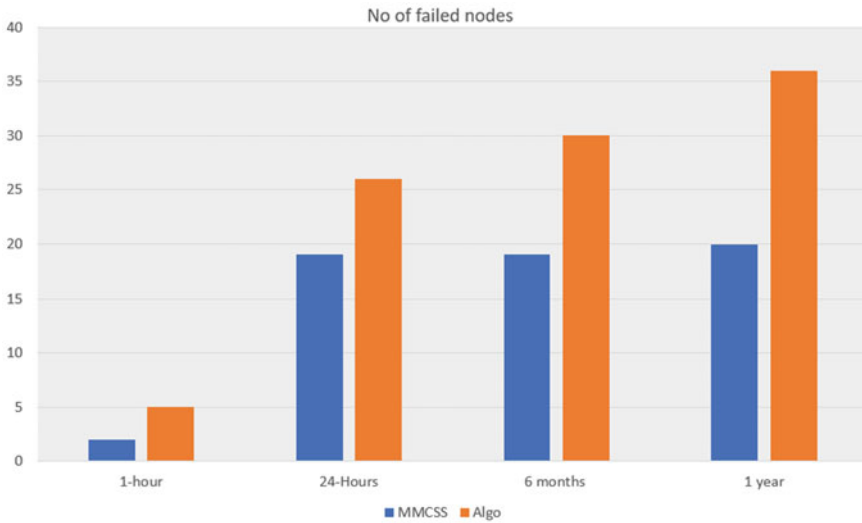
### 4.1 Number of Nodes Die Versus Simulation Time

Lifetime of WSN is the time till the first sensor node in the network dies once the mobile charger has started from the base station. We improved the network lifetime by multiple mobile charging scheduling scheme (MMCSS) algorithm and compared the simulations with the existing algorithms (RCSS). In Fig. 9a, if we increase the simulation time in MMCSS, the number of failed nodes was very less compared to the existing algorithm. This is measured continuously by recording the number of sensor nodes that die by increasing the simulation time. Here, in Fig. 10, as the simulation



**Fig. 9** Illustration of figure. **a** (MMCSS) Simulation time versus number of failed nodes **b** (Existing algo) Simulation time versus number of failed nodes

time of MMCSS increases, the number of failed nodes decreases drastically, whereas in the existing algorithm, as the simulation time increases, the number of failed nodes increases drastically. Here, we have randomly generated sensor nodes in the area of  $150\text{ m} \times 150\text{ m}$ , and the total number of nodes here in this paper we have taken are 70 to 100. It is clear in Fig. 10 that the total number of sensor nodes that were survived using this algorithm multi-mobile charging scheduling scheme (MMCSS) are always greater than the existing algorithm even there are large number of sensor nodes. This is why we have taken distance, energy consumption of each and every node in to consideration for the selection rate which were not done in the existing algorithms.



**Fig. 10** No. of failed nodes of MMCSS versus existing algorithms

## 4.2 Performance Evaluation

In MMCSS algorithm, while selecting the request charging node that was initially in the charging matrix, we took both energy consumption and distance as important factors to formulate the problem as shown in Eqs. 2 and 3 which explains that MMCSS is better than the existing algorithm (RCSS). We have tested MMCSS with the following parameters (inputs). We did not take the parameter beta which is decided by the network as shown in the existing algorithm. For this algorithm, we have taken different kinds of data sets that were medium, denser and sparse data sets. With this input, we performed different kinds of simulations. Here, in this algorithm, we have taken total number nodes as 100 as an example, and MMCSS survives maximum number of nodes in the experiment as shown in Fig. 9a, while in the existing algorithm, the failed number of nodes was increased drastically as shown in Fig. 9b, which shows that MMCSS has improved the total number of failed sensor nodes as shown in Fig. 9a.

## 5 Conclusion

In this paper, we have given an algorithm called multi-mobile charging scheduling scheme (MMCSS) for wireless rechargeable sensor networks. In this MMCSS scheme, we have taken lot of parameters in to consideration like MCV's threshold, sensor node features like energy consumption, distance between sensor nodes,

MCV's residual energy and MCV's energy. Here, in this algorithm, while calculating the selection rate, we have taken the energy consumption and distance in to consideration that makes this algorithm more efficient. The selection of MCVs and charging the sensors within due time has proven that MMCSS is better than the existing algorithms.

## References

1. P. Zhong, Y. Zhang, S. Ma, X. Kui, J. Gao, RCSS: a real-time on-demand charging scheduling scheme for wireless rechargeable sensor networks. *Sensors* **18**(5), 1601 (2018)
2. W. Xu, W. Liang, X. Jia, Z. Xu, Z. Li, Y. Liu, Maximizing sensor lifetime with the minimal service cost of a mobile charger in wireless sensor networks. *IEEE Trans. Mob. Comput.* **17**(11), 2564–2577 (2018)
3. C. Lin, S. Wei, J. Deng, M.S. Obaidat, H. Song, L. Wang, G. Wu, GTCCS: a game theoretical collaborative charging scheduling for on-demand charging architecture. *IEEE Trans. Veh. Technol.* **67**(12), 12124–12136 (2018)
4. P. Zhong, Y.-T. Li, W.-R. Liu, G.-H. Duan, Y.-W. Chen, N. Xiong, Joint mobile data collection and wireless energy transfer in wireless rechargeable sensor networks. *Sensors* **17**(8), 1881 (2017)
5. A. Liu, M. Huang, M. Zhao, T. Wang, A smart high-speed backbone path construction approach for energy and delay optimization in wsns. *IEEE Access* **6**, 13836–13854 (2018)
6. J. Gao, J. Wang, P. Zhong, H. Wang, On threshold-free error detection for industrial wireless sensor networks. *IEEE Trans. Ind. Inform.* **14**(5), 2199–2209 (2017)
7. Y. Liu, K. Ota, K. Zhang, M. Ma, N. Xiong, A. Liu, J. Long, QTSAC: an energy-efficient mac protocol for delay minimization in wireless sensor networks. *IEEE Access* **6**, 8273–8291 (2018)
8. J. Wang, T. Si, X. Wu, X. Hu, Y. Yang, *Sustaining a perpetual wireless sensor network by multiple on-demand mobile wireless chargers*, in *2015 IEEE 12th International Conference on Networking, Sensing and Control* (IEEE, New York, 2015), pp. 533–538
9. X. Liu, N. Xiong, N. Zhang, A. Liu, H. Shen, C. Huang, A trust with abstract information verified routing scheme for cyber-physical network. *IEEE Access* **6**, 3882–3898 (2018)
10. K. Ota, M. Dong, J. Gui, A. Liu, QUOIN: incentive mechanisms for crowd sensing networks. *IEEE Network* **32**(2), 114–119 (2018)
11. M. Dong, K. Ota, A. Liu, RMER: reliable and energy-efficient data collection for large-scale wireless sensor networks. *IEEE Internet Things J.* **3**(4), 511–519 (2016)

# CRAWL: Cloud-Based Real-Time Interconnections of Agricultural Water Sources Using LoRa



P. Sree Harshitha, Raja VaraPrasad, and Hrishikesh Venkataraman

**Abstract** Water wells are traditional sources of water for agricultural needs. Particularly, due to ever-increasing demand from the exponentially growing population, there arises a need to balance the water demand and conserve water resources. The current systems adopted are very rudimentary and cannot be scaled. Hence, a major challenge is to investigate equitable allocation of water from excess sources to deficit ones. In this regard, this work proposes an Internet of things (IoT)-based technique to effectively manage and utilize water resources by connecting wells, ponds, lakes, etc., with a smart network of pipelines. The interconnected wells are configured with a sensors-actuation mechanism and communication devices that sense the water scarcity among wells in a network and then redistribute water accordingly. A low-cost and low-power IoT technology is used for data acquisition from sensors to auto control the actuators. A long-range wireless communication between water sources is achieved by deploying long-range (LoRa) modules, a prototype is developed, and a cloud-based app is deployed. The CRAWL—cloud-based real-time interconnections of agricultural water sources using LoRa system—is scalable and hence is capable of being developed as a rugged and robust system that can solve problems of floods, burst of rains and water shortages at not only panchayat/Taluka level, but also scaled upto district and state levels in the country.

**Keywords** Water-sharing system · Water resource management · IoT · LoRa

---

P. Sree Harshitha (✉) · R. VaraPrasad · H. Venkataraman  
Indian Institute of Information Technology, Sri City, India  
e-mail: [sreeharshitha.p@iiits.in](mailto:sreeharshitha.p@iiits.in)

R. VaraPrasad  
e-mail: [yrv.prasad@iiits.in](mailto:yrv.prasad@iiits.in)

H. Venkataraman  
e-mail: [hvraman@iiits.in](mailto:hvraman@iiits.in)

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2021  
S. M. Thampi et al. (eds.), *Advances in Computing and Network Communications*,  
Lecture Notes in Electrical Engineering 735,  
[https://doi.org/10.1007/978-981-33-6977-1\\_38](https://doi.org/10.1007/978-981-33-6977-1_38)

## 1 Introduction

Water has always been an essential asset for all forms of life, where agriculture demands enormous water resources. In the present days, most of the agriculture fields do not have sufficient amount of water for farming, while some of the fields have excess amount of water. Farmers depend either on open water wells or on bore wells. Extensive dependency and usage of bore wells have resulted in depletion of the water table beyond its threshold lower limits. Critical drawback of this practice is that if water is deficit at the times of crop yielding can hugely dent their harvest. To mitigate this effect, water transfer mechanisms are effective solutions to balance water demands and ordeals of dry season. Good water scheduling system requires the planning, water-saving agricultural systems to improve farming water usage efficiency in the dry and semi-arid regions with close collaboration among farmers [1]. Arranging and usage of already laid pipeline system efficiently are expensive and time taking. In such manner, a paper with a novel advisory structure named Zobhana Jala Sambaddha (ZJS) to provide an automated arrangement when compared with manual pipe laying mechanism [2]. Some farmers in Maharashtra connected 30 wells to create a perennial water bank to irrigate 150 acres of agricultural land. However, it failed to take off because of the lack of planning and control units. IoT technology, including radio frequency identification technology, sensor technology, smart technology, nanotechnology and other innovations, is a modern system of information processing and acquisition. IoT is a connectivity between people and things, using any network and any service at any time, in any location, and is therefore a large, diverse global network infrastructure [3]. The strong motivation behind this work is that every farmer should have access to water for their irrigation needs at required times. The proposed CRAWL IoT architecture can defuse the crisis if water is redistributed from surplus bodies to deficit one by monitoring the water level of each remote node and by remote actuation of valves and motors using LoRa as communication mode which has distinct features than existing ones, i.e. Wi-Fi, 3G. The following are the novelty benefits for smart agriculture using CRAWL system:

- Long coverage
- Low power consumption
- Low maintenance system.

This paper is organized as follows. Section 2 tells about related work, and Sect. 3 tells about real-time modelling architecture and implementation. The experimental results are discussed in Sect. 4, conclusion in Sect. 5 and future work in Sect. 6.

## 2 Related Work

The significant challenge recognized in the current space is to give farmers needed water for the crop and convenient support at the major crop yielding period. The inconsistent distribution of water is uplifted by political changes, unequal resource management and climatic irregularities.

In large-scale networks, water is transported along the open water channels under the power of gravity alone, where interlinking of five river basins of Rajasthan was established with out pumping systems [4]. A similar system was implemented in Heihe river basin [5]. A water allocation scheme was implemented with more than 60 water decision-makers at more than 40 locations to implement and maintain real-time responses based on the current relationship between supply and demand of water. An integrated environment was developed to manage data and facilitate cooperation among different levels of users. Notably, a model was developed to predict water consumption and improve water management on irrigation schemes in semi-arid Brazil [6]. Authors asserted the model's potential to be implemented in 62 public irrigation schemes in 730,000 ha of irrigable area to regulate irrigation water consumption. In Italy, the Ofanto water irrigation plot among the biggest multi-cropped watered field is integrated with open-branched distribution network in which every channel has a control unit and flow sensors using supervisory control and data acquisition (SCADA) [7]. A customary SCADA framework would have cost 100,000 dollars to set up and more to maintain.

One of the authors proposed a technique to built a real-time water level monitoring of storage in water distributed networks (WDNs) and remote actuation of valves utilizing the IoT empowered devices to regulate water supply across the water distribution network from the nodal reservoir [8]. Other author proposed a smart irrigation and tank automated monitoring system using ultrasonic sensors to measure the water level inside the tank which switches ON/OFF the motor compared with the threshold level [9]. The work proposed by the author for an automated drinking water system to each home unit is provided with a separate flow sensor which calculates flow rate and arrests the water using valves [10]. The efficient use of water resources for stability of the agriculture irrigation systems is modelled based on the IoT using hardware and network architectures with software process control [11]. IoT application for remote monitoring and control of water in agricultural fields is based on the analysis of data collected by the wireless sensor network [12]. The system overcomes limitations of traditional agricultural procedures by utilizing water resource efficiently and also reducing labour cost, and a prototype of the mechanism is carried out using TICC3200 launch pad interconnected sensors modules with other necessary electronic devices [13]. In large-scale irrigation network management, closed-loop control is adopted in open water channels where flow is controlled on the basis of the water level [14]. The IoT cloud uses both the HTTP and MQTT protocols to provide users with visual and timely sensor data [15].

The above-mentioned plan empowers data to the users and form a network through Wi-Fi or 3G where as the CRAWL system is identified to come up with great solutions for water-sharing mechanism among farmers and from excess water resource available for sustainable growth of agriculture.



### 3 CRAWL—Real-Time Modelling Architecture and Implementation

Automated water shared systems focus on proper sharing of water to all water storage in farms and shut water supply with a predefined threshold limit. Here, architecture conceptualized on the water requirement for each farm, and accordingly, water is shared and supplied using valves and motors. The system allows the flow of water in concurrence with threshold levels, and real-time parameters are measured and regulated with flow sensor, level sensor and Arduino UNO. The real-time data would be stored in an aggregator on a temporary basis and later will be sent to the cloud. Figure 1 shows the architecture of the proposed system, in which automated water shared between water bodies is achieved using Arduino, water level sensor, flow sensors, valves and motors. Any discrepancy is intimated to the main hub, i.e. aggregator using LoRa connected to the Arduino board. The main function of the Arduino UNO is to read the data from the sensor and send the data to the aggregator node and to optionally process it. The long-range communication in real time is essential as two water bodies are far from each other, and the application is mainly used in rural areas, as Internet application is not a viable option. LoRa module is the most efficient mode of communication in long ranges in the absence of Internet-enabled services.

Arduino acts as the remote node and node MCU as the aggregator for data aggregation, a transformation which is connected to the Internet to inject the data into the cloud. For administrator visualization, a GUI and mobile app has also been developed that conveys information and reflect actions which the user can take. The previously mentioned drawbacks make a genuine difficulty in settling the current challenging situations of the country individuals in farming area.

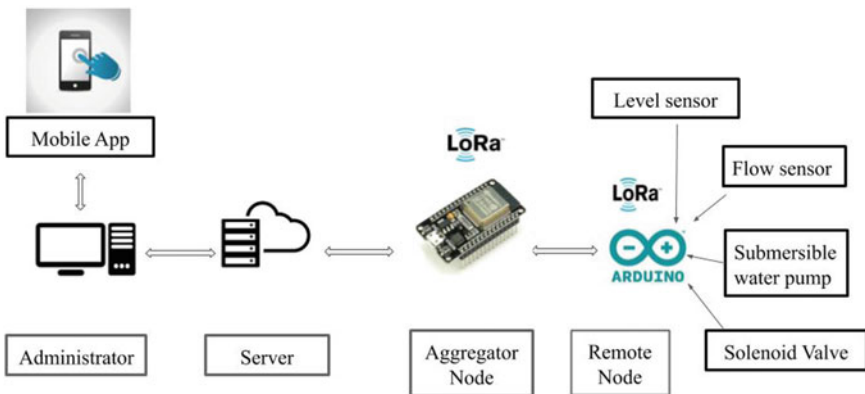


Fig. 1 CRAWL architecture

### 3.1 Implementation of CRAWL

Figure 2 shows the network of pipelines from the top view in which all farmlands are interconnected with each other, each farmland has a node with a submersible water pump motor, and solenoid valves are actuators which can automatically pump and fill water by communicating with server. The supply of water for irrigation water storage can be well executed using pipeline networks and in an automated way is by embedding all the components and fixing the threshold level to the Arduino. The Arduino UNO board plays a vital role, and it is interfaced with a level and flow sensor, valves and actuators of each farm unit separately. The sensors monitor the water level in the nodes and send appropriate signals to the Arduino. It sends the level sensor data to the server where node MCU is used as microcontroller which decides and sends back particular actuation messages to the Arduino to turn ON/OFF the motor and solenoid valve. It processes these values, and based on the actuation messages, actuators are triggered at particular nodes.

To supply water to the deficit node whenever needed in an optimized and well-designed manner from different water bodies all over the network. The nodes which pump water have to switch ON both motor pump and solenoid valve, where the receiver node has to switch ON only valve to get water. Now, server decides which node has to be chosen to pump water from. To do so, the data required is the water level of all the nodes in the network frame, and then, a particular node could be chosen for actuating the valves and motors accordingly. To pump the excess amount of water from the upper threshold water body using the submersible water pump

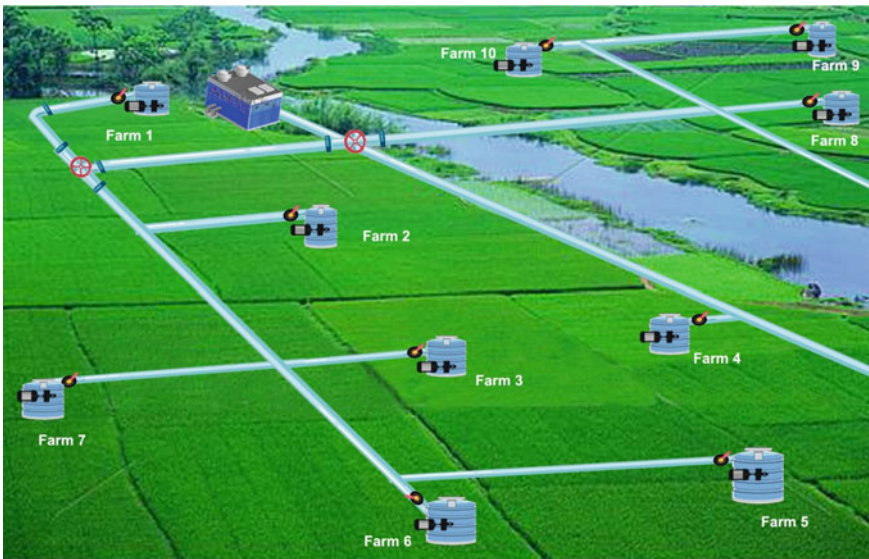


Fig. 2 Typical implementation layout of the CRAWL architecture

to the deficient water body to reach the required threshold level, flow sensors are deployed in the system to check whether the same flow rate is observed at both water bodies, where one is a sender of water and the other is the receiver. So, even water leakage can be detected.

Hence, adopting this method of interconnected network in a geographical distributed water body such that surplus water from one source of water could be pumped into the other source of deficit farm in a sound economical manner.

### **3.2 Flow chart**

Flow chart is a diagram of the sequence of actions of nodes involved in complete network. Figure 3 shows a basic understanding of the work flow in the proposed framework. Every node sends its water level sensor data for particular defined interval, and server will store the data of every node of a network and sends back the actuation messages depend on their level compared to threshold levels. If any request from the deficit source received by the server, then it compares with other water levels and selects and sends actuation messages to the surplus node to share water. The actuation message sent to the surplus node is to pump water through submersible motor, and solenoid valve should be actuated. The actuated message sent to deficit source is to switch ON the solenoid valve to receive water from any other surplus node. The real-time comparison of the process parameters is continuous one, so surplus water body should not fall below the deficit level. Once the requirements achieved by the network, then valve and pump of all nodes will be turned OFF as per the server instructions.

## **4 Experimental Set-Up and Results**

The main aim of our work is to develop and demonstrate algorithms to come up with the best distribution regimes using IoTs to get real-time data about water level in each well and storage tank, and therefore, a solution is provided by constructing an electronic system that has the capability of supplying water to a certain water body when its water level is below a threshold. Generally, SX1301 is used as an aggregator LoRa module as it is an 8-channel gateway, but very costly, compared to it the cost of LG01N which is very less and a single channel. LG01N is used as a LoRa aggregator that supports multiple working modes with 50–300 sensor nodes and also bridges LoRa wireless network to an IP network with Wi-Fi, Ethernet, 3G/4G cellular. An algorithm is written at aggregator node in such a way to avoid the repetitive conditions and written the final possible conditions in Table 1 remained are only 10 with 1 exceptional condition, i.e. all 3 nodes with deficit water bodies then at that time, the comparison was made among the water levels of 3 nodes and water will be shared equally among those nodes.

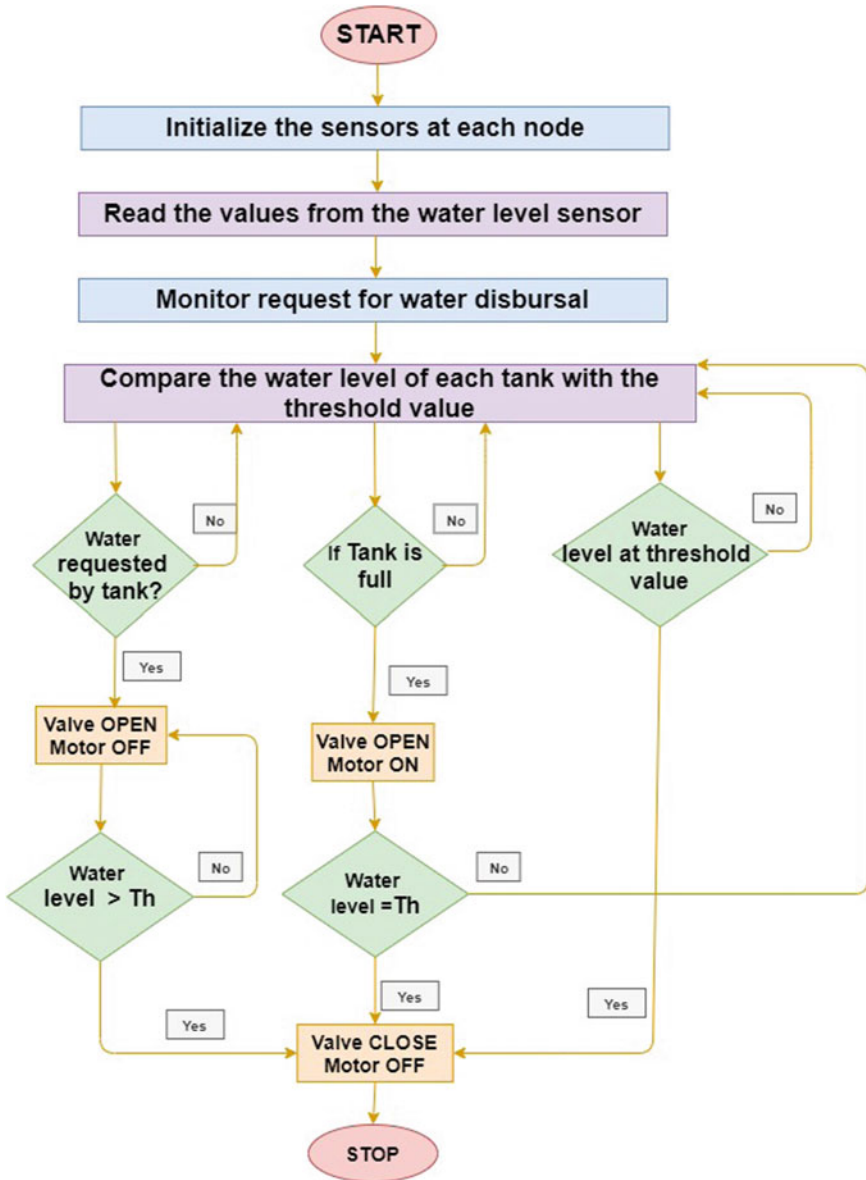


Fig. 3 Flow chart for the CRAWL process

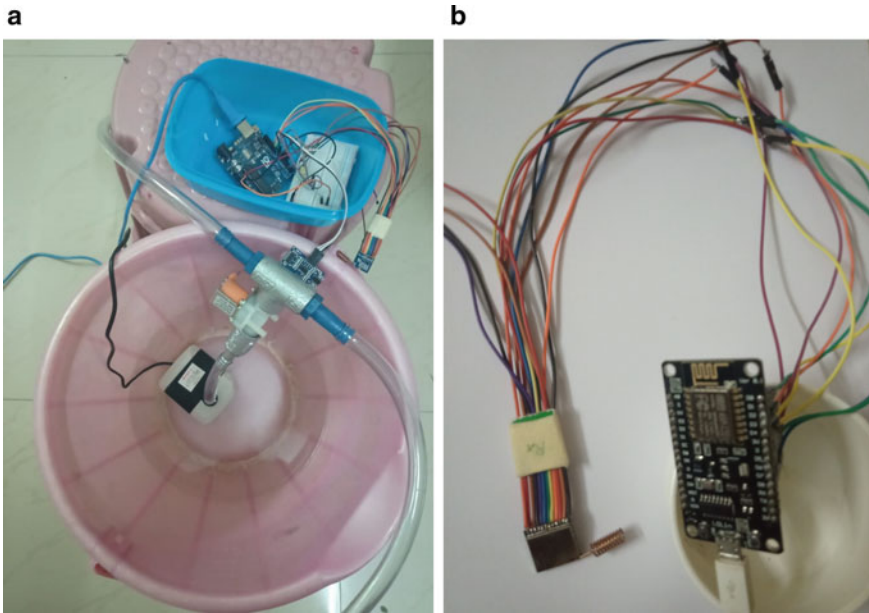
**Table 1** Switching response of remote nodes towards water level conditions

Node1	Valve1	Motor1	Node2	Valve2	Motor2	Node3	Valve3	Motor3
Deficit	Open	OFF	Surplus	Open	ON	Surplus	Open	ON
Surplus	Open	ON	Deficit	Open	OFF	Deficit	Open	OFF
Ok	Open	ON	Deficit	Open	OFF	Deficit	Open	OFF
Ok	Close	OFF	Surplus	Open	ON	Deficit	Open	OFF
Deficit	Open	OFF	Ok	Open	ON	Ok	Open	ON
Surplus	Close	OFF	Ok	Close	OFF	Ok	Close	OFF
Ok	Close	OFF	Surplus	Close	OFF	Surplus	Close	OFF
Surplus	Close	OFF	Surplus	Close	OFF	Surplus	Close	OFF
Ok	Close	OFF	Ok	Close	OFF	Ok	Close	OFF
Deficit	Open	OFF	Deficit	Open	OFF	Deficit	Open	OFF

#### 4.1 Experimental Set-Up

Figure 4a shows remote node which consists of an Arduino MCU, LoRa module, ultrasonic sensor, flow sensors, submersible pump and solenoid valves. Arduino collects the water level in the wells every minute from an ultrasonic sensor and sends it to the aggregator node using LoRa which is connected to Arduino using a serial peripheral interface (SPI), and power is taken from A 3.3 V output. The power supply is designed to 5 V to Arduino, ultrasonic sensor, flow sensor 12 V external power supply is used for motor and valve. Remote nodes take reading of the level sensor and check the status of the valve fully open or fully closed and motor ON/OFF for every minute and send them to the aggregator using LoRa. Communication between remote nodes and aggregator is half-duplex. Based on the command received, Arduino can energize or de-energize the valve to OPEN/CLOSE and the motor to ON/OFF.

In Fig. 4b, the aggregator node collects the data from sensor nodes, processes and pushes the data to the central server whenever it receives data from any of the remote nodes. Again, it receives command from the centralized server and transmits to the sensor node to OPEN/CLOSE the valves and motors to ON/OFF. Aggregator LoRa is always in receiving mode to receive data from data from remote nodes. The LoRa wireless network allows users to send or receive data for long ranges at low data-rates and provides high interference immunity. In the proposed system, the expected range between the two nodes is approximately 2 km. The range of communication has become a basic part of the IoT framework. Lora gives long-range communication up to 10–40 km in rural zones and 1–5 km in urban zones.



**Fig. 4** Types of nodes used

## 4.2 Experimental Results

The nodes will send the water level of each water well to the server which responds accordingly. The threshold limit with 0–60 cm then it's a surplus condition, if it is between 60–150 cm then it's considered as a safe node, if the threshold limit is more 150 cm then it is in deficit condition. Server will be ready with all nodes data so that it can easily compare and select the condition from which nodes the water has to pump to equilibrium all the nodes to form all nodes to be in safe condition. Here, a case is taken with node-1 235 cm which is below the lower threshold limit and is in deficit condition, so that it requests server for water need. Now, server compares and sends the message to node-1 to open the valve to get water from any other surplus water bodies. Now, the server checks for node-2 and node-3 which is of water 80 cm and 30 cm, respectively. So, the water from node-3 which is in surplus condition is pumped to node-1, but noting the condition that the threshold limit of node-3 should not fall below the safe condition and node-2 will also contribute if the comparison of the node-3 is below node-2. The water will be pumped to the node-1 from all the nodes which is of upper threshold level in the shortest path to balance all the water levels of the nodes.

Figure 5 shows the snapshot of node-1 sending data to server using LoRa with time stamp. The data of water level and flow rate sent from node-1 to server, and as a result, comparison is made in server with the remaining nodes, so that node

```

COM10
14:07:13.809 -> Node1 Receive  D1/V/ON/M/OFF
14:07:13.809 -> Node1_valve_on/MOTOR_OFF
14:07:15.587 -> Send Message
14:07:15.587 -> Send Message
14:07:21.578 -> Send Message
14:07:27.579 -> Send Message
14:07:35.631 -> TxDone
14:07:37.614 -> TxDone
14:07:39.604 -> TxDone
14:07:39.779 -> Node1 Receive  D1/V/ON/M/OFF
14:07:39.779 -> Node1_valve_on/MOTOR_OFF
14:07:41.626 -> Send Message
14:07:41.626 -> Send Message
14:07:47.614 -> Send Message
14:07:53.604 -> Send Message
  
```

Fig. 5 Nodes to aggregator time stamp

never falls into the deficit condition. Now, the node-1 sends the data in the format of  $N1/235/95$  where  $N1$  is node-1 and 235 is water level data and 95 is flow rate of node-1.

As 235 cm comes under lower threshold value in the server, the actuation messages sent to the node-1 is “Valve-ON and Motor-OFF” in the following format  $D1/V/ON/M/OFF$  for which node-1 reacts accordingly. Here, flow rate of all nodes also compared to know the reliability of the system.

Figure 6 is the snapshot of serial monitor of server part the receiving data from nodes and sending back the actuation messages to the nodes, respectively. Water level data received from Node 1, 2, 3 is 235 cm, 80 cm and 30 cm, respectively. Now, server algorithm decides that

- Node-1 is in deficit condition
- Node-2 is in safe condition
- Node-3 is in surplus condition

Next sequence of operation is sending the actuation messages to all the three nodes to react accordingly. Algorithm is programmed in such a way for every node where the node which receives water has only to switch ON the valve, where the node which pumps water from that node has to switch ON both valve and motor, and if the node is in safe state, then the valve and motor have to be in OFF condition.

Now accordingly, actuation messages were sent to the three nodes which indicate the decision messages  $D1$ ,  $D2$  and  $D3$  are as follows:

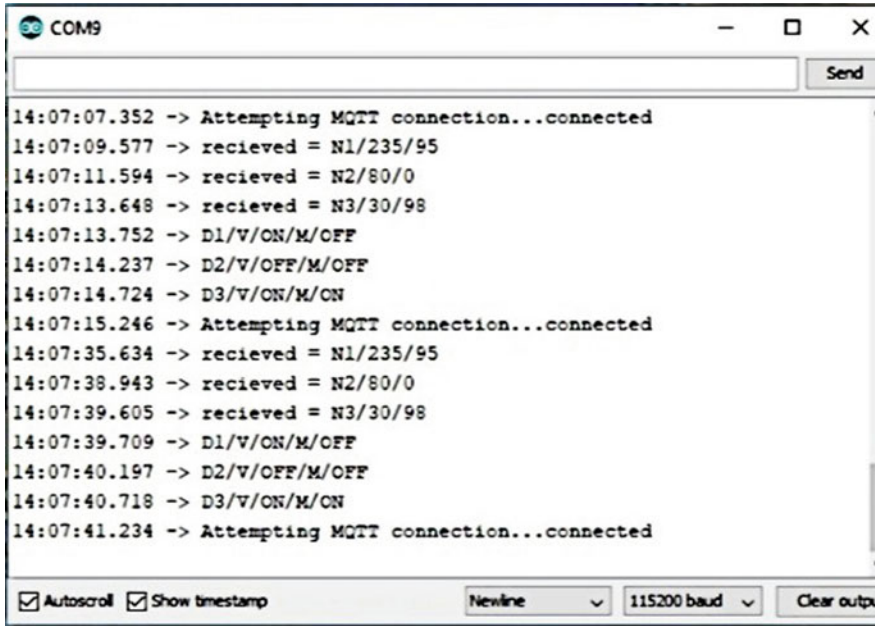


Fig. 6 Time stamp from aggregator to cloud

- D1/V/ON/M/OFF
- D2/V/OFF/M/OFF
- D3/V/ON/M/ON

Figure 7 shows the messages received from server to cloud using Hive MQTT (Message Queuing Telemetry Transport), and it is a free open source and works efficiently where the data is updated in faster manner. The data will be updated in to the cloud in which Hive MQTT acts as a central distribution hub for publishing and subscribing messages using Wi-Fi mode.

Later for administrator viewing purpose, a graphical user interface (GU) is developed to update continuously using Java script as shown in Fig. 8. The flow rate and water level of each node are measured and displayed in Web page using hyper text markup language (HTML) and cascading style sheets (CSS). This displayed data can be easily understood by even layman.

In addition to it, a mobile app has also been developed in the view of farmers to understand the whole procedure clearly as shown in Fig. 9. Nodes will be in safe condition even after sharing their sources and will also get water from any surplus body when they are in need of water during crop yielding as it is critical requirement. The data from node to server and from server to cloud and from cloud to GUI and finally from GUI to mobile app has tested many number of times with different conditions works efficiently for CRAWL network.



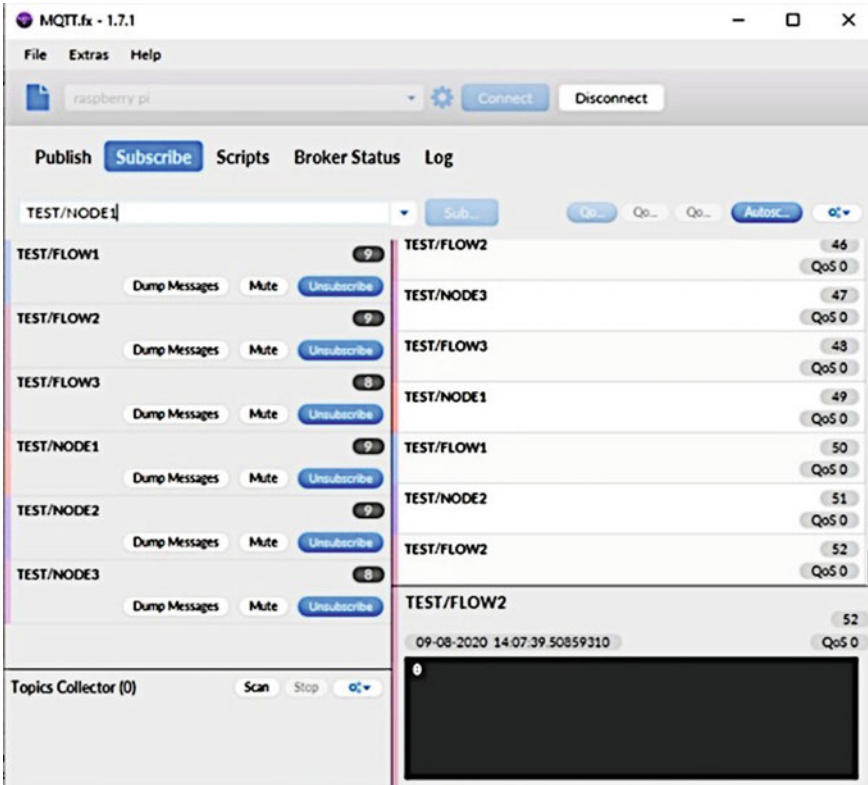


Fig. 7 Data received at cloud through MQTT

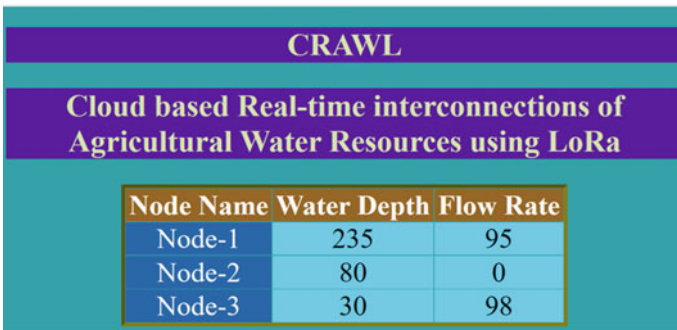
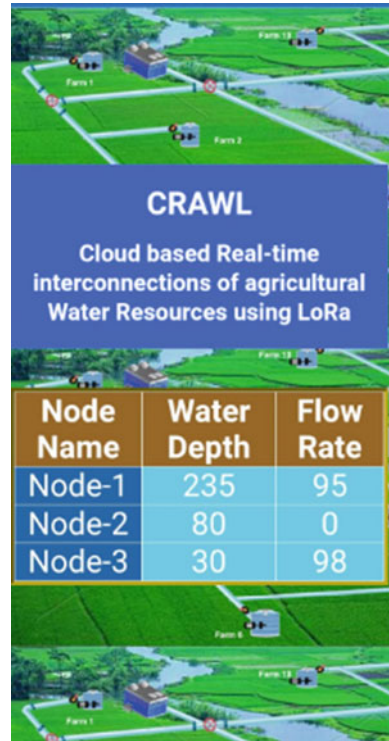


Fig. 8 Snapshot of data received from cloud to in built GUI

**Fig. 9** Snapshot of mobile app



## 5 Conclusions

The rapidly increasing population has led to the need for innovative methods in irrigation to provide food security for future generations through water-sharing mechanisms. Related works are mostly on a single node implementation even if it is extended to form a network, the working mechanisms are semiautomatic or manual, and the large networks work with gravity or using hydraulic-based system. This paper presents a low-cost IoT-based solution using LoRa for long-range communication for monitoring and controlling water-sharing networks in fully autonomous manner in efficient manner. Our CRAWL system covers long coverage 2 km between two nodes without Internet facility at node level which is also energy efficient system as the server used is node MCU with LoRa which is of low power consumption and highly reliable system as the flow sensors are used for clear comparison. This optimized model shares water in a robust manner with appropriate quality and quantity at the right time to safeguard the crop. This not only shows that distribution of water is scalable across a village, but also opens up an opportunity for work sharing and management across a Panchayat or municipal level, leading to low cost yet near optimum water distribution.

## 6 Future Work

Simulating a multi-hop cluster-based communication between nodes. The expected working range between the two LoRa modules is 2 km, but practically, the distance between server and node might be more than above stated value, so to enhance the range to communicate with the server, a cluster-based multi-hop communication is proposed using LoRa. A clustering mechanism is proposed for communication between nodes. Further, a LoRa-Gateway design approach is considered for wireless communication. The next step is to develop an architecture for multiple clusters in hardware and compare that with simulating results.

**Acknowledgements** The authors acknowledge Department of Science and Technology (DST), Govt. of India, for their kind support.

## References

1. X. Deng, L. Shan, H. Zhang, N.C. Turner, Improving agricultural water use efficiency in arid and semiarid areas of China. *Agric. Water Manage.* pp. 23–40 (2006). <https://doi.org/10.1016/j.agwat.2005.07.021>
2. J.S. Sunil, K. Manasa, V.P. Raja, V. Hrishikesh, Advisory framework to interconnect distributed water bodies targeting agriculture farms, in *International Conference on Applied Computing and Information Technology* (IEEE, Deggendorf, Germany, 2020)
3. A. Ghasempour, Internet of things in smart grid: architecture, applications, services, key technologies, and challenges. *Inventions* **4**(1):22, 1–12 (2019). <https://doi.org/10.3390/inventions4010022>
4. S. Vyas, G. Sharma, Y.P. Mathur, V. Chandwani, Interlinking feasibility of five river basins of Rajasthan in India. *Perspect. Sci.* **8**, 83–86 (2016). <https://doi.org/10.1016/j.pisc.2016.04.002>
5. Y. Ge, X. Li, C. Huang, Z. Nan, A decision support system for irrigation water allocation along the middle reaches of the Heihe River Basin. Northwest China: *Environ. Modell. Software* **47**, 182–192 (2013). <https://doi.org/10.1016/j.envsoft.2013.05.010>
6. M.T. Folhes, C.D. Renno, J.V. Soares, Remote sensing for irrigation water management in the semi-arid Northeast of Brazil. *Agric. Water Manage.* pp. 1398–1408 (2009). <https://doi.org/10.1016/j.agwat.2009.04.021>
7. L. Levidow, D. Zaccaria, R. Maia, E. Vivas, M. Todorovic, A. Scardigno, Improving water-efficient irrigation: prospects and difficulties of innovative practices. *Agric. Water Manage.* pp. 84–94 (2014). <https://doi.org/10.1016/j.agwat.2014.07.012>
8. S. Chinnusamy, P. Mohandoss, P. Paul, R. Rohit, N. Murali, S.M. Bhallamudi, S. Narasimhan, S. Narasimhan, IoT enabled monitoring and control of water distribution network, in *1st International WDSA/CCWI 2018 Joint Conference*, Kingston, Ontario, Canada, July 23–25 (2018)
9. K. Kumar, H. Md. Azhar, N. Srinivasan, J. Albert Mayan, Smart irrigation and tank monitoring system, in *IOP Conference Series: Materials Science and Engineering*, Bangkok, Thailand, pp. 1–7 (2019). <https://doi.org/10.1088/1757-899X/590/1/012035>
10. M. Nivetha, S. Sundaresan, Automated drinking water distribution using arduino. *Int. J. Trend Sci. Res. Dev. (IJTSRD)* pp. 698–702 (2017). <https://doi.org/10.31142/ijtsrd26414>
11. S. Li, Application of the internet of things technology in precision agriculture irrigation systems, in *International Conference on Computer Science and Service System*, United States, pp. 1009–1013 (2012). <https://doi.org/10.1109/CSSS.2012.256>

12. F. Karim, F. Karim, A. Frihida, Monitoring system using web of things in precision agriculture, in *The 12th International Conference on Future Networks and Communications*, Leuven, Belgium, pp. 402–409 (2017). <https://doi.org/10.1016/j.procs.2017.06.083>
13. I. Mohanraj, A. Kirthika, J. Naren, Field monitoring and automation using IOT in agriculture domain, in *6th International Conference On Advances In Computing and Communications*, ICACC 2016, Cochin, India, pp. 931–939 (2016). <https://doi.org/10.1016/j.procs.2016.07.275>
14. M. Cantoni, E. Weyer, Y. Li, S.K. Ooi, I. Mareels, M. Ryan, Control of large-scale irrigation networks. *Proc. IEEE* **95**(1), 75–91 (2007). <https://doi.org/10.1109/JPROC.2006.887289>
15. Z. Yang, Q. Zhou, L. Lei, K. Zheng, W. Xiang, An IoT-cloud based wearable ECG monitoring system for smart healthcare. *J. Med. Syst.* **40**(12), 286, 1–11 (2016). <https://doi.org/10.1007/s10916-016-0644-9>

# Link Prediction Analysis on Directed Complex Network



Salam Jayachitra Devi and Buddha Singh

**Abstract** Link prediction helps in the analysis of complex network and predicts the future possible links. Researchers developed various link prediction methods using the network topological information. The topological information depends on different types of network such as undirected network, directed network, weighted network, etc. So, designing a link prediction method based on the types of complex network is a challenging task. Methods which are suitable for undirected network cannot be applied to a directed network. Hence, for every method associated with undirected network, corresponding methods of directed network can be developed by considering the topological information associated with the directed network. In this paper, we have designed a link prediction method known as modified resource allocation (MRA) for directed complex network. In the existing directed resource allocation (DRA) method, the immediate neighbors in the path length of two is considered. Here, this resource allocation index has been extended by considering neighbors in the path length of at most three. The proposed MRA method is primarily designed to predict the probability of formation of links between the disconnected nodes in a directed network by considering the longer path length. Area under the receiver operating characteristic curve (AUC) metric is used for evaluating the performance. Based on parameter  $\sigma$ , the AUC value is calculated and selected the most ideal solution. The comparative analysis of the proposed method with existing link prediction methods is performed to determine that the proposed MRA method provides better results than the existing link prediction models.

**Keywords** Directed complex network · Topological information · Similarity indices · Link prediction · AUC

---

S. J. Devi (✉) · B. Singh  
School of Computer & Systems Sciences, Jawaharlal Nehru University, New Delhi 110067, India  
e-mail: [Jayachitra.salam@gmail.com](mailto:Jayachitra.salam@gmail.com)

B. Singh  
e-mail: [b.singh.jnu@gmail.com](mailto:b.singh.jnu@gmail.com)

## 1 Introduction

Complex network includes a rising multidisciplinary research area that triggers much consideration from physicists, mathematicians, scholars, engineering, computer science, and numerous others [1]. Complex network structures describe a variety of complex systems of high technology and intellectual importance, such as communication networks, World Wide Web, Internet, mobile social network, and friends' network. [2, 3, 34–36]. It has also shown its noteworthy power in chemical and biological systems [4]. With the rapid development of complex network theory, several systems related to society and nature are portrayed as a complex network. Among the various fields of research on the complex network, link prediction has become an open issue in the field of data mining and knowledge discovery.

In recent years, it has been widely studied by researchers from different scientific communities [5]. The main aim of link prediction in complex network is to estimate the probability of existence of links among the nodes [6]. Link prediction has various applications in disparate fields. Due to this, it has become a research hotspot.

In case of biological networks, prediction of missing links helps in discovering unknown protein–protein interaction in the network and reduces the experimental cost [7, 8]. Not only this, it also helps in online social networks by recommending new friends to the users [23, 39, 40]. Link prediction also helps in identifying spurious connections and missing interaction in authors network [9, 10]. Apart from these, link prediction algorithm can also be applied in partially labeled networks for prediction of research areas or protein function [11].

Several link prediction algorithms are firmly related to the issue of network developing mechanisms. Recently, several authors proposed various link prediction methods based on different backgrounds. Among these previous methods, link prediction based on similarity indices which are also based on topology is the simplest and effective [12]. Due to its simpleness, it received close attention from the researchers. Similarity-based methods are classified as path-based and neighbor-based methods. The similarity of nodes can be determined from the links among them and the path in which resources are transferred [13, 14].

Some of the similarity indices for link prediction based on neighbor are common neighbor index, Jaccard's coefficient index, Adamic–Adar index, resource allocation index, etc. [15]. These methods are applicable in undirected network. Such methods have been extended for directed network structure considering the in-degree and out-degree of the nodes in the network [9]. In case of path based, some of the similarity indices include Katz index, significant path index, effective path index, ACT index, etc. [1]. These methods are proposed based on the global information of the nodes. These methods are also named as a global similarity index. They are mostly suitable in real undirected networks. Besides, it is not applicable to large network because of computational complexity and not suitable in directed networks. From previous studies, link prediction methods which are based on local paths gives lower complexity, and it also provides better performances in real networks. Hence,

we considered the local information of the nodes present in a directed network to develop a link prediction model.

In this paper, we focus on the resources being transferred between unconnected nodes in the network, which are likely to form links in the future. So, a link prediction model has been designed in this paper which is mainly applied to a directed network structure where the local topological information such as in-degree and out-degree is considered. This proposed model has been extended from the method developed by Shuxin Liu et.al. for directed network considering the topological information of the nodes [14]. We consider different directed network databases collected from various fields to perform the predictive analysis. We also consider a parameter which adjusts the amount of resources that are transferred to the longer paths of the directed networks. In this modified method, we also consider both the common neighbor as well as non-common neighbors of each pair of unconnected nodes in the network. From the simulation results, it has been shown that the modified method provides better AUC values. These results have been obtained from the comparative analysis of the proposed method with some existing methods on the basis of various real-world directed networks.

The organization of this work is noted as in Sect. 2; some of the existing link prediction methods based on directed network structure and its drawbacks are discussed. The problem description of our research work is discussed in Sect. 3. Evaluation metrics are described in Sect. 4. The detail description of the proposed method is given in Sect. 5. Later, the description of the database used in our experimental analysis is illustrated in Sect. 6. Simulation results and the comparative analysis of the proposed and existing methods are discussed in Sect. 7, and finally, Sect. 8 summarizes the conclusion along with future direction.

## 2 Related Work

In this section, some of the well-known link prediction methods of directed network are given. The neighbors in directed network can be represented in two ways as in-degree neighbor and out-degree neighbor. Some of the existing methods which have been described below for predicting the links are mainly based on common neighbors and the immediate neighbors of the node.

In common neighbors (CN) [15], the nodes having more common neighbors are most probable to form future link. The prediction of such kind of links is used in numerous fields such as social network, biological network, etc. As this method is the simplest, it is used widely in social network and it performs well too. The directed common neighbor index (DCN) is defined by considering out-degree of one node and the in-degree of the other node [9]. The common neighbor is computed by taking into consideration of the node that lies between the outgoing links of the initial node and the incoming links of the destination node as  $DCN(x \rightarrow y) = |\Gamma_{out}(x) \cap \Gamma_{in}(y)|$ .  $\Gamma_{out}$  denotes the out-degree neighbors of the corresponding node, and  $\Gamma_{in}$  denotes the in-degree neighbors of the corresponding node. Resource allocation (RA) [16]

has been inspired by the resource allocation dynamics in complex networks. This method defines the transmission of resources from node  $x$  to  $y$  passing through their common neighbor. The same method has an extended version of directed network [9]  $DRA(x \rightarrow y) = \sum_{z \in \Gamma_{out}(x) \cap \Gamma_{in}(y)} \frac{1}{k_{out}(z)}$ . Adamic–Adar (AA) [17] is mainly designed for social network analysis, and it defines the enhancement of common neighbor index. And this method was extended to make it applicable to directed network [9] as  $DAA(x \rightarrow y) = \sum_{z \in \Gamma_{out}(x) \cap \Gamma_{in}(y)} \frac{1}{\log k_{out}(z)}$ . Preferential attachment [18] is mainly used for generating evolving scale-free networks. For directed networks [9], this method is defined as  $DPA(x \rightarrow y) = |\Gamma_{out}(x)| * |\Gamma_{in}(y)|$ . Jaccard’s coefficient [19] considers the total number of neighbors as well as the common neighbors of the unconnected pairs of nodes. The Jaccard’s coefficient for directed network [9] can be computed as  $DJC(x \rightarrow y) = \frac{|\Gamma_{out}(x) \cap \Gamma_{in}(y)|}{|\Gamma_{out}(x) \cup \Gamma_{in}(y)|}$ . Salton [2] estimate common neighbors and the respective nodes degree to determine the similarity. In case of directed network structure, this can be defined as [9]  $DSA(x \rightarrow y) = \frac{|\Gamma_{out}(x) \cap \Gamma_{in}(y)|}{\sqrt{k_{out}(x) * k_{in}(y)}}$ . Sorensen [20] is also constructed based on the common neighbors and the degree of the nodes. The extension of this method for directed network [9] is  $SO(x \rightarrow y) = \frac{2|\Gamma_{out}(x) \cap \Gamma_{in}(y)|}{k_{out}(x) + k_{in}(y)}$ . Hub promoted [21] is mainly used for quantity determination of the overlapping pairs of substrates in the metabolic networks. This index can be extended for directed network [9] as  $DHP(x \rightarrow y) = \frac{|\Gamma_{out}(x) \cap \Gamma_{in}(y)|}{\min\{k_{out}(x), k_{in}(y)\}}$ . Hub depressed is exactly the opposite of the above index. The hub depressed for directed network [9] is  $DHD(x \rightarrow y) = \frac{|\Gamma_{out}(x) \cap \Gamma_{in}(y)|}{\max\{k_{out}(x), k_{in}(y)\}}$ . Leicht–Holme–Newman [22] considers both the common neighbor and their respective degrees. The extended version of this method based on directed network structure [9] is computed as  $DLH(x \rightarrow y) = \frac{|\Gamma_{out}(x) \cap \Gamma_{in}(y)|}{k_{out}(x) * k_{in}(y)}$ . Common neighbors plus preferential attachment (CN + PA) [37] is the combination of both common neighbor and preferential attachment index. This method has extended version of the directed network structure [38] and defined mathematically as  $D(CN + PA)(x \rightarrow y) = |\Gamma_{out}(x) \cap \Gamma_{in}(y)| + \in \frac{|\Gamma_{out}(x)| \times |\Gamma_{in}(y)|}{\sum_{z \in V} (|\Gamma_{out}(z)| + |\Gamma_{in}(z)|)}$ . ERA (extended resource allocation) [14] for undirected and unweighted network consist of all the resources transferred between the two nodes say  $x$  and  $y$ . The main drawback in this method is that it is applicable only in the undirected unweighted network. From the literature survey, we determine that some of the existing methods based on the undirected network structure have their extended version of the directed network structure too. As this paper deals with various directed complex network structure, we have considered only the existing methods which are suitable for the directed network model.

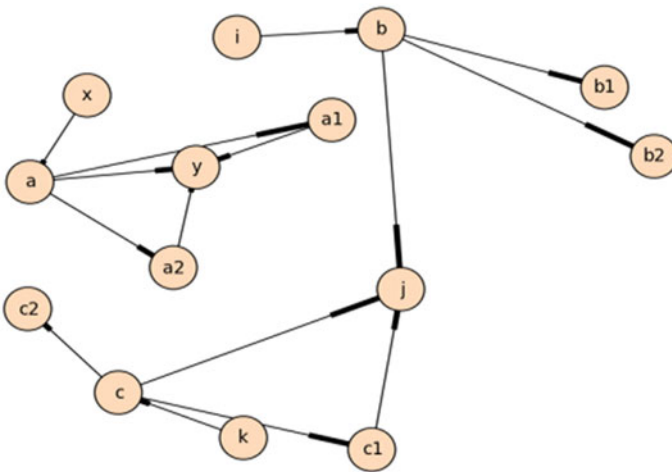
### 3 Problem Statement

This section will describe about the propagation of resources between the unconnected nodes through their common neighbors as well as non-common neighbors



in directed network structure. The more the resource transfer among the unconnected nodes, the higher the probability to form a future link between them. Let us consider the directed network as shown in Fig. 1. In this network, there are three pairs of nodes say  $(x, y)$ ,  $(i, j)$ , and  $(k, j)$  which are considered to be unconnected in the observed network. The common neighbor of  $(x, y)$  is determined by taking the common among the out-degree of  $x$  and in-degree of  $y$ . Here, node  $a$  is the common neighbor. Likewise, for the remaining pair of nodes such as  $(i, j)$  and  $(k, j)$ , the common neighbor of these nodes has also been determined and found that  $b$  and  $c$  are the common neighbor. Considering the resource allocation index for directed network, the similarity scores for the three pairs of nodes are same, but according to the concept of resource allocation index, node  $x$  sends resource to node  $y$  through their common neighbors as transmitters, these transmitters will acquire a unit of resource, and it will be distributed equally to all its corresponding neighbors. So, from Fig. 1, we can say that node  $(x, y)$  will have the high probability to form a link in the future because there are several other ways to transfer the resources from node  $a$  to node  $y$  via node  $a1$  and  $a2$  which results in receiving more resources. Here,  $a1$  and  $a2$  are the non-common neighbor node. Finally, we can conclude that resources transfer through longer paths also plays a vital role in determining the similarity of the unconnected nodes in the network.

This motivates us to modify the ERA index [14] to make it applicable to the directed network structure. Like ERA index, we also consider the parameter for adjusting the amount of resources that are transferred through longer paths in different directed networks.



**Fig. 1** Directed network model to show transfer of resources using directed common neighbor and non-common neighbor

## 4 Evaluation Metric

**Consider a directed network**  $G(V, E)$ , where  $V$  is the total number of vertices and  $E$  is the total number of edges. In directed network, the in-degree of a node is the number of edges heading toward the node, and the out-degree of a node is the number of edges driving away from the node. Mathematically, it is represented as  $\Gamma_{in}(x) = \{y | (y, x) \in E\}$  and  $\Gamma_{out}(x) = \{y | (x, y) \in E\}$ . Assume that multiple links or loop is not allowed. Let  $U$  be the universal set of all possible links in the directed network. It is determined as  $|v|(|v| - 1)$  where  $|v|$  denotes the set of nodes (vertices) in the network. Since  $E$  represent the set of edges in the directed network, the number of nonexistence links (edges) is determined as  $U - E$ . Our main aim is to find out future link by assigning scores for each node pairs that occur in nonexistence links. The score for each node pair will arrange in decreasing order, and the topmost pair of nodes will be most probable for the future link. Let  $x$  and  $y$  be the two nodes in the network. The similarity score for  $(x, y)$  will measure the probability of existence of links between them.

To evaluate the performance of the algorithm, we divide the set of observed links  $E$  into the ratio 90:10. This means 90% of the links will be treated as training sets and denoted as  $E^{train}$  and 10% of the links will be probe set, denoted as  $E^{probe}$ . The training set is known as information of links, and the probe set is mainly used for testing the algorithms. Hence,  $E^{train} \cup E^{probe} = E$  and  $E^{train} \cap E^{probe} = \emptyset$  [24].

In order to quantify the prediction accuracy of our proposed method from baseline methods, we consider area under the receiver operating characteristic curve (AUC) as evaluation metric. AUC evaluates the performance of the method by considering the probability of the randomly chosen missing links giving a higher score as comparable to the randomly chosen nonexistent link [2]. It is defined as  $AUC = \frac{n' + 0.5n''}{n}$ , where  $n'$  indicates many a times the missing links (testing set) giving higher score and  $n''$  indicates many a times training and testing set have the same score in  $n$  independent comparisons.  $AUC \approx 0.5$ . Initially, the similarity scores are randomly generated for both the set. Hence, if the AUC value exceeds 0.5, then it indicates the algorithm performing better.

## 5 Proposed Method

Directed resource allocation defines the amount of resources transferred through their common neighbor in a particular direction. Considering a directed network, the common neighbor of two nodes  $(x, y)$  is evaluated by considering the out-degree of  $x$  and in-degree of  $y$ . This indicates resources being transferred from node  $x$  and propagates through node  $y$  via their common neighbor. Here, we assumed that the resource transfer through their out-degree links is distributed equally. So, in case of real-time networks, if the path between two nodes is long enough, then they are likely to transfer more resources as the resources can be distributed equally to all the

connected nodes. Since all real-world directed networks have different properties, the resource transfer through longer paths will not be same. So, in order to solve this problem, we considered some set of adjusting parameters.

In our proposed method, we considered the resource transferred through the path length of at most three. We assumed that the resources transfer through their outgoing links is equally distributed. Further, the nodes which received the resources are again distributed equally to each out-going link. So, the amount of resources received by the target node through their incoming links from a particular node differed. Therefore, the neighbors of two unconnected nodes in different real-world directed network is classified into two groups as directed common neighbors and directed non-common neighbors. The proposed method is designed in the following steps.

**Step1:** In directed network  $G = (V, E)$ , let us assume that  $x$  and  $y$  are the two unconnected nodes where  $x, y \in V$ . And  $z \in \{\Gamma_{out}(x)\} - \{\Gamma_{out}(x) \cap \Gamma_{in}(y)\}$  denotes the set of nodes connected by the out-going links of node  $x$  but non-common neighbor of node  $x$  and  $y$ . Even the non-common neighbor nodes will also play a role in our proposed method because each out-going links from node  $x$  act as a resource carrier.

**Step2:** Let  $x, y \in V$  be the two unconnected pair of nodes. In this step, we defined the importance of directed common neighbor in building a part of our proposed method. Here  $z \in \{\Gamma_{out}(x) \cap \Gamma_{in}(y)\}$  denotes the common neighbor between node  $x$  and  $y$  for directed network. As resources are transferred through their common neighbor, we defined mathematically as follows

$$S_{di}(xy) = \sum_{z \in \{\Gamma_{out}(x) \cap \Gamma_{in}(y)\}} \frac{1 + \sigma \cdot N_{z \rightarrow y}}{k_{out}(z)} \quad (1)$$

where  $N_{z \rightarrow y}$  represent the number of common neighbors of node  $z$  and node  $y$ . It is expressed mathematically as  $N_{z \rightarrow y} = |\Gamma_{out}(z) \cap \Gamma_{in}(y)|$ .  $\sigma$  is the set of adjusting parameter, and  $k_{out}(z)$  indicates the number of out-degree of node  $z$ .

**Step3:** Let  $x, y \in V$  be the two unconnected pair of nodes. In this step, we defined the importance of directed non-common neighbor in building a part of our proposed method. Here,  $z \in \{\Gamma_{out}(x)\} - \{\Gamma_{out}(x) \cap \Gamma_{in}(y)\}$  denote the neighbor of node  $x$  but non-common neighbor of node  $x$  and  $y$ . The total amount of resources which are transferred through non-common neighbor in directed network is defined mathematically as follows

$$S'_{di}(xy) = \sum_{z \in \{\Gamma_{out}(x)\} - \{\Gamma_{out}(x) \cap \Gamma_{in}(y)\}} \frac{\sigma \cdot N_{z \rightarrow y}}{k_{out}(z)} \quad (2)$$

Here, our proposed methods consider the path length of exactly three.

**Step 4:** This step computes the formation of a link between the unconnected nodes present in the network. The proposed link prediction method performed prediction of the link on the basis of resource transferred, including common neighbors as well

non-common neighbors. The probability of predicting links between two nodes in the directed complex network is represented as

$$\begin{aligned} \text{MRA}_{\text{di}}(xy) = S_{\text{di}}(xy) + S'_{\text{di}}(xy) = & \sum_{z \in \{\Gamma_{\text{out}}(x) \cap \Gamma_{\text{in}}(y)\}} \frac{1 + \sigma \cdot N_{z \rightarrow y}}{k_{\text{out}}(z)} \\ & + \sum_{z \in \{\Gamma_{\text{out}}(x)\} - \{\Gamma_{\text{out}}(x) \cap \Gamma_{\text{in}}(y)\}} \frac{\sigma \cdot N_{z \rightarrow y}}{k_{\text{out}}(z)} \end{aligned} \quad (3)$$

where values of  $\sigma$  can be greater than or equal to zero. The proposed method will act same as directed resource allocation index if  $\sigma = 0$ . For analyzing the performance of the proposed method, 11 existing similarity methods for link prediction in directed complex network has been considered, and the performance of these methods has been compared with the proposed method based on the AUC values.

## 6 Databases

The analysis of the link prediction performance is done on the basis of 12 real complex network datasets which are collected from different fields. The details of the real-world database are as follows:

ModMath\_directed [25] network represents friendships among superintendents of school in Allegheny County (Pennsylvania, USA). World\_trade [26] network represents a trade of “trade miscellaneous manufactures of metal”, among 80 countries which show high technology along with machinery. Sampson’s monastery [27] network represents social interactions between organization of men (novices) getting ready to join a New England monastery, as surveyed by Samuel F. Small\_and\_Griffith\_Garfield [28] network represents citations among the scientific publications, determined from the Web of Science database. SanJuanSur\_Turrialba [25] denotes the visiting relationships of two networks among the families resided in the Attiro and San Juan Sur neighborhoods in the Turrialba region of Costa Rica. Centrality\_literature [29] is citations network form among the published papers based on the subject of centrality scores of the network. Flying\_teams [25] network shows preference for the co-pilots among the cadets at a US Army Air Force (USAAF). Mixed.species\_brain\_1 [30] network shows the interactions among the cortical regions of the cat brain, as measured by tract studies. C.elegans\_neurons [31] network shows the interactions of neural of the Caenorhabditiselegans nematode. Political\_blogs [32] represent a directed network of hyperlinks among weblogs based on US politics, which was documented by Glance and Adamic in 2005. Product\_co-purchasing (Amazon network) [32] e-commerce networks is mainly built on customers purchasing the items to purchase Amazon Web site’s feature. Suppose, if the item  $p$  is frequently co-purchased another item  $q$ , formerly the network has an

**Table 1** Properties for networks

Network	$ V $	$ E $	$k$	$d$	$k_{in}(max)$	$k_{out}(max)$	$C$
ModMath	43	179	4.1628	2.6436	16	11	0.1596
World_Trade	89	1012	11.3708	1.7472	21	77	0.6758
SampSon	32	381	11.9062	1.4400	30	24	0.5928
SmaGri	642	4926	7.6729	2.6063	172	983	0.4002
San	82	478	5.8293	2.4545	23	8	0.2288
Central	132	615	4.6591	2.3831	30	65	0.3345
Flying Team	51	355	6.9608	1.8869	16	16	0.3654
Mixed	65	1139	17.5231	1.6995	40	44	0.6613
Celegans	297	2359	7.9428	2.4553	139	39	0.2923
Polblogs	1018	7748	7.6110	2.8722	218	138	0.1958
Amazon	405	2998	7.4025	4.0020	54	10	0.5676
Email	960	7879	8.2073	3.2944	88	125	0.2026

edge directed to  $q$  from  $p$ . Communication network (email-EuAll) [32] was generated from a large European institution for research. For certain period of time, information regarding the outgoing and incoming email of the research institution was anonymized.

The properties of the different real-world networks are given in Table 1. The values given in the table are combined together to identify the complexity of the network. These values can be utilized to recognize the structural properties of the network [33]. This includes the properties of 12 real directed complex networks located from different fields.  $|V|$  indicates the total nodes present in the network.  $|E|$  denotes the total number of edges present in the network.  $k$  represent the average degree of the network and  $d$  represent the average shortest distance of a pair of nodes in the network.  $k_{in}(max)$  and  $k_{out}(max)$  are the maximum in-degree and out-degree of nodes present in the network.  $C$  denotes the clustering coefficient w.r.t. the network.

## 7 Simulation Analysis

### 7.1 Comparative Analysis of the Existing Methods and Proposed Method

The implementation of the proposed method and existing methods has been done on a Windows 10 operating system with 12 GB RAM using the Python 2.7 platform. The performance analysis of the existing methods as well as our proposed method is done on the basis of AUC. The experimental analysis based on AUC with different  $\sigma$  values is performed on 12 real networks. The value of  $\sigma$  is taken in the range

of 0 to 1 with a difference of 0.05. The value of AUC is obtained by taking the average of 20 iterations for each method based on independent division of  $E^{\text{train}}$  and  $E^{\text{probe}}$ . From Fig. 2, it is clear that in 12 real networks, there is a variation of AUC value corresponding to each  $\sigma$  value. The highest AUC value obtained within the range of 0 to 1 of  $\sigma$  values is considered to be the ideal solution for that network. When  $\sigma = 0$ , our proposed index behaves exactly like directed resource allocation (DRA) index. It is shown clearly in Fig. 2 that there is an increase in the AUC value when  $\sigma > 0$ . In case of mixed and Celegans network there is a decrease in the AUC value after reaching a certain point. But in almost all the network, there is not much variation between each AUC value for  $\sigma > 0$ . For the network San and Amazon, the AUC values are almost stable. These analyses proved that our proposed index is more effective as compared to the DRA index. Finally, we can say that not only immediate neighbor plays important role in link prediction, but also longer paths plays an important role, and it is much more effective than the immediate neighbors.

Table 2 presents the AUC values of 11 existing directed link prediction methods and the proposed method. For each network, the method which provides high AUC value is highlighted in bold. From the table, we can see that in all the 12 real networks, our proposed method (MRA) gives highest AUC values. Existing methods give low AUC value than our proposed method because the existing directed link prediction methods focus only on the immediate neighbors, whereas our proposed model considers immediate neighbors as well as neighbors which lie at longer path. As existing methods consider only the immediate neighbors, it does not show the importance of the nodes which lie at longer paths. But our proposed methods solved this problem by considering a longer path length of less than or equal to three (at most three). The difference in the path length makes the difference in the AUC value between existing methods and proposed method. Among the existing methods, directed resource allocation (DRA) and direct Adamic–Adar (DAA) are more competitive. Next, we can say that directed preferential attachment (DPA) and directed Jaccard’s coefficient (DJC), which is the simplest among these existing methods are also competitive. However, our proposed method (MRA) performed better than DRA in all the real network, and it is shown in Fig. 3. In Fig. 3, the difference in AUC results obtained by DRA and MRA based on 12 real-time networks is given as follows: ModMath network (+0.113), World Trade (+0.056), SampSon (+0.059), SmaGri (+0.057), San (+0.026), Central (+0.074), Flying Team (+0.086), Mixed (+0.035), Celegans (+0.016), Polblogs (+0.076), Amazon (+0.020), and Email (+0.024). The positive sign implies that the MRA AUC value is higher than the DRA AUC value. Comparison is done with a DRA index because our proposed index is based on this existing method, and Fig. 4 shows the comparative analysis of existing methods.

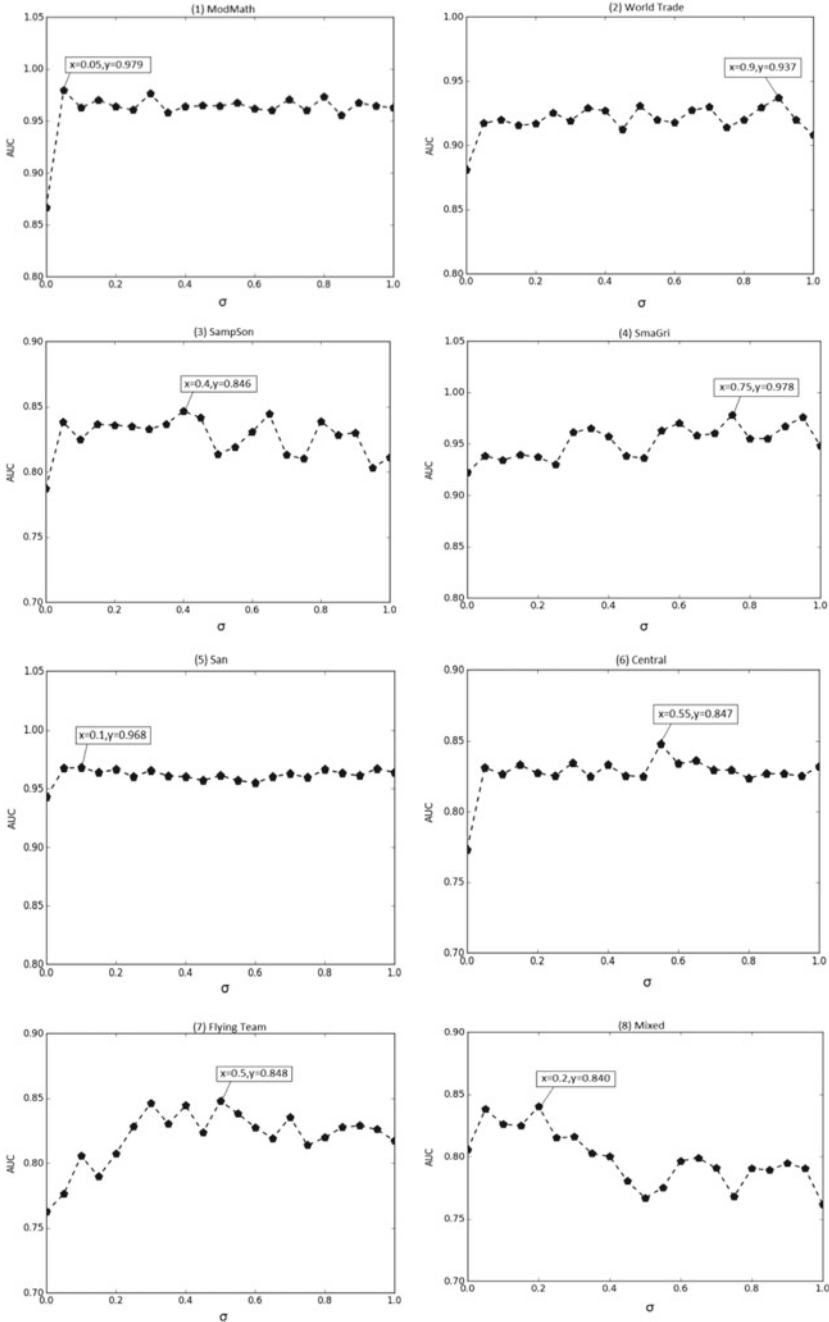


Fig. 2 Simulation results of MRA index based on AUC metric considering 12 real networks with 11 different values for  $\sigma$

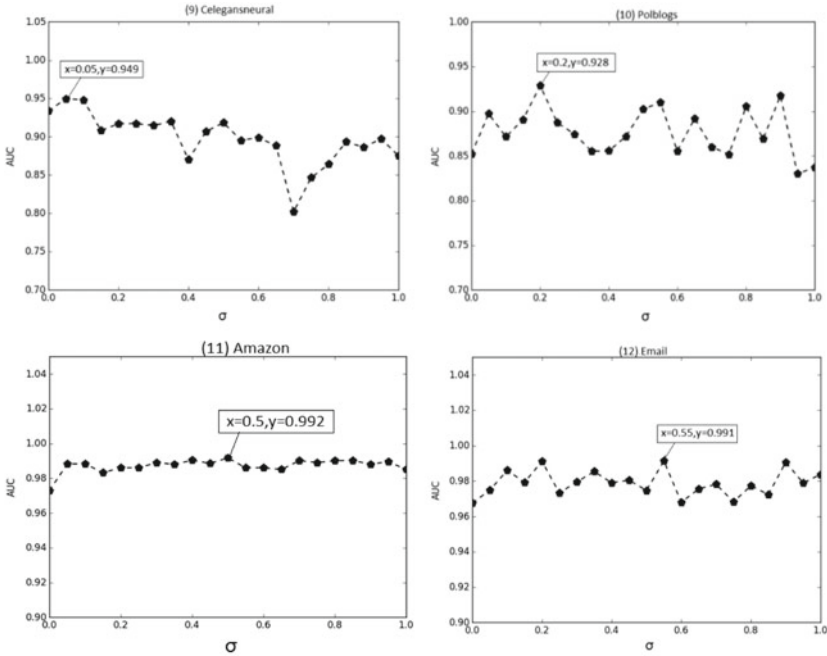


Fig. 2 (continued)

### 7.2 Comparative Analysis of Various Existing Directed Link Prediction Methods

This section mainly deals with the existing directed link prediction methods. We have considered 11 existing methods which are applicable only to directed network and determine their AUC values on the basis of various directed networks.

From Fig. 4, it has been clearly shown that none of the existing methods work well on all the networks. DJC works well on ModMath, and Mixed. D(CN + PA) works well in SampSon and SmaGri network. DAA provides acceptable performance in the network such as San, Celegans, and Amazon. For the network such as Flying Team and Email, DRA provides best performance. For World Trade network, DHP provides above 0.9 AUC value, whereas DHD provides best performance for Polblogs network, and lastly for Central network, DPA provides the best performance. The AUC value obtained by the methods which acquire the best performance is above 0.80 except for Flying Team which is above 0.75. In final comparison, D(CN + PA) provides better AUC values than DCN and DPA in almost all the network. From the analysis of the existing methods, we can conclude that different methods are suitable for different types of network.



**Table 2** Result of proposed and some existing methods based on AUC value

Networks	DCN	DPA	DJC	DSA	DSO	DLH	DHP	DHD	D(CN + PA)	DAA	DRA	MRA
ModMath	0.918	0.842	0.980	0.796	0.959	0.954	0.974	0.957	0.951	0.950	0.866	0.979
WorldTrade	0.876	0.889	0.845	0.518	0.871	0.747	0.912	0.805	0.896	0.884	0.881	0.937
SampSon	0.810	0.815	0.827	0.538	0.773	0.601	0.735	0.781	0.848	0.757	0.787	0.846
SmaGri	0.899	0.911	0.915	0.434	0.905	0.844	0.905	0.868	0.944	0.902	0.921	0.978
San	0.940	0.724	0.880	0.882	0.940	0.937	0.936	0.940	0.956	0.961	0.942	0.968
Central	0.574	0.820	0.581	0.394	0.595	0.688	0.625	0.544	0.787	0.610	0.773	0.847
Flying Team	0.615	0.736	0.667	0.591	0.625	0.631	0.662	0.661	0.752	0.672	0.762	0.848
Mixed	0.776	0.737	0.859	0.780	0.796	0.730	0.724	0.789	0.801	0.783	0.805	0.840
Celegans	0.731	0.843	0.721	0.542	0.718	0.715	0.658	0.836	0.776	0.942	0.933	0.949
Polblogs	0.913	0.874	0.900	0.538	0.890	0.869	0.782	0.927	0.924	0.831	0.852	0.928
Amazon	0.921	0.740	0.928	0.955	0.904	0.954	0.938	0.939	0.942	0.974	0.972	0.992
Email	0.948	0.938	0.965	0.629	0.940	0.896	0.919	0.923	0.958	0.966	0.967	0.991

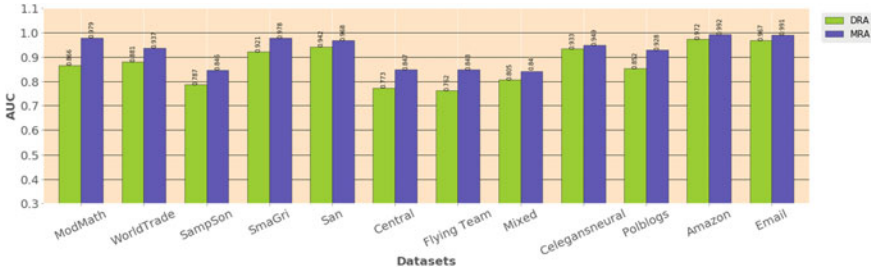


Fig. 3 Comparison of AUC values for DRA method and MRA method based on 12 real networks

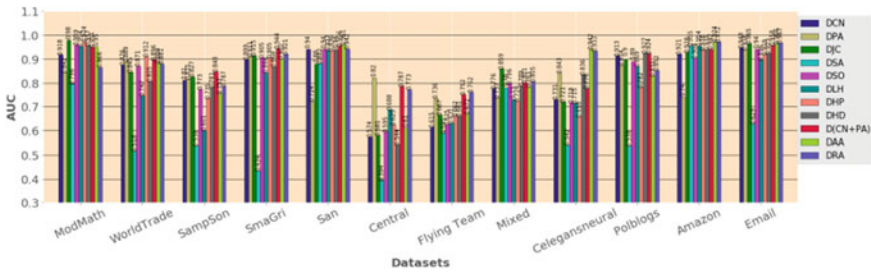


Fig. 4 Illustration of existing directed link prediction methods based on AUC values

## 8 Conclusion

We found that the topological information of a network plays a major role in executing a link prediction task. The proposed modified resource allocation (MRA) method for predicting links in a directed complex network considered the path length of at most three between two unconnected nodes in order to perform link prediction task. This method computes common neighbor as well as non-common neighbor for each pair of unconnected nodes, say  $(i, j)$ , in the network by considering outgoing neighbors and incoming neighbors. For each node  $z$  in the set of common neighbors as well as non-common neighbors, we further calculate the common neighbors of  $(z, j)$ . This is the main difference in topological information about our proposed modified resource allocation (MRA) method and existing link prediction methods. Each outgoing link carries an equal amount of resources from the initial node and transfer through the outgoing links of the intermediate nodes, and the resource has been distributed equally to all the adjacent nodes. Finally, the incoming links of the destination nodes will collect the resources coming from different incoming links. The experimental analysis of our proposed modified resource allocation (MRA) method is performed on 12 real networks based on area under the receiver operating characteristic curve (AUC) values considering a set of adjusting parameters  $\sigma$ . The adjusting parameter is taken in the range of 0 to 1 with a difference of 0.05, and these parameters help to find an ideal solution in various types of networks. When

the values of the adjusting parameters ( $\sigma$ ) increase, the AUC values also increase to a certain point. Once they reach at a certain point, it either becomes stable or there is a decrease in the AUC values. We choose the most desirable solution of our proposed model from the results given by the set of adjusting parameters. We have seen that our proposed modified resource allocation (MRA) method performs better and achieved good accuracy as compared to the existing link prediction methods. This is mainly due to the main differences between the topological information that is considered in our proposed MRA index and the existing methods. Further, we have also included a comparative analysis of only the existing methods to understand the difference in their performance according to types of network. Hence, from the comparative analysis of the proposed method and existing method, we can wind up that our proposed method is much more effective, and it is applicable to various other networks mainly in large real networks. In future direction, we can develop a link prediction model for weighted network structure considering the longer path length.

## References

1. Y. Liu, C. Zhao, X. Wang, Q. Huang, X. Zhang, D. Yi, The degree-related clustering coefficient and its application to link prediction. *Phys. A* **15**(454), 24–33 (2016)
2. L. Lü, T. Zhou, Link prediction in complex networks: A survey. *Phys. A* **390**(6), 1150–1170 (2011)
3. M.W. Ahn, W.S. Jung, Accuracy test for link prediction in terms of similarity index: the case of WS and BA models. *Phys. A* **1**(429), 177–183 (2015)
4. Z. Wu, Y. Lin, J. Wang, S. Gregory, Link prediction with node clustering coefficient. *Phys. A* **15**(452), 1–8 (2016)
5. Z. Liaghat, A.H. Rasekh, A. Mahdavi, Application of data mining methods for link prediction in social networks. *Social Network Anal. Min.* **3**(2), 143–150 (2013)
6. İ Güneş, Ş Gündüz-Öğüdücü, Z. Çataltepe, Link prediction using time series of neighborhood-based node similarity scores. *Data Min. Knowl. Disc.* **30**(1), 147–180 (2016)
7. K.K. Shang, M. Small, W.S. Yan, Link direction for link prediction. *Phys. A* **1**(469), 767–776 (2017)
8. P. Pei, B. Liu, L. Jiao, Link prediction in complex networks based on an information allocation index. *Phys. A* **15**(470), 1–1 (2017)
9. X. Zhang, C. Zhao, X. Wang, D. Yi, Identifying missing and spurious interactions in directed networks. *Int. J. Distrib. Sens. Netw.* **11**(9), 507386 (2015)
10. S. Scellato, A. Noulas, C. Mascolo, Exploiting place features in link prediction on location-based social networks, in *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining 2011 Aug 21* (ACM, ), pp. 1046–1054
11. P. Holme, M. Huss, Role-similarity based functional prediction in networked systems: application to the yeast proteome. *J. R. Soc. Interface* **2**(4), 327–333 (2005)
12. X. Feng, J.C. Zhao, K. Xu, Link prediction in complex networks: a clustering perspective. *Euro. Phys. J. B.* **85**(1), 3 (2012)
13. X. Zhu, H. Tian, S. Cai, Predicting missing links via effective paths. *Phys. A* **1**(413), 515–522 (2014)
14. S. Liu, X. Ji, C. Liu, Y. Bai, Extended resource allocation index for link prediction of complex network. *Phys. A* **1**(479), 174–183 (2017)
15. D. Liben-Nowell, J. Kleinberg, The link-prediction problem for social networks. *J. Assoc. Inform. Sci. Technol.* **58**(7), 1019–1031 (2007)

16. T. Zhou, L. Lü, Y.C. Zhang, Predicting missing links via local information. *Euro. Phys. J. B* **71**(4), 623–630 (2009)
17. L.A. Adamic, E. Adar, Friends and neighbors on the web. *Social Networks* **25**(3), 211–230 (2003)
18. A.L. Barabási, R. Albert, Emergence of scaling in random networks. *Science* **286**(5439), 509–512 (1999)
19. P. Jaccard, Étude comparative de la distribution florale dans une portion des Alpes et des Jura. *Bull Soc Vaudoise Sci Nat.* **37**, 547–579 (1901)
20. T. Sørensen, A method establishing groups of equal amplitude in plant sociology based on similarity of species content and its application to analyses of the vegetation on Danish commons. *K. Dansk. Vidensk. Selsk. Biol. Skr.* **5**, 34 (1948)
21. E. Ravasz, A.L. Somera, D.A. Mongru, Z.N. Oltvai, A.L. Barabási, Hierarchical organization of modularity in metabolic networks. *Science* **297**(5586), 1551–1555 (2002)
22. E.A. Leicht, P. Holme, M.E. Newman, Vertex similarity in networks. *Phys. Rev. E* **73**(2), 026120 (2006)
23. J.L. Herlocker, J.A. Konstan, L.G. Terveen, J.T. Riedl, Evaluating collaborative filtering recommender systems. *ACM Trans. Inform. Syst. (TOIS)*. **22**(1), 5–3 (2004)
24. S. Zeng, Link prediction based on local information considering preferential attachment. *Phys. A* **1**(443), 537–542 (2016)
25. W. De Nooy, A. Mrvar, V. Batagelj, *Exploratory Social Network Analysis with Pajek*, vol. 27 (Cambridge University Press, Cambridge, 2011)
26. W. de Nooy, A. Mrvar, V. Batagelj, *Exploratory Social Network Analysis with Pajek*, Chap 2 (Cambridge University Press, Cambridge, 2004)
27. S.F. Sampson, A Novitiate in a Period of Change. An Experimental and Case Study of Social Relationships. Ph.D. thesis Cornell University (1968)
28. E. Garfield, From Computational Linguistics to Algorithmic Historiography. Symp. Honor of Casimir Borkowski at U. Pittsburgh School of Information Sciences (2001)
29. H. Norman, P. Doreian, L. Freeman, Analyzing the structure of the centrality-productivity literature created between 1948 and 1979. *Sci. Commun.* **11**(4), 459–480 (1990)
30. M.A. de Reus, M.P. van den Heuvel, Rich Club Organization and intermodule communication in the Cat Connectome. *J. Neurosci.* **33**(32), 12929–12939 (2013)
31. D.J. Watts, S.H. Strogatz, Collective dynamics of ‘small-world’ networks. *Nature* **393**, 440–442 (1998)
32. Datasets available at <https://snap.stanford.edu/data>
33. S. Ferretti, On the complex network structure of musical pieces: analysis of some use cases from different music genres. *Multimedia Tools Appl.* **13**, 1–27 (2017)
34. X. Liu, J. Sun, W. Yang, M. Jiang, F. Yang, Ensuring efficient multimedia message sharing in mobile social network, *Multimedia Tools Appl.* 1–5 (2017)
35. E. Anshelevich, O. Bhardwaj, M. Usher, Friend of my friend: Network formation with two-hop benefit. *Theory Comput. Syst.* **57**(3), 711–752 (2015)
36. S. Haghani, M.R. Keyvanpour, A systemic analysis of link prediction in social network. *Artif. Intell. Rev.* 1–35 ((2017))
37. S. Zeng, Link prediction based on local information considering preferential attachment. *Phys. A* **443**, 537–542 (2016)
38. S.J. Devi, B. Singh, Link Prediction analysis for directed complex Network based on local information, in *INDIAcom Conference Proceeding* (2018)
39. V.S. Anoop, S. Asharaf, A topic modeling guided approach for semantic knowledge discovery in e-commerce. *Int. J. Interact. Multimedia Artif. Intell* **4**(6) (2017)
40. F.Z. Benkaddour et al., An adapted approach for user profiling in a recommendation system: application to industrial diagnosis. *IJIMAI* **5**(3), 118–113 (2018)

# Energy-Efficient VM Management in OpenStack-Based Private Cloud



P. K. Prameela, Priyanka Gadagi, Revathi Gudi, Somashekar Patil, and D. G. Narayan

**Abstract** The increase in usage of cloud data centers in recent years has led to high utilization of resources. Allocating of new virtual machines (VMs), disabling of existing hosts, and existing VM being removed are the reasons due to which the resource utilization in data centers vary over time. For efficient resource utilization, detection of the host's load using prediction techniques is an important issue. Furthermore, host load detection enhances the scheduling which results in higher utilization of the compute, network, and storage resources. Default scheduler in OpenStack uses a worst fit algorithm for VM allocation which leaves large fragments of RAM in compute nodes. In this work, VMs are scheduled based on user request using modified best fit algorithm depending on the prediction results that minimize the unnecessary large fragments of RAM in compute nodes. The prediction of host load is carried out using machine learning and statistical models. Furthermore, as a part of continuous resource monitoring and server consolidation, load balancing is performed based on the prediction result to balance the load on all servers. This helps in the energy-efficient consolidation to optimize energy consumption of hosts. We conduct the experimentation of proposed work using OpenStack based cloud testbed in a multi-node environment. Experimental results show that machine learning model

---

P. K. Prameela · P. Gadagi · R. Gudi (✉) · S. Patil · D. G. Narayan  
School of Computer Science & Engineering, KLE Technological University, Hubli, Karnataka,  
India

e-mail: [revathi.u.gudi@gmail.com](mailto:revathi.u.gudi@gmail.com)

P. K. Prameela  
e-mail: [prameelapk1998@gmail.com](mailto:prameelapk1998@gmail.com)

P. Gadagi  
e-mail: [priyankang29@gmail.com](mailto:priyankang29@gmail.com)

S. Patil  
e-mail: [skpatil@kletech.ac.in](mailto:skpatil@kletech.ac.in)

D. G. Narayan  
e-mail: [narayan\\_dg@kletech.ac.in](mailto:narayan_dg@kletech.ac.in)

LSTM and the statistical ARIMA model give comparatively good results for PlanetLab CPU trace data set. Also, the results reveal that the load among the servers is fairly distributed, and there is a significant improvement in energy saving.

**Keywords** OpenStack · LSTM · ARIMA · Resource scheduling · Energy efficiency

## 1 Introduction

Cloud computing is a distributed computing paradigm which delivers computing services on demand like processing power for computing, applications, and storage mainly over the Internet on the basis of pay-as-you-go method. Resources in the cloud can be provisioned rapidly and can be released with less management effort. Cloud computing depends on sharing of resources to achieve consistency. Cloud computing mainly comprises three types of service models: Software as a Service (SaaS), Infrastructure as a Service (IaaS) and Platform as a Service (PaaS). Cloud consumers can easily utilize the resources without the need to physically install them. We have several advantages by opting cloud resources as it reduces the capital expenses, enhances speed where vast amount of computing resources can be provisioned within minutes, improves performance as it reduces the latency, and reliable as it ensures disaster recovery and data backup; with the help of cloud computing it is possible to create cloud-native applications, test and build the applications, store huge amount of data, and deliver software on demand.

In recent years, resource utilization in the data center network has become an important research issue because it is very dynamic in nature. New VMs and/or hosts are created; existing hosts/VMs are removed due to failure. Various cloud resources are used to enhance the performance. The three major resources in cloud data centers are compute, storage, and network, and these resources need to be managed in the most efficient way. This requires VMs and hosts to reorganize to provide a better resource utilization and lower SLA violation, hence making dynamic resource provision a challenge. In this work, we propose a pre-step for dynamic resource provisioning by using prediction models. Machine learning techniques and statistical models are implemented to check their performance against PlanetLab datasets and real data. Reduction in the energy usage can be achieved by dynamic VM consolidation. The VMs residing in physical servers can be migrated to other servers, and this can be switched to sleep mode or can be turned off. This results in saving a huge amount of energy consumption by the servers in idle or standby state.

The contributions of this paper are:

1. We used prediction techniques to estimate the load on compute nodes.
2. We designed a resource management algorithm for energy-efficient VM management by performing load balancing and server consolidation.
3. We carried out the evaluation of the proposed technique using OpenStack-based testbed.

The remaining sections of this paper are organized in this manner: After the introduction, the next section discusses the literature survey done. The major elements of proposed work, operations of each along with flowchart, and algorithms are given in Sect. 3. Section 4 discusses and analyzes the results, taking different scenarios into account. Section 5 gives the conclusion of the paper along with the view for future work.

## 2 Background Study

### 2.1 Related Work

Over the decades, significant research in data centers regarding resource management and allocating VM during user requests has been noticed. To generate optimal computing resource utilization and energy consumption, several overloaded/underloaded host detection algorithms have been proposed. Authors in [1] have proposed energy-efficient consolidation which sets a benchmark. The framework is configured with the OpenStack multi-node configuration, and dynamic VM consolidation is performed which results in nearly 33% energy saving. Authors in [2] have proposed a host load detection algorithm, a new VM placement algorithm which is based on a proposed robust simple linear regression prediction model. The experiment results show that they have successfully reduced SLA violation by 99.16% and also energy consumption by 25.43% for the workload.

Authors in [3] have proposed work with a multi-objective optimization formulation for server-side energy saving and time to migrate virtual machines which is introduced. Further, their results show that two-stage greedy heuristic algorithms consume around 57.6 KJ while DRS and base consume 64.8 and 83 KJ, respectively. Authors in [4] inspired by host susceptibility and symbiotic coefficient among symbionts have designed two heuristics that are proposed for better resource utilization via VM consolidations. The experimental results show that average ESV by the proposed methodology is 14–92% lower than that of other benchmarking mechanisms over ten simulated days.

In [5], authors have proposed a classification model based on multiple parameters like current workload, memory, and CPU utilization of the hosts using machine learning. Further, the decision is used by the proposed scheduler to implement the best fit algorithm. The result of the experiment shows that underlying resources are being utilized efficiently by the proposed algorithm when compared to the default scheduler. Authors in [6] have proposed a SLA-aware resource-scheduling strategy to provide gain to cloud customers as well as service providers. The result of experiments using real-time multi-node setup shows the effective usage of resources. Authors in [7] have proposed dynamic live VM migration which determines the best way for migration of VMs from an overloaded host to an underloaded host. The experimental results

show that there is a fair distribution of load and the advantage of energy-efficient VM consolidation which results in up to 15% energy savings.

In [8], authors provide the critical analysis on the basis of research on energy-efficient dynamic allocation of virtual machines to hosts in a data center as per variable workload demands of different applications running on VMs. The goal of the paper is to enhance the overall utilization of computing resources. Authors in [9] have proposed energy-aware and maintaining QoS with efficient management of cloud computing environments. The results show that there is enhancement of energy efficiency under dynamic workload. Authors in [10] have made use of methods from the theory of robust optimization to measure the effects of uncertainty observed in modern data centers. The results reveal that by using the model, higher total energy consumption can be calculated by cloud operators while migrating the VMs.

In [11], authors have implemented an energy-efficient resource management system for virtualized cloud data centers which will reduce the operational costs and promise to provide required Quality of Service (QoS). Continuous consolidation helps in saving the energy. The results of the paper show that the techniques put forward bring considerable savings in the energy and ensure QoS. In [12], authors have proposed an approximate MDP-based dynamic VM management method which is called MadVM. They have proved that convergence in MadVM with maximum of two times the ideal migration cost. The results show that the proposed system archives remarkable performance reap over existing approaches in power consumption.

Authors in [13] have come up with an approach named dynamic switching probability (DSP). They have balanced the exploration of global search. The paper claims to outperform the first fit decreasing with 24.9% enhancing the environmental sustainability. Work of authors in [14] is mainly focused on the factors for consolidation like power, CPU, and networking resource sharing. They have proposed virtual machines' performance with different specifications. Authors in [15] have proposed different techniques in scheduling known as network-aware scheduling and mapping of different management objectives to the controller which handles resource allocation to achieve management objectives.

## 2.2 *OpenStack Architecture*

OpenStack is a cloud operating system which is used for cloud deployment. It is a set of software tools for building and managing cloud resources. Complete infrastructure can be built with the help of OpenStack. It is used to control a large pool of compute, storage, and networking resources. It also acts as a development environment and testing environment where we can run instances on these environments. It basically provides a platform on which we can build our own application and infrastructure.

Figure 1 shows the three independent parts of the OpenStack architecture called the OpenStack services: compute service, the networking service, and the storage service.



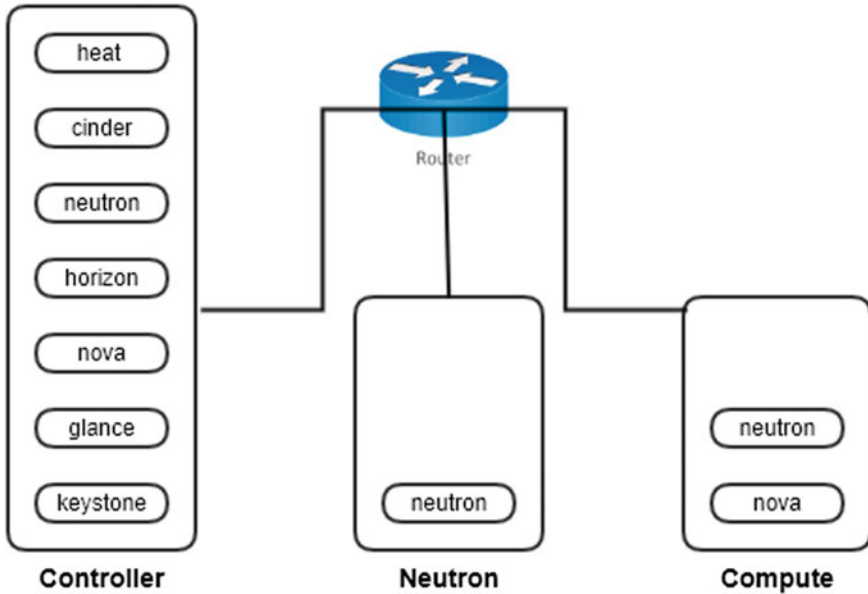


Fig. 1 OpenStack architecture

**Compute Service:** The compute service manages pools of computer resources and works with virtualization technologies. It is also called NOVA which acts like the brain of OpenStack. It is used to manage numerous virtual machines and other instances that handle various computing tasks.

**Networking Service:** The networking service is also known as the Neutron and has networking capability like managing networks and IP addresses for OpenStack. It allows users to create their own networks and connect devices and servers to one or more networks. Communication on neutron is through ports, different ports are defined, and the VM requests are listened to on each port.

**Storage Service:** Storage is used to manage a set of well-defined remotely accessed APIs. The main services provided are storage, backup, and document sharing. Three types of storage services are object storage also known as swift, block storage also known as cinder, and shared file system.

**Scheduling in OpenStack:** The NOVA component is responsible for storing, retrieving images from glance, and running the instances. NOVA-scheduler is one of the components of NOVA that determines on which host/compute the particular instance should be made to run on. The requests for computing instances are dispatched using the NOVA-scheduler by the compute service in OpenStack. The scheduler driver is by default configured as a filtered scheduler.

There are basically three types of schedulers: simple, chance, and filter.

**Simple scheduler:** The host with least available load is considered.

Chance scheduler: The characteristics are not considered, and it chooses randomly among all the compute nodes.

Filter scheduler: There are nearly 14 types of filters available that can be applied to filter the hosts to get eligible hosts. The default scheduler applies some filters to get the eligible hosts, and these hosts should meet the conditions as imposed by the filters such as

Availability zone: A random host within the availability zone is selected.

Compute filter: It checks whether the compute can service the request.

Compute capability filter: It checks whether the instance specifications can be satisfied by the compute.

There are other filters applied on hosts before scheduling. Administrators can configure or discard any of the filters, and this option is made available by OpenStack. Among the eligible hosts, the most suitable host is selected to launch the new VM requested. This procedure is done mainly in two steps filtering and weighing.

Filtering—Internally, the `get_filtered_hosts()` function returns hosts after filtering, and the list contains the ones passing the filters and by eliminating the ones which cannot accommodate this instance.

Weighing—The function `get_weighed_hosts()` performs weighing of the filtered hosts with regard to the weights which are set by the weigh handler. The host with the highest available memory is chosen.

The existing scheduler depends on a weighing strategy based on only RAM, and it does not consider the dynamic workload characteristics of the host. Hence, there is a need for a new strategy which takes into consideration the workload on the hosts.

### 3 Proposed System

The compelling factors in providing cloud resources are the cost of resource allocation and optimal energy consumption. Energy consumption depends on multiple factors like service-level agreement, workload, etc. The two main objectives of system are VM scheduling and server consolidation of servers. System ensures the schedule of the VM requested by the user and also consolidates all the servers as a way to balance the load on these servers. The following section describes the proposed system model, implementation methodology, and the different modules of the systems along with supporting algorithms used to implement these modules.

Figure 2 shows the proposed system model. Whenever the request for a new virtual machine is received, the data is fetched and fed to the prediction model which predicts the state of eligible hosts. Based on the prediction, VM is scheduled using the best fit strategy. Continuous resource management is done by capturing resource utilization of all servers followed by server consolidation, and load balancing is done if the state of host is overloaded and underloaded, respectively.

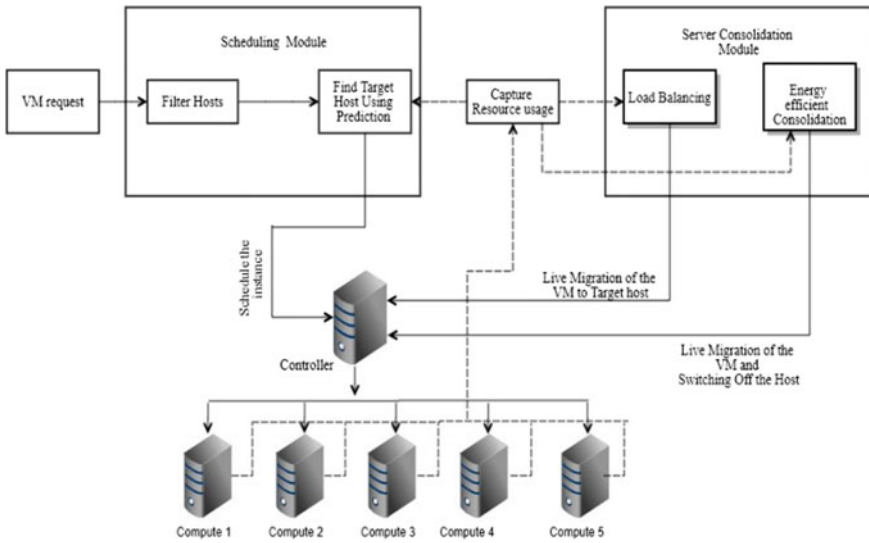


Fig. 2 Proposed system model

### 3.1 Modules

In this subsection, we discuss the major modules. There are three modules in our proposed system namely load prediction, VM scheduling, and server consolidation. These are described as follows.

#### Load Prediction

Analyzing resource utilization primarily deals with deploying multi-node setup in the system. In our multi-node setup, there are five compute nodes where VMs can be launched based on user specifications. We have considered CPU utilization as a parameter for describing load on the server. We have taken history data for one day that consists of 144 values fetched for every 10 min. A model is chosen based on its performance analysis and is trained based on history data/training data. While scheduling the requested VM, the present data of CPU utilization is fetched and is appended to the training data, and the future load on each server is predicted using the built model. LSTM and ARIMA are tested for their performance on PlanetLab datasets and real data. ARIMA gave better results in terms of efficiency and RMSE compared to LSTM. Hence, ARIMA is chosen for load prediction of servers.

#### VM Scheduling

After predicting the load of filtered hosts, we get the states of hosts in terms of overloaded, normal loaded, or underloaded. Further the requested VM is scheduled on the host which is normal loaded. Suppose at a particular instance of time, the load on all the servers is zero, then the scheduling is performed considering RAM as a

parameter to select the target host. Modified best fit technique is implemented for VM scheduling in the latter case if all the hosts underloaded.

### **Server Consolidation**

Server consolidation is the process where the VMs from the underloaded hosts and normal loaded hosts are migrated to normal loaded hosts. This is an approach to ensure efficient usage of servers by reducing the energy being consumed by the servers. For every 10 min, load on the host machine is predicted, and the VMs from overloaded host machines are migrated to the target host in order to achieve load balancing and the servers which are underloaded are powered off by migrating all the instances from this host to other hosts.

## **3.2 Algorithms**

Algorithms for modules discussed above are given below.

Module 1: The first module explains about the classification of servers based on their CPU utilization. Two thresholds are set to classify servers as underloaded, normal loaded, and overloaded hosts. It also explains about the filtering of hosts. Whenever there arrives a VM request, the hosts are filtered based on RAM specification. The algorithms are given below.

**Module 1 Algorithm 1****Algorithm 1: Host State Detection****Input:** Host's CPU utilization, upper threshold=80 and lower threshold =20**Output:** X<- State of Host

```

Begin
  if CPU_Util<lower_threshold then
    X<-Underloaded
  else if CPU_Util>Upper_threshold then
    X<-Overloaded
  else
    X<-Normal Loaded
  end if
  return X
end Begin

```

**Module 1 Algorithm 2****Algorithm 2: Host Filtering Module****Input:** User Requirements, host\_List [ ],input\_ram**Output:** Filtered host\_List[ ]

```

Begin
for each host in host_List do:
  if host satisfy User Requirements then do:
    if vcpu>0 then do:
      if free_ram_available>input_ram then do:
        Filtered hostlist = host
      else:
        Filtered host
      end if
    end if
  end if
end for
end begin

```

Module 2 explains about the prediction of future load of each filtered host and VM scheduling. Based on the predicted load, the host is decided as either underloaded, normal loaded, or overloaded. After prediction, the specified VM will be scheduled on a normal loaded host. Suppose if no normal loaded host is formed, then it will be scheduled on an underloaded host. And if all hosts are underloaded, then the host with less available RAM will be allocated with the VM.

## Module 2 Algorithm

---

### Algorithm 3: Load prediction and Scheduling

---

**Input:** Filtered host\_List, CPU utilization and number of VMs present in each host

**Output:** State of host

```

Begin
  for each host in Filtered host_List do
    Fetch the current parameters of Host
    Predict the Host state with model
  if (state ==normal)
desiredHost<-host
else if(state==underloaded and state==empty)
desiredHost<-host
  end for
  if desiredHost not found then
select host with less available RAM
desiredHost<-host
  Schedule VM on desired host
End Begin

```

---

Module 3 is load balancing and VM consolidation. This module consists of two main conditions namely energy-efficient migration and load balancing.

## Module 3 Algorithm

---

### Algorithm4: Energy Efficient Migration and Load Balancing

---

**Input:** Resource utilization of Host ,Host\_List[]

**Output:** Consolidation and Load Balancing

```

Begin
If (Host_util<Lower Threshold)
  Live Migrate the VMs to available hosts if any
  Turnoff the host
  XUnderloaded
else if (Host_util>Upper Threshold)
  For each item in VMList
    Cpu_util.append(cpu utilization of item)
  end for
  maxvm=VM with maximum cpu utilization
  for each host in Host_List
    if(available_RAM>= maxvm_RAM and VCPU>0)
      Sum_CPU<-host_CPU +maxvm_CPU
      Predict the state of host with sum_CPU
      if (state!=overloaded)
        Migrate the maxvm to this host
      else
        Continue
    endif
  endif
endfor
endif
End Begin

```

---

Algorithm 4 helps in reduced energy consumption. This is achieved by migrating all the VMs which belong to underloaded hosts to other suitable hosts, and the underloaded host is powered off. This also ensures that the hosts to which these instances/VMs are migrated will not be overloaded. And the load on each host machine is balanced where we have considered the number of vCPUs and RAM as our parameters. VMs with maximum CPU utilization of overloaded hosts are considered as eligible for migrating on other host machine which has equal or more number of vCPUs and RAM availability.

The power consumption of a single physical machine is expressed as

$$p_{server} = p_{active} + p_{dynamic}(Util_{cpu}).$$

where  $p_{server}$  is the total power consumed by the server,  $p_{dynamic}$  is the dynamic power consumption of the CPU,  $(Util_{cpu})$  is the average CPU utilization, and  $p_{active}$  is the power consumption when the CPU is idle. A physical machine in standby mode consumes  $p_{standby}$ , meanwhile, an active physical machine consumes  $p_{active}$  in addition to the power consumed by each virtual machine hosted by that physical machine  $p_{vm}$ .

## 4 Results and Discussions

### 4.1 Experimental Setup

The multi-node testbed setup has the following components: one controller, one neutron, and five compute nodes. Controller node supplies and manages services like API, scheduling, and load balancing of the cloud. And it controls workflows like scheduling and load balancing. The virtual networking and networking services to NOVA instances are provided by neutron node using the neutron layer 3 and DHCP network services. It also assigns IP addresses and proxy addresses. Table 1 shows

**Table 1** Specifications of the five compute nodes

Nodes	IP address	RAM (GB)	Disk (GB)	VCPUs
Controller	192.168.31.2	4	50	2
Neutron	192.168.31.3	2	50	2
Compute 01	192.168.31.4	4	50	2
Compute 02	192.168.31.5	8	50	2
Compute 03	192.168.31.6	8	50	4
Compute 04	192.168.31.7	8	50	8
Compute 05	192.168.31.8	8	50	8

the specifications of the five compute nodes in terms of RAM, disk, and VCPUs. In the following testbed, compute nodes 1, 2, 3, 4, and 5 have the same RAM and disk size. This setup is being built on OpenStack.

### 4.2 Prediction Result Analysis

We collected the data from our test node. CPU traces are collected from our test node for every 10 min. The data collected is stored in a history file. Prediction of future load on the host with the given present load is the first aim, and the present load is appended to history values. After predicting the load on the host machine, the file is saved by appending the present data and by popping the first CPU utilization.

As mentioned earlier, ARIMA and LSTM techniques are considered as predictive techniques to predict the future load on the servers. The performance of both techniques is analyzed by implementing these techniques on two datasets of PlanetLab and real dataset. The values for real dataset is fetched from the “ceilometer” component that is present in our OpenStack setup. Ceilometer keeps track of the VMs that are up and the amount of time each service on the VM is being used. It keeps a meter running which runs every time when you avail a service or every time a VM is running. The one with highest accuracy and which takes less execution time is chosen for predicting load on hosts.

Figure 3 shows the performance of ARIMA and LSTM algorithms on three datasets. The three datasets are PlanetLab01, PlanetLab02, and realdata. For PlanetLab01 dataset, accuracy obtained by implementing ARIMA is 91.64%, whereas

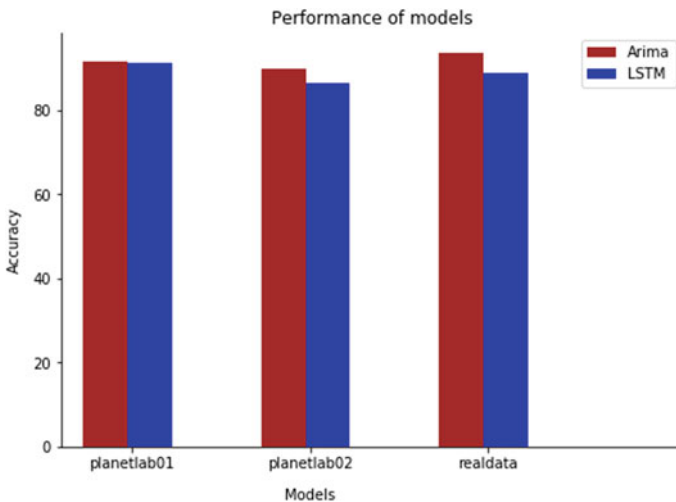


Fig. 3 Prediction result for PlanetLab01, PlanetLab02, and real datasets



for LSTM, it is 91.33% which are almost equal to each other. And the accuracies obtained for PlanetLab02 are 89.8% and 86.35%, respectively. Real dataset achieved an accuracy of 93.877% for ARIMA and 88.88% for LSTM. From the above results, it is clear that ARIMA achieved the highest accuracy when compared to LSTM. Apart from comparing the results on the basis of accuracy, we also measured the execution time for each algorithm. Execution time for both ARIMA and LSTM was calculated on a real dataset. ARIMA took 2.56 s to build the model for prediction, whereas LSTM took 10.37 s. Execution time had a noticeable difference in their output. Hence, ARIMA is befitting for predicting the load on servers in terms of execution time and accuracy.

### 4.3 Scheduling Result Analysis

Scheduling is carried out in two ways.

1. According to the load present on each server.
2. Based on availability of RAM.

The first one is performed when there is sufficient load on servers. The second one is performed when at a particular instance of time the load present on all servers is zero.

Figure 4 shows the load on each compute node. Here, the compute 1 and 2 are in standby mode, i.e., they are not running. Based on the thresholds considered according to the algorithms, compute 3 is in underloaded state, whereas compute 4 and compute 5 are in normal loaded state. We now consider an example of getting a VM request for 2 GB RAM. After filtering the hosts based on RAM according to the proposed algorithm 2, compute 3, 4, and 5 are eligible for VM allocation.

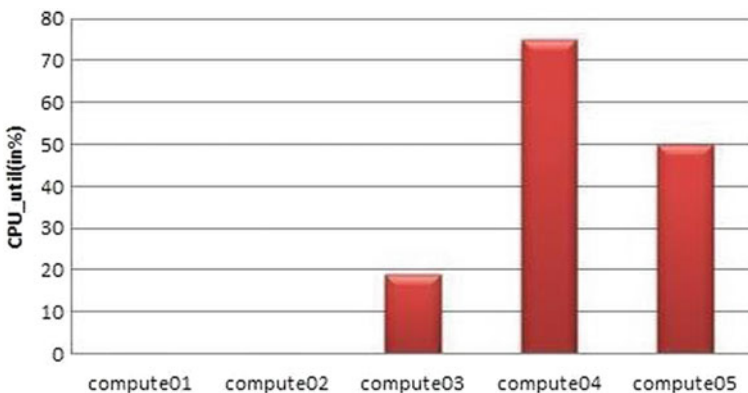
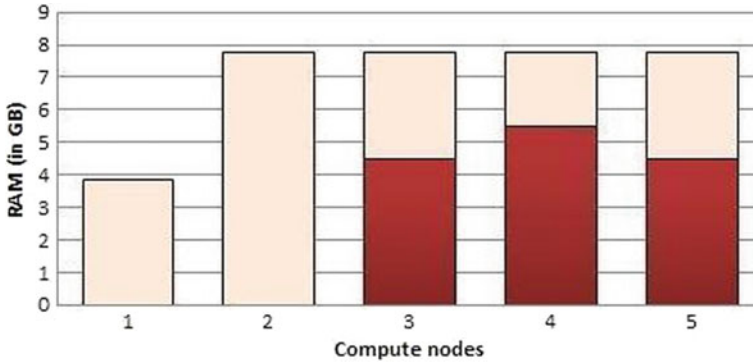


Fig. 4 Scheduling new request based on load



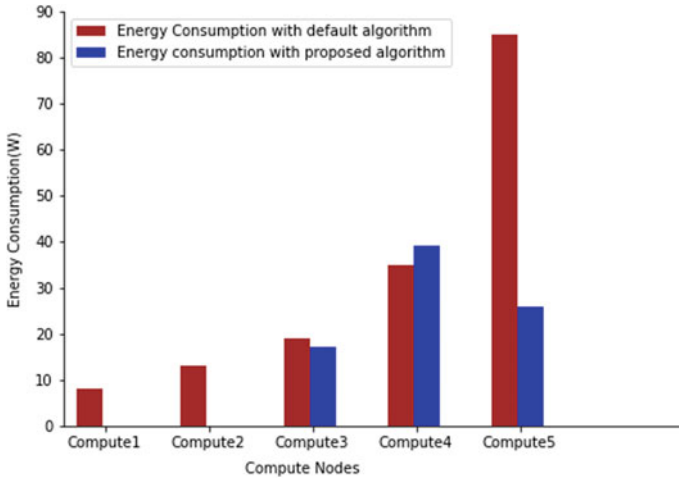
**Fig. 5** Scheduling new request based on RAM

In this case, since compute 4 and 5 are normal loaded and the first available compute/host is compute 4, VM will be launched on compute 4.

Figure 5 shows the RAM occupied by each compute node. Here, again the compute 1 and compute 2 are in standby mode. Total RAM of compute 3, 4, and 5 are 7.8 GB each. RAM occupied by compute 3, compute 4, and compute 5 are 4.5 GB, 5.5 GB, and 4.5 GB, respectively. Available RAM is 3.3 GB, 2.3 GB, and 3.3 GB, respectively. We now consider an example of getting a VM request of 2 GB RAM. After filtering the hosts based on RAM, compute 3, 4, and 5 are eligible for VM allocation. In this case, VM allocation is implemented using modified best fit scheduling as proposed in Algorithm 3. Compute 3 will leave a fragment of 1.3 GB, compute 4 will leave a fragment of 0.3 GB, and compute 5 will leave a fragment of 1.3 GB. As the algorithm considers the host that has less available RAM to avoid the generation of large fragments, the requested VM will be launched in compute 4.

#### 4.4 Load Balancing Result Analysis

Figure 6 shows the comparison of energy consumption (W) by VMs of each host with respect to the default algorithm which uses the worst fit algorithm versus proposed algorithm which uses modified best fit algorithm. Worst fit algorithm has a major drawback of having large fragments after allocating the resources demanded by the user, whereas the modified best fit algorithm ensures to have smaller fragments so that the storage is used efficiently. It was predicted that compute 2 was an underloaded host. Hence, after performing the proposed algorithm, the instances from compute 2 were migrated to compute 4 and compute 2 which were powered off. And compute 5 was predicted as an overloaded host, and therefore based on the proposed algorithm, some of the instances were chosen to migrate to target hosts. High energy consumption of compute nodes is observed by the usage of the default algorithm of OpenStack and the inclusion of proposed algorithm resulted in less and optimal



**Fig. 6** Energy consumption using default algorithm and proposed algorithm

energy consumption. Hence, the energy consumption of the ideal server is reduced by switching it off.

## 5 Conclusion and Future Scope

Energy-saving problems are considered to be important for cloud service providers due to the current growth in size of data centers. Opportunities for energy saving are provided by the development of virtualization technologies. In this paper, we present a framework for managing and controlling virtual machines placement on physical servers to reduce the energy consumed by data centers. A vital role in utilization of resources in the data centers is played by VM consolidation by applying dynamic live VM migration. Worst fit algorithm is used by OpenStack as the default scheduling strategy, which does not indulge in efficient utilization of underlying resources. When examined, NOVA-schedulers of NOVA components use the first fit strategy for live VM migration, and there is no consideration of different dynamic characteristics of the VM to be migrated and the host. From the obtained results, it can be concluded that our proposed system provides an efficient usage of the underlying resources when compared to the existing one. Fair distribution of the resources of CPU and RAM is observed. 10–15% of the overall consumption is reduced by performing energy-efficient consolidation.

As future work, we play to appraise the load of compute servers over a period of time using the transfer learning technique and propose an energy-efficient VM and cluster migration mechanism with a minimized load within the server.

## References

1. A. Beloglazov, R. Buyya, OpenStack neat: a framework for dynamic and energy-efficient consolidation of virtual machines in OpenStack clouds. *Concurr. Comput. Pract. Exp.* **27**(5), 1310–1333 (2015)
2. F. Farahnakian, et al., Energy-aware VM consolidation in cloud data centers using utilization prediction model. *IEEE Trans. Cloud Comput.* (2016)
3. M. Al-Tarazi, J. Morris Chang, Network-aware energy saving multi-objective optimization in virtualized data centers. *Clust. Comput.* **22**(2), 635–647 (2019)
4. J.V. Wang, et al., Bio-inspired heuristics for vm consolidation in cloud data centers. *IEEE Syst. J.* (2019)
5. N.V. Janagoudar, D.G. Narayan, M.M. Mulla, Multi-objective scheduling using logistic regression for openstack-based cloud, in *Third International Conference on Computing and Network Communications (CoCoNet' 2019)*
6. D.G. Preeti Parakh, D.G. Narayan, M.M. Mulla, V.P. Baligar, SLA-aware virtual machine scheduling in openstack-based private cloud, in *3rd IEEE International Conference on Computational Systems and Information Technology for Sustainable Solutions* (2018)
7. M. Pyati, D.G. Narayan, S. Kengond, Energy-efficient and dynamic consolidation of virtual machines in openstack-based private cloud, in *Third International Conference on Computing and Network Communications (CoCoNet' 2019)*
8. A. Choudhary, S. Rana, K.J. Matahai, A critical analysis of energy efficient virtual machine placement techniques and its optimization in a cloud computing environment, in *International Conference on Information Security & Privacy* (2015)
9. A. Beloglazov, J. Abawajy, R. Buyya, Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing. *Future Gener. Comput. Syst.* (2012)
10. E.Z. RobayetNasim, A.J. Kassler, Robust optimisation for energy-efficient virtual machine consolidation in modern data centers. *Clust. Comput.* **21**, 1681–1709 (2018)
11. A. Beloglazov, R. Buyya, Energy efficient resource management in virtualized cloud data centers, in *IEEE/ACM International Conference on Cluster, Cloud and Grid Computing* (2016)
12. Z. Han, H. Tan, R. Guihai Chen, Y. Li, F. Chi Moon Lau, Energy-efficient dynamic virtual machine management in data centers. *IEEE/ACM Trans. Netw.* (2019)
13. M.J. Usman, A.S. Ismail, H. Chizari, G. Abdul-Salaam, A.M. Usman, A.Y. Gital, O. Kaiwartya, A. Aliyu, Energy-efficient virtual machine allocation technique using flower pollination algorithm in cloud datacenter: a panacea to green computing. *J. Bionic Eng.* (2019)
14. M.F. Corradi, Antonio, L. Foschini, VM consolidation: a real case based on OpenStack cloud, in *Future Generation Computer Systems*, vol. 32 (IEEE, 2014), pp. 118–127
15. F. Wuhib, R. Stadler, H. Lindgren, Dynamic resource allocation with management objectives—implementation for an OpenStack cloud, in *International Conference on Network and Service Management* (2012)

# Intelligent Transportation System: The Applicability of Reinforcement Learning Algorithms and Models



S. P. Krishnendhu and Prabu Mohandas

**Abstract** Nowadays, many research works that associate real-time data widely use an unsupervised artificial intelligence (AI) technique, namely reinforcement learning (RL). Its fast adaptiveness to the dynamicity draws the attention of researchers who works in real-time traffic signal control systems. The scope of RL in most of the research problems remains remarkable with its peculiar characteristics. This paper reviews the basic concepts of RL, along with RL algorithms and models with an emphasis on traffic signal control (TSC). TSC is one among the trending applications of RL. Traffic congestion control with less human intervention is a challenging task of the intelligent transportation system (ITS). It not only helps traffic managers to get a grip over the traffic operation situation and analyze congestion, but also assists travelers to avoid congestion. Considering its significance, we have chosen TSC as the basis to explain the RL algorithms and models presented in this paper. In addition to such a comprehensive review, we have also provided a list of open challenges which when addressed can take the research in this area to considerable heights.

**Keywords** Traffic signal control · Reinforcement learning · Intelligent transportation system · Artificial intelligence · Supervised learning

## 1 Introduction

Traffic congestion has become an annoying and a complicated issue in most of the urban areas. A smart and efficient traffic controlling mechanism is the solution to this problem. Moreover, such a system can provide abundant advantages such as smooth traffic flow, and reducing unwanted waiting time in traffic junctions. Better managing of traffic at bottleneck junctions is essential as the traffic demands rise, failure in which is sure to cause congestions. Congestion can mostly occur in a junction if most

---

S. P. Krishnendhu (✉) · P. Mohandas  
National Institute of Technology, Calicut, Kerala, India  
e-mail: [krishnendhusp@gmail.com](mailto:krishnendhusp@gmail.com)

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2021  
S. M. Thampi et al. (eds.), *Advances in Computing and Network Communications*,  
Lecture Notes in Electrical Engineering 735,  
[https://doi.org/10.1007/978-981-33-6977-1\\_41](https://doi.org/10.1007/978-981-33-6977-1_41)

557

of the vehicles are waiting for the signal to turn green. Unfortunately, the current traffic systems fail to consider real-time parameters that affect traffic congestion.

Thus, many research works are ongoing in traffic regulatory systems to avoid the challenge of traffic congestion. The automation of traffic regulatory systems is related to many fields such as *image processing (IP)*, *machine learning (ML)*, and *Internet of things (IoT)*. Previously, *traffic signal control (TSC)* models did not significantly address the inconveniences caused by over-saturation, delays due to unexpected events, and climate change. Data collected from traffic networks at different times were used to control green signals based on the Webster formula [1]. However, they were not adequate to control the fast-moving traffic. Scientific and technical studies that are consistent with the fact that queue size plays a vital role in traffic control [2–4] have also failed to address green-signal vegetate, cross-blocking, and occlusion.

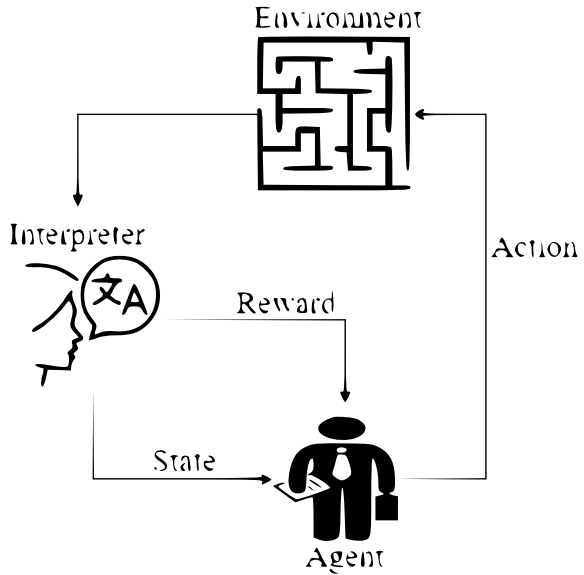
Nowadays, the world is witnessing a few exciting research pieces that strive to automate an optimized traffic signal that overcomes the shortcoming of existing ones by considering all the real-time facts learned from the system's surroundings, including the driver behavior [5]. The research in traffic automation is fastened right from the introduction of *Reinforcement Learning (RL)*. However, utilizing RL, the traffic regulatory system can be modified in an effective way such as the green light duration is shortened or lengthened, or even skipped according to the dynamic traffic conditions [6]. RL is highly adaptable to the dynamicity of traffic conditions irrespective of the time. This peculiar property increases the possibility of producing such a system in real.

Figure 1 shows the typical framework of an RL scenario. Here, an agent takes action (say A) in an environment. This action is interpreted into a reward (say R) and a representation of the state (say S). The result is fed back into the agent. The main problem arises in deciding an algorithm that suits the current situation. One must have a clear idea of the algorithms to select an appropriate one for the case under study. Also, when RL algorithms utilize the advantages of other techniques, the results obtained are mindblowing.

This paper focuses on different RL algorithms. The discussion introduces some of the well-known algorithms and familiarizes the environments where these algorithms can be used. The paper also addresses the advantages and disadvantages of algorithms under consideration.

The study is organized as follows. Section 2 focuses on the preliminary knowledge of the RL algorithms. Section 3 provides a foundation to lay a better understanding of the existing systems, which will act as the basis for this paper. A detailed analysis of *RL models (RLMs)* is presented in Sect. 4. Section 5 gives an overview of datasets, simulation platforms and performance metrics used in *RL-based vehicular traffic control models (RL-VTCMs)*. The open challenges and recommendations of the system and the conclusion are given in Sects. 6 and 7, respectively.

**Fig. 1** Framework of the RL scenario



## 2 Reinforcement Learning

*Reinforcement learning (RL)* refers to a kind of ML method which analyzes how the software agents are ought to take actions in their environments to maximize the cumulative reward. Nowadays, it is used in various software and machines to find the most suitable path or state it should take while considering the present scenario. This section provides the preliminaries regarding RL.

The environment means the object on which the agent is acting. The agent is the RL algorithm. Initially, without any prior knowledge of how to behave, the agent starts interacting with its environment. The input is sent to the agent by the environment. The input is a state. Then, the agent takes action based on the knowledge it gained as a response to the received state. Via an interpreter, the environment sends a pair of next state and reward back to the agent. The reward, which is either positive or negative, solemnly depends on the agent's action. The negative reward is usually referred to as punishment to the agent. Also, to evaluate its last action, the agent updates its knowledge using the reward obtained. The agent iteratively learns and reaches the optimal condition.

Even now, few are confused with *Supervised Learning (SL)* and RL. Table 1 gives a comparison between RL and SL to bring in more clarity.

The sequential nature of RL can be explained as follows. The output depends on the state of current input, which depends on the previous output. i.e., the input at a particular time always considers the output of the previous cycle. Thus, a chain is formed. To predict future output, the SL algorithms apply the knowledge gained to the new data as labeled examples.

**Table 1** Reinforced learning versus supervised learning

Reinforced learning	Supervised learning
Makes decisions sequentially	Decision entirely depends on the initial input
Trial and error search method and delayed reward	Starts functioning by analyzing a known training dataset
The learning algorithm interacts with its environment	The learning algorithm produces an inferred function
Labeling of the sequence of dependent decisions	Labeling of each decision since they are independent
Example: Games	Example: Object recognition

The continuous interaction with the environment benefits software agents and the machines to automatically determine the specific context's quintessential behavior to maximize its performance. After sufficient training, the ideal output is attained for any new input if the system is provided with a suitable dataset.

RL requires a reward feedback method known as the reinforcement signal. The agent learns using this reinforcement signal. The action corresponding to each reward is analyzed to find the best. The learning algorithm compares its obtained output with the correct, intended output and thereby calculate the error. Accordingly, the necessary modifications are made in the model.

## 2.1 Classification of Reinforcement Learning Algorithms

The RL algorithms can be classified based on different factors such as the reward, model, action space, policy. These classifications are explained below:

**Based on Reward** The reward-based classification mainly depends on the nature of the reward. In practical cases, RL is categorized into positive RL and negative RL. An RL algorithm is said to be positive reinforcement when an event increases the strength of the behavior; i.e., an event occurs because of a particular behavior of the agent. A reward is assigned to the agent. If that reward helps to maximize performance, then it is positive reinforcement. Alternatively, in other words, in positive RL, the reward said to be a positive effect on the behavior. In negative RL, the system is trained to stop or avoid unfavorable conditions, which reduces strength. Such an action strengthens behavior. It resists the minimum standard of performance.

**Based on Model** Model-free and model-based are the two classifications of RL algorithms based on the model. The *transition probability distribution (TPD)* is also known as the transition model. The model of the environment contains both TPD and *Reward Function (RF)*. In the model-free algorithm, the TPD and the RF associated with the environment are not utilized.

Let the current state be  $s_0$  and action be  $a$ . Performing  $a$ , the model reaches  $s_1$  from  $s_0$ . The model analyzes and learns the transition probability function  $T$ . In



this case,  $T(s1|(s0, a))$ ). By successfully analyzing, the agent determines the chance to enter into a particular state from the current state by taking a specific action. In model-free algorithms, the peculiar trial-and-error method of RL algorithms is used. In each trial, the model gains some knowledge. Correct action helps the model to optimize the output. The wrong action helps the model to update itself to stay away from entering into unfavorable states. Because the model earns some knowledge from all its trials, there is no need to store the transitions.

The absence of state space and action space makes the model-free RL algorithms more demanding than model-based. The cases where the transitions have to be saved uses model-based RL algorithms. As the state-space and action-space grow, the effective utilization of storage space became impractical. Model-based RL algorithms are preferred in scenarios where the system could decide the next move based on a trained model, without interacting with the current environment. Conversely, if the system decision needs a continuous interaction with the environment, such as a real-time traffic regulation system, model-free RL algorithms will perform well.

**Based on Action Space** Based on the action space, RL agents can have two categories of action spaces, namely discrete and continuous action space. If the agent decides the next action from a finite action set, it is called discrete action space algorithms. Instead, in the continuous action space, a single real-value vector is used to represent the entire action space. The difference in actions cannot be expressed because of the single vector representation. In discrete action space, the fine-tuning of action selection is done. Also, discrete action space is more suited for value-based approaches. Further, a discrete action space approach is engaged in cases where small action space is required. The continuous action space is required when the size of action space grows to infinity.

**Based on Policy** In the policy-based RL algorithms, the main objective is to maximize the reward. The policy defines the behavior of an agent at a particular time. In other words, it is a mapping from learned states to actions to be taken when the agent reaches those states. These algorithms try to determine the action to be taken at a state to attain the maximum reward in the forthcoming steps.

The algorithm fine-tunes a vector of parameters to attain the objective. For example, to select the best action to be taken under the policy  $\pi$ , a vector of parameters say  $\theta$  is adjusted. This example is mathematically represented as follows:

$$\pi(a|s, \theta) = Pr \{A_t = a | S_t = s, \theta_t = \theta\} \quad (1)$$

Right-hand side (RHS) of Eq. 1 means that, at time interval  $t$ , the best action to be taken is  $a$  from state  $s$  by tuning the parameter  $\theta$ . Left-hand side (LHS) implies that the agent learns this knowledge. The entire learning or training phase follows the same policy.

The two types of policy-based RL algorithms are on-policy and off-policy algorithms. The agent learns the Q-function so that the probability of goodness of each action is determined. Among the results, the best one selected stochastically. Such a learning approach is known as on-policy RL algorithms. On the other hand, a

**Table 2** Reinforced learning algorithms: comparison

Algorithm	Model	Policy	Action space	State space
Q-learning	Model-Free	Off-policy	Discrete	Discrete
SARSA	Model-Free	On-policy	Discrete	Discrete
Q-learning- $\lambda$	Model-Free	Off-policy	Discrete	Discrete
SARSA- $\lambda$	Model-Free	On-policy	Discrete	Discrete
DQN	Model-Free	Off-policy	Discrete	Continuous
DDPG	Model-Free	Off-policy	Continuous	Continuous
Actor-critic	Model-Free	On-policy	Continuous	Continuous
A3C	Model-Free	On-policy	Continuous	Continuous
NAF	Model-Free	Off-policy	Continuous	Continuous
TRPO	Model-Free	On-policy	Continuous	Continuous
PPO	Model-Free	On-policy	Continuous	Continuous
TD3	Model-Free	Off-policy	Continuous	Continuous
SAC	Model-Free	Off-policy	Continuous	Continuous

greedy decision is taken to take action with the best Q-value. The Q-value is learned by using other different algorithms. Such algorithms are called off-policy RL algorithms. Based on their nature, these kinds of reinforcement algorithms are sometimes referred to as stochastic and deterministic reinforcement algorithms, respectively.

Policy-based algorithms can exhibit better convergence. These algorithms are suitable even in higher-dimensional action spaces. The more attractive characteristics of these algorithms are their stochastic nature. Though the policy-based algorithms have many advantages, they still possess some disadvantages. Rather than converging into the global optimum, these algorithms converge to the local optimum. In mathematics and computer science, global optimum gives the optimal solution among every possibility. Local optimum is not preferred because it is the best solution to a problem only within a small neighborhood of possible solutions.

Also, the policy-based algorithms have higher variance. But a small variance characterizes an efficient estimation. The important reinforcement algorithms with the properties mentioned above have been compared. The main traits of them are given in Table 2.

### 3 Related Work

Researchers have proposed several solutions to solve conventional TSC problems. This section discusses the important among such solutions put forward by the researchers.

An RLM that uses the Q-learning algorithm with action-value approximation has been used to build an online model-free traffic signal controller [7]. This work focuses

on both average delay and queue length, rather than considering only the average delay. Also, it utilized the advantage of ANN to train the model according to the temporal difference. However, [7] failed to consider unexpected dynamic scenarios in real time.

The model-free RL algorithm can contribute highly to the traffic signal control problem by combining the prior traffic knowledge with a deep RL approach, which is shown in [8]. Here, Q-learning is used to train another approach named *Mixed Q-network (MQN)*. A model can learn traffic patterns and then find out the most suitable agent. The biggest failure of such a system is finding a suitable traffic pattern detector according to the dynamic traffic structures.

Most of the earlier studies have succeeded in developing traffic signal controllers (with restricted action selection) with the help of peculiar properties of basic RL algorithms. Two RL adaptive traffic signal controllers were designed to analyze their learned policies and compare them to a Webster's controller [9]. The controllers were implemented by using asynchronous Q-learning and advanced adaptive actor-critic algorithms. The neural network function approximation has also been added to the design. Interval became constant due to the fixed green signal duration for the scenario under observation. If the action selection is made dynamic, the agent could control the environment better. Also, in the testing scenario, each intersection was controlled by an isolated RL agent. Hence, the model cannot be considered as a multiagent RL system.

Q-learning techniques maximize the number of vehicles passing a junction and adjust the roads' signals by observing the variation of queue lengths and throughput as the key parameters [10]. However, this system fails to evaluate the accuracy of the model in multiple intersection roads. Also, the data transfer between the traffic island have not been considered in this study.

The time delay, the number of idle vehicles, and the combined saturation were estimated from the experience to learn and determine the optimal actions preserving the traffic signal timing efficiently [11]. The work modularized the actual continuous traffic states for simplification purposes.

The spectacular properties of *Deep Q-Network (DQN)* have a lot to help with TSC models [12]. Further, DQN is used in learning models in modern ride-sharing platforms [13]. The model-free DQN learns the optimal vehicle dispatch policies from its interaction with the environment. However, some crucial detailing is missing in this study. Scalability, fault tolerance, reliability, and availability of shared data also have to be considered.

European countries are well versed with the advantage of group-based signal control that provides flexible phase structures. Most of the existing systems used simple timing logic in implementation. Jin and Ma [14] try to formulate the existing system as an adaptive multi-agent system by incorporating Q-learning and SARSA. Nevertheless, the work lacks the handling of real-time scenarios and the issues associated.

*R-Markov average reward technique (RMART)* is suited for an environment among signal controllers in a connected vehicle environment [15]. The research took eighteen signalized intersections to implement the idea in a hypothetical network by assuming the learning parameter and discount factor to be arbitrary. Aragon-Gómez

and Clempner [16] address a multiagent continuous-time schedule problem and proposes a learning scheme for it. Thus introduced an RLM (based on the temporal difference method) by observing traffic signal control problems as *continuous-time Markov games (CTMG)*. Transition rates and reward points are calculated accordingly. However, some shortcomings in this work include the lack of incorporation of a collaborative approach and the method's robustness when exposed to a real-time environment.

Vehicles are used not only for travel. They are also used for goods transportation. Therefore, traffic control is one of the primary demands for manufacturing companies too. In the future, more emphasis will be given on automation. Therefore, the product's timely delivery to the consumer is also a factor that affects the product's quality and production cost. The *deep reinforcement learning (DRL)* model paves a solution to this via dynamic routing strategy [17]. The traffic states and actions can be predicted using DRL combined with a Q-learning step and a *recurrent neural network (RNN)*. Hence by reducing the delivery time and delay, the different combinations of states, actions, rewards are utilized for the modeling. Still, the model failed to consider a few other dependent factors/causes of traffic congestion.

Lack of proper traffic control not only creates traffic congestion but also adversely affects safety, time, efficiency, and energy. These problems are also heating up with the advent of autonomous cars and electric cars. Therefore, ongoing research work has begun using RL techniques to address these issues [18–20]. RL techniques can also be used intelligently and appropriately to facilitate learning and problem-solving in many other traffic-related areas [21, 22].

## 4 RL Models in Vehicular Traffic

This section presents some of the RLMs that are used in vehicular traffic for traffic automation purposes. A review of RLMs and their strengths to address traffic control challenges is given in Table 3. Also, Table 4 reports RLMs and the attributes for the vehicular traffic regulation system.

### 4.1 Multiagent Reinforcement Learning

Most intelligent systems nowadays highly depend on multiple agents competing with each other to improve the system's overall behavior. Such a process that incorporates RL algorithms is known as *multiagent reinforcement learning (MARL) Algorithm*. The combinational availability of *state-action pairs (SAPs)* increases exponentially with the number of agents. In other words, the number of agents is directly proportional to the number of SAPs. In MARL, the agents exchange information. Based on all the available and received data, the agents coordinate their actions to achieve global Q-value optimization. The most attractive feature of MARL is its scalability (i.e., adding new agents quickly) [12, 23–25].

**Table 3** Summary of RL models

RLM	Strengths	Challenges
MARL	Highly scalable, inherently robust, follows top-down approach	Exponential complexity, curse of dimensionality, difficult to state the learning goal
MBRL	Heeds the longer-term effects of an action under a state	Exponential decay in eligibility trace due to trace decay parameter
MPRL	Follows top-down approach, addresses the curse of dimensionality	The algorithms converge to a point after a finite number of iterations
RLFA	Addresses the curse of dimensionality, saves computation time and memory space	May produce an inconvenient result, adjustable weights oscillate within a region

**Table 4** Summary of RL models for vehicular traffic regulation systems

Representation	Attributes	MARL [2]	MBRL [27]	MPRL [3]	RLFA [4]
Agent	Traffic signal control	Yes	No	Yes	Yes
	Traffic movement	No	Yes	No	No
State	Queue size	Yes	No	Yes	Yes
	Current traffic phase	No	Yes	Yes	No
	Traffic phase split	No	Yes	Yes	No
Action	Traffic phase type	Yes	Yes	Yes	Yes
	Traffic phase split	No	Yes	No	No
Reward	Variation of vehicular delay	No	Yes	No	No
	Waiting time	Yes	No	No	Yes
	Variation of queue size	Yes	No	Yes	Yes

The main challenge for the agents in a shared dynamic environment lies in learning the situation and making a better decision. The same is the case with traffic also. In vehicular traffic scenario, the action of an agent at an intersection point can affect and vary with the agent’s decision at the neighboring intersection point, which may also affect the agent’s self-performance. In case of a wrong decision, there is a high probability of having high congestion in the nearby intersections. Hence, each agent should take and communicate optimal actions and coordinate with each other. MARL is a helpful model in such cases [2, 26]. MARL that tries to optimize the global Q-value is used for the traffic regulation system [2]. The inappropriateness of the traffic phase is tackled using a distributed model [26].

## 4.2 *Multistep Backup Reinforcement Learning*

The optimal action decided by a typical RL algorithm highly depends on the present state. Usually, an action affects a consecutive series of states. In *multistep backup reinforcement learning (MBRL)*, the average outcomes of the temporal differences are calculated inside an episode (a series of time instants). Based on this data, the agent updates the Q-values. MBRL focuses on the long-term payoff for an action related to a state. The average effects of temporal difference are attained using most fitting traces.

The MBRL model cut down the average hold-up time by considering the phase sequence and phase split of traffic in an intersection with a single lane traffic network [27]. Traffic phases with grouped individual traffic give the traffic phase split for processing. An episode is a duration between activation and termination of the green signals with the combination of traffic movements. Each time a SAP is visited, its value is set to one. This value gets updated for all the visits, and the eligibility trace adds more credit to recent SAPs. The temporal difference is being weighted using the eligibility traces. The Q-value of an episode is updated using this temporal difference. A trace decay parameter exponentially decays the eligibility trace of an unvisited SAP.

## 4.3 *Max-Plus Reinforcement Learning*

Agents in a coordination graph are interconnected. A max-plus algorithm calculates and exchanges the local and global payoffs among these agents. As part of the optimal joint action, agents use the payoff values to determine their corresponding action. A *max-plus reinforcement learning (MPRL)* follows a top-down approach. This modularization helps them to confront the challenge of dimensionality. The probability of better results in an oversaturated network is calculated by incorporating MRPL in the reward structure of Q-learning agent in the design of a traffic signal control [3].

The agent  $i$  sends locally optimized payoffs to its neighbor  $j$  via the edges connecting them. The action taken by  $j$  determines the payoff. After a finite number of iterations, the algorithm converges to a fixed point. It is possible to increase the throughput of traffic signal and reduce the number of stops per vehicle to some extent [3].

## 4.4 *Reinforcement Learning with Function Approximation*

Commonly, in a shared dynamic space, the number of SAPs can be huge in number. The SAPs increase exponentially when the number of agents increases, leading to a diminishing scalability scope. Thus, RL faces the challenge of dimensionality. This issue can be solved to some extent by introducing *function approximation (FA)* logic

in RL. Instead of many SAPs, FA stores and pays attention to an appreciably smaller amount of features. Thus reduces memory/storage capacity, improve scalability, and reduce learning time. In *RL with FA (RLFA)*, Q-values are represented using tunable weight vectors and feature vectors [1, 4, 28, 29].

Consider a real-world traffic network based on Bangalore,  $2 \times 2$  and  $3 \times 3$  grids, and sixteen trivial streets traffic network using a centralized model. Here, the RLFA approach addresses the challenge in traffic phase sequence in a two-way intersection by optimizing the global system performance. RLFA helps to increase throughput and reduce waiting time [4, 28].

## 5 Datasets, Simulation Platforms, and Performance Metrics Analysis of RL-VTCMs

This section includes analyzing performance metrics used in traffic-related research and simulation platforms used in such studies. Also, it investigates the datasets used in RL-VTCMs. Table 5 gives a summary of the performance metrics.

### 5.1 Benchmarked Datasets for RL-VTCMs

Some of the benchmarked datasets that focus on autonomous navigation are ADE20K [30], *Berkeley Deep Drive (BDD)* [31], Cityscapes [32], Camvid [33], Daimler [34], IDD [35], KITTI [36], Leuven [37], and Mapillary Vistas [38]. The different lighting circumstances and the multiple cameras and sensors in the cities help the Cityscapes provide a large amount of data. The Mapillary Vistas Dataset creates the imagery of street scenes. Images from different angles of the road and its surroundings are present

**Table 5** Summary of performance measures

Performance measures	MARL [25]	MBRL [27]	MPRL [3]	RLFA [4]
Lower average waiting time	✓			✓
Lower average delay	✓	✓		
Lower number of stops per vehicle			✓	
Smaller queue size		✓		
Higher throughput	✓		✓	✓

in this dataset, irrespective of the cameras that captured them. They have no video data. The Berkeley Deep Drive Dataset concentrates on autonomous navigation. For ADE20K, the general locale parsing issue is the main area of interest. Dashboard cameras are used on the BDD100K to capture images. The glass in front of the cameras adversely affects the image quality. It can get worse in rainy conditions. IDD can be used to ensure security and reliability in unusual and extreme cases.

## 5.2 *Simulation Platforms*

Some discrete-event simulators are developed using programming languages such as C/C++ and tools such as MATLAB. There exist macroscopic and microscopic approaches for traffic simulators with a graphical user interface (GUI). Most traffic simulators embrace the microscopic approach, including VISSIM, SUMO, TSIS, and ITSUMO.

## 5.3 *Performance Measures*

Appropriate performance measures are required to assess the merits of any traffic control system. These parameters are essential in RL based TSC; because an agent needs to assess his own performance to learn from experience. Some of the performance measures used in vehicular traffic are reduction of fuel consumption, reduction of emissions, the number of stops in a journey, percentage of stopped vehicles, average delay, *average trip waiting time (ATWT)*, vehicle density at different parts of the network, queue length, and average vehicle speed. Table 5 reports some of the performance measures accomplished by the RLs and algorithms.

# 6 **Open Challenges and Recommendations**

After discussing the major algorithms and models in RL, here we examine various challenges that need to be addressed during their usage. This section throws light into the important hurdles in using RLs and algorithms in ITS. It also includes suggestions for handling these challenges.

- **Injecting RL in unfitting circumstances-** RL is propitious and fastly advancing technique in a variety of fields such as Resources management in computer clusters, Traffic Light Control, Robotics, Games, and Chemistry. Too much reinforcement leads to states overload, followed by the diminishment of results. The inappropriate parameters and assemblage of payoff messages lead to poor system performance, even during the initial learning phase.



- **Availability of data** When enough data are available, SL methods are preferred. This is due to the fact that when action space is large enough, the RL algorithm becomes time-consuming.
- **Real-time environment** In a shared and dynamic environment like traffic regulation, RL algorithms and models have to include the recent advances in ITS to exhibit their full strength. A better traffic regulatory system comprises almost all the dynamic parameters such as traffic density, road utilization, and vehicles.
- **Self adaptiveness** Aim of the current researches is to build an automated traffic regulatory system that performs self-configuration of the dependent parameters to adapt with the dynamicity of traffic. The interoperability is usually affected by the communication overhead. Hence, the exchange of control messages needs a limit by eliminating unwanted control messages, by which the learning rate of the system also improves. The agent is expected to learn new and unexpected actions and states in the operating environment.
- **External impediments** The weather conditions such as rain, flood, fog are the factors that pull down the hope of a fully automated self-paced traffic regulatory system. Not only this, but also the traffic flow(in and out) and the disturbance in traffic flow make the problem worse. In upcoming traffic regulation proposals, all such situations have to be taken care of.

RL enhances system performance in scenarios with fewer data, such as in traffic regulatory systems. Hence, in developing countries with very few publicly available traffic datasets, RL has a huge impact in developing better VTCMs. Integrating RL with advanced technologies such as fuzzy logic, game theory, and AI; fastens the ride towards an extremely self-paced traffic regulatory system. These technologies help to include prior knowledge and obtain optimal actions. The analysis of prior traffic data, gained knowledge, approximation, and conventional control systems are required for a better traffic control model. Agents in the model use the traffic observer's information for increasing the learning rate in the (re)learning phase to achieve enhanced system performance.

## 7 Conclusion

In this paper, we have reviewed the RLMs and algorithms with an emphasis on the applicability in traffic regulation systems. The ability of RL algorithms to determine actions that yield highest rewards can be regarded as the prime reason for their wide acceptability. Consequently, a study on the RL algorithms can reveal the intrinsic features which in turn can be utilized effectively for handling traffic regulation issues. The paper provides such a detailed review of the RL algorithms, but it is not limited to that.

In addition to providing an in-depth analysis of various RL algorithms, the paper also discusses the issues that need to be rectified for its hurdle-free application in traffic regulation systems. These issues demand immediate attention of researchers, especially considering the fact that we are fast progressing towards a world which is 'smart' in all aspects.

## References

1. B. Yin, Traffic network micro-simulation model and control algorithm based on approximate dynamic programming. *IET Intell. Transport Syst.* **10**, 186–196 (2016). <https://digital-library.theiet.org/content/journals/10.1049/iet-its.2015.0108>
2. K.J. Prabuchandran, A.N. Hemanth Kumar, S. Bhatnagar, Decentralized learning for traffic signal control, in *2015 7th International Conference on Communication Systems and Networks (COMSNETS)*, pp. 1–6 (2015)
3. J.C. Medina, R.F. Benekohal, Traffic signal control using reinforcement learning and the max-plus algorithm as a coordinating strategy, in *2012 15th International IEEE Conference on Intelligent Transportation Systems*, pp. 596–601 (2012)
4. L.A. Prashanth, S. Bhatnagar, Threshold tuning using stochastic optimization for graded signal control **61**, 3865–3880 (2012)
5. Y. Matsumoto, K. Nishio, Reinforcement learning of driver receiving traffic signal information for passing through signalized intersection at arterial road. *Transp. Res. Proc.* **37**, 449–456 (2019)
6. M.A. Wiering, Multi-agent reinforcement learning for traffic light control, in *Machine Learning: Proceedings of the Seventeenth International Conference (ICML'2000)*, pp. 1151–1158 (2000)
7. G. Tolebi, N.S. Dairbekov, D. Kurmankhojayev, R. Mussabayev, Reinforcement learning intersection controller, in *2018 14th International Conference on Electronics Computer and Computation (ICECCO)*, pp. 206–212 (2018)
8. J. Zeng, J. Hu, Y. Zhang, Training reinforcement learning agent for traffic signal control under different traffic conditions, in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pp. 4248–4254 (2019)
9. W. Genders, S. Razavi, Policy analysis of adaptive traffic signal control using reinforcement learning. *J. Comput. Civ. Eng.* **34**(1), 04019046 (2020)
10. H. Joo, S.H. Ahmed, Y. Lim, Traffic signal control for smart cities using reinforcement learning. *Comput. Commun.* (2020)
11. X. Zhou, F. Zhu, Q. Liu, Y. Fu, W. Huang, A Sarsa ( $\lambda$ )-based control model for real-time traffic light coordination. *Sci. World J.* vol. 2014, (2014)
12. Y. Gong, M. Abdel-Aty, Q. Cai, M.S. Rahman, Decentralized network level adaptive signal control by multi-agent deep reinforcement learning. *Transp. Res. Interdisciplinary Perspect.* **1**, 100020 (2019)
13. A.O. Al-Abbasi, A. Ghosh, V. Aggarwal, DeepPool: distributed model-free algorithm for ride-sharing using deep reinforcement learning. *IEEE Trans. Intell. Transp. Syst.* **20**(12), 4714–4727 (2019)
14. J. Jin, X. Ma, Adaptive group-based signal control by reinforcement learning. *Transp. Res. Proc.* **10**, 207–216 (2015). <http://www.sciencedirect.com/science/article/pii/S2352146515002574>, 18th Euro Working Group on Transportation, EWGT, 14–16 July 2015 (Delft, The Netherlands, 2015)
15. H.A. Aziz, F. Zhu, S.V. Ukkusuri, Learning-based traffic signal control algorithms with neighborhood information sharing: an application for sustainable mobility. *J. Intell. Transp. Syst.* **22**(1), 40–52 (2018)

16. R. Aragon-Gómez, J.B. Clempner, Traffic-signal control reinforcement learning approach for continuous-time markov games. *Eng. Appl. Artif. Intell.* **89**, 103415 (2020). <http://www.sciencedirect.com/science/article/pii/S0952197619303239>
17. Y. Kang, S. Lyu, J. Kim, B. Park, S. Cho, Dynamic vehicle traffic control using deep reinforcement learning in automated material handling system, in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 9949–9950 (2019)
18. X. Qi, Y. Luo, G. Wu, K. Boriboonsomsin, M. Barth, Deep reinforcement learning enabled self-learning control for energy efficient driving. *Transp. Res. Part C: Emerg. Technol.* **99**, 67–81 (2019)
19. Y. Wu, H. Tan, J. Peng, H. Zhang, H. He, Deep reinforcement learning of energy management with continuous control strategy and traffic information for a series-parallel plug-in hybrid electric bus. *Appl. Energy* **247**, 454–466 (2019)
20. C. You, J. Lu, D. Filev, P. Tsiotras, Advanced planning for autonomous vehicles using reinforcement learning and deep inverse reinforcement learning. *Robot. Autonomous Syst.* **114**, 1–18 (2019)
21. D.M. Vlachogiannis, E.I. Vlahogianni, J. Golias, A reinforcement learning model for personalized driving policies identification. *Int. J. Transp. Sci. Technol.* (2020)
22. Y. Ye, X. Zhang, J. Sun, Automated vehicle's behavior decision making using deep reinforcement learning and high-fidelity simulation environment. *Transp. Res. Part C: Emerg. Technol.* **107**, 155–170 (2019)
23. L. Busoni, R. Babuska, B. De Schutter, Multi-agent reinforcement learning: a survey, in *2006 9th International Conference on Control, Automation, Robotics and Vision* (IEEE, New York, 2006), pp. 1–6
24. L.L. Lemos, A.L. Bazzan, Combining adaptation at supply and demand levels in microscopic traffic simulation: a multiagent learning approach. *Transp. Res. Proc.* **37**, 465–472 (2019)
25. S. El-Tantawy, B. Abdulhai, H. Abdelgawad, Multiagent reinforcement learning for integrated network of adaptive traffic signal controllers (MARLIN-ATSC): methodology and large-scale application on downtown Toronto. *IEEE Trans. Intell. Transp. Syst.* **14**(3), 1140–1150 (2013)
26. K. Zhang, Z. Yang, T. Başar, Multi-agent reinforcement learning: a selective overview of theories and algorithms. *arXiv preprint arXiv:1911.10635* (2019)
27. J. Jin, X. Ma, Adaptive group-based signal control using reinforcement learning with eligibility traces, in *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, pp. 2412–2417 (2015)
28. L.A. Prashanth, S. Bhatnagar, Reinforcement learning with function approximation for traffic signal control **12**, 412–421 (2011)
29. T. Chu, J. Wang, Traffic signal control with macroscopic fundamental diagrams, in *2015 American Control Conference (ACC)*, pp. 4380–4385 (2015)
30. B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, A. Torralba, Scene parsing through ADE20K dataset, in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5122–5130 (2017)
31. F. Yu, W. Xian, Y. Chen, F. Liu, M. Liao, V. Madhavan, T. Darrell, Bdd100k: a diverse driving video database with scalable annotation tooling. *arXiv preprint arXiv:1805.04687* 2(5), 6 (2018)
32. M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, B. Schiele, The cityscapes dataset for semantic urban scene understanding. *CoRR* <http://arxiv.org/abs/1604.01685> (2016)
33. G.J. Brostow, J. Fauqueur, R. Cipolla, Semantic object classes in video: a high-definition ground truth database. *Pattern Recogn. Lett.* **30**(2), 88–97 (2009)
34. T. Scharwächter, M. Enzweiler, U. Franke, S. Roth, Efficient multi-cue scene segmentation, in *German Conference on Pattern Recognition* (Springer, Berlin, 2013), pp. 435–445
35. G. Varma, A. Subramanian, A. Nambodiri, M. Chandraker, C. Jawahar, IDD: a dataset for exploring problems of autonomous navigation in unconstrained environments, in *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)* (IEEE, New York, 2019), pp. 1743–1751

36. M. Menze, A. Geiger, Object scene flow for autonomous vehicles, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3061–3070 (2015)
37. B. Leibe, N. Cornelis, K. Cornelis, L. Van Gool, Dynamic 3D scene analysis from a moving vehicle, in *2007 IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, New York, 2007), pp. 1–8
38. G. Neuhold, T. Ollmann, S. Rota Bulo, P. Kotschieder, The mapillary vistas dataset for semantic understanding of street scenes, in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4990–4999 (2017)

# Speaker Identification Approach for the Post-pandemic Era of Internet of Things



A. Saleema and Sabu M. Thampi

**Abstract** With the rapid aggravation of COVID-19 pandemic, the organizations, industries, institutions, etc., are forced to rapidly adapt to social distancing by limiting physical contact and thereby limit the person to person contamination. The scenario of Internet of things in the post COVID era will be interesting indeed. In order to ensure public health, the social distancing and semi lockdown will continue in the foreseeable future, and therefore, the need of secure remote person authentication methods is being more and more critical especially in the Internet of things which is a multitude of networks consisting of a huge number of uniquely identifiable devices. As far as human\_device authentication strategies are concerned, the one which needs less human involvement will be preferable in a post COVID-19 IoT scenario. Moreover, since the range of IoT devices may span from tiny sensors to complex machines, an authentication method which will be adaptable to each and every type of device will be more welcomed. Considering these facts, voice biometric authentication seems to be the most suitable one which can provide a balanced mix of security, adaptivity and convenience to such an advanced world of connectivity. Here, we introduce a lightweight text independent voice biometric method for IoT using extreme learning machines, and we perform a comparative analysis with a deep learning-based method of speaker identification using 3D convolutional neural networks. We have performed experimental study using different datasets and concluded that the extreme learning-based method is more suitable for IoT, considering the trade-off between the recognition accuracy and the training time requirements.

**Keywords** Speaker identification · Authentication · Voice biometrics · Post covid 19 authentication · Authentication in the internet of things

---

A. Saleema (✉)

Cochin University of Science and Technology, Kochi, India

e-mail: [saleema.res17@iiitmk.ac.in](mailto:saleema.res17@iiitmk.ac.in)

A. Saleema · S. M. Thampi

School of Computer Science & Engineering, Indian Institute of Information Technology and Management-Kerala (IIITM-K), Thiruvananthapuram, Kerala, India

e-mail: [sabu.thampi@iiitmk.ac.in](mailto:sabu.thampi@iiitmk.ac.in)

## 1 Introduction

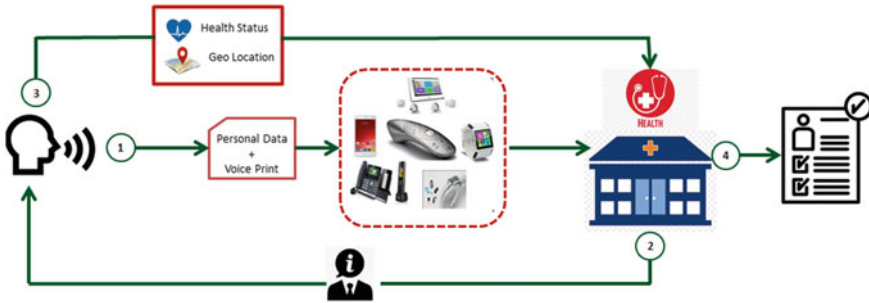
Since the novel Coronavirus, COVID-19 has aggravated the entire world, and our digital society is in high demand of rapid technological advancements that can help fight the spread of the disease at an international level. Although lockdowns and social distancing measures have been implemented, it cannot be continued in a long run since these will lead to the largest global economic crisis until otherwise advanced technological solutions that can adapt to the situation are deployed. The Internet of things entwined with artificial intelligence seems to extend an indispensable role in fighting the pandemic since IoT allows quick data collection on a vast scale and AI supports rapid data analysis and processing.

In the context of COVID-19, the business, financial, academic units which have never executed remote work are now required to operate in a fully remote mode. Since there arise serious risks and threats due to these work from home and remote connectivity policies, remote and live biometric authentication methods are of significant demand. Moreover, the enforcement of social distancing and less human interaction demands touchless biometric authentication methods in the post COVID-19 era. Considering these facts, the biometric research community has been oriented toward unveiling the potential of live voice authentication methods, which is the major motive of this work.

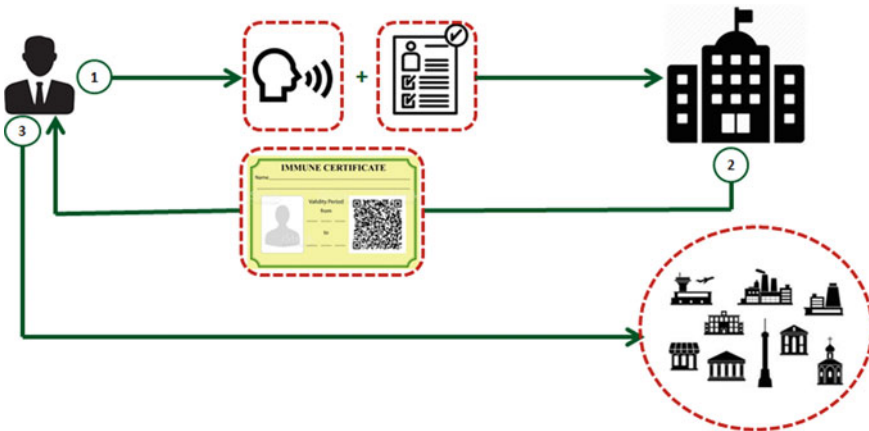
The health department as well as other government authorities has already begun implementing biometric-based quarantine management systems. Several crisis management technologies have been deployed as mobile applications which can perform identity assurance in order to aggregate personal health data at the national level. These applications are obligatory for action planning, decision making and the provision of essential aids to the citizens. Figure 1 shows a pandemic crisis management solution developed by Maharaja Associate and Tech5, reconceptualized with voice biometrics [1]. Citizens are allowed to confirm proof of presence, and thereby, the administrative authorities can keep track of the health status, especially when they are in quarantine/isolation. By making use of the IoT sensors, smart devices, etc., quick and accurate remote identity verification, health status monitoring, geo location verification, etc., can be performed in an excellent manner.

Nations like South Korea have already tried the mass testing approach of COVID-19 antibodies, in order to document “herd immunity” at a regional level. The physical documents are progressively replaced with the digital ones, and the concept of immunity certificate/passport will be of significant importance in the post-pandemic era. Digital immunity certificates can be defined as a person’s essential digital id which integrates his/her own biometrics, that allows a person to freely move around, to end self-isolation, etc., that is to get back to the normal lifestyle. Figure 2 shows the illustration of implementing the concept of immunity passport adapted from [2], reconceptualized with voice biometrics.

The potential of the Internet of things in combating the post-pandemic era also has been explored by researchers, especially in building the future healthcare systems. The use of wearables, drones, robots, IoT buttons, smart applications will be



**Fig. 1** Voice recognition-based COVID-19 quarantine management system. Step 1: Self-enrollment with the application using live voice and personal data through one’s smart device. Step 2: Upon successful enrollment, the health department entity requests the person under quarantine for the proof of presence. Step 3: The person provides the geolocation and health status in his/her live voice, ensuring the identity as well as proof of presence. Step 4: The health department verifies the identity by comparing the voice template with the enrolled voice print and prepares the health report to be submitted to the higher government authority



**Fig. 2** Notion of immunity passport based on voice biometric authentication. Step 1: The person submits the personal data and the antibody test result (which he received from an authorized authority) with his live voice Step 2: The government entity verifies the received data and generates the “immunity passport” which is a high definition barcode consisting of all the gathered data encrypted with PKI, so that the right owner can only unlock it with his biometrics. Step 3: This digital immunity passport can be carried by the person and can submit wherever it is asked for

vital in providing higher quality of services and advanced user experiences in the healthcare field. The tracking of quarantine, isolation, fast diagnosis, data collection, user identification/authentication, reducing contamination, smart disinfecting, etc., can be deployed in the near future which makes IoT the frontier to contend with the post-pandemic era. However, IoT has to overcome challenges like security, privacy, implementation, interoperability, connectivity, compatibility, etc., with the

enormously increasing connected smart devices being deployed in homes, offices and other places. Among these, security remains as the biggest challenge since a wrongly authenticated human to the devices may bring the entire system down. Therefore, identity verification/authentication is the most important thing to be considered in such a context. The traditional authentication methods like login/password on a smart node will become a bottleneck to the IoT technology while considering the efficiency as well as user convenience. The use of biometrics for IoT authentication eliminates the shortcomings of the traditional ones, and it offers accuracy, accountability, security, convenience, versatility and scalability like properties, which an IoT infrastructure demands. The existing biometric authentication methods include physiological as well as behavioral methods, each of which having some merits as well as demerits.

The range of IoT devices span from simple sensor only devices such as smart refrigerators and wearables to complex autonomous intelligent devices like smart cars. Human to device authentication systems that can well adapt to each type of device, each use case and each context will be mission critical in the Internet of Things. Apart from mobile computing and desktop computing, an IoT device can be almost any object from a simple light bulb to complex manufacturing equipment. Considering these facts, voice biometric authentication seems to be the most suitable one which can provide a balanced mix of security, adaptivity and convenience to such an advanced world of connectivity.

Voice biometrics can be described as a process of extracting the voice prints from the voice samples of individuals that can serve as unique identifiers. Commonly referred to as speaker recognition, this process can be categorized into speaker identification, speaker verification, speaker tracking, etc. Depending on whether the voice print is taken from a particular text phrase or not, these can again be categorized into text dependent and text independent speaker recognition methods [3]. The process of speaker recognition is not only relied upon the physical characteristics of individuals, but also to the behavioral characteristics, which makes it the most suitable to ensure security in the Internet of things context. Most often, the voice biometric has been used in security applications to access control to buildings or sensitive data. Banking and financial institutions in case of telephone initiated transfers of huge amounts of money can rely open voice biometric-based authentication systems, if securely deployed.

Researches on speech and speaker recognition have been progressing from several decades. Although many techniques have been proposed for speaker recognition, methods which can be adapted to the Internet of things' challenges are yet to come. Recently, the use of deep neural networks is gaining impression in the field of speaker recognition. Although deep learning can be used for solving intractable problems, they are not well suited to address the challenges in IoT since it demands substantial computing power, which can be a limited resource on many IoT devices. Recently, some techniques like network compression, approximate computing and hardware accelerators have been suggested to employ deep neural nets in IoT devices. But these are proven to be compromised for producing unacceptable drops in accuracy and precision [4].



The major drawbacks of adopting the deep neural network based method are its slow learning speed, trivial human intervention and computational complexity. It is inferred from the existing literature that extreme learning machines can provide better generalization performance at a much faster learning speed and with least human intervention which is more suited with Internet of things. In this paper, we propose two speaker verification methods based on deep convolutional neural networks and extreme learning machines in combination with support vector machines in order to reap high performance.

The major contributions of this research work can be listed as follows

1. To the best of our knowledge, this is the first work to compare the performance of speaker identification based on deep learning (3D convolutional neural networks) and extreme learning machines in terms of recognition rate and training time .
2. We have developed a deep learning framework based on 3D convolutional neural networks with support vector machines for highly accurate speaker verification.
3. We have proposed a fast and highly accurate speaker verification method for the Internet of things based on extreme learning machine entwined with support vector machine.
4. We have analyzed the suitability of our approach for the Internet of things scenario in terms of recognition accuracy and training time requirements.
5. The proposed approaches have been experimented in three datasets and inferred that the extreme learning-based method is more suitable for IoT, while the trade-off between the recognition accuracy and the training time requirements are considered.

The organization of the remaining sections of this paper is as follows. Section 2 gives an overview of some relevant existing research works on speaker recognition. Section 3 describes the details of the proposed approaches. The datasets, experimental setup , results and discussions are provided in Sects. 4 and 5 concludes the whole work.

## 2 Related Works

Automatic speaker recognition methods had its beginning from human aural and spectrogram comparisons and then turned to simple template matching and dynamic time warping approaches and further spanned to modern statistical pattern recognition methods such as neural networks and hidden Markov models [5]. The evolution of speaker recognition models from the traditional to the most recent is illustrated in Fig. 3. The existing speaker modeling methods in the literature consist of spectrogram-based methods [6, 7], Gaussian mixture models [8–10], dynamic time warping [11], vector quantization [12, 13], neural networks [14, 15], hidden Markov models [16, 17] and so on. Mel frequency cepstral coefficients [18], linear predictive coding [19], linear predictive cepstral coefficients [20], perceptual linear predictive



**Fig. 3** Evolution of speaker recognition models from the traditional to the state-of-the-art methods

[21], gammatone frequency cepstral coefficients [22] are some of the successful feature extraction methods for speaker recognition. An overview of the traditional as well as modern techniques for feature extraction and speaker modeling is presented in [3]. The paper details the current state of affairs of the voice biometrics-based recognition, how voice biometrics is connected to the Internet of things, how the concepts of cloud and fog computing are beneficial in connecting voice to the Internet of things, the future trends in voice biometrics research area, etc.

A substantial amount of work has been done regarding the use of deep neural networks in speech as well as speaker recognition. Variani et al. [23] investigate the use of deep neural networks for a small footprint text dependent speaker verification task, in which specific feature named d-vector is extracted and taken as the speaker model. The method is compared to i-vector method [24] and infers that d-vector is more reasonable and the fusion of i-vector and d-vector achieves much better results than a stand-alone i-vector-based system. Later, a method using 3D CNN was proposed in [25], demonstrating that the system can outperform the traditional d-vector methods by 6% in equal error rates. Text dependent setup based on locally connected and convolutional neural networks have also been experimented for speaker verification [26]. Another work which utilizes DNN as a feature extractor and then uses it for speaker modeling is proposed in [27].

A more recent work based on deep CNNs with self-attention [28] shows that the self-attention mechanisms have gained improved performance than traditional i-vector based methods as well as other baseline CNN models. They have used residual neural networks (ResNets) and visual geometry group (VGG) which are the two representations of CNNs. A couple of researchers have attempted the fusion of traditional short-term feature extraction methods with deep learning and gained significant improvements in accuracy. An example of such research work is presented in [29] which combines the MFCC and MFCCT(time-based features) with deep neural networks for text independent speaker identification. The model was compared with five other machine learning algorithms and found deep learning to be the most efficient. The recently introduced conditional adversarial generative networks(CGAN) have also been experimented for speaker identification by some researchers and have proven a great reduction in classification error rate. Also, when compared to the traditional i-vector-based methods as well as the baseline deep learning models, the CGAN showed surprisingly greater improvement [30]. Moreover, the superiority of this model was gained under constrained circumstances with very limited training data, which makes this model promising for short utterance speaker identification. In the future, the other variants of GAN (e.g. CycleGAN) are expected to give astonishingly greater performance than the state-of-the-art models.

Over and above, several cascaded models have gained popularity in text independent speaker identification research. A kind of such model is discussed in [31], which fuses the Gaussian mixture model with the deep neural net, experimented in the emotionally talking environments. Their model has shown better performance than both GMM and DNN in isolation.

The development of speaker verification/identification for real-time applications is very much confronting. Some innovative cutting edge research works regarding the application of speaker identification have been progressing recently. In [32], a speaker identification system was developed for aeronautical applications. They have used multiresolution analysis for feature extraction making use of stationary wavelet transform bands. Their method has earned higher identification accuracy besides high noise reduction. Another intelligent identification system was developed in [33] for the real-time monitoring of sports training. In addition to voice, they have heart rate detection, motion gesture recognition, etc. A smart voice assistance system was presented in [34], which can accept voice commands from a home environment. It is a short speech speaker identification approach, and it made use of the vector quantization, mel frequency cepstral coefficients and principal component analysis ensued with the Gaussian mixture classification model.

A combination of sound processing and machine learning algorithms are applied in [35] for real-time speaker identification. They have utilized Markov chain classifier and real-time MFCC in their study. Another architecture developed for real-time speaker recognition is presented in [36], in which a novel pipeline method is construed. They have identified the requirements of a real-time voice identification system by analyzing the challenges in this field. Comparing with the AlexNet architecture, their method shows superior performance in terms of accuracy, sensitivity, specificity, etc.

### 3 Proposed Speaker Identification Approach

This section details the feature extraction process, the design of the proposed 3D convolution neural networks-based speaker identification approach and the proposed extreme learning machine-based fast learning method for speaker identification. Initially, feature extraction is done using the process Mel frequency cepstral coefficient extraction. Then, some additional features (chroma features) are extracted in order to improve the learning speed and recognition accuracy of the proposed models. These features include

- short-time Fourier transform (stft)
- constant q-transform (cqt) and
- chroma energy normalized statistics (cens).

These features are extremely powerful in summarizing the audio wave, and it consists of the short-time energy distribution of the input signals. For instance, a 12-dimensional chroma feature will encode the short-time energy distribution of the

signal over the 12 chroma bands which correspond to the 12 traditional pitch classes. The stft represents the complex amplitude versus time and frequency of a signal. This is obtained by windowing and applying discrete Fourier transform in each window of the signal. This gives a physical and intuitive representation of the audio signal which has great power in speech/speaker audio analysis. The cqt or constant q-transform will transform a time-domain signal into the time-frequency domain so that the central frequencies of the frequency bins are geometrically spaced and their q factors are all equal. The cens is a robust and scalable feature that captures the dynamics as well as the temporal micro deviations of the audio signal. These are features with low temporal resolution and can be processed efficiently.

### 3.1 Feature Extraction from Input Speech

Voice features that carry speaker-specific information are to be extracted in order to perform speaker identification/verification. The information contained in a speech signal can be high level like dialect, accent, talking style, the subject manner of context, phonetics, prosodic and lexical information or low levels like fundamental frequency, formant frequency, pitch, intensity, rhythm, tone, spectral magnitude and bandwidth for an individual's voice. Among these, features are selected by assigning priority to those having lower intra-speaker variability and higher inter-speaker variability. The other concerns while selecting a feature include robustness against noise and distortion, frequency of occurrence in natural speech, difficulty in mimicry, unaffected by health and easiness in measurability.

Among these, short-term spectral features are found to be the most powerful one as it carries the resonance properties of the supralaryngeal vocal tract. Based on the literature, Mel frequency cepstral coefficients are rated as the best short-term spectral feature due to its high success rate of recognition and strong robustness against noise in the lower frequency regions. The process of extraction of MFCC in our proposed work is illustrated in Fig. 4.

As a preprocessing step, before feature extraction, we can do down sampling of the audio samples if we do not have efficient computational power. For each enrollment as well as evaluation samples, MFCC vectors, which represent the short-term power spectrum of a sound, are extracted. Since ConvNets cannot handle sequence data, we have converted the features extracted from all audio samples to fixed length vectors.

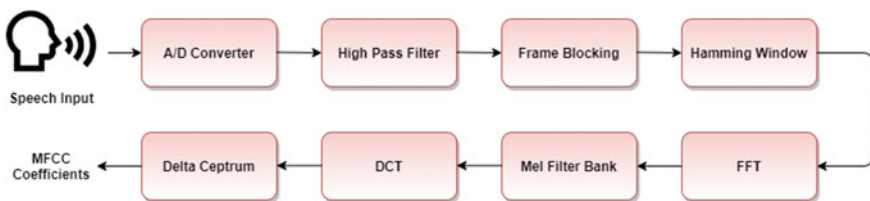


Fig. 4 Block diagram of the process of extraction of MFCC

### 3.2 Proposed Deep Learning Approach for Speaker Identification

The MFCC vectors appended with the additional chroma features (stft, cqt and cens) of the enrollment samples are reshaped to four-dimensional tensors and given as input to the designed 3D convolutional framework. The framework is configured with two convolution layers, two max-pooling layers and a fully connected layer. We have used a filter size of 32 in the convolution layers with a kernel size of  $2 \times 2 \times 2$ . Since the real-world audio data that is to be learned by our ConvNet are nonlinear in nature and the convolution is a linear operation, to account for nonlinearity, a nonlinear function, rectified linear operation has been used as the activation function. In the pooling step, we have used max-pooling to reduce the dimensionality of the feature map produced after convolution, which in turn reduces the number of parameters and computations in the network. Also, pooling makes the network invariant to small variations, distortions and noises in our input audio. For max-pooling also, the kernel size is set as  $2 \times 2 \times 2$ . Figure 5 depicts the architecture of 3D CNN.

Given the target classes which are the actual identities for each audio sample, the network is trained, and the knowledge learned is transferred to an SVM in the form of new feature vectors. That is, we have cut off the fully connected layers of the 3D CNN and used the flattened output to feed the SVM as SVMs are supposed to perform well with smaller amounts of data and high dimensions compared to neural networks. For training the SVM, we have used the same target data given to the 3D CNN. So in the case of testing, the input features in the form of 4-D tensors are given to the trained deep learning framework, and the output vector from the fully connected layers is taken. This vector is then given to the trained SVM to identify the speaker.

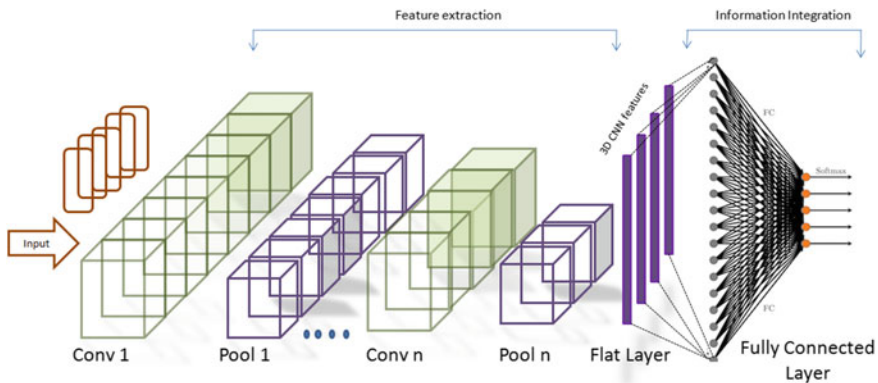


Fig. 5 Architecture of the three-dimensional convolutional neural network used in the experiment

### 3.3 Proposed extreme learning machine-based approach for speaker identification

Extreme learning machines are feed-forward neural networks which avoid time-consuming iterative training process and improve the generalization performance. The ELM randomly sets all the network parameters and easily generates the local optimal solution. In the ELM-based fast learning approach, input layer weights  $W$  and biases  $b$  are set randomly and never adjusted so that the output weights are independent of them (unlike in deep learning networks where back-propagation is used as the training method) and have a direct solution without iteration. Figure 6 depicts the architecture of the extreme learning machine we have used in our experiment.

The detailed description of the ELM architecture we have used is as follows Let  $(x_i, t_i)$  be the set of  $N$  training samples where  $x_i \in R^d$  and  $t_i \in R^c$  and  $L$  being the number of hidden neurons. The output of  $L$  hidden neurons is calculated by Eq. 1,

$$\sum_{j=1}^L \beta_j \phi(w_j x_i + b_j); i \in [[1, N]] \tag{1}$$

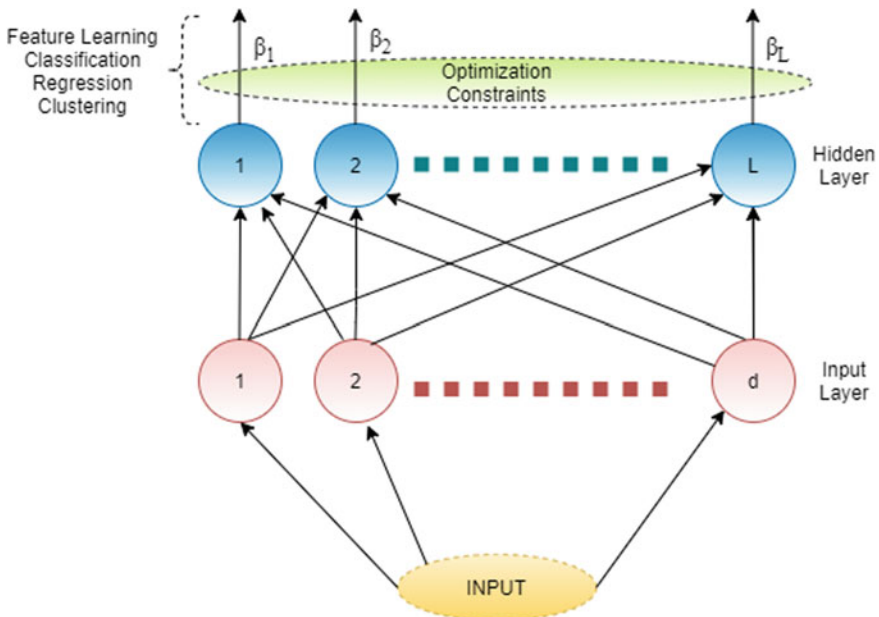


Fig. 6 Architecture of the extreme learning machine used in the experiment

where  $W_i$  = input weights,  $b_i$  = biases and  $\beta_i$  = output weights. The activation function is the sigmoid function, defined by Eq. 2,

$$\phi(x) = \frac{1}{(1 + e^{-x})} \tag{2}$$

Gathering the outputs of all hidden neurons in a matrix H, the matrix form of ELM can be represented as in Eq. 3

$$H = \begin{bmatrix} \phi(w_1, x_1 + b_1) & \dots & \phi(w_L, x_1 + b_L) \\ \dots & \dots & \dots \\ \phi(w_1, x_N + b_1) & \dots & \phi(w_L, x_N + b_L) \end{bmatrix} \tag{3}$$

where  $\beta = (\beta_1^T, \dots, \beta_L^T)$ , input is  $XW$  and output is  $H\beta$   
 $T = (y_1^T, \dots, y_N^T)$  Simply,

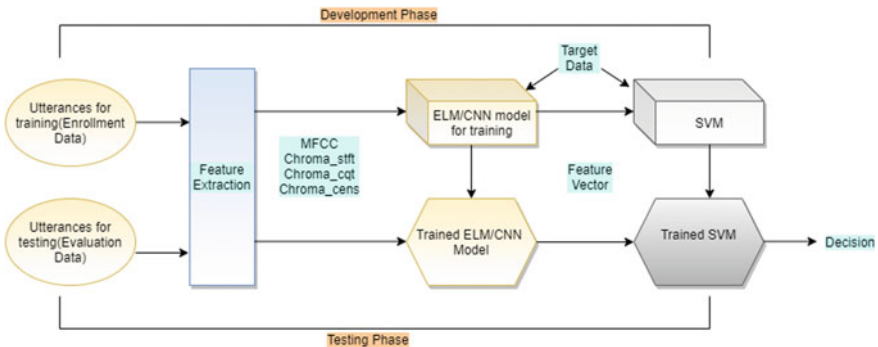
$$H\beta = T \tag{4}$$

Equations 5 and 6 represent solving using the pseudo-inverse method,

$$H\beta = T \tag{5}$$

$$\beta = H \dagger T \tag{6}$$

After training the ELM, the feature vector from the layer just before the final output layer is taken and fed as input to an SVM. The same target data are used to train the SVM. During the testing phase, initially, the MFCC and the chroma features are extracted and fed to the trained ELM and taken the intermediate feature vector. Then, this feature vector is fed to the trained SVM to predict the speaker. The block diagrams of the proposed method are depicted in Fig. 7.



**Fig. 7** Detailed block diagram showing the steps in development and testing phase of speaker identification

## 4 Experimental Evaluation

### 4.1 Dataset Description and Experimental Set up

The datasets we have used to assess our methods are English language speech database for speaker recognition (ELSDSR), THUYG-20-SRE database and LibriSpeech. ELSDSR is spoken by 20 Danes, one Icelander and one Canadian. It consists of a total of 154 utterances (7 from 22 speakers each), for training and 44 utterances (2 from each speaker) for testing [37]. THUYG-20-SRE is an open and free database for Uyghur speaker recognition. The entire database is split into three datasets: The training set involves 4771 utterances spoken by 200 speakers [38]. LibriSpeech is a corpus of English speech consisting of 1000 hours of speech of 16 kHz. It was prepared by Vassil Panayotov with the assistance of Daniel Povey. The data from reading audiobooks from the LibriVox project are segmented and aligned in this dataset [39]. The training set of this dataset is divided into three subsets with 100, 360 and 500 hours, respectively. Table 1 shows a detailed description of the datasets used in our experiment.

The experiments were implemented in Python language using Tensor Flow and Keras libraries on a workstation with Windows 10 Operating System with 32 GB RAM and Intel Xeron CPU E5-1620 v4 @ 3.50 GHz.

The two proposed models have experimented with MFCC vector alone as well as the MFCC combined with the chroma features, using each of the three datasets, and the results are analyzed. In our 3D CNN-based approach, the vectors are resized to four-dimensional tensors and the convolutional layers used 32 filters with a filter size of  $2 \times 2 \times 2$ . We have performed batch normalization, the normalization of layers by adjusting and scaling the activation, which allows each layer of a network to learn by itself a little bit more independently of other layers. After each convolution Layer, we have done max-pooling with a filter size of  $2 \times 2 \times 2$ .

In our ELM-based approach also, the number of hidden neurons is made greater than the number of input neurons. We can select the number of hidden neurons randomly however for all datasets, we have used approximately 15000 hidden neurons for better performance.

**Table 1** Details of the dataset

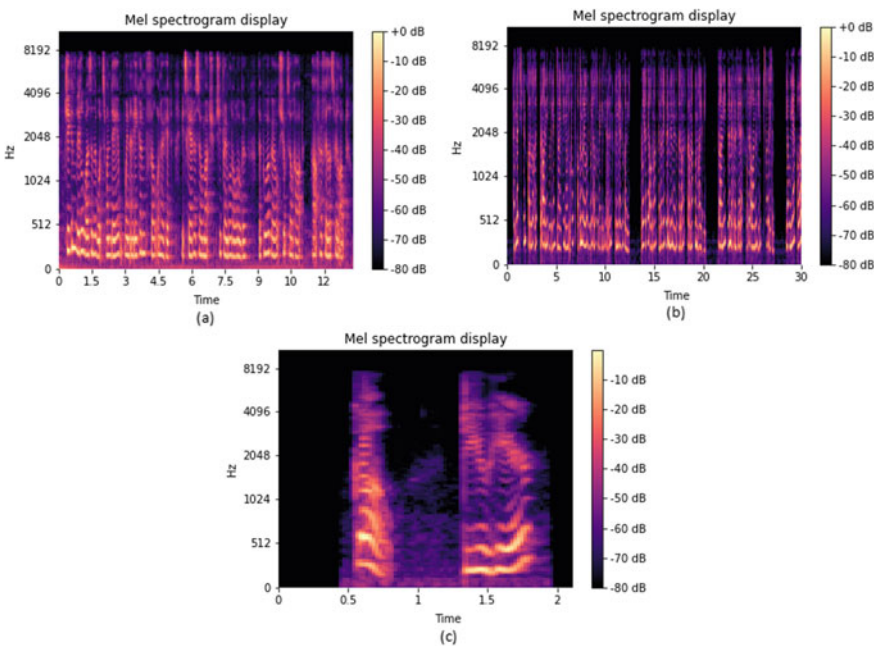
Dataset	Language	Developers
ELSDSR	English	Technical University of Denmark (DTU)
THUYG-20-SRE	Uyghur	CSLT@Tsinghua University and Xinjiang University
LIBRISPEECH	English	Vassil Panayotov, Guoguo Chen, Daniel Povey



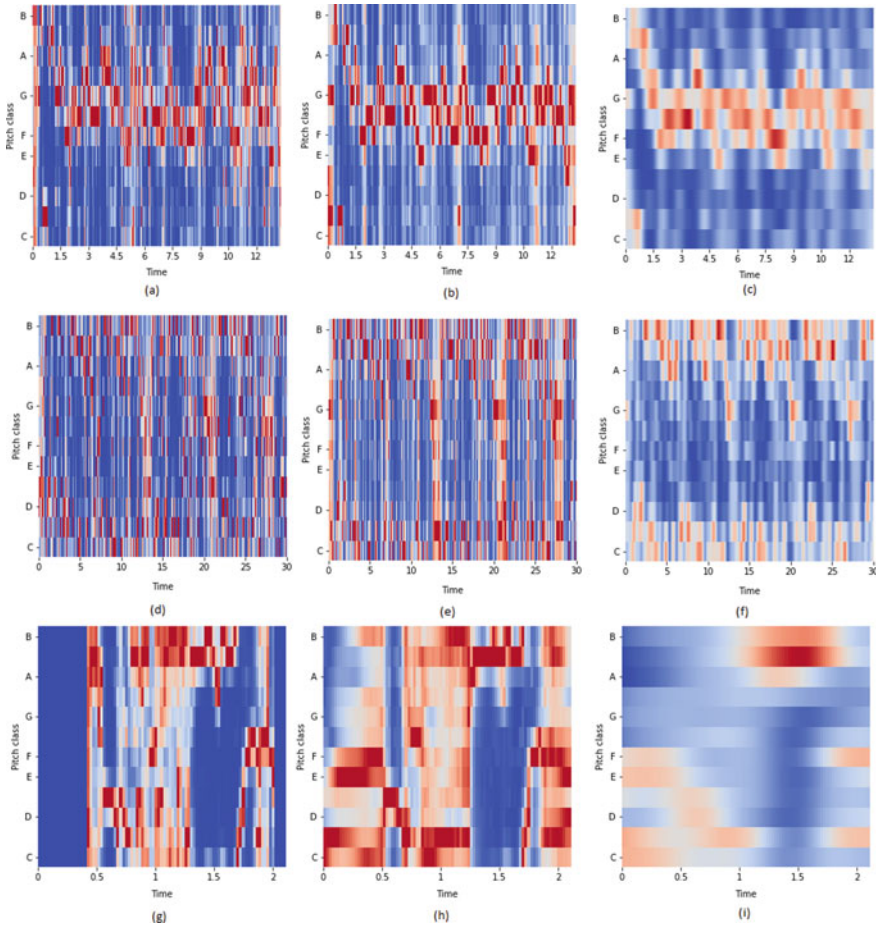
## 5 Results and Discussion

Recognition accuracy is the most important evaluation metric used for speaker recognition techniques, which refers to the percentage of samples correctly recognized by the system. We have repeated our experiments using MFCC in combination with the chroma features and acquired much better results for the three datasets, especially when using 3D CNN because the convolutional neural networks are supposed to work well with image data rather than other less spatially correlated data.

Figure 8 shows the Mel spectrogram display of sample audio clips from each of the three datasets. The chroma features short-time fourier transform (stft), constant q-transform (cqt) and chroma energy normalized statistics (cens) of sample audios from each of the three datasets are displayed in Fig. 9 using matplotlib. Specifically, the chroma vector is a 12 element feature vector which indicates the amount of energy of each pitch class present in the signal, say C, C#, D, D#, E, ..., B. The constant q-transform varies from the fourier transform in the fact that it uses a logarithmically spaced frequency axis. The chroma energy normalized statistics, typically used for audio matching and similarity analysis, takes the statistics over large windows, smooths local deviation in articulation, tempo, trills and arpeggiated chords, etc. Figure 9 shows the plots of chroma features of sample audio clips from each of the three datasets.



**Fig. 8** Mel\_spectrogram display of sample audios from each of the three datasets, plotted using matplotlib. **a** Mel spectrograms of FAML\_Sa.wav **b** F101\_train.wav **c** 84-121123-0000.flac



**Fig. 9** Plots (a), (d) and (g) denote the chroma\_stft, chroma\_cqt, chroma\_cens of FAML\_Sa.wav sample from the ELSDSR dataset. Plots (b), (e) and (h) denote the chroma\_stft, chroma\_cqt, chroma\_cens of F101\_train.wav sample of the THYUG-20 SRE dataset. Plots (c), (f) and (i) denote the chroma\_stft, chroma\_cqt, chroma\_cens of 84-121123-0000.flac sample of the LibriSpeech dataset

The recognition rates and training times while using the two methods with MFCC feature and MFCC combined with the other features are depicted in Tables 2, 3, 4 and 5. Table 2 presents the comparison of recognition rates of the ELSDSR dataset from various speaker identification models. The method specified in [40] acquires a recognition rate of 91.3% which is shown to be surpassed by the proposed methods. The greater recognition rate is achieved while using the 3D CNN with SVM with the MFCC as well as the additional features(stft, cqt and cens). With MFCC alone, the 3D CNN gives 94% recognition rate. The extreme learning machine gives a recognition rate of 93% and 94.79% with MFCC alone and MFCC with additional features,

**Table 2** Comparison of recognition rates of the ELSDSR dataset from various speaker identification models

Speaker identification model	Recognition rate (%)
Hossen et al. [40]	91.3
3D CNN with SVM (MFCC)	94
3D CNN with SVM (MFCC, stft, cqt and cens)	96.7
ELM with SVM (MFCC)	93
ELM with SVM (MFCC, stft, cqt and cens)	94.79

**Table 3** Comparison of training time requirement of the ELSDSR dataset with various speaker identification models

Speaker identification model	Training time (s)
3D CNN with SVM (MFCC)	490
3D CNN with SVM (MFCC, stft, cqt and cens)	503
ELM with SVM (MFCC)	3.5
ELM with SVM (MFCC, stft, cqt and cens)	~4

**Table 4** Comparison of recognition rates and training time requirement of the THYUG-20-SRE dataset with various speaker identification models

Speaker identification model	Recognition rate (%)	Training time
3D CNN with SVM (MFCC)	92.17	2 h
3D CNN with SVM (MFCC, stft, cqt and cens)	94	2.6 h
ELM with SVM (MFCC)	91.75	157 s
ELM with SVM (MFCC, stft, cqt and cens)	~92	~200 s

respectively. Table 3 shows the comparison of training times required for the ELSDSR dataset while using various speaker identification models. It clearly shows that the extreme learning machines give outstanding results compared to the deep learning-based method. The training time required for extreme learning machine limits to a maximum of 4 s, while it takes above 500 s for the deep learning approach.

As we know, using deep learning, it is unlikely to outperform other approaches unless we train it with a huge amount of data. But with the concatenation of SVM, we have acquired about 96.7% accuracy of recognition with such a small dataset of 22 speakers. The obstacle in adopting our deep learning approach in IoT is the training time which may exceed one or two days for large datasets, however, this can successfully be used with small datasets in the case of a smart home/office or other environments equipped with sufficient GPU. To tackle these constraints, we propose our second approach in which we replaced the deep convolutional network with an extreme learning machine for feature learning.

**Table 5** Comparison of recognition rates and training time requirement of the LibriSpeech dataset with various speaker identification models

Speaker identification model	Recognition rate (%)	Training time
3D CNN with SVM (MFCC)	91.56	~1.5 h
3D CNN with SVM (MFCC, stft, cqt and cens)	93.29	~1.27 h
ELM with SVM (MFCC)	92.75	~180 s
ELM with SVM (MFCC, stft, cqt and cens)	~93	~192 s

The comparison of recognition rates and time required for training the THYUG-20 SRE dataset with the proposed models is depicted in Table 4. Similar to the ELSDSR dataset, the greater recognition rate is achieved by the 3D CNN model when MFCC along with the additional features is given. The performance of 3D CNN while using MFCC alone is also appreciable. Coming to the extreme learning machines, while MFCC and the additional features are given, the approximate recognition rate is 92% which is on par with the 3D CNN-based approach. The training time required for the 3D CNN was around 2.6 h, while it takes only 200 s for the ELM. So considering the trade-off between recognition rate and the training time needed, the extreme learning machines are found superior to the 3D CNN-based approach, especially for real-time applications.

Table 5 presents the comparison of the recognition rates and the training time requirements of the LibriSpeech dataset with the proposed approaches. The method presented in [41] gives an accuracy of 86%. The method in [29] uses MFCC and time-based features, and these features are fed to a deep neural network, and the accuracy of speaker identification was up to 89%. Another method in [42] uses MFCCsMap as a feature and a deep learning network as the model for recognition, attaining 90% accuracy. The proposed 3D CNN with MFCC as well as the chroma features gives a recognition rate of 93.29%, while the ELM-based method could give an approximate of 93%.

Regarding the training time, while deep learning takes hours for training, the extreme learning-based approach takes only a few seconds. As we compare the training time while using all the three datasets, we can notice this huge difference. The two key reasons behind the slower learning of the deep neural networks are

- The slow gradient-based learning algorithms used for training
- Iterative tuning of the parameters of the network by the learning algorithms.

Table 6 summarizes the features and model used for recognition in the recent state-of-the-art techniques for speaker identification. It is clear that the Mel frequency cepstral coefficients are the most powerful and increasingly popular feature extraction method for speaker identification. Combining the MFCC with other speech features can result in astounding results when modern classification models are utilized. Researchers have already proven the cogency of various types of deep neural net-

**Table 6** Comparison of features and recognition models used in the recent approaches for speaker identification

Recent approaches	Features	Model for recognition
Bose et al. [43]	MFCC	Gaussian mixture Model
Chen et al. [30]	FBank (Mel Filter Bank Co-efficients) MFCC	ConditionalGenerative Adversarial Network
Hong et al. [44]	Acoustic feature embedding from CNN based UBM, articulatory features from multilayer perceptron	Fully connected Neural Network
An et al. [28]	MFCC	Two representative CNNs-Visual Geometry Group (VGG) and Residual neural networks
Dhakal et al. [36]	Univariate feature selection from statistical features, CNN and Gabor features	Support vector machine Random forest classifier Deep neural network
Borandag [35]	MFCC	Markov chain model
Sekkate et al. [32]	SMFCC (MFCC from each sub band)	i-vector modeling framework
Shahin et al. [31]	MFCC	Cascaded model of Gaussian mixture model & Deep neural network
Tiwari et al. [34]	MFCC and other speech features like energy, pitch and LPC	Gaussian mixture model
<b>Proposed Approach</b>	<b>MFCC Short time Fourier Transform (stft) Constant q-transform (cqt) Chroma energy normalized statistics (cens)</b>	<b>CNN-SVM ELM-SVM</b>

works for efficient classification. Some contemporary works rely upon the coalesce of deep neural network architecture with traditional methods, providing rather more superior systems. The proposed approach combines MFCC with additional features like stft, cqt and cens conducive to attain increased recognition rates. We have used two cascade models, the CNN-SVM and the ELM-SVM model and demonstrated that the ELM-SVM model surpasses the other while a trade-off between recognition rate and training time is considered.

## 6 Conclusion and Future Directions

The paper presents a comparative study of two speaker identification models based on deep learning and extreme learning machines. Also, the paper describes in detail how voice biometrics can be utilized in combating the post-pandemic era. The significance of voice biometrics in the Internet of things scenario is also discussed.

The major barrier in adopting the current state-of-the-art speaker recognition algorithms in the Internet of things is the inability to achieve a better trade-off between the accuracy/recognition rate and time\_space requirement of algorithms. We have proposed two speaker recognition methods based on deep learning and extreme learning machines. We could infer that the extreme learning machines are faster than the deep neural networks by more than 50% and therefore more suitable for IoT authentication. Also, the deep convolutional neural networks are found out to be best suited for chroma features just as how MFCC feature works. Improving the extreme learning machine accuracy by using restricted Boltzmann's machine is in consideration as future work. Furthermore, the voice feature values we obtain from different devices differ for the same speaker itself. Also, the same speaker's voice through the same device differs in different pathological conditions. Considering these facts, the development of a fuzzy fusion system to model the speaker recognition task is a future scope since fuzzy logic can well deal with imprecise data.

**Acknowledgements** This research work was supported by the Kerala State Council for Science, Technology and Environment [KSCSTE/5623/2017-FSHP-ES].

## References

1. Tech 5. *Touchless Biometric Technologies And Innovative Solutions For Covid-19 Management And The Post-Pandemic Era*, 2020. Accessed 1 Oct 2020
2. Tech 5. *The Potential Of Touchless Biometric Technologies And Solutions For Covid-19 Management And The Post-Pandemic Era* *Biometric Technologies And Innovative Solutions For Covid-19 Management And The Post-Pandemic Era*, 2020. Accessed 1 Oct 2020
3. A. Saleema, S.M. Thampi, Voice biometrics: the promising future of authentication in the internet of things, in *Handbook of Research on Cloud and Fog Computing Infrastructures for Data Science* (IGI Global, 2018), pp. 360–389
4. Samsung. *Deep IoT*, 2020. Accessed 1 Jan 2020
5. T. Kinnunen, H. Li, An overview of text-independent speaker recognition: from features to supervectors. *Speech Commun.* **52**
6. M. Espi, M. Fujimoto, Y. Kubo, T. Nakatani, Spectrogram patch based acoustic event detection and classification in speech overlapping conditions, in *2014 4th Joint Workshop on Hands-free Speech Communication and Microphone Arrays (HSCMA)* (IEEE, 2014), pp 117–121
7. O. Vinyals, G. Friedland, Modulation spectrogram features for improved speaker diarization, in *Ninth Annual Conference of the International Speech Communication Association* (2008)
8. C.-L. Huang, J.-C. Wang, B. Ma, Ensemble based speaker recognition using unsupervised data selection. *APSIPA Trans. Sign. Inform. Process* **5** (2016)
9. H.C. Bao, Z.C. Juan, The research of speaker recognition based on GMM and SVM, in *2012 International Conference on System Science and Engineering (ICSSE)* (IEEE, New York, 2012), pp. 373–375

10. M. Ferras, K. Shinoda, S. Furui, Structural map adaptation in GMM-supervector based speaker recognition, in *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (IEEE, New York, 2011), pp. 5432–5435
11. Y. Chen, E. Heimark, D. Gligoroski, Personal threshold in a small scale text-dependent speaker recognition, in *2013 International Symposium on Biometrics and Security Technologies* (IEEE, New York, 2013), pp. 162–170
12. S. Singh, E.G. Rajan, Vector quantization approach for speaker recognition using MFCC and inverted MFCC. *Int. J. Comput. Appl.* **17**(1), 1–7 (2011)
13. D. Handaya, H. Fakhruroja, E.M.I. Hidayat, C. Machbub, Comparison of Indonesian speaker recognition using vector quantization and hidden Markov model for unclear pronunciation problem, in *2016 6th International Conference on System Engineering and Technology (ICSET)* (IEEE, New York, 2016), pp. 39–45
14. F. Richardson, D. Reynolds, N. Dehak, Deep neural network approaches to speaker and language recognition. *IEEE Sign. Process. Lett.* **22**(10), 1671–1675 (2015)
15. O. Ghahabi, J. Hernando, Deep learning backend for single and multisession i-vector speaker recognition. *IEEE/ACM Trans. Audio, Speech, Language Process.* **25**(4), 807–817 (2017)
16. H. Zeinali, H. Sameti, L. Burget, HMM-based phrase-independent i-vector extractor for text-dependent speaker verification. *IEEE/ACM Trans. Audio, Speech, Language Process.* **25**(7), 1421–1435 (2017)
17. N.S. Dey, R. Mohanty, K.L. Chugh, Speech and speaker recognition system using artificial neural networks and hidden Markov model, in *2012 International Conference on Communication Systems and Network Technologies* (IEEE, New York, 2012), pp. 311–315
18. Y. Wang, B. Lawlor, Speaker recognition based on MFCC and BP neural networks, in *2017 28th Irish Signals and Systems Conference (ISSC)* (IEEE, New York, 2017), pp. 1–4
19. S.S. Tirumala, S.R. Shahamiri, A.S. Garhwal, R. Wang, Speaker identification features extraction methods: a systematic review. *Expert Syst. Appl.* **90**, 250–271 (2017)
20. Y. Yujin, Z. Peihua, Z. Qun, Research of speaker recognition based on combination of LPCC and MFCC, in *2010 IEEE International Conference on Intelligent Computing and Intelligent Systems*, vol. 3 (IEEE, New York, 2010), pp. 765–767
21. W.H. Abdulla, Robust speaker modeling using perceptually motivated feature. *Pattern Recogn. Lett.* **28**(11), 1333–1342 (2007)
22. X. Shi, H. Yang, P. Zhou, Robust speaker recognition based on improved GFCC, in *2016 2nd IEEE International Conference on Computer and Communications (ICCC)* (IEEE, New York, 2016), pp. 1927–1931
23. E. Variansi, X. Lei, E. McDermott, I.L. Moreno, J. Gonzalez-Dominguez, Deep neural networks for small footprint text-dependent speaker verification, in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (IEEE, New York, 2014), pp. 4052–4056
24. N. Dehak, P.J. Kenny, R. Dehak, P. Dumouchel, P. Ouellet, Front-end factor analysis for speaker verification. *IEEE Trans. Audio, Speech, Language Process.* **19**(4):788–798 (2010)
25. A. Torfi, J. Dawson, N.M. Nasrabadi, Text-independent speaker verification using 3D convolutional neural networks, in *2018 IEEE International Conference on Multimedia and Expo (ICME)* (IEEE, New York, 2018), pp. 1–6
26. Y. Chen, I. Lopez-Moreno, T.N. Sainath, M. Visontai, R. Alvarez, C. Parada, Locally-connected and convolutional neural networks for small footprint speaker recognition, in *Sixteenth Annual Conference of the International Speech Communication Association* (2015)
27. G. Heigold, I. Moreno, S. Bengio, N. Shazeer, End-to-end text-dependent speaker verification, in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (IEEE, New York, 2016), pp. 5115–5119
28. N.N. An, N.Q. Thanh, Y. Liu, Deep CNNs with self-attention for speaker identification. *IEEE Access* **7**, 85327–85337 (2019)
29. R. Jahangir, Y.W. Teh, N.A. Memon, G. Mujtaba, M. Zareei, U. Ishtiaq, M.Z. Akhtar, I. Ali, Text-independent speaker identification through feature fusion and deep neural network. *IEEE Access* **8**, 32187–32202 (2020)

30. L. Chen, Y. Liu, W. Xiao, Y. Wang, H. Xie, Speakergan: speaker identification with conditional generative adversarial network. *Neurocomputing* **418**, 211–220 (2020)
31. I. Shahin, A.B. Nassif, S. Hamsa, Novel cascaded Gaussian mixture model-deep neural network classifier for speaker identification in emotional talking environments. *Neural Comput. Appl.* **32**(7), 2575–2587 (2020)
32. S. Sekkate, M. Khalil, A. Adib, Speaker identification for OFDM-based aeronautical communication system. *Circuits, Systems, Sign. Process.* **38**(8), 3743–3761 (2019)
33. Y. Yue, Y. Yang, Mobile intelligent terminal speaker identification for real-time monitoring system of sports training. *Evol. Intell.* pp. 1–12 (2020)
34. V. Tiwari, M.F. Hashmi, A. Keskar, N.C. Shivaprakash, Virtual home assistant for voice based controlling and scheduling with short speech speaker identification. *Multimedia Tools Appl.* **79**(7), 5243–5268 (2020)
35. E. Borandağ, Markov model based real time speaker recognition using k-means, fast fourier transform and mel frequency cepstral coefficients. *Celal Bayar Üniversitesi Fen Bilimleri Dergisi* **15**(3), 287–292 (2019)
36. P. Dhakal, P. Damacharla, A.Y. Javaid, V. Devabhaktuni, A near real-time automatic speaker recognition architecture for voice-based user interface. *Mach. Learn. Knowl. Extract.* **1**(1), 504–520 (2019)
37. L. Feng, L.K. Hansen, *A new database for speaker recognition* (IMM, Informatik og Matematisk Modelling, DTU, 2005)
38. Open SLR. *Dataset-Thyug-20 SRE*, 2020. Accessed 15 Sept 2020
39. V. Panayotov, G. Chen, D. Povey, S. Khudanpur, Librispeech: an ASR corpus based on public domain audio books, in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (IEEE, New York, 2015), pp. 5206–5210
40. A. Hossen, S. Al-Rawahi, A text-independent speaker identification system based on the Zak transform. *Sign. Process.: Int. J.* **4**, 68–74 (2010)
41. S. Chakraborty, R. Parekh, An improved approach to open set text-independent speaker identification (OSTI-SI), in *2017 Third International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN)* (IEEE, New York, 2017), pp. 51–56
42. T. Lin, Y. Zhang, Speaker recognition based on long-term acoustic features with analysis sparse representation. *IEEE Access* **7**, 87439–87447 (2019)
43. S. Bose, A. Pal, A. Mukherjee, D. Das, Improved language-independent speaker identification in a non-contemporaneous setup. *Int. J. Mach. Learn. Comput.* **10**(5) (2020)
44. Q.-B. Hong, C.-H. Wu, H.-M. Wang, C.-L. Huang, Combining deep embeddings of acoustic and articulatory features for speaker identification, in *ICASSP 2020–2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (IEEE, New York, 2020), pp. 7589–7593



# Random Permutation-Based Linear Discriminant Analysis for Cancelable Biometric Recognition



P. Punithavathi and S. Geetha

**Abstract** The increased use of biometrics in the present scenario has led to the concerns over security and privacy of the enrolled users. This is because the biometric traits like face, iris, ear, etc., are not cancelable or revocable. In case if the templates are compromised, the imposters may gain illegitimate access. To resolve such issues, a simple yet powerful technique called “random permutation-based linear discriminant analysis” for cancelable biometric recognition has been proposed in this paper. The proposed technique is established on the notion of a cancelable biometric system through which the biometric templates can be revoked and renewed. The proposed technique accepts the cancelable biometric template and a key (called PIN) issued to the user. The user’s identity is recognized only when both cancelable biometric template and PIN are valid, else the user is prohibited. The performance of the proposed technique is demonstrated on the freely available face (ORL), iris (UBIRIS), and ear (IITD) datasets against state-of-the-art methods. The key benefits of the proposed technique are (i) classification accuracy remains unaffected by using random permutation and (ii) robustness across different biometric traits.

**Keywords** Biometric template security · Cancelable biometric recognition · Linear discriminant analysis · Random permutation · Template revocation

## 1 Introduction

Biometrics is a unique attribute possessed by every individual. It can be either physiological (e.g., face, iris, fingerprint, palmprint, etc.) or behavioral (e.g., gait, keystroke

---

P. Punithavathi · S. Geetha (✉)

School of Computer Science and Engineering, Vellore Institute of Technology, Chennai Campus, Chennai, Tamil Nadu, India

e-mail: [geethabaalan@gmail.com](mailto:geethabaalan@gmail.com)

P. Punithavathi

e-mail: [p.punithavathi2015@vit.ac.in](mailto:p.punithavathi2015@vit.ac.in)

dynamics, mouse dynamics, etc.) attribute to identify the user. With rapid technological advancements, biometrics have replaced passwords or personal identification number (PIN) in several access control applications like financial, healthcare, immigration, surveillance, etc. Principal component analysis (PCA) [1] and linear discriminant analysis (LDA) [2, 3] are two popular approaches used in the biometric recognition process. PCA is an unsupervised learning approach which learns eigenfaces from covariance matrix generated by unlabeled training data. PCA preserves the distributions in training data but has no information about the classification. Hence, it fails miserably in pattern recognition (PR) applications. On the other hand, LDA is a supervised learning approach. It generates optimal projection by maximizing the between-class distance and minimizing the within-class distance, thereby LDA is more influential than PCA in PR applications.

During biometric recognition, it is obvious that the dimension of the modality like face, iris, etc., is higher than the number of sample images in the central database. This leads to “Small Sample Size” (3S) problem [4]. The 3S problem can be alleviated by increasing the number of sample images per person or by using a dimension reduction technique. The former one becomes infeasible due to the storage cost, effort, and time to be spent in collecting several sample images, hence selecting a dimension reduction technique is better than increasing the number of sample images per person. Among several linear dimension reduction techniques, random projection (RP) [5, 6] is a feasible approach as it maps a set of points in a high-dimensional space to a new set of points in lower-dimensional space while approximately preserving the pairwise distances between them.

Cancelable biometric system (CBS) [7] is a template securing mechanism which generates cancelable biometric templates out of original biometric attributes for a user-specific key. The cancelable biometric template is generated out of original biometric attributes using a secure and discriminability-preserving transformation function and user-specific key. In case of a database breach, the cancelable biometric templates are alone compromised. Hence, the privacy of the biometric attributes is preserved during attacks. The CBS also provides an added advantage of generating numerous and diverse cancelable biometric templates for various applications just by changing the transformation function and/or user-specific key, thereby preventing privacy threats. In this way, CBS provides high level of security, privacy, and revocability to biometrics that may help to increase public confidence for acceptance of biometric-based systems.

## 2 Literature Survey

CBS is a biometric template securing technique in which enrollment and authentication are performed in the transformed domain. The CBS transformation techniques have been broadly classified into biometric salting and non-invertible transforms as in [8]. Apart from these two techniques, several other categories of CBS have emerged in recent times.

Biometric salting can be defined as a transformation technique which generates cancelable biometric templates by mixing in an artificial pattern. The mixing patterns can be a random/synthetic pattern or a pure random noise. Non-invertible transformation technique can be defined as a cancelable biometric template generation technique which uses a secret key as a constraint for the transformation function. Several other techniques have risen in recent times. For instance, hashing-based transformation [9–11] uses the index positions of the biometric feature template derived from several basic hashing techniques to generate cancelable templates. The Bloom filter-based transformations were introduced in [12]. The Bloom filters were initially introduced by Rathgeb et al. in [13–15] which map multiple code words to identical position to generate cancelable biometric templates.

The following research gaps have been identified from the literature survey.

- The transformation is mostly applied at feature level which becomes time. There exists a research gap for generating cancelable templates by applying the transforms at the signal level
- The 3S problem persists if the dataset contains less number of sample images
- The security, privacy, revocability, and diversity of the biometric information of user must be preserved simultaneously while achieving good recognition rate

This work proposes a novel cancelable biometric recognition technique called *Random Permutation-based Linear Discriminant Analysis (RPLDA) method* that addresses the above issues. The approach aims to generate secure, revocable, non-invertible, privacy-preserving, and performance preserving templates even in the stolen token scenario.

The research contributions of the proposed system have been listed as follows:

- The proposed cancelable biometric template generation system—RPLDA is capable of generating cancelable biometric templates which have been transformed at the signal level
- The 3S problem has been alleviated by employing LDA
- The recognition performance of RPLDA has been proved to be better in the transformed domain
- The RPLDA satisfies the basic properties of CBS such as non-invertibility, diversity, unlinkability, and revocability which are evident from the research outcomes

The rest of the paper is organized as follows. Section 3 explains the proposed RPLDA technique, analysis of the relationship between LDA and RPLDA and its applicability as a cancelable biometric system. Section 4 mentions the experimental setup. Section 5 describes the results. Section 6 gives a brief conclusion.

### 3 Proposed Technique

Intermediate templates are generated by employing a random permutation matrix on the given training images of the biometric traits. The random permutation matrix is chosen to be the PIN. The random permutation matrix is selected to be a matrix whose entries are “0” and “1”, distributed randomly. Through the proposed technique, the cancelable features in the intermediate templates are recognized using LDA [2, 3]. The finally extracted cancelable features are the discriminant vectors which comprise the cancelable template.

#### 3.1 Preliminaries of LDA

LDA is a popular feature extraction technique. The main objective of the LDA is to derive the direction along which the variance in the data is high.

Assume that  $x \in \mathbb{R}^d$  is a column vector and it represents each image in “d” dimensional space. If there are “c” users, i.e.,  $\{1, 2, 3, \dots, c\}$ , such that each user has  $N_i$  images. Thus, the total number of training images is given by Eq. (1)

$$N = \sum_{i=1}^N N_i \quad (1)$$

Let  $\hat{x}$  represent the mean image vector of the training data as shown in Eq. (2)

$$\hat{x} = \frac{1}{N} \sum_{i=1}^N N_i \hat{x}_i \quad (2)$$

where the value of  $\hat{x}_i$  is given by Eq. (3)

$$\hat{x}_i = \frac{1}{N_i} \sum_{i=1}^N x \quad (3)$$

The total scatter matrix ( $M$ ) for the training data is given by Eq. (4) as below:

$$M = M_w + M_b \quad (4)$$

where  $M_w$  and  $M_b$  are within-class scatter matrix and between-class scatter matrix, respectively as given by Eqs. (5) and (6).

$$M_w = \sum_{i=1}^c (x - \hat{x}_i)(x - \hat{x}_i)^t \quad (5)$$

$$M_b = \sum_{i=1}^c N_i (\hat{x}_i - \hat{x})(\hat{x}_i - \hat{x})^t \quad (6)$$

The criterion function ( $J$ ) for a projection matrix  $\psi = \{w_1|w_2|\dots|w_{c-1}\}$  is given by Eq. (7)

$$J(\psi) = \frac{|\psi^t M_b \psi|}{|\psi^t M_w \psi|} \quad (7)$$

For an optimal projection matrix, say  $\psi^*$  the eigenvectors corresponding to the largest eigenvalues and can be modeled as

$$\psi^* = \{w_1^*|w_2^*|\dots|w_{c-1}^*\}.$$

Thus, the transformed data points ( $T$ ) are given by Eq. (8)

$$T = \psi^t x \quad (8)$$

### 3.2 Proposed RPLDA Scheme

A random permutation matrix is projected on the sample image of the biometric traits of the users thereby generating an intermediate template which is a column vector. The random permutation matrix is chosen to be an involutory matrix to convert the given sample image into a column vector. The cancelable templates are generated by extracting the LDA features out of the intermediate template. Thus, the biometric patterns can be renewed or revoked whenever required just by changing the random permutation matrix.

If  $x$  is an input sample, then a random permuted image ( $x'$ ) or an intermediate template can be generated from the input sample  $x$  using a random permutation matrix  $R$ , as represented in Eq. (10).

$$x' = Rx \quad (10)$$

The LDA features of the intermediate template are determined to construct the cancelable template.

The various entities involved in the generation of the cancelable templates using the proposed RPLDA scheme have been illustrated in Fig. 1. The users are enrolled in an application during the enrollment phase. The users are verified during the verification phase. During the enrollment phase, the biometric image sensor captures the image samples of the biometric trait of the user. The random permutation matrix projection unit generates a random permutation matrix which is an involutory matrix.

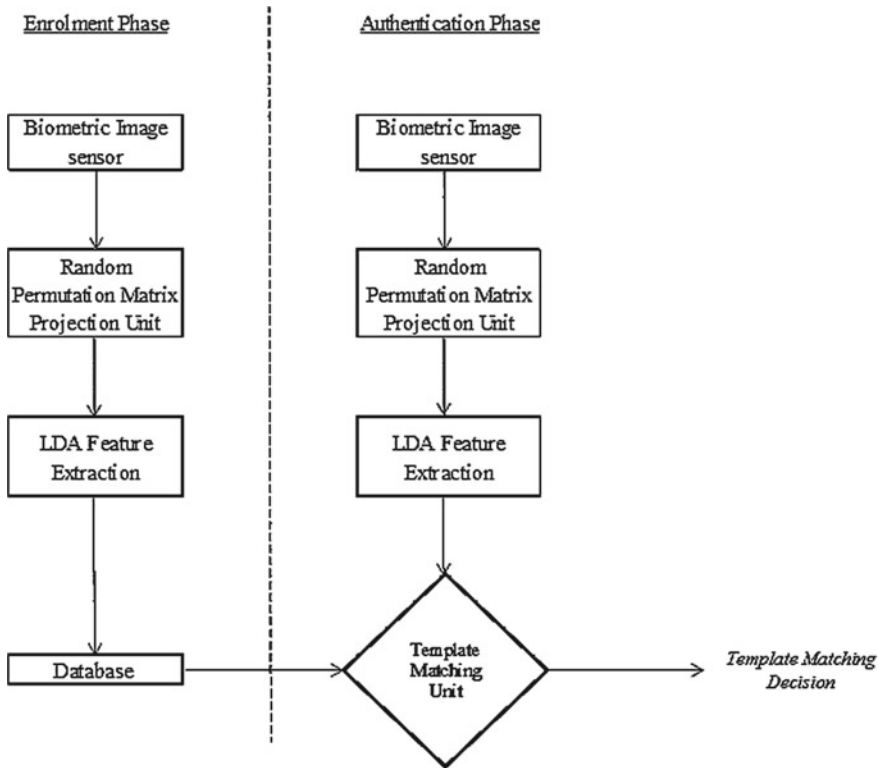
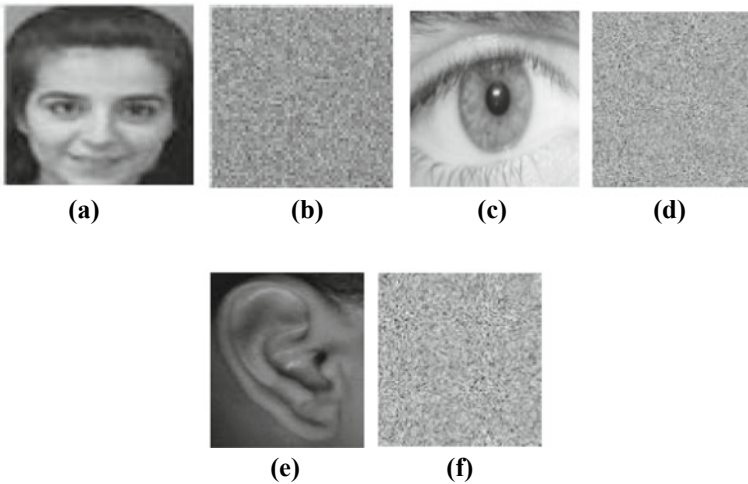


Fig. 1 Enrollment phase and authentication phase in proposed system

This random permutation matrix is projected on the image sample of the user to generate an intermediate template. The LDA feature extraction unit extracts the features from the intermediate template to generate a cancelable template which is stored in the database.

During the authentication phase, the user again produces the biometric trait to the biometric imaging sensor. The random permutation matrix projection uses the same matrix which was generated in the enrollment phase from the user to generate a query intermediate template. The LDA feature extraction unit extracts LDA features from the query intermediate template and generates a query cancelable template. This query template is matched with the cancelable template stored in the database by the matching unit to grant access to the application, in case if the templates match.

A sample face image and its corresponding cancelable template are shown in Fig. 2. The major contribution of the research work is the relationship between eigenvalues and eigenvectors of original biometric images and cancelable templates. The eigenvalues possessed by the original biometric patterns and the cancelable templates are the same. But the eigenvectors of the cancelable template are randomly



**Fig. 2** **a** Sample face image from ORL dataset; **b** cancelable template generated from a sample face image in **(a)** using proposed RPLDA; **c** sample iris image from UBIRIS dataset; **d** cancelable template generated from sample iris image in **(c)** using proposed RPLDA; **e** sample ear image from IITD dataset; **f** cancelable template generated from sample ear image in **(e)** using proposed RPLDA

permuted form of the eigenvectors of the original biometric images. The randomly permuted intermediate templates correspond to the random permutation matrix.

### 4 Experimental Setup

The cancelable template has been generated using proposed RPLDA technique using the iris, face and ear biometrics are chosen from publicly available datasets like UBIRIS [16], ORL [17], and IITD [18] databases, respectively. Table 1 displays the details of the databases.

The performance of the proposed techniques is evaluated in terms of the security provided by the proposed technique, average classification accuracy, and average training time of the algorithm. The training image from each identity is selected randomly and the remaining images are used as testing set. This process is repeated 40 times to achieve a stable classification accuracy and average training time. All

**Table 1** Summary of datasets used in experiments

Dataset	No. of subjects	Image size	Total
ORL	40	112 × 92	400
UBIRIS	241	150 × 200	1877
IITD	125	180 × 50	493

the experiments are performed on Intel Xeon E3 CPU 2.4 GHz with Windows 7 and 8 GB memory.

## 5 Results and Discussions

In this section, the performance of the proposed technique has been analysed both qualitatively and quantitatively with other state-of-the-art methods, viz. Gray-Salt (GS) PCA, Block-Remapping (BR) PCA, and RPPCA [19]. The cancelable templates were generated independently using each technique. The classification accuracy for each technique has been determined using nearest neighborhood technique. Then the classification accuracies have been compared with the classification accuracy of the proposed RPLDA technique.

The classification accuracy measures the percentage of identities correctly classified by some technique or algorithm. It depends on factors such as the number of training used and the number of dimensions in the transformed representation. The classification accuracy of the proposed RPLDA technique has been reported in Table 2. The classification accuracy of the proposed technique is higher when compared to the state-of-art techniques like GSPCA, BRPCA and RPPCA as shown in Table 2.

The equal error ratio (EER) is the metric which is used to measure the matching performance of a CBS. The EER value must be as low as possible to indicate that CBS has a good matching performance. The EER value of the proposed system is calculated and compared with the EER values of the RPPCA, GSPCA, and BRPCA. The EER comparative results have been listed in Table 3. It can be inferred from Table 3 that the EER of the proposed system is lower than the other state-of-the-art techniques,

**Table 2** Classification accuracy achieved by proposed RPLDA technique

Techniques	Classification accuracy (%)		
	ORL	UBIRIS	IITD
RPPCA	93.84	92.98	90.24
GSPCA	87.68	85.35	83.66
BRPCA	86.99	84.96	83.13
Proposed RPLDA	95.33	94.89	92.67

**Table 3** EER (%) achieved by the proposed RPLDA technique

Techniques	EER (%)		
	ORL	UBIRIS	IITD
RPPCA	6.72	6.28	8.33
GSPCA	11.43	14.13	15.39
BRPCA	12.38	17.38	14.36
Proposed RPLDA	4.21	5.43	6.52



**Table 4** Training time (in seconds) required by the proposed RPLDA technique

Dataset	No. of training images			
	2	3	4	5
	Training time (in seconds)			
ORL	0.165	0.269	0.612	1.478
UBIRIS	5.111	6.547	9.124	12.96
IITD	0.412	0.613	0.948	1.631

The average training time required by the proposed techniques is shown in Table 4. The RPLDA has been trained using only few training images of each user to alleviate 3S problem [20]. The average training time over 40 runs of the proposed algorithm was measured on all the datasets as shown in Table 4. It is observed that the training time of the proposed technique increases with an increase in the number of training images. It can be readily observed from Table 4 that the training time required by RPLDA is significantly less and feasible.

The cancelable templates generated using the proposed system are non-invertible. If a brute-force attack has been simulated against the cancelable templates generated using the proposed system, then the number of iterations required to reverse-engineer the cancelable templates is completely dependent on the input image size. It will take  $(112 \times 92)!$ ,  $(150 \times 200)!$ , and  $(180 \times 50)!$  for achieving a preimage of the original biometric image. It is very difficult to reverse-engineer the cancelable templates generated using the proposed system. Hence, the cancelable templates generated using the proposed system are non-invertible.

The cancelable templates generated using the proposed system are diverse. A user can register to different applications using different cancelable templates which correspond to a single biometric pattern. This is achieved just by changing the random permutation matrix. With the change in the pattern of the entries in the random permutation matrix, the cancelable templates also change. Assume if there are “ $n$ ” entries in a random permutation matrix, then “ $n$ ” different cancelable templates can be generated from the input biometric pattern.

## 6 Conclusion

In this research work, a simple yet powerful technique called as random permutation-based linear discriminant analysis (RPLDA) has been proposed. The proposed technique is applicable in cancelable biometric recognition. The users are required to provide their biometric trait and the PIN which is the random permutation matrix. The random permutation matrix is projected on the biometric image and an intermediate template is generated. The LDA extracts cancelable features from the intermediate template and generates the cancelable template. The effectiveness of the proposed technique has been illustrated by the results of the experiments conducted

on different datasets. The classification accuracy of the proposed technique is found to be better than the state-of-the-art techniques. The training time is also found to be very less. The proposed technique is also proved to be effective against 3S problem. The future study can be directed toward application of diagonal LDA for cancelable biometric recognition.

**Acknowledgements** The authors would like to thank the Management and Staff of Vellore Institute of Technology, Chennai Campus. The first author is supported by Visvesvaraya Ph.D. Scheme, sponsored by Digital India Corporation, held by the Ministry of Electronics and Information Technology (MeitY), Government of India.

## References

1. S. Wold, K. Esbensen, P. Geladi, Principal component analysis. *Chemometrics Intell. Lab. Syst.* **2**(1–3), 37–52 (1987)
2. K. Fukunaga, *Introduction to Statistical Pattern Recognition* (Academic, New York, 2013)
3. R.O. Duda, P.E. Hart, D.G. Stork, *Pattern Classification*, vol. 2 (Wiley, New York, 1973)
4. W.J. Krzanowski, P. Jonathan, W.V. McCarthy, M.R. Thomas, Discriminant analysis with singular covariance matrices: methods and applications to spectroscopic data. *Appl. Statistics* **101**–115 (1995)
5. E. Bingham, H. Mannila, Random projection in dimensionality reduction: applications to image and text data, in *Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2001)
6. P. Punithavathi, S. Geetha, Dynamic sectored random projection for cancelable Iris template, in *2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI)* (IEEE, 2016), pp. 711–715
7. N.K.C.J.H. Ratha, R.M. Bolle, Enhancing security and privacy in biometrics-based authentication systems. *IBM Syst. J.* **40**(3), 614–634 (2001)
8. H. Kaur, P. Khanna, Biometric template protection using cancelable biometrics and visual cryptography. *Multimedia Tools Appl.* **75**(23), 16333–16361 (2016)
9. A.B.J. Teoh, D.N. C. Ling, A. Goh, Biohashing: two factor authentication featuring fingerprint data and tokenised random number. *Pattern Recogn.* **37**, 2245–2255 (2004)
10. A.B.J.C.S. Teoh, J. Kim, Random permutation Maxout transform for cancellable facial template protection. *Multimedia Tools Appl.* **77**, 1–27 (2018)
11. H. Li, J. Qiu, A.B.J. Teoh, Palmprint template protection scheme based on randomized cuckoo hashing and MinHash. *Multimedia Tools Appl.* **1**(1), 1–25 (2020)
12. J. Bringer, C. Morel, C. Rathgeb, Security analysis and improvement of some biometric protected templates based on Bloom filters. *Image Vis. Comput.* **58**, 239–253 (2017)
13. C. Busch, C. Rathgeb, F. Breiting, H. Baier, in *On the application of Bloom filters to Iris biometrics*. *IET Biometrics* (2014)
14. S. Ajish, K.S. AnilKumar, Iris template protection using double bloom filter based feature transformation. *Comput. Secur.* **97**, 101985 (2020)
15. M. Gomez-Barrero, C. Rathgeb, J. Galbally, J. Fierrez, C. Busch, Protected facial biometric templates based on local gabor patterns and adaptive bloom filters, in *2014 22nd International Conference on Pattern Recognition (ICPR)* (2014)
16. H. Proenca, L. Alexandre, UBIRIS: a noisy iris image database, in *13th International Conference on Image Analysis and Processing (ICIAP 2005)* (2005)
17. F. Samaria, A. Harter, Parameterisation of a stochastic model for human face identification, in *Proceedings of 2nd IEEE Workshop on Applications of Computer Vision*, Sarasota, FL (1994)

18. A. Kumar, C. Wu, Automated human identification using ear imaging. *Pattern Recogn.* **41**(5), 956–968 (2012)
19. N. Kumar, S. Singh, A. Kumar, Random permutation principal component analysis for cancelable biometric recognition. *Appl. Intell.* **48**(9), 2824–2836 (2018)
20. R. Bellman, *Adaptive Control Processes* (Princeton, Princeton University Press, 1961).

# A Deep Learning-Based Framework for Distributed Denial-of-Service Attacks Detection in Cloud Environment



Amit V. Kachavimath and D. G. Narayan

**Abstract** The widespread cyberattack is distributed denial of service (DDoS). The unauthorized users will target the specific server or network infrastructure by flooding with malicious Internet traffic by creating the interruption in the normal traffic. The victim server will not be able to respond to legitimate traffic. The DDoS attacks recognition in real time is one of the challenging problems. The predictable solutions analyze the traffic and detect the different types of activities from captured traffic based on attributes of statistical differences. The alternate approach for identifying the performance of DDoS attacks is the analysis of the statistical features using machine learning algorithms. These detection techniques have a low detection rate and time delay. The new approach for the DDoS attack detection was proposed by capturing different patterns of sequences from the captured traffic and analysis of the high-level features using deep learning and can be used with a high detection rate. The results of the proposed methodology have demonstrated the better performance of long short-term memory (LSTM) approach with good accuracy compared to the convolutional neural network (CNN) and multilayer perceptron (MLP).

**Keywords** Distributed denial of service · Deep learning · Long short-term memory

## 1 Introduction

Cloud computing is storage, management, computing, and networking with the advent of the Internet. Cloud computing is obtaining higher momentum because

---

A. V. Kachavimath (✉)

Department of Master of Computer Applications, KLE Technological University, Hubli, Karnataka, India

e-mail: [amitk@kletech.ac.in](mailto:amitk@kletech.ac.in)

D. G. Narayan

School of Computer Science & Engineering, KLE Technological University, Hubli, Karnataka, India

e-mail: [narayan\\_dg@kletech.ac.in](mailto:narayan_dg@kletech.ac.in)

of the growing technology trends and rapid business migration in computing infrastructure. Google, Amazon, and Microsoft servers are being distributed that are flexible and secured geographically located all over the world. The DDoS attacks will consume the cloud resources like bandwidth and computation capacity and cause destruction for an entire cloud application with a short span of time.

The DDoS attacks are an illegal attempt to interrupt the normal flow of traffic for the network infrastructure like host, virtual machine, and servers by flooding the target victims with annoying Internet traffic. The exploited systems can be the combination of different network components like Internet of things (IoT) devices, switch, router, host, and controller in a software-defined network [1]. The traffic arriving from these exploited devices will interrupt legitimate traffic and will not be able to provide the service for the authenticated service request.

The Web applications and network services are flooded with network traffic from unauthorized requests. The cloud is a pool of resources that are shared among the browser, host, server, and network level that leads to the open eye for DDoS attacks [2]. The two different objectives for DDoS attacks are to reduce the performance of the server and hide the identification of the attackers—the highlight for the requirement of a complete distributed and cooperative defense approach by using a machine learning approach. The different machine learning algorithms like support vector machine, random forest, Naïve Bayes, and decision tree are used for the classification of DDoS attacks [3]. These algorithms are used to parse data, learn from it, and make informative decisions of the prior knowledge acquired by it. The machine learning approach has limited capability for tuning of hyperparameter.

Deep learning will enable a machine to effectively analyze issues with the architecture of hidden layers that are composite of being programmed manually. The deep learning approach gets a superior hand compared to machine learning while handling the colossal volumes of unstructured data. The tuning of hyperparameter can be done effectively compared to machine learning [4]. The DDoS attacks are increased by 500% from unauthorized requests of 26Gbps, by the reference of the Q2 2018 report of the Nexus guard's. It is affecting the organization's performance financially and also disrupts the authenticated requests [5]. The identification of the anonymous traffic is challenging as the traffic is geographically distributed from different locations, and if the IP is spoofed of the attacking device, then tracking the geographic location is difficult. The detection of the attacks can be done by collecting the network traffic and analyzing the statistics of the attributes [6]. The challenging problem in network security is the mitigation of DDoS attacks.

The challenge was addressed by implementing the DDoS attacks detection system using the deep learning approach. The proposed framework consists of three modules: the first module is data preprocessing, the second module is the detection of the attacks using deep learning classification model, and the last module is the performance evaluation. The framework was implemented using a benchmark dataset. The first module is the priority processing of the dataset, data cleaning by filtering, and modifying the data that makes the exploration of the data convincing and selection of the specific features for DDoS attacks detection with higher priority—the three

different approaches of deep learning MLP, LSTM, and CNN. The performance of the implemented model is measured using the accuracy and confusion matrix.

The remaining content of paper is devised as follows: Sect. 2, an overview for the earlier research work. Section 3 consists of an overview of the implementation of the detection of DDoS attacks. Section 4 consists of the experimental results and in Sect. 5, conclusion and future scope.

## 2 Related Work

The DDoS attacks are being identified as the second most attacks of cybercrime compared to the stealing of confidential data. It will consume cloud resources like bandwidth and CPU utilization and reduce the overall performance of the cloud. The detection and mitigation of such attacks in cloud computing must be of high priority for maintaining eHealth of the cloud. Sahi et al. [7] have proposed a new classifier system for the prevention of the TCP flooding for the public cloud, and the proposed system offers classification of the incoming network traffic and classification of the attacks using support vector machine (SVM). The wide range usage for the standard IEEE 802.11 has been one of the acting key solutions for supporting security threats against TCP flooding. Ruswin et al. [8] denoted that anomaly detection using the decision tree in the network, and KDD Cup'99 benchmark dataset is used to perform the experiment, and they also exploit integration of good pattern recognition of classification proficiencies using the random forest prand J48. The real-time detection of TCP-based attacks that extracts the effective features for the TCP traffic and classifies the malicious traffic from legitimate by using the two different decision classifiers. The proposed approach is implemented using the simulated ISCX IDS, CAIDA 2007, and real-time dataset from Baidu cloud platform. The experiments have shown a better detection rate for the attacks and lower alarm rate [9].

The proposed approach is implemented using the simulated ISCX IDS, CAIDA 2007, and real-time dataset from Baidu cloud platform. The experiments have shown a better detection rate for the attacks and lower alarm rate. Zekri et al. [10] have proposed the DDoS attacks detection system using C4.5 algorithm, and it is integrated with signature-based identification that generates the decision tree for performance evaluation. The experiment was carried out in Open Stack Juno for constructing the public, private, and hybrid cloud, and the detection rate was 98%. The seminal contributions have been used with the approach of data mining for the identification of DDoS attacks; threshold-based value follows the relearning of the algorithm that increases the value for classification of traffic accuracy. The proposed approach consists of two steps collection of network traffic using NetFlow protocol and data mining algorithm for the analysis of new traffic. The collected network traffic is based on the threshold value of data mining and segregates legitimate and malicious traffic [11]. The low rate of DDoS attacks will exploit the vulnerability for control of TCP congestion by flooding the illegitimate traffic at a lower constant rate and decrease the performance of the victim machine. The higher and lower threshold value will

be able to efficiently filter around 79% of the network traffic with a better detection rate [12].

The identification of the attacks at the early stage is one of the emerging challenges of cybersecurity. The analysis of communication between the compromised bots and the legitimate server is one of the key features in the detection of attacks. The support vector machine and principal component analysis (PCA) are used for the extraction of features and random forest for building the classification model [13]. The detection of DDoS attacks in the application layer is a complex concern for Web security; many of the authors have proposed the different techniques for the detection of the DDoS attacks in the network and transport layer compared to the application layer. In this approach, deep learning is introduced for analyzing the features of application-layer attacks. The proposed deep learning framework consists of more than three layers; an autoencoder is used. The performance of the proposed approach has been represented by a better detection rate [14]. Kim et al. [15] have represented the different approaches for intrusion detection systems (IDS) using artificial intelligence. The experiments were performed using the benchmark dataset KDD Cup 99. The four hidden layers, hundred hidden units, ReLU activation, and Adam optimizer, were used to perform the experiments. The series of recent studies have indicated that early detection and the isolation policy will help in the identification of legitimate clients at the early stage.

### 3 Proposed Methodology

In this section, we give a summary of deep learning algorithms that are used for implementation. The key features and statistical analysis of the dataset are represented, different steps for data preprocessing, the framework of deep learning, and evaluation metrics for the framework of DDoS attack detection.

#### 3.1 Introduction to Deep Learning Models

Artificial neural network (ANN) is an efficient computing system for the analogy of biological neural networks. ANN captures a huge amount of information that is interconnected with the specific pattern to communicate among multiple neurons and hidden layers. The nodes are the processors operating in parallel to provide the communication between the input and output layer. The neurons are interconnected with the link that is associated with a weight that contains information of the input signal. The nodes in the neural network have an internal state that includes the knowledge of the activation signal. The output signals are generated after the combination of input signals and activation function result being communicated to other nodes.

### 3.1.1 Long Short-Term Memory

LSTM is the specific type of recurrent neural network (RNN) used for the sequence predictions of different complex problems that are capable of learning with long-term dependency. The LSTM is popularized by many people in deep learning that they work tremendously on the various varieties of problems. Most of the layers in the neural network will have multiple chain repetitive modules that contain the structure of the tanh layer. The looping arrows of the LSTM cell define the recursive feature of the cell, and parameters of the LSTM are also called as cell state. The recursive feature allows storing the information of the previous intervals. The LSTM will store the information from the previous interval and will proceed with the next state. The state of the cell is being upgraded by using forget gate that is being placed below the cell state by the modulation of the input gate. The equation specified of the cell state will forget by multiplying using forget gate and will add new information through the output of the input gates.

The prime feature of LSTM is a memory cell, also called as cell state; the gates are components in LSTM, which are used for synchronizing by adding or eliminating information. LSTM contains a neural layer of sigmoid and multiplication operation among vectors of input data. The LSTM architecture is as shown in Fig. 1. The input time for the step ( $X_t$ ), hidden state by using the time step of previous data ( $S_{t-1}$ ). Hidden state ( $S_t$ ) is calculated as follows (Fig. 2):

Forget gate ( $f_t$ ):

$$f_t = \sigma(X_t U_f + S_{t-1} W_f + b_f) \tag{1}$$

The values of the layer input gate ( $i_t$ ) decides that it needs to be simplified in a second way. The representation of the two layers using tanh is given by the following steps

$$i_t = \sigma(X_t U_i + S_{t-1} W_i + b_i) \tag{2}$$

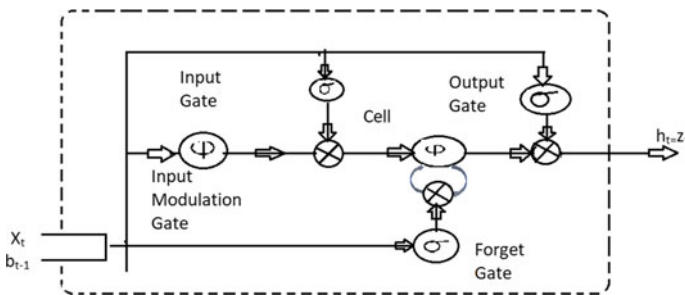
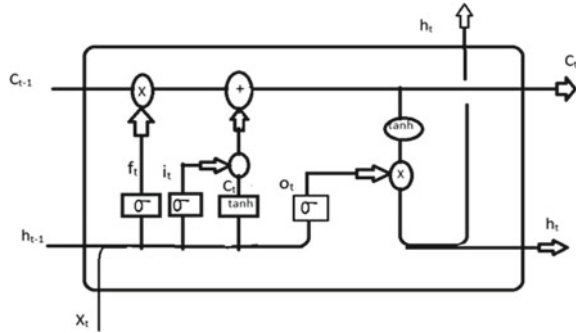


Fig. 1 Architecture of LSTM



Fig. 2 Forget gate equation



$$C_t = \tanh(X_t U_c + S_{t-1} W_c + b_c) \tag{3}$$

The previous state of the  $C_{t-1}$  is reduced to the new state  $C_t$  that is represented by:

$$C_t = C_{t-1} \otimes f_t \oplus i_t \otimes \tilde{C}_t \tag{4}$$

The value of the output gate ( $o_t$ ) is represented by:

$$o_t = \sigma(X_t U_o + S_{t-1} W_o + b_o) \tag{5}$$

$$S_t = o_t \otimes \tanh(C_t) \tag{6}$$

The forget gate is also called a remember vector. The output obtained for the forget gate will provide information for the cell state by multiplying the value 0 for the position of a given matrix [16].

The output for the forget gate will act as input for the next stage that updates the information to the cell state by multiplication of the value zero in a given matrix; after processing, the production is one for the forget gate, and information is stored in the cell state. The sigmoid activation function is being applied with the hidden state and weighted observation. The input gate is also called a save vector. The sigmoid function is used in the input gate to obtain the range [0, 1]. The sigmoid function will be able to add the memory.

### 3.1.2 Convolution Neural Networks (CNNs)

CNN is regularized version of a multilayer perceptron. CNN is developed based on the working of the animal visual cortex neurons. CNN is the composition of the multiple layers that represents the complex attributes of the data. The applications of CNN are autonomous vehicles, robotic applications, image processing, and intrusion detection system. The convolution is the integration of multiple functions that shows

how the input of one function will modify the values of the other function. The three features of this process are data input, feature reduction, and feature map. The feature detector is referred to as a kernel. The input data is multiplied with elements for the matrix representation of the given input data to perform feature reduction. The activation map is used to increase the throughput by reducing the input features. The features are the unique characters used to identify that specific object.

### 3.1.3 Multilayer Perceptron (MLP)

The multilayer perceptron contains the class of neural network that represents at least three nodes for I/O processing. The MLP is the combination of the input layer, multiple hidden layers, and the output layer, the hidden and output layer uses a nonlinear activation function. The hidden layer acts as an interface between the input and output layers. The MLP uses back propagation for the training of the data and nonlinear activation function for neurons to model the behavior like the human brain. The backpropagation consists of two important steps forward pass and backward pass. The forward pass is the evaluation of expected output respective of the given input. The backward pass is the process of the partial derivative of different features that are propagated back to the network.

## 3.2 Framework for Detection of DDoS Attacks

The framework for recognition of DDoS attacks is represented in Fig. 3, with the three different deep learning algorithms. The framework comprises of three important steps: preprocessing of data, deep learning approach, and performance metrics evaluation of the proposed algorithms.

The proposed framework includes the processing of the input data from the benchmark dataset and real-time dataset captured from the network traffic. The attributes of the benchmark dataset are being refined. The feature selection of the attributes is made by using the rank correlation approach. The benchmark dataset contains 41 features, and after applying rank correlation, eight essential features that are highly

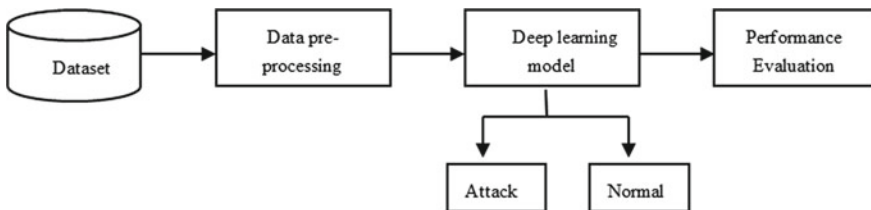


Fig. 3 Proposed framework for the recognition of DDoS attacks

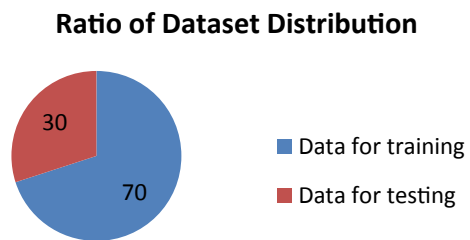
correlated are being extracted. The extracted features are count, destination bytes, protocol type, same srv rate, service, src bytes, srv count, and label.

### 3.3 Overview of Dataset

The benchmark dataset contains 41 features in it, and it is classified into four different types based upon their behavior of attacks. The root to local attack is launched to gain unauthorized access of any victim machine in the network infrastructure, user to root attacks in which the attacker gains the different access rights from a normal instance to gain the access of the root system and probe attack is the new threat for collaborative in the intrusion detection system. The data analysis of network traffic is performed by fetching the different attributes from the network infrastructure. The attacks of the label are categorized into two types 0 and 1. The label 0 indicates normal behavior, and 1 represents an attack. The distribution of the dataset is represented in Fig. 4. 70% of the data is used for training the algorithm, and 30% is used for testing. Table 1 represents the statistical distribution of the train and test dataset.

The rank correlation is used for feature extraction, and among 41 features, eight are being extracted. The dataset of eight features count, srv\_count, destination\_bytes, src\_bytes, protocol\_type,same\_srv\_rate, service and label is being filtered by the elimination of nan values. The rank correlation is the robust measure used to measure the linear association of multiple variables. The properties are robust with outliers and being immutable under the monotonic incremental transformations of data. The empirical values of data are used to estimate the relationship among variables. The coefficient of the rank correlation is in the interval between  $-1$  and  $1$ . The higher value represents a better correlation among variables.

**Fig. 4** Dataset distribution for train and test data



**Table 1** Statistical representation of the dataset

Sl. no.	Benchmark dataset	Total no. of samples	No. of features	Total no. of training samples	Total no. of testing samples
1	KDD Cup 99	494,021	41	345,815	148,206
2	NSL-KDD	29,176	41	20,424	8752

## 4 Experimental Results

The benchmark datasets and real-time network traffic captured from the private cloud deployed with configuration of 6 servers of 128 cores, 512 GB RAM and 30 TB of virtualized storage from multiple virtual machines has been classified into two parts, namely train and test dataset. The low orbit ion cannon (LOIC) tool is used to simulate the DDoS flooding attacks in a real-time environment. The confusion matrix is the tabular representation for the classification model evaluation with parameters true positive (TP), true negative (TN), false positive (FP), and false negative (FN). It provides an analysis of correct and incorrect predictions. We also compute the recall, accuracy, precision, and F-measure using the following equations [17],

$$\text{Acc} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (7)$$

$$\text{Rec} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (8)$$

$$\text{Prec} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (9)$$

$$\text{F - meas} = \frac{2 * \text{Rec} * \text{Prec}}{\text{Rec} + \text{Prec}} \quad (10)$$

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}} \quad (11)$$

The proposed methodology for the DDoS attacks detection is evaluated by carrying out the experiment with the benchmark and real-time dataset. The performance of the proposed method is evaluated using the accuracy of the metric, F-measure, precision, and recall. The accuracy obtained for the LSTM is 93.29% for NSL-KDD, 95.66% for real time and 99.09% for KDD Cup 99 dataset. The accuracy is used to represent the rate of correct classification of the entire data over the incorrect classification results. The performance of LSTM is evaluated by comparing it with CNN and MLP. The false positive rate (FPR) for KDD Cup 99 is 0.0196, NSL KDD is 0.037, and real time is 0.029 using LSTM. The recall, F-measure, and precision of LSTM are better compared to CNN and MLP (Figs. 5 and 6).

The visualization of the performance for the proposed deep learning approach will provide a better sense of data that is poured into the model and make the informed decision for the changes that need to be made in hyper parameters. The benefits of visualization are to evaluate the underfitting or overfitting of the curve and updating the hyperparameters to increase the detection rate.

The graph of model accuracy and loss with NSL KDD is represented in Figs. 7 and 8, KDD Cup 99 in Figs. 9 and 10 for the benchmark and real-time dataset in Figs. 11 and 12.

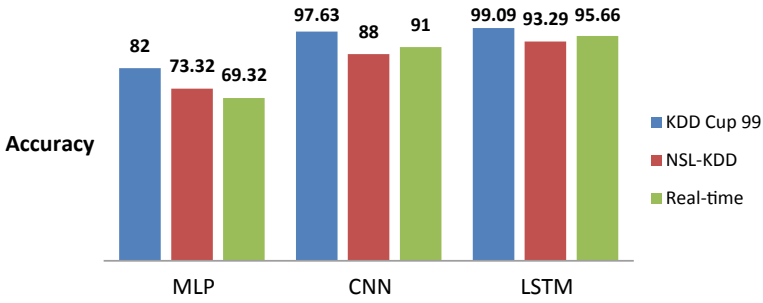


Fig. 5 Comparison of accuracy of deep learning algorithms

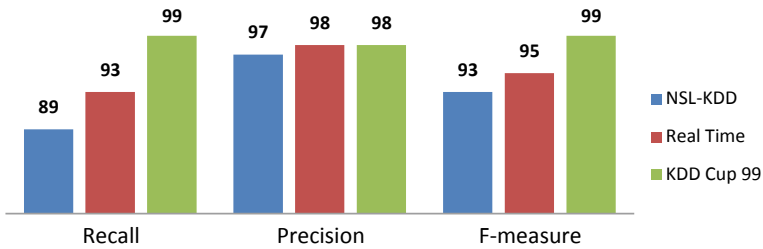
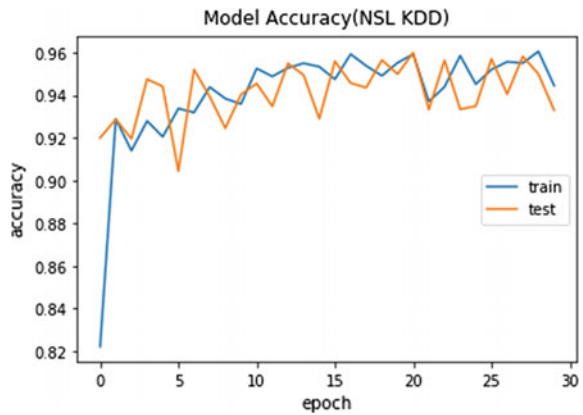


Fig. 6 Comparison of recall, F-measure, and precision of LSTM

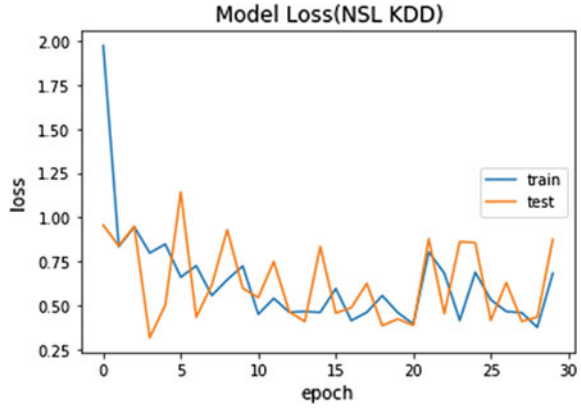
Fig. 7 Model accuracy of NSL KDD



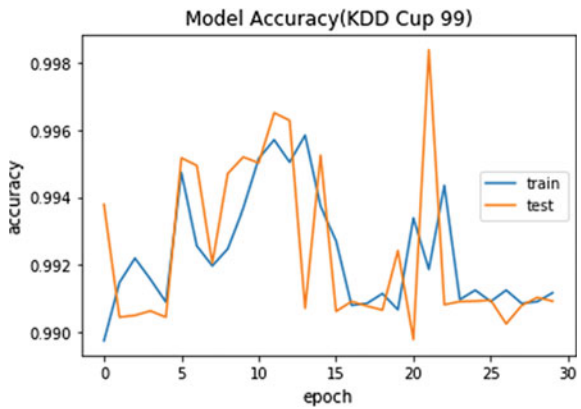
## 5 Conclusion and Future Scope

In this paper, we propose a novel mechanism for handling DDoS attacks detection in cloud environment. The proposed mechanism safeguards the cloud infrastructure that is being flooded with unsolicited traffic and provides a better quality of assurance in service for different cloud consumers—the detection of flooding attacks in the

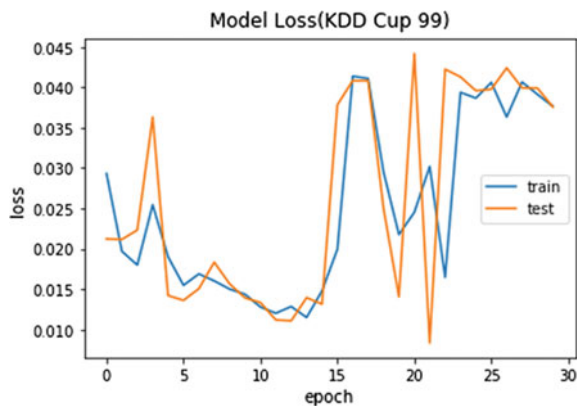
**Fig. 8** Model loss of NSL KDD



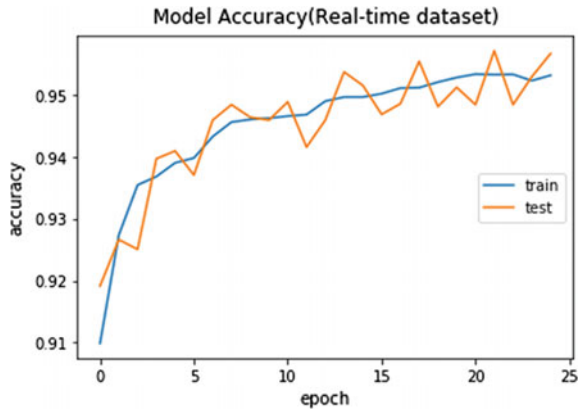
**Fig. 9** Model accuracy of KDD Cup 99



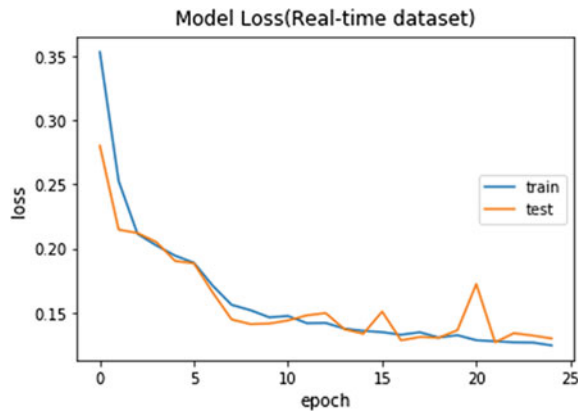
**Fig. 10** Model loss of KDD Cup 99



**Fig. 11** Model accuracy of real time



**Fig. 12** Model loss of real time



private cloud using the machine and deep learning approach. The experimentation was carried out using an open stack operating system [18]. The detection of DDoS attacks uses a software-defined network (SDN) using two levels of security. The first level detection of signature-based attacks using Snort, and the second level using machine learning and deep neural network to detect anomaly-based attacks [19]. The new approach of rank correlation for feature selection and LSTM for classification of malicious and legitimate traffic. The LSTM was compared with the two other deep learning models CNN and MLP. The performance and accuracy of LSTM are better compared to the other two models. The accuracy of 99.09% with KDD Cup 99', 93.29% with NSL-KDD, and 95.66% with the real-time dataset is obtained. The performance is better compared with the state-of-the-art approach. Through experimental results being conducted with different DDoS attacks, we prove the proposed novel mechanism is an effective and innovative mechanism for the supervision of DDoS attacks in a cloud environment.

As future work, we plan to design and implement a scalable deep learning-based DDoS attacks recognition system and deploy in a software-defined network controller with the incremental learning technique. It can be deployed as the virtualized function. The proposed approach can be extended for building the new scheme of the multimachine recognition in data center networks that plays the role of an efficient recovery algorithm.

## References

1. S. Dong, K. Abbas, R. Jain, A survey on distributed denial of service (DDoS) attacks in SDN and Cloud Computing Environments. *IEEE Access* **7**, 80813–80828 (2019)
2. B. Zhang, T. Zhang, Z. Yu, DDoS detection and prevention based on artificial intelligence techniques, in *2017 3rd IEEE International Conference on Computer and Communications (ICCC)*, Chengdu (2017), pp. 1276–1280
3. S.T. Zargar, J. Joshi, D. Tipper, A survey of defense mechanisms against distributed denial of service (DDoS) flooding attacks. *IEEE Commun. Surv. Tutor.* **15**(4), 2046–2069 (2013)
4. N. Agrawal, S. Tapaswi, Defense mechanisms against DDoS attacks in a cloud computing environment: state-of-the-art and research challenges. *IEEE Commun. Surveys Tutori.* **21**(4), 3769–3795 (2019)
5. K. Wehbi, L. Hong, T. Al-salah, A.A. Bhutta, A survey on machine learning based detection on DDoS attacks for IoT systems, in *SoutheastCon*, Huntsville, AL, USA (2019), pp. 120–136
6. W. Zhijun, L. Wenjing, L. Liang, Y. Meng, Low-rate DoS attacks, detection, defense, and challenges: A survey. *IEEE Access* **8**, 43920–43943 (2020)
7. A. Sahi, D. Lai, Y. Li, M. Diyk, An efficient DDoS TCP flood attack detection and prevention system in a cloud environment. *IEEE Access* **5**, 6036–6048 (2017)
8. S. Lakshminarasimman, S. Ruswin, K. Sundarakantham, Detecting DDoS attacks using decision tree algorithm, in *2017 Fourth International Conference on Signal Processing, Communication and Networking (ICSCN)*, Chennai (2017), pp. 1–6
9. J. Jiao et al., Detecting TCP-based DDoS attacks in Baidu Cloud Computing Data Centers, in *2017 IEEE 36th Symposium on Reliable Distributed Systems (SRDS)*, Hong Kong (2017), pp. 256–258
10. M. Zekri, S. E. Kafhali, N. Aboutabit, Y. Saadi, DDoS attack detection using machine learning techniques in cloud computing environments, in *2017 3rd International Conference of Cloud Computing Technologies and Applications (CloudTech)*, Rabat (2017), pp. 1–7
11. A. Rukavitsyn, K. Borisenko, A. Shorov, Self-learning method for DDoS detection model in cloud computing. In: *2017 IEEE conference of Russian young researchers in electrical and electronic engineering (EIConRus)*, St. Petersburg (2017), 544–547
12. N. Zhang, F. Jaafar, Y. Malik, Low-rate DoS attack detection using PSD based entropy and machine learning, in *2019 6th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/2019 5th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom)*, Paris, France, 2019, pp. 59–62
13. Y. Feng, H. Akiyama, L. Lu, K. Sakurai, Feature selection for machine learning-based early detection of distributed cyber attacks, in *2018 IEEE 16th Intl Conf on Dependable, Autonomic and Secure Computing, 16th Intl Conf on Pervasive Intelligence and Computing, 4th Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech)*, Athens (2018), pp. 173–180
14. S. Yadav, S. Subramanian, Detection of application layer DDoS attack by feature learning using stacked AutoEncoder, in *2016 International Conference on Computational Techniques in Information and Communication Technologies (ICCTICT)*, New Delhi (2016), pp. 361–366



15. J. Kim, N. Shin, S.Y. Jo, S.H. Kim, Method of intrusion detection using deep neural network, in *2017 IEEE International Conference on Big Data and Smart Computing (BigComp)*, Jeju (2017), pp. 313–316
16. LSTM equations for input output and forget gate [Online]. Available: [https://en.wikipedia.org/wiki/Gated\\_recurrent\\_unit](https://en.wikipedia.org/wiki/Gated_recurrent_unit)
17. M.D. Hossain, H. Ochiai, D. Fall, Y. Kadobayashi, LSTM-based network attack detection: performance comparison by hyper-parameter values tuning, in *2020 7th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/2020 6th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom)*, New York, NY, USA (2020), pp. 62–69
18. K.B. Virupakshar, M. Asundi, K. Channal, P. Shettar, S. Patil, D.G. Narayan, Distributed denial of service (DDoS) attacks detection system for OpenStack-based private cloud. *Procedia Comput. Sci.* **167**(2020), pp. 2297–2307. ISSN 1877-0509
19. B.V. Karan, D.G. Narayan, P.S. Hiremath, Detection of DDoS attacks in software defined networks, in *2018 3rd International Conference on Computational Systems and Information Technology for Sustainable Solutions (CSITSS)*, Bengaluru, India (2018), pp. 265–270

# Automation for Furnace in Thermal Power Station Using Public Key Cryptography



M. Prathyusha, Padmanabha Nikitha, S. Rajashree,  
and B. Prasad Honnavalli

**Abstract** Data is facts and statistics collected together for reference or analysis and it must be protected from corruption or unauthorized access. Data security is practice as well as technology of securing or protecting valuable and sensitive information by means of encryption of data. It is also known as information security. Data can be guarded using various hardware and software technologies. Some common ones include antivirus, encryption, firewalls, etc. Safeguarding the sensitive data from corruption and unauthorized access protects from malicious use of the data. Cryptography refers to securing information using mathematical concepts and techniques. Data encryption software enhances data security with more efficiency. To an authorized person, the encrypted form is absolutely unreadable. The main objective of this paper is encryption and decryption of data received by temperature sensor and motion sensor that has been done using the ECC method.

**Keywords** Encryption · Decryption · Elliptic curves · Packet tracer · Temperature sensor · Motion sensor

---

M. Prathyusha (✉)  
Electrical Department, PES University, Bengaluru, India  
e-mail: [prathyushapreethu@gmail.com](mailto:prathyushapreethu@gmail.com)

P. Nikitha · S. Rajashree  
PES University, Bengaluru, India  
e-mail: [p.nikitha1399@gmail.com](mailto:p.nikitha1399@gmail.com)

S. Rajashree  
e-mail: [rajashrees@pes.edu](mailto:rajashrees@pes.edu)

B. P. Honnavalli  
ISFCR Center, PES University, Bengaluru, India  
e-mail: [prasadhb@pes.edu](mailto:prasadhb@pes.edu)

## 1 Introduction

Power plants are much safer than they were before. There are strict norms. Power plants have improved employee safety to a great extent. Safety requirements, rules and regulations have been established to create a safe environment. However, the employees still encounter hazards. The most common hazards that occur to the power plant employees are electrical shocks, burns, boiler fires, explosions and contact with dangerous chemicals. Standing directly in front of the doors is dangerous. Furnace pulsations produced by firing conditions, soot blower operations or tube failure can blow hot furnace gases out of the open door.

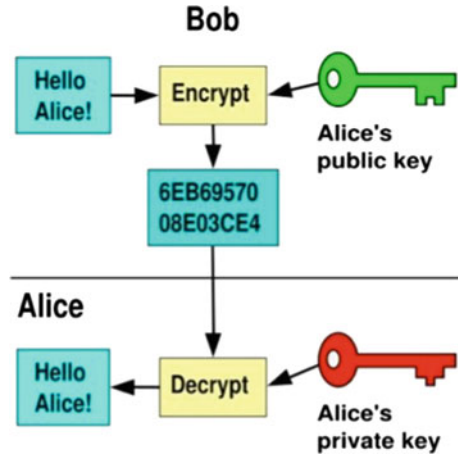
Cryptography is a really powerful method of securing sensitive data. It is a method of protecting information and conversations using codes so that only those who are authorized can read and process the data. The prefix “crypt-” means “hidden” or “vault” and the suffix “-graphy” means “writing.” It provides the four most basic services of information security—confidentiality, authentication, data integrity and non-repudiation.

In general, there are three types of cryptography—symmetric key cryptography, public key cryptography and hash functions. Sensor network services use the sensor data from low end-IoT device of the types widely developed over long distance. In our case, actual temperature is not private. There might be a theft where attacker may provide false temperature and it can cause an undesirable response such as, turning on the furnace unnecessarily. As the plant comprise of high temperature, to ensure no human activity near the power plant motion sensor is considered.

## 2 Concepts Used

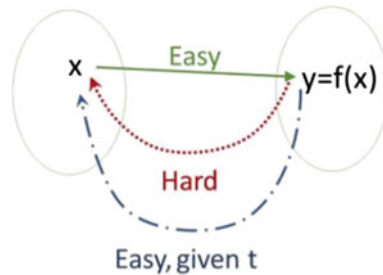
- A. *Plain Text*  
The text in normal form without any encryptions applied yet is called as plain text.
- B. *Cipher Text*  
It is the result of encryption algorithm applied to the plain text. It is not readable until it is converted back to plain text with help of a key.
- C. *Public Key Cryptography*  
This is a type of cryptography where we use two keys. A public key which everyone will know and a private key which only you will know (Fig. 1).
- D. *Symmetric Key Cryptography*  
This is the second type of cryptography. Here, we use only one key to both encrypt and decrypt the information (Fig. 2).
- E. *Trapdoor Function*  
A function  $f$  is trapdoor if it is easy to compute  $f(x)$ , given  $x$ . But hard to compute  $x$ , given  $f(x)$ . An extra trapdoor information makes it easy and then  $x$  can be computed (Fig. 3).

**Fig. 1** Public key cryptography



**Fig. 2** Symmetric key cryptography

**Fig. 3** Trapdoor function



F. *Elliptic Curve Cryptography*

It is a powerful cryptography approach where it uses mathematical tools like elliptic curves to generate the key to encrypt and decrypt the data.

G. *Routing*

It is defined as the process of selecting a path for traffic in a network or between or across networks. Routing is a high-level decision making that directs network packets from the source to destination through intermediate nodes.

The elliptic curve cryptography algorithm is used to encrypt and decrypt the data generated by the sensors—temperature sensor and motion sensor and is being displayed.

### 3 Proposed Method

Elliptic curves have been studied for a long time in the field of number theory and algebraic geometry. An elliptic curve is a set of points that satisfy a particular mathematical equation. This method can offer security with shorter keys with the security level same as in the RSA algorithm. It is designed for devices with limited compute power. Since the key sizes are small, elliptic curve cryptography algorithms can be implemented on smartcard without mathematical coprocessors. Hence, it is becoming widely used and important in the wireless communication areas. Similarly, it is also expected to become increasingly important for wireless sensor networks.

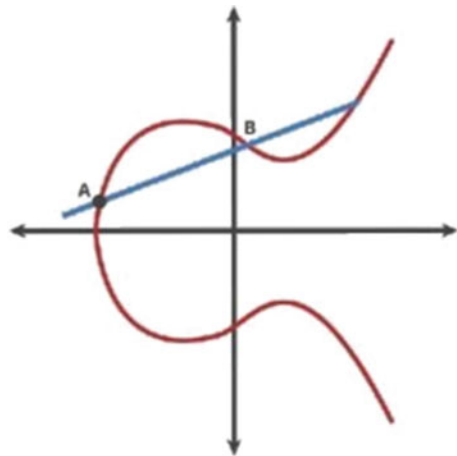
The main advantages of elliptic curve cryptography are—it uses smaller keys, cipher texts, supports very fast key generation. It scores over RSA because of its moderately fast encryption and decryption. ECC computations use less memory and CPU cycles compared to RSA.

Elliptic curve cryptography is based on the difficulty of solving the Elliptic Curve Discrete Logarithm Problem (ECDLP). Though it is a hard problem, it is easy to state: Given two points, A and B, on an elliptic curve, integer n has to be found out such that:

$$B = nA$$

An elliptic curve is defined by an equation in two variables with coefficients. Elliptic curves are not ellipses. Ellipses are formed by quadratic curves. Elliptic curves are always cubic. The curves can be defined over any field of numbers (i.e., real, integer, complex, etc.). Cryptography makes use of such elliptic curves in which the variables and coefficients are all restricted to elements of a finite field. The elliptic curve equation is (Fig. 4)

**Fig. 4** Elliptic curve



$$y^2 = x^3 + Ax + B$$

This equation is also referred to as Weierstrass equation of characteristic 0.

### A. Algorithm

Let  $E(a,b)$  be the elliptic curve. Consider the equation

$$B = K * A$$

Here, if  $K$  and  $A$  are known, it is easy to find  $B$ . But we have defined  $A$  and  $B$ . Hence, it is difficult to find  $K$ . “ $n$ ” is the limit point on elliptic curve.

- Define the global elements  $E$  and  $G$  where  $G$  is the point on elliptical curve which is greater than  $n$
- User A key generation:

1. Select private key  $N_a$  which should be less than  $n$ .
2. Calculate the public key  $P_a$ :

$$P_a = N_a * G$$

3. Calculate the secret key  $K_a$ :

$$K_a = N_a * P_b$$

- User B key generation:

1. Select private key  $N_b$  which should be less than  $n$
2. Calculate public key  $P_b$ :

$$P_b = N_b * G$$

3. Calculate secret key  $K_b$ :

$$K_b = N_b * P_a$$

- Encryption:

1.  $P_m$  is the sensor output data which has to be encrypted and is lying on the elliptic curve.
2. Cipher point is calculated using

$$\{(K * G), (P_m + K * P_b)\},$$

where  $K$  is a random integer.

- Decryption:

1. Take the first co-ordinate from the cipher point and multiply it by B's private key:

$$K * G * Nb$$

2. Take the second co-ordinate from cipher point, subtract  $K * G * Nb$  from it.

$$Pm + K * Pb - (K * G * Nb)$$

We know that,  $G * Nb = Pb$

$$Pm + K * Pb - (K * Pb)$$

Hence,  $Pm$  is the original text.

3. Decryption is successful.

## B. Basic Code

As explained in the algorithm, the values are initialized and coded as follows:

```
#Make E and G global variables
global Eq,G
G=205
#UserA private key
Na=180
#UserB private key
Nb=185
#K is random integer less than n
K=150
#find User A public key
Pa=Na*G
#find User B public key
Pb=Nb*G
#find User A secret key
Ka=Na*Pb
#find User B secret key
Kb=Nb*Pa
#limit 'n'
n=194
#Pm is the list of values that has to encrypted
Pm= []
j=135
while j<=n:
Pm.append(j)
j+=1
print Pm
```

```

#find first co-ordinate in cipher list
FCo=K*G
#Cipher is the list of the encrypted values
cipher=[]
#coordinates and cipher text
print"Elliptical coordinates are:"
for i in Pm:
#second co-ordinate in Cipher list
SCo=i+(K*Pb)
print (FCo,SCo)
cipher.append(SCo)
print "Encrypted values=",cipher
original=[]
#decryption
D=Nb*FCo
print "D",D
for i in cipher:
D1=i-D
original.append(D1)
#original is the list of decrypted values
print "Decrypted values=",original

```

In this code, the values that are appended to the Pm list are encrypted and added into the cipher list. Then, the values from the cipher list are decrypted and appended into the original list for displaying.

By making a few changes in the code, it can be used for a longer time or an even shorter time.

### C. Technology used

Cisco Packet Tracer is an effective education simulation software which supports computer networks for experimenting and practicing network related projects. In this software, the networking devices appear as they would in real life and that enables thorough understanding in various networking devices and techniques. The path of a packet can be tracked when it moves from source to destination. Various tests can be run so that different kinds of network failures can be understood and troubleshooting them also can be learnt. The packet tracer can also be utilized to learn different networking devices like hubs, switches, server, etc. and the appropriate manner to use them (Fig. 5).

Packet tracer allows users to experience network simulation using various devices such as routers, switches, wireless access points, microcontroller boards and single board computers in a user-friendly environment. It is an open-source software that anyone can download from the internet for free of cost and can be used for any use.

It allows usage of three languages within the software—Python, Java and Visual. Hence, it is very convenient to code and simulate networks.





**Fig. 5** Cisco Packet Tracer

Language used throughout the experiment is Python. The main reason Python was used was that it was very easy and fast to develop. It has all the libraries which is needed by the experiment.

## 4 Experiment Data

### A. *Network Design*

The design consists of single board computer (SBC) and microcontroller boards to operate and control the actuators and sensors, LCD which displays the output, router to enable sending of data from one network to other and actuators which take an electrical input and turn it into physical action. Heating element and LCD display are actuators (Fig. 6).

### B. *Components*

In the model, the concept of socket programming is heavily used to send the data from the client to server. Client is defined as a computer hardware or software that accesses a service made available by a server and a server is defined as a computer hardware or software that provides functionality for other programs or devices like clients (Figs. 7, 8 and 9).

### C. *IP Addressing*

An IP address is a human readable number assigned to device connected to the network which uses the Internet protocol for communication. Another number

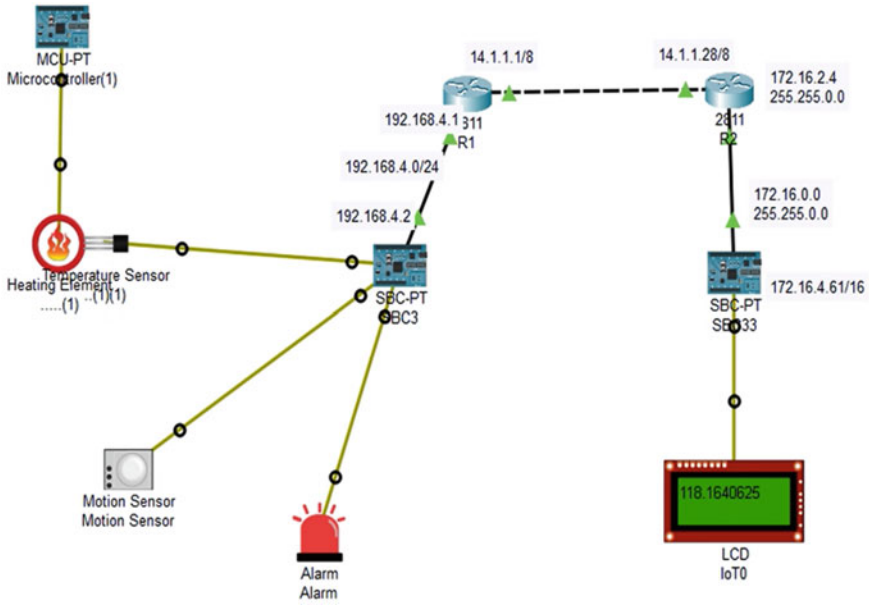
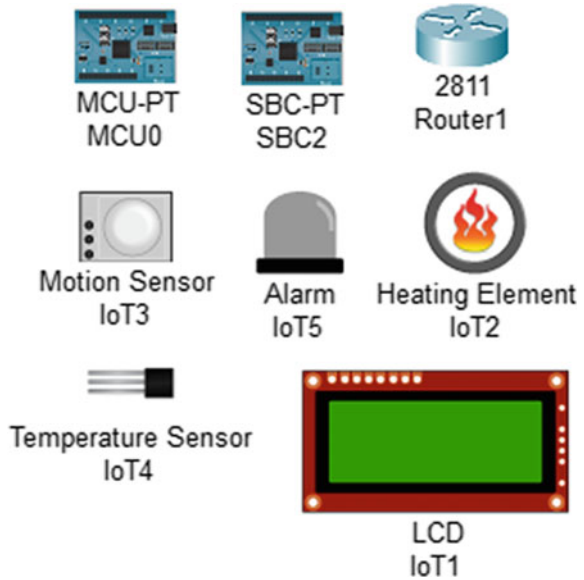
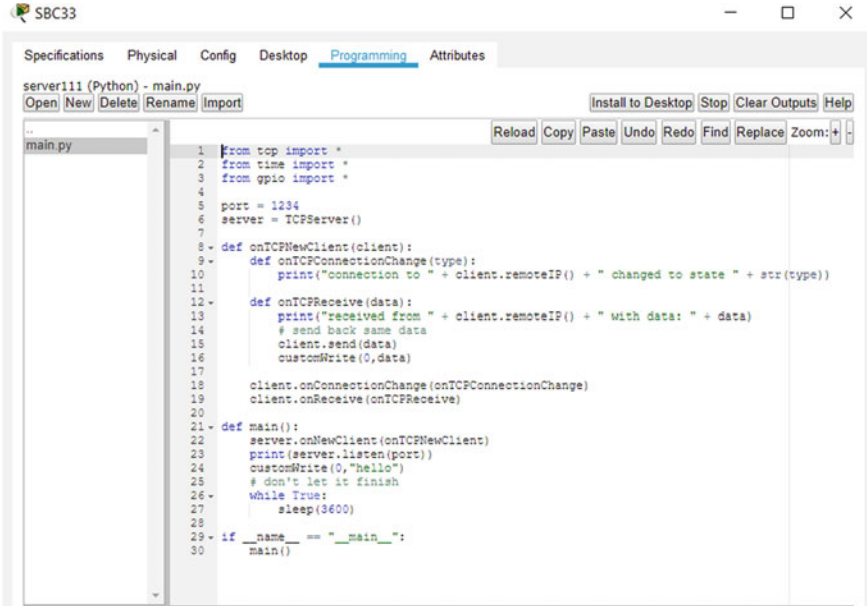


Fig. 6 Screenshot of the working model

Fig. 7 Components used in the topology

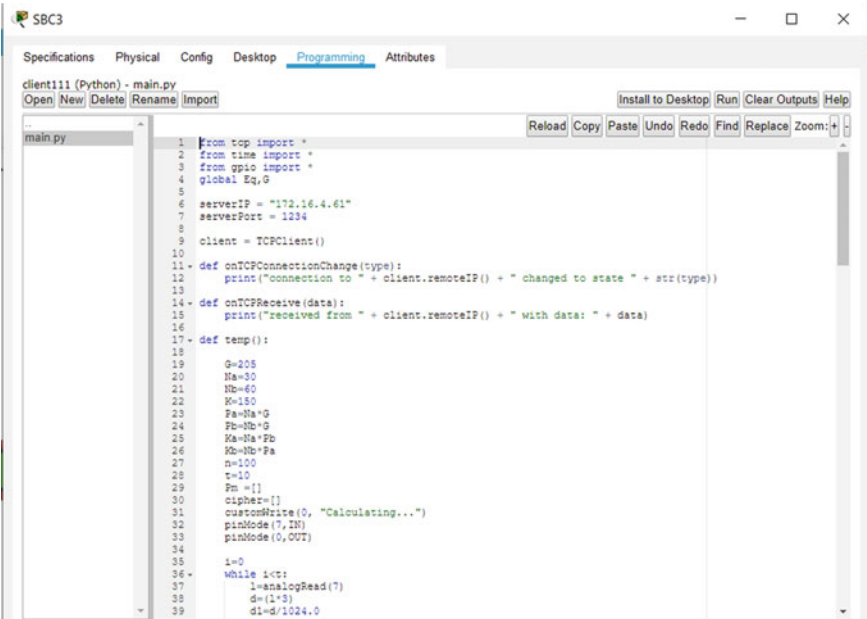




The screenshot shows an IDE window titled "server111 (Python) - main.py". The code defines a TCP server using the `TCPServer` class. It includes a `main` function that starts the server on port 1234 and sends a "hello" message to the first client. The server handles connection changes and receives data from clients.

```
1 from top import *
2 from time import *
3 from gpio import *
4
5 port = 1234
6 server = TCPServer()
7
8 def onTCPNewClient(client):
9     def onTCPConnectionChange(type):
10        print("connection to " + client.remoteIP() + " changed to state " + str(type))
11
12    def onTCPReceive(data):
13        print("received from " + client.remoteIP() + " with data: " + data)
14        # send back same data
15        client.send(data)
16        customWrite(0,data)
17
18    client.onConnectionChange(onTCPConnectionChange)
19    client.onReceive(onTCPReceive)
20
21 def main():
22     server.onNewClient(onTCPNewClient)
23     print(server.listen(port))
24     customWrite(0,"hello")
25     # don't let it finish
26     while True:
27         sleep(3600)
28
29 if __name__ == "__main__":
30     main()
```

Fig. 8 Screenshot of the code for server part

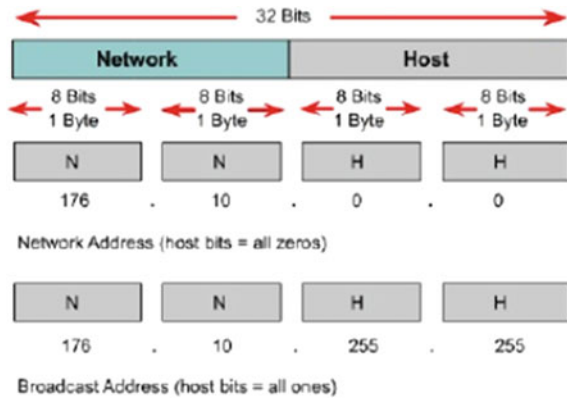


The screenshot shows an IDE window titled "client111 (Python) - main.py". The code defines a TCP client that connects to a server at IP 172.16.4.61 on port 1234. It sets up GPIO pins for an analog-to-digital converter (ADC) and performs a calculation, displaying the result on the serial monitor.

```
1 from top import *
2 from time import *
3 from gpio import *
4 global Eq,G
5
6 serverIP = "172.16.4.61"
7 serverPort = 1234
8
9 client = TCPClient()
10
11 def onTCPConnectionChange(type):
12     print("connection to " + client.remoteIP() + " changed to state " + str(type))
13
14 def onTCPReceive(data):
15     print("received from " + client.remoteIP() + " with data: " + data)
16
17 def temp():
18
19     G=205
20     Na=30
21     Nb=60
22     M=150
23     Fa=Na*G
24     Fb=Nb*G
25     Ka=Na*Fb
26     Kb=Nb*Fa
27     n=100
28     t=10
29     Fm=[]
30     cipher=[]
31     customWrite(0, "Calculating...")
32     pinMode(7,IN)
33     pinMode(0,OUT)
34
35     i=0
36     while i<t:
37         i=analogRead(7)
38         d=(2*3)
39         di=d/1024.0
```

Fig. 9 Screenshot of the code for the client part

Fig. 10 IP addressing



defines the range of IP address included in the network called Subnetmask. IP addresses are classified into five classes, namely Class A till E. Each of them has a specific range and can be used for particular purposes only.

Cisco Packet Tracer generates the subnet mask by itself. In the topology, client side lies in the network with the IP address 192.168.X.X and the server side with the sequence 172.16.X.X. There also exists a network between the two routers since it needs to communicate to send and receive data and its IP address sequence will be 14.1.X.X (Fig. 10).

#### D. Results

The temperature values and motion sensor data that are being sensed in client network are sent to the server network via the routers to be displayed on the LCD. With an appropriate delay time, temperature values are seen to vary. When sensed temperature exceeds the set limit, the alarm rings and alerts everyone beforehand and hence makes sure that there is less casualty and normal operations at the power station can resume again without much delay (Figs. 11 and 12).

As seen in picture, the values sensed by the motion sensor are either 0 or 1023 only. Here, 0 indicates that there is not motion detected by the sensor and 1023 indicates that motion has been detected and the alarm rings because of that.

#### E. Outcome

In the Cisco Packet Tracer simulation tool, the sensors have detected the temperature values and motion values and then those values have been sent to the server side via the routers and then it has been displayed to the user on the LCD.

When used in the thermal power station, it is expected to be in operation at the door area near the furnace. In furnace of thermal power stations, temperature will be extremely high in range of 600 °C which is dangerous, and hence, it is necessary to make sure that there will be no human activity near it for the safety of the employees.

Sensing part of the network which will continuously keep monitoring the temperature to cool down to a level which is not harmful and motion to avoid anyone to come

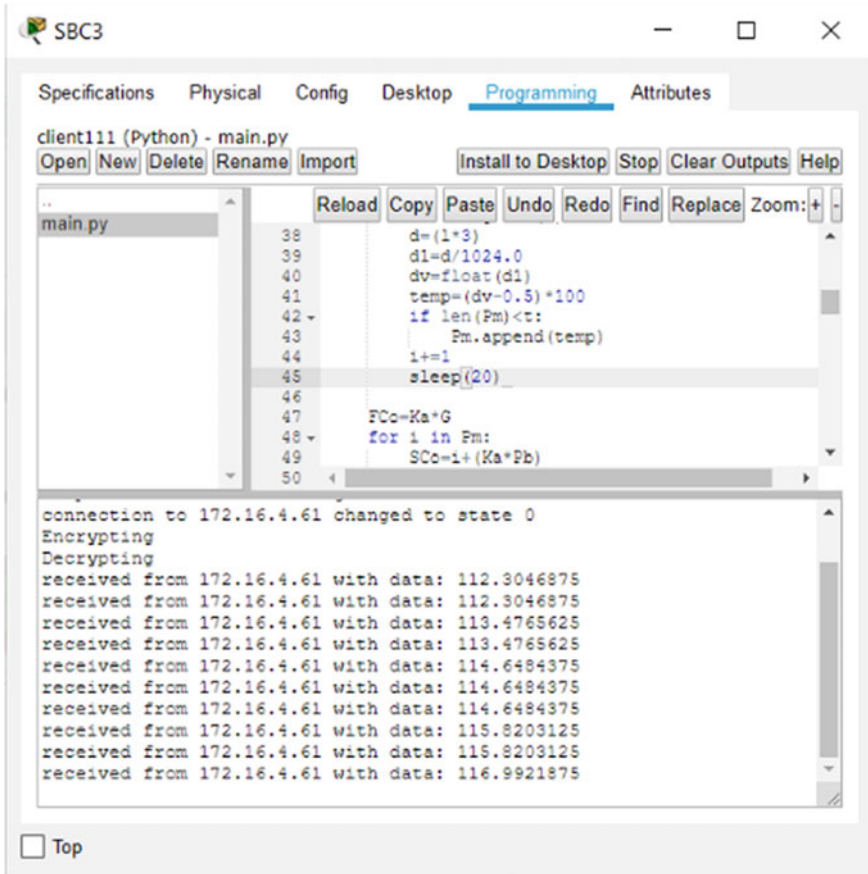


Fig. 11 Screenshot of the working temperature sensor

in contact with the hazardous gases and chemicals will be operating in the furnace room and another network that is connected to the LCD display will be connected outside enabling the employees to go in at the right time without any accidents.

This topology works for a fixed ten values of temperature and motion sensing. This number can be increased or decreased according to the requirement. By modifying the code, it can be used with time constraints as well.

As the operating person can see the sensed and decrypted values from outside of the furnace room, mishaps can be prevented as much as possible. As the values are encrypted, there will be no fear of theft of such sensitive data.

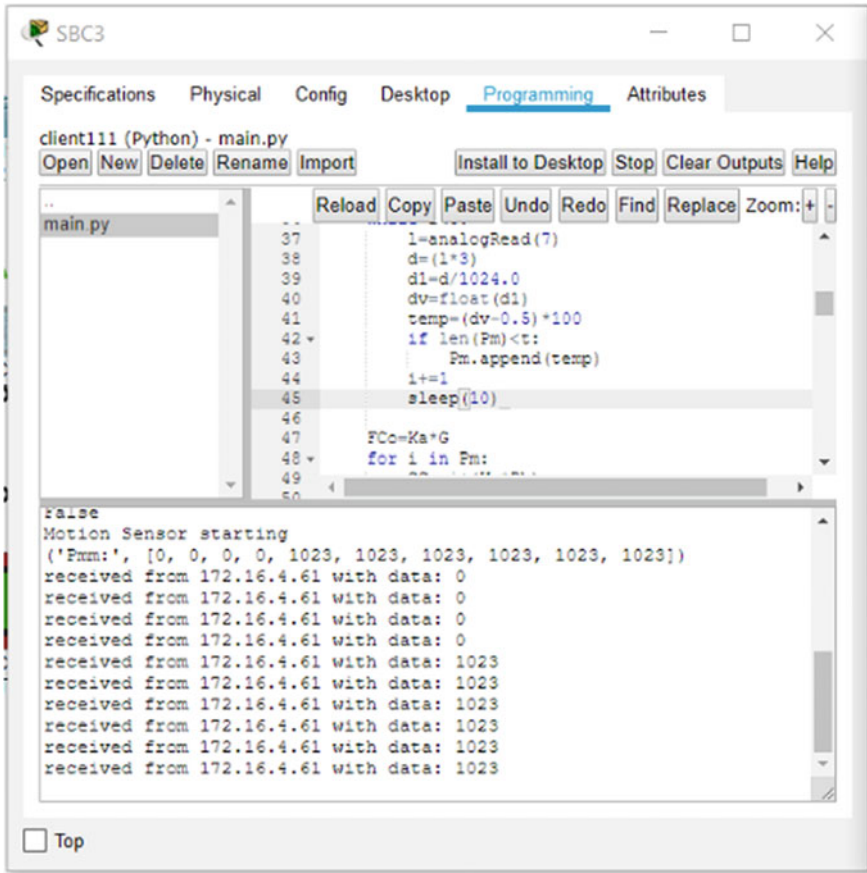


Fig. 12 Screenshot of the working motion sensor

## 5 Conclusion

This paper mainly describes how cryptography can be used to ensure that data is transmitted without any alterations keeping it confidential. With the help of a type of public key cryptography that is elliptic curve cryptography, the encryption and decryption of the data that is obtained by the temperature and motion sensors have been demonstrated.

The temperature in the furnace area is kept track of all the time, the values are closely monitored, if in case it exceeds a set limit, the alarm rings alerting the employees. Similarly, the values from the motion sensor near the entrance for the furnace area are monitored so that no one comes in contact with the hazardous gases and chemicals.

The experiment can further be improved by adding immediate operating of water sprinkler in case temperature does exceed. A variety of sensors like smoke sensor to detect which gas is harmful, radiation sensor to detect the level of radiation so the employees are informed priorly can be utilized to further increase safety for everyone in the power station.

This topology could be also used in cases like, large-scale cooking, metallurgy and mining. As certain temperature is to be maintained in these areas, temperature range can be adjusted as per the requirement.

**Acknowledgements** First and foremost, we would like to express our sincere gratitude to our guides Mrs. Rajashree Soman and Mrs. Vineetha B. for the continuous support for our project, for your patience, motivation and encouragement to do better. Their guidance helped us throughout the project and hence we could successfully complete it.

We would also like to express our deepest appreciation to our university to have given us this wonderful opportunity to learn and write a paper.

## References

1. Avinash Kak, Purdue University's paper can be found at <https://engineering.purdue.edu/kak/compsec/NewLectures/Lecture14.pdf>
2. SHEIKH RAASHID JAVID PHd,RK University's paper can be found at [https://www.researchgate.net/profile/Sheikh\\_Javid/publication/264233874\\_Role\\_of\\_Packet\\_Tracer\\_in\\_learning\\_Computer\\_Networks/links/53d33b6a0cf228d363e97376/Role-of-Packet-Tracer-in-learning-Computer-Networks.pdf](https://www.researchgate.net/profile/Sheikh_Javid/publication/264233874_Role_of_Packet_Tracer_in_learning_Computer_Networks/links/53d33b6a0cf228d363e97376/Role-of-Packet-Tracer-in-learning-Computer-Networks.pdf)
3. John Mitchell's work can be found at <https://crypto.stanford.edu/cs155old/cs155-spring03/lecture8.pdf>

# Active Dictionary Attack on WPA3-SAE



Manthan Patel, P.P Amritha, and R. Sam jasper

**Abstract** In wireless network, we have different protocols like WEP, WPA, and WPA2. WPA3 is currently used standard protocol in WIFI to authenticate the client with access point. In the WPA3, Simultaneous Authentication of Equals protocol downgrade attack is already discovered. With the downgrade attack, we are able to do offline dictionary attack on WPA3-SAE protocol. WPA3-SAE is also known as WPA3-Personal. Dictionary attack is classified into active dictionary attack and passive dictionary attack. Passive dictionary attack is also known as offline dictionary attack. In this paper, we proposed active attack model in which software will try different password from given dictionary word list until it connect with the Access Point. In this model, computer will change their MAC address continuously so that access point won't detect as an attack. To speed up the process, we can use multiple virtual machines that will work as a separate wireless client to the access point.

**Keywords** Active dictionary attack · WPA3-SAE · Wi-Fi Security

## 1 Introduction

After WPA2 failure Wi-Fi Alliance published WPA3-SAE and WPA3-Enterprise protocol. WPA3 is not a new protocol but it is a modification of existing protocol [1]. In WPA2, Wi-Fi Alliance adds the Dragonfly handshake that is known as WPA3 protocol. WPA3 protocol also providing the backward compatibility so WPA2 device can also pair up with access point. Dragonfly handshake is used in EAP-pwd and

---

M. Patel · P. Amritha (✉) · R. Sam jasper (✉)  
TIFAC-CORE in Cyber Security, Amrita School of Engineering,  
Amrita Vishva Vidyapeetham, Coimbatore, India  
e-mail: [pp.amritha@cb.amrita.edu](mailto:pp.amritha@cb.amrita.edu)

R. Sam jasper  
e-mail: [samjasper@protonmail.com](mailto:samjasper@protonmail.com)

M. Patel  
e-mail: [cb.en.p2cys19011@cb.students.amrita.edu](mailto:cb.en.p2cys19011@cb.students.amrita.edu)

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2021  
S. M. Thampi et al. (eds.), *Advances in Computing and Network Communications*,  
Lecture Notes in Electrical Engineering 735,  
[https://doi.org/10.1007/978-981-33-6977-1\\_46](https://doi.org/10.1007/978-981-33-6977-1_46)



WPA3-SAE, where EAP-pwd protocol is used in enterprise Wi-Fi networks and WPA3-SAE is used in Home Network. This both protocols are providing forward secrecy and offline dictionary attack security [3], which devices are not supporting WPA3-SAE protocol authentication for that devices to give backward compatibility access point will work in a transition mode. In this mode, access point will work with WPA2-PSK and WPA3-SAE which are simultaneously using same username and passphrase. With the use of auditing, we can identify vulnerabilities and performing the penetration testing to see the issues in wireless network [10]. Dictionary attacks are very well-known attacks in WPA2 protocol. In WPA2 protocol, PSK is the main key which will protect this WLAN communication. In offline dictionary attack, attacker can capture the initial four-way handshake frames and try to perform offline dictionary attack. That is also known as passive dictionary attack [2]. We analyzed that one time password generator is already proposed for WPA2 but in WPA3 this method is not used to connect wireless client to access point [13]. In this paper, we presented a new model which will perform Online dictionary attack using transition mode on WPA3-SAE protocol. We are using this attack since offline dictionary attack is not possible because dragonfly handshake is providing core security to the communication. In transition mode, WPA3-SAE is also providing downgrade security with robust security network element (RSNE) which will generate unauthenticated beacons to advertise the network. That beacons will verify during four-way handshake. In access point, there is feature of MAC-filtering but to overcome on this security, we are frequently changing MAC of computer. We presented a model which will take one by one different password from words list dictionary and try to connect with access point. To speed this process, we are proposing two methods, first one is attack should be in transition mode and second one is we can use different virtual machines which will act as a separate wireless client for an access point.

## 2 Related Work

In transition mode, access point will accept WPA2 four-way handshake to connect with wireless client. Most common technique to bypass is WPA2 security by capturing initial frames of authentication between wireless client and access point. After getting that handshake frame attacker can perform offline dictionary attack on it because these handshake packets are containing passphrase of the wireless access point. But to secure this offline attack WPA3-SAE protocol is providing downgrade security with RSNE which will generate unauthenticated beacons periodically to discover the presence of SSID in the network. In this session, we are explaining how access point and client will exchange different keys to protect WPA2 communication (Fig. 1).

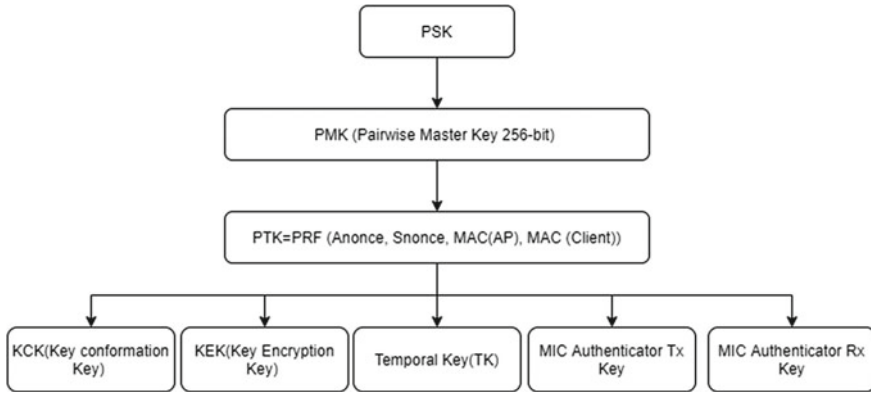


Fig. 1 Key generation in WPA2

### 2.1 Key Generation in WPA2-PSK

Pre-shared-key (PSK) is nothing but the password of the access point. This PSK is used to derive different keys which will help to secure wireless communication. In WPA2-PSK, when a client authenticates with an access point, this pre-shared-key will become a Pairwise Master Key (PMK) which is derived from a key derivation function and that function will use SHAH-1 as a hash function. PMK is used to generate a Pairwise Transit Key (PTK). PTK is generated with the use of a pseudorandom function, which uses a combination of PMK, Anonce (which is a 32-bit random number generated by the access point), Snonce (which is a 32-bit random number generated by the client), MAC address of the access point, and MAC address of the wireless client. That combination will give a 512-bit PTK, and this key is used for encryption of all unicast communication between the access point and the wireless client. This PTK will be used to derive five separate keys. In PTK, the first 128 bits act as a Key confirmation Key (KCK) which is used to compute a Message Integrity Code (MIC). The second 128 bits in PTK act as a Key Encryption Key (KEK), and this key is used for encrypting the data between the access point and the wireless client. The third 128 bits act as a Temporal Key (TK), and it is used for encrypting and decrypting the unicast traffic. The fourth 128 bits are used to compute MIC Authenticator Tx Key and MIC Authenticator Rx Key, and both keys are 64-bit long. Another key is the Group Temporal Key (GTK) which is used to encrypt and decrypt all the multicast and broadcast communication between the access point and the client. Every access point will always have a different GTK which is shared with the connected wireless client.

## 2.2 Key Exchange in WPA2-PSK

Both the access point and the wireless client is dependent on the four-way handshake communication to ensure the having control of PSK. After the wireless client authenticates and associates process with access point, four-way handshake will start. Four-way handshake consists of four messages. Extensible authentication protocol over LAN (EAPOL) which will help to complete this four-way handshaking messages between both wireless client and access point. Firstly, the access point sends Message 1 which will contain Anonce and this message is going through unicast traffic to the client.

After getting this message 1 from access point, client will have all the parameters to derive Pairwise Transit Key from Pre-shared-key. From this, PTK client will generate remaining keys like KCK, KEK, and TK. From Message Integrity Code (MIC) and Snonce (32-bit random number), client will generate message 2 and sends to the access point. MIC is required to make sure that the Message 2 is tampered or not while transferring. After getting Message 2 from client-side, access point is able to derive PTK. Hence, PTK access point will get all the remaining keys like KCK, KEK, and TK. Now access point will compute MIC from the PTK and compare with MIC which is obtained from the wireless client in Message 2; in this way, access point can check the integrity of the Message 2. Now access point will encrypt GTK with the use of KCK. With this encrypted GTK and MIC access point will generate Message 3 and send to the client. Message 4 will be sent by the wireless client to the access point for the acknowledgment of successfully completing four-way handshake as shown in Fig. 2. In this process, if attacker generates Message 2 by guessing password and if they gets Message 3 as a reply from access point, attacker can ensure that the password which is used to generate the Message 2 is correct password to authenticate with access point.

## 3 Active Dictionary Attack

The given model in Fig. 3 is able to perform an active dictionary attack on WPA3-SAE. The goal of this attack is to find out the password without capturing the four-way handshake packets between the access point and wireless client. In this attack, first we will force to access point to use WPA2 four-way handshake to connect with wireless client with the use of transition mode. After that, software will try to guess the passwords automatically from a given dictionary file and it will generate Message 2 of the four-way handshake communication. The software then sends that generated message to the access point, and it will wait for the reply. If it gets Message 3 as a reply from access point, it means that guessed password is correct one. If it gets

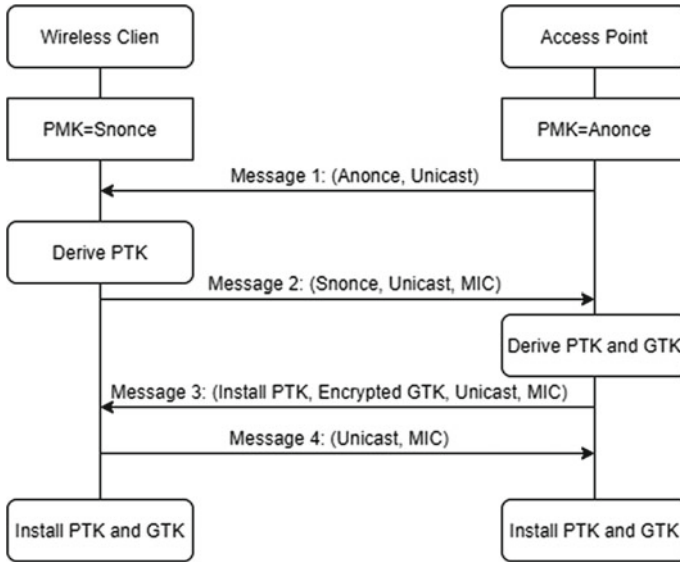
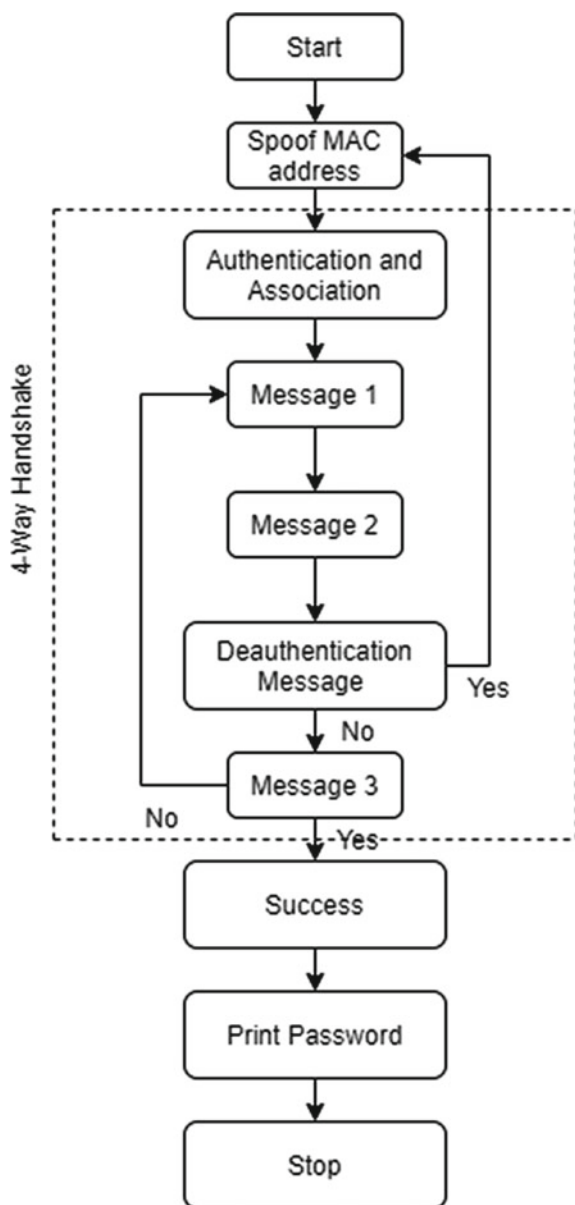


Fig. 2 Key exchange in WPA2

Message 1 in a reply of Message 2 from the access point, it means password which is used to create Message 2 is not correct one. Software is monitoring every reply which is coming from the access point. To speed up this process, computer will try different password from the given dictionary in the same session with access point. If an access point has a security that particular device has limited attempt for trying incorrect password, then after some incorrect tries access point will send deauthentication message. The computer will be disengaged from the access point, and it will create a new session with the access point after changing MAC address. After a computer passes the association and authentication stages of the access point, the computer will begin the four-way handshake with the access point. Then computer can apply different password to connect with the access point from a dictionary file. If the access point responds with Message 3 then the passphrase which is guessed, that is correct otherwise the software will keep trying different password from the dictionary file in the same session. If access point reply with message 3, then attack is successful and password will print on the display. To speed up this attack, we can use different virtual machines at a same time. For access point, these virtual machines will act as a separate wireless client which will try different password from the dictionary. MAC address of this VMs will also change after frequent time.

**Fig. 3** Active dictionary attack proposed model



### 3.1 Proposed Software Program

**Listing 1.1** Implementation software in Python pseudo-code.

---

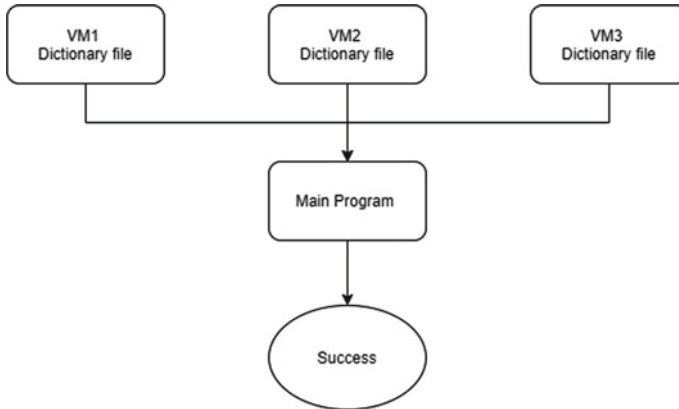
```

1  IMPORT OS, platform, pyWi-Fi
2  from pyWi-Fi IMPORT PyWi-Fi
3  from pyWi-Fi IMPORT Profile
4  SET number TO 0
5  DEFINE FUNCTION MAC_changer():
6      os.system macshift.exe -i "eth0"
7  DEFINE FUNCTION main(ssid, password, number):
8      SET profile.ssid TO ssid
9      profile.akm.append(const.AKM_TYPE_WPA2PSK)
10     SET profile.key TO password
11     iface.remove_all_network_profiles
12     SET tmp_profile TO iface.add_network_profile(profile)
13     iface.connect(tmp_profile) # trying to Connect
14     message_temp = reply from access point
15     if message_temp == Deauthentication
16         MAC_change()
17         menu()
18     elsif ifaces.status() EQUALS const.IFACE_CONNECTED:
19         OUTPUT(Crack success!',RESET)
20         OUTPUT('password is ' + password, RESET)
21         exit()
22     ELSE:
23         OUTPUT(Crack Failed using this password)
24 DEFINE FUNCTION pwd(ssid, file):
25     with open file with read permission as words:
26         FOR line IN words:
27             number += 1
28             SET line TO line.split("\n")
29             SET pwd TO line[0]
30             main(ssid, pwd, number)
31 DEFINE FUNCTION menu():
32     IF args.wordlist and args.ssid:
33         SET ssid TO args.ssid
34         SET filee TO args.wordlist
35     ELSE:
36         SET ssid TO INPUT(" SSID: ")
37         SET filee TO INPUT("password file: ")
38     IF os.path.exists(filee):
39         pwd(ssid, filee)
40     ELSE:
41         OUTPUT "No Such File."
42 menu()

```

---

In this pseudo-code, first we are taking name of the Wi-Fi (SSID) and path of the dictionary file. If file path is not proper, software will stop. If file path is proper, it will check status of wireless interface of computer which is connected or not. After that, this software will take one by one password from the dictionary file and try to connect the access point. This software will continuously monitor the status of the wireless interface. This process will run until status of wireless interface counter will become one or password list will be completed. Software will check all the replies which are coming from access point. If deauthentication packet will come, software will change MAC address and try different password. If attack is successful, it will print the password and process will stop.



**Fig. 4** Active dictionary attack proposed model with different VMs

### 3.2 Proposed Method with Different Virtual Machines

In proposed model, we used different VMs at a same time which will reduce the attack time. Here in Fig. 4, we can see that all 3 VMs will act as a legitimate user for a wireless access point and try the password from the dictionary file. In this process, VMs will use same software but we divided the content of dictionary file. By this process, we can amplify the attack efficiency. In a same way, we can use as many as possible which will help to reduce the time of attack.

## 4 Conclusion

Active dictionary attack can recover WPA3-SAE password in transition mode even when attacker is not able to capture the four-way handshake frames between wireless client and access point. In this paper, we proposed a method to attack on WPA3-SAE protocol to recover the password. To speed the attack, we proposed to use different machines at a same time. For access point, all virtual machines will act as a legitimate wireless client. VMs will start picking up passwords from the dictionary file and try different password at a same time in same session. This method is much faster than the traditional active dictionary attack.

## References

1. M. Vanhoef, E. Ronen, Dragonblood: analyzing the dragonfly handshake of WPA3 and EAP-pwd, in *Proceedings of the 2020 IEEE Symposium on Security and Privacy (S&P 2020)* (IEEE, 2020)

2. O. Nakhila, A. Attiah, Y. Jin, C. Zou, Parallel active dictionary attack on WPA2-PSK Wi-Fi networks, in *2015 IEEE Military Communications Conference (MILCOM 2015)* (IEEE, 2015), pp. 665–670
3. C.P. Kohlios, T. Hayajneh, A comprehensive attack flow model and security analysis for Wi-Fi and WPA3. *Electronics* **7**(11), 284 (2018)
4. <http://www.aircrack-ng.org>
5. D. Fehér, B. Sandor, Effects of the WPA2 krack attack in real environment, in *2018 IEEE 16th International Symposium on Intelligent Systems and Informatics (SISY)* (IEEE, 2018)
6. M.A. Abo-Soliman, M.A. Azer, A study in WPA2 enterprise recent attacks, in *2017 13th International Computer Engineering Conference (ICENCO)* (IEEE, 2017)
7. T. Radivilova, H.A. Hassan, Test for penetration in Wi-Fi network: attacks on WPA2-PSK and WPA2-enterprise, in *2017 International Conference on Information and Telecommunication Technologies and Radio Electronics (UkrMiCo)* (IEEE, 2017)
8. C.-M. Chen, T.-H. Chang, The cryptanalysis of WPA & WPA2 in the rule-based brute force attack, an Advanced and efficient method, in *2015 10th Asia Joint Conference on Information Security* (IEEE, 2015)
9. <https://linuxconfig.org/how-to-change-mac-address-using-macchanger-on-kali-linux>
10. A.K. Mohan, M. Sethumadhavan, Wireless security auditing: attack vectors and mitigation strategies. *Procedia Comput. Sci.* **115**, 674–682 (2017)
11. A.A. Kumar, A.K. Mohan, P.P. Amritha, Deceiving attackers in wireless local area networks using decoys. *J. Cyber Secur. Mob.* **7**(1), 201–214 (2018)
12. A. Raghuprasad, S. Padmanabhan, M. Arjun Babu, P.K. Binu, Security analysis and prevention of attacks on IoT devices, in *2020 International Conference on Communication and Signal Processing (ICCS)* (IEEE, 2020), pp. 0876–0880
13. C. Sudar, S.K. Arjun, L.R. Deepthi, Time-based one-time password for Wi-Fi authentication and security, in *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)* (IEEE, 2017), pp. 1212–1216



# Multiple Hashing Using SHA-256 and MD5



Gautham P. Reddy, Anoop Narayana, P. Karan Keerthan, B. Vineetha, and Prasad Honnavalli

**Abstract** Message Digest 5 (MD5) is a hashing function with numerous vulnerabilities such as pre-image vulnerability and collision vulnerability which restrict the usage of MD5. Therefore, by using other hashing functions such as SHA prior to hashing with MD5, we can use MD5 for various applications such as data integrity without compromising the security of the hash. MD5 is widely used in file transfer or storage applications because it produces a smaller hash value of 128 bits when compared with other hashing algorithms. Also, it is simpler to implement in hardware and as a program. We propose a technique of hashing the original message (or string) with secure hashing algorithms such as SHA-256 followed by hashing the hash value of SHA-256 with MD5 to get the resultant hash which is less prone to various security attacks such as collision attacks. By hashing the string twice, we make it more secure and tackle the pre-image vulnerability and collision vulnerability of MD5. This makes the hashing algorithm more secure for file transfer applications. Multiple iterations will produce more secure hash values but our simulation uses two iterations, where we upload a file onto a cloud server and check if it has been tampered with or modified.

**Keywords** MD5 · SHA-256 · Multiple hashing · Cryptography · Salt · File integrity

---

G. P. Reddy · A. Narayana · P. K. Keerthan  
Department of Electronics and Communication Engineering, PES University, Bengaluru, India

B. Vineetha (✉) · P. Honnavalli  
Department of Computer Science and Engineering, PES University, Bengaluru, India  
e-mail: [vineethab@pesu.pes.edu](mailto:vineethab@pesu.pes.edu)

PES University, Bengaluru, India

P. Honnavalli  
e-mail: [prasadhb@pes.edu](mailto:prasadhb@pes.edu)

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2021  
S. M. Thampi et al. (eds.), *Advances in Computing and Network Communications*,  
Lecture Notes in Electrical Engineering 735,  
[https://doi.org/10.1007/978-981-33-6977-1\\_47](https://doi.org/10.1007/978-981-33-6977-1_47)

## 1 Introduction

MD5 was developed in 1991 by Ronald Rivest of MIT for the purpose of cryptographic hash functions, and it joined the series of message digest algorithms such as MD4. However, the security of MD5 has been severely compromised by various collision attacks, and as early as 1993, it was theorized that MD5 was vulnerable to collision attacks while it was successfully demonstrated in 2005 [1]. MD5 and sha-1 hash were shown to be vulnerable to collision attacks. Despite being vulnerable, numerous content management systems use MD5 which can compromise their security. Several hashing algorithms such as Secure Hash algorithm (SHA) and Whirlpool have been developed but the message digest or hash value in these is 256 and 512 bits, respectively, thus, consuming more space. Hence, using a stronger algorithm such as SHA before using MD5 can bolster MD5. When data is downloaded or transferred, there is a possibility for corruption of data due to errors in the communication systems. Checking each bit of the transferred data against each bit of the original data is imprudent because for large files; it takes a prolonged amount of time. Thus, we use a hashing function which generates a small hash value to be compared with the hash value of the original file. Currently, MD5 can be used to only check unintentional corruption of data, but by using multiple hashing functions before MD5, we can make it more secure. Therefore, by using stronger hashing techniques before using MD5, we can improve the security of MD5 and also retain its advantage of producing a hash value of a relatively small size. Addition of a random salt to the original string can further improve its security [2]. In our simulation, we appended a random salt (a string of random letters) and also appended the content of the file (the message string) to further enhance security. Using multiple hashes is a common practice in storing passwords in databases but we propose the same for file storage in a server. A user can upload a file along with its multiple-hashed value. While downloading it from a server, he can calculate the multiple-hash value and check if it is equal to the hash value generated by the user who uploaded the file. If the file has not been modified intentionally or corrupted unintentionally, we should obtain the same hash value on both the sides. By creating a C program that calculates the SHA-256 hash value and the MD5 hash value of a given string, we hashed the contents of a given file by adding a random salt to it. This paper presents a data integrity method based on MD5 and SHA-256 algorithm called as a multiple hashing [3].

## 2 MD5 Architecture and SHA Architecture for Multiple Hashing.

### 2.1 MD5 Architecture

A hashing algorithm converts data of any length into a value of a fixed length, and the fixed length varies based on the algorithm [4]. MD5 is a part of the series of message

digest algorithms developed after its predecessor MD4. The MD hash family shares a common structure of the compression function which consists of message expansion and the consecutive evaluation of a number of similar operations called steps. These steps are usually grouped together into 3–5 rounds.

The message expansion ensures that every single message block is used multiple times. To increase the diffusion of the message, words, a recursive message expansion technique was used. Every input of each step is highly correlated to the original message so even a small alteration in the message affects numerous steps leading to a different hash value or message digest.

The operations used are:

1. Bitwise Boolean operations.
2. Integer addition.
3. Bit shift operations or bit rotations.

These operations have been chosen since they can be efficiently evaluated and they are cryptographically strong. Therefore, they can be implemented in hardware as well.

**STEP\_1:** MD5 is a hash function that processes or hashes the message by breaking it into blocks or data chunks of 512 bits of data. The final 64 bits (that is bit –448 to bit –512) are meant to store the original length of the message (before breaking it into chunks). Thus, the length of the message cannot exceed 2 raised to 64 bits. Hence, the message is padded with zeros until its equal to  $(\equiv) 448$  modulo 512. However, the first bit which is padded is 1 instead of a 0. Following this 1, every other bit is a zero. The final 64 bits (left most bits) contain the length of the initial or original message.

**STEP\_2:** The registers used for MD5 are initialized as shown below:

$$A = 0x67452301$$

$$B = 0xEFCDAB89$$

$$C = 0x98BADCFE$$

$$D = 0x10325476$$

**STEP\_3:** The most crucial step in the MD5 algorithm is processing of each message block. For each input block, there are four rounds of operation, and each operation uses a different Boolean function which is shown below:

$$M(X, Y, Z) = (X \text{ AND } Y) \text{ OR } (\text{NOT } X \text{ AND } Z) \quad (1)$$

$$N(X, Y, Z) = (X \text{ AND } Z \text{ OR } (\text{NOT } Y \text{ AND } Z)) \quad (2)$$

$$Q(X, Y, Z) = X \text{ XOR } Y \text{ XOR } Z \quad (3)$$

$$P(X, Y, Z) = Y \text{ XOR } (X \text{ OR } \text{NOT } Z) \quad (4)$$

Each of these rounds has 16 operations which mean that there are effectively 64 rounds in MD5. Each of these 64 rounds also uses a constant and stores the result in A, B, C, and D buffers. Further, the message block of 512 bits is broken down into words of 32 bit length which are used in each of these rounds. By using the equation shown below, we update the buffer values in which I(b, c, d) refers to the Boolean functions M, N, Q, and P while X refers to the 32 bit words formed by the 512 bit message blocks, and T is the constants specified earlier. The variable s specifies the number of times it is left shifted to obtain the final value.

$$a = b + (a + I(b, c, d) + X + T[i]) \ll s \quad (5)$$

The resultant hash value is stored in A, B, C, and D buffers each of which is 32 bits. Therefore, the final hash value is computed by appending the buffers B, C, and D to the buffer A.

## 2.2 Properties of a Hash Function:

- (1) **Pre-image Resistance:** This property implies that a hash function cannot be inverted; that is, it is computationally very difficult (ideally, it must be impossible) to obtain the original message from the provided hash value. The more difficult it is to obtain the message, the higher is the pre-image resistance.
- (2) **Second Pre-image Resistance:** This property implies that given an input along with its hash value, it must be computationally very difficult to obtain another input which produces the same hash value. It ensures that attackers who want to replace an existing value with another input of the same hash value are unable to do so.
- (3) **Collision Resistance:** This property implies that for a hash function, it must be computationally very difficult to obtain two inputs which produce the same hash value. A hash function is defined to be a compression function that reduces the size of the input to a fixed length. Therefore, it is impossible to completely avoid collisions, but it must be extremely difficult for a particular hash function to do so.

## 2.3 SHA-256 Algorithm Structure

SHA or the secure hash algorithm is used as a cryptographic hash which is more secure than MD5 and widely used in digital signatures and password storage [5]. This hash function takes a string and outputs a hash value that is 256 bits long. The whole algorithm can be divided into four parts.

### 1. Insert Padding bits

The SHA-256 algorithm works on the block of 512 bits which is the standard length for this particular algorithm. The padding is done by adding some extra bits to the original message. The appending starts with 1 and the other bits following it are zero. The last 64 bits is left out of multiple of 512 which is filled in the next step.

The length of the original message is M, while the number of padding bits appended or added to the initial message is L. From the following equation, the number of padding bits can be calculated.

$$M + L + 64 = n \times 512 \tag{6}$$

### 2. Add Length Bits

The appending to the original message is done in the first step, and the remaining 64 bits are added in this step. The original message length multiplied by 8 (8 bit ASCII) is added in binary to the padded message. The message has a length that is a multiple of 512 bits. The big-endian convention used in the algorithm which indicates the left most bit is stored in the most significant bit position.

### 3. Buffer Initialization

The predefined values are initialized which is used in the algorithm that is implemented in the next step. The initialization includes eight hash values and 63 keys. The 64 key values are used in the 64 rounds of SHA-256 algorithm.

- Hash\_1 = 0x6a09e667
- Hash\_2 = 0xbb67ae85
- Hash\_3 = 0x3c6ef372
- Hash\_4 = 0xa54ff53a
- Hash\_5 = 0x510e527f
- Hash\_6 = 0x9b05688c
- Hash\_7 = 0x1f83d9ab
- Hash\_8 = 0x5be0cd19

Key[64] = {0x428a2f98, 0x71374491, 0xb5c0fbef, 0xe9b5dba5, 0x3956c25b,  
 0x59f111f1, 0x923f82a4, 0xab1c5ed5, 0xd807aa98, 0x12835b01, 0x243185be,  
 0x550c7dc3, 0x72be5d74, 0x80deb1fe, 0x9bdc06a7, 0xc19bf174, 0xe49b69c1,  
 0xefbe4786, 0x0fc19dc6, 0x240ca1cc, 0x2de92c6f, 0x4a7484aa, 0x5cb0a9dc,  
 0x76f988da, 0x983e5152, 0xa831c66d, 0xb00327c8, 0xbf597fc7, 0xc6e00bf3,  
 0xd5a79147, 0x06ca6351, 0x14292967, 0x27b70a85, 0x2e1b2138, 0x4d2c6dfc,  
 0x53380d13, 0x650a7354, 0x766a0abb, 0x81c2c92e, 0x92722c85, 0xa2bfe8a1,  
 0xa81a664b, 0xc24b8b70, 0xc76c51a3, 0xd192e819, 0xdf6990624, 0xf40e3585,  
 0x106aa070, 0x19a4c116, 0x1e376c08, 0x2748774c, 0x34b0bcb5, 0x391c0cb3,

0x4ed8aa4a, 0x5b9cca4f, 0x682e6ff3, 0x748f82ee, 0x78a5636f, 0x84c87814,  
 0x8cc70208, 0x90befffa, 0xa4506ceb, 0xbef9a3f7, 0xc67178f2}

The buffers are initialized as shown above along with the constant values which are used.

#### 4. Compression Algorithm

The padded message and the default predefined values are used in this step, and this is the important part of the hashing algorithm. The whole message is divided into “N” 512 bits blocks and passed the block into compression algorithm. The block of 512 bits undergoes a 64 round (Fig. 1).

The each round takes 32 bit words [i] and key [i] as input. For the first sixteen rounds, 512 bit block is divided into 16 blocks of 32 bits. These 16 blocks of 32 bits act as input for the first sixteen rounds. The words[i] for the remaining rounds are calculated using following formulae

$$W_j = \sigma_1(W_{j-2}) + W_{j-7} + \sigma_0(W_{j-15}) + W_{j-16} \tag{7}$$

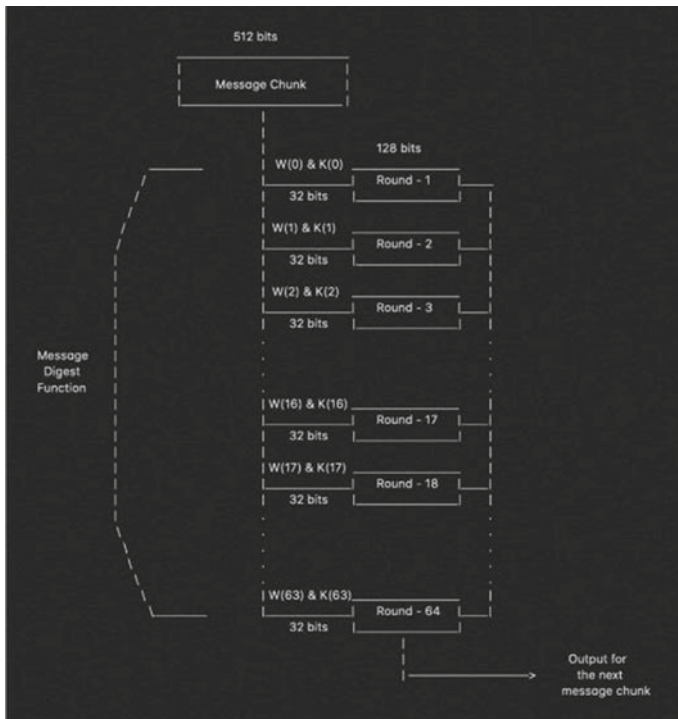


Fig. 1 Algorithm structure

Total of six logical functions are used in SHA-256 algorithm. The logical functions are defined below.

$$ch(x, y, z) = (x \& y) \wedge (\sim x \& z) \tag{8}$$

$$Maj(x, y, z) = (x \& y) \wedge (x \wedge z) \wedge (y \& z) \tag{9}$$

$$\sum_0(x) = S^2(x) \wedge S^{13} \wedge S^{22} \tag{10}$$

$$\sum_1(x) = S^6(x) \wedge S^{11} \wedge S^{25} \tag{11}$$

$$\sigma_0 = S^7 \wedge S^{18} \wedge R^3 \tag{12}$$

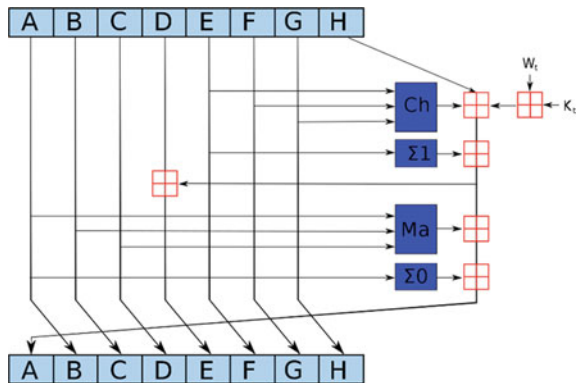
$$\sigma_1 = S^{17} \wedge S^{19} \wedge R^{10} \tag{13}$$

In Eqs. (10)–(13),  $S$  means rotate right  $x$  by  $n$  bits. The above functions are used in each of the 64 rounds, and it is also applied on “ $N$ ” 512 bits block. The below image shows the operations happening in each round (Fig. 2).

We get eight hash values by processing one 512 bits block, the result we obtained is added with the previous hash value, and it is given as input to the next round.

$$\begin{aligned} \text{Hash1}_i &= a + \text{Hash1}_{i-1} \\ \text{Hash2}_i &= a + \text{Hash2}_{i-1} \\ \text{Hash3}_i &= a + \text{Hash3}_{i-1} \\ &\vdots \\ \text{Hash8}_i &= a + \text{Hash8}_{i-1} \end{aligned}$$

**Fig. 2** Each round description. *Source* Adapted from [6]



The process of adding hash with the previous one keeps continuing till the last bit of the message. The result of the last round of the  $N$ th 512 bits message gives the result. The final result is 256 bits.

## ***2.4 A Comparison of SHA-256 and MD5***

- (1) MD5 is faster to compute when compared with SHA-256 which is much slower.
- (2) MD5 checksums or hash values are more likely to suffer from collision when compared to SHA-256 hash values. Therefore, it is easier to find messages with the same checksum in MD5 than in SHA-256.
- (3) The hash value of MD5 is smaller (128 bits) as compared to SHA-256 (256 bits).

It is therefore obvious that MD5 is less secure but computationally faster when compared to SHA-256 which is more secure but slower. In order to make MD5 more secure, we made use of SHA-256 which computes a 256 bit hash value and that is hashed again to produce a 128 bit hash value to produce an MD5 checksum.

## **3 Multiple Hashing Using MD5 and SHA-256**

### ***3.1 Security Issues of Hash Functions:***

Hash functions such as MD5 are more prone to collision attacks, where an attacker can easily find another file which has the same hash value as the original file and substitute it in its place. MD5's vulnerabilities were exposed by Wang who found pairs of 1024 bit messages which produced the same hash value thus resulting in collisions. Hence, MD5 was vulnerable to collisions, but it can be used to detect unintentional corruption of data while downloading files. However, MD5 is a fast hash which can be employed to store files and download files from a server but not useful to store data that is sensitive to attacks such as passwords. Usually, passwords are stored as their hash values instead of the regular strings since if the passwords are stolen, the attacker will be left with the hash values, and the resistance of a hash prevents the attacker from obtaining the password.

Dictionary attack is another weakness of hashing algorithms. Here, the attacker might try the brute force approach, where he has the hash value of every word in a dictionary and compares it with the hash value stored in order to obtain the original text.

A rainbow table attack is another form of attack on a server which stores sensitive data. It is a pre-computed dictionary of passwords in plain text along with their hash value which is used to obtain the original message with ease.



### 3.2 *Multiple Hashing*

Using a single hash function like MD5 to obtain the hash value of a file can be effective but it compromises security making it easier for an attacker to predict the contents of the file. Therefore, instead of MD5 hashing the contents of the file directly, we propose MD5 hashing the SHA hash value of the contents of the file which may make it more secure as MD5 is less secure as compared to SHA-256. This way, we obtain a hash value which is 128 bits and also secure due to SHA-256.

The resulting hash value is shown in the equation below:

$$\text{MD5}(\text{SHA}(\text{Message})) = \text{ResultingHashValue}$$

Multiple hashing may or may not make the hash function more secure which is why we have limited it to two hash functions, namely SHA and MD5. The gain in security is a question for cryptographers but running the contents of the file through two hash functions which improves the security compared to a scenario, where we run the contents through a single MD5 hash function. MD5 (message) is less secure than MD5( SHA-256(message)). However, multiple hashing and the security it adds are still questionable as we cannot guarantee how much security multiple hashing adds to a system; however, it makes it slightly more secure provided we use the correct hash functions.

Kerckoff's principle states that a cryptosystem must be stable even if the working of the system is known. Therefore, we must assume that any attacker is aware of the method of hashing used for our files before making it secure. One must be cautious while running multiple hashes consecutively as running multiple hashes means that the final hash is only as strong as the weakest hash function used. Also, the intricate and complex working of these hash functions may result in an overall weaker hash function due to the complex interactions between them. Hence, we have restricted the usage to just two multiple hash functions, that is, SHA-256 and MD5. Although the gain in security is less, it is more secure than using a single MD5 hash function on the file. Running it on multiple iterations, that is, hashing it hundreds of times or more does not guarantee an increase in security, but it adds complexity to the code. Therefore, we have restricted it to just one iteration of hashing. Also, since the hash is only as strong as the weakest hash function, if the attacker breaks the weakest hash function, it becomes easier to obtain the original text. Thus, we use the equation shown below:

$$\text{MD5}(\text{SHA}(\text{Message})) = \text{ResultingHashValue}$$

Now, even if the attacker breaks SHA-256, he has to still know what "message" is in order to successfully guess what the message is and break MD5.

### 3.3 Addition of Salt

While storing passwords in a database, it is a common practice to add a randomly generated string to the original password which is called as salt before hashing it and storing it in the database. We can extend this for file applications as well by adding a random set of characters before hashing it.

Salt is nothing but a string that is added with the contents of a file before hashing it with a hash function. Before uploading a file on a server, salt is added to the contents of the file and hashed. When a person downloads the file, he can verify its contents by adding the same salt to the file and hash it before comparing the two hash values to check if they are equal. If they are equal, the data has not been tampered with or corrupted while downloading. The salt added protects the contents of the file from dictionary attacks. Since files have a large amount of content present within them, dictionary attacks are less likely to cause problems but if they are smaller files, salt plays a crucial role in safeguarding the contents of the file from being revealed from its hash value.

Salts can be appended to the message/string, inserted in the beginning instead or both. It is crucial to ensure that the salt chosen must be random and have no correlation with the contents of the file. Thus, the equation for the resultant hash that we generated is:

$$\text{MD5}(\text{SHA}(\text{Message})) = \text{ResultingHashValue}$$

Here, salt1 is appended to the message before calculating the SHA-256 value, and salt2 is added to the hash value before calculating the MD5 value of the resulting string. This is a normal practice in password storing but we have extended it to hashing of files as well. The salt that is used need not be encrypted or hashed since it is already a random set of characters. Creating a cryptographically strong salt is a cryptographers task, hence for the purpose of demonstration, we have randomly chosen two salts of random characters.

## 4 Improvement from Traditional Methods

Traditional hashing techniques employ one powerful and unbreakable hashing algorithm which can protect the hashing algorithm's properties pre-image resistance and collision resistance. However, we believe that hashing a file twice by adding an appropriate salt and pepper to it will enhance its security and make it more collision resistant and more pre-image resistant. SHA-256 eliminates the collision attack weakness that MD5 exhibits while the advantage of obtaining a smaller number of bytes in the output is also retained by using MD5. Hence, the purpose of hashing it with SHA-256 is to improve security, while the purpose of MD5 is to reduce the number of bytes in the message digest. This is the reason behind using the two

algorithms. It is also worth mentioning that repeated hashing might destroy the security that hashing algorithms provide. Every hashing algorithm diffuses the message as much as possible. If we use multiple hashing algorithms, repeated diffusion of the input message might destroy the algorithm's security. Hence, we have restricted ourselves to only using two hashing algorithms, namely MD5 and SHA 256. Using more algorithms, more number of times affects the performance and the diffusion property of each algorithm.

## 5 Experimental Results

A function which returns the MD5 hash value was developed from scratch by using the "stdint.h" library file in C. C provides a wide range of data types of specific sizes and is much faster when compared to languages such as Python.

Therefore, C was chosen over other languages as it is more preferred for cryptography.

### 5.1 Hashing the File Twice

A simple text file with a random text is chosen as the input. The text file contains the string "MULTIPLE HASHING WITH MD5 AND SHA-256."

A random salt "AHYFRTU" is added, and the same salt is used for both the stages. (i.e., salt1 = salt2 = AHYFRTU").

SHA-256(text+salt1) is found to be (Fig. 3):

```
f684f0822a46c178d6f2279a4533664bf6f9da6e4f6708af843146b419c66eea
```

MD5(SHA-256(text+salt1)+text+salt2) is found to be:

```
90b2ae9d3ecf940ac7e4beba8930d62e
```

### 5.2 Uploading the Files

In order to test our hash function, the original file was uploaded onto Google drive along with another file that contained the twice-hashed value of the file. A small python program uploads the two files onto Google drive using the API provided by Google. In order to distinguish the originally uploaded file from the file containing the hash value, a suffix ("md") was added to the file name.

```

gautham@gautham-HP-Pavillion-Notebook-15-bc5xxx:~/Desktop$ ./a.out
Enter the filename to be opened
text.txt
The contents of the file are:
MULTIPLE HASHING WITH MD5 AND SHA256.
The contents of the file with salt:
MULTIPLE HASHING WITH MD5 AND SHA256.
AHYFRT
SHA(text+salt)+salt+text=
f684f0822a46c178d6f2279a4533664bf6f9da6e4f6708af843146b419c66eeaAHYFRTMULTIPLE H
ASHING WITH MD5 AND SHA256.
AHYFRT
SHA value is
f684f0822a46c178d6f2279a4533664bf6f9da6e4f6708af843146b419c66eea
double hashing output :
90b2ae9d3ecf940ac7e4beba8930d62egautham@gautham-HP-Pavillion-Notebook-15-
~/Desktop$

```

Fig. 3 File hashing

### 5.3 Downloading the Files

The two files can be downloaded using the file ID of both the files. Before this, you must ensure that you have permission to download the file. Two files must be created without any contents in order to dump the contents of the files from Google Drive. Thus, we now have two files, one which contains the original contents of the uploaded file and another which contains the hash value of the originally uploaded file.

### 5.4 Checking the Integrity of the Downloaded Files

The downloaded file is passed through the same hash function that we used prior to uploading. If the data has not been tampered with or corrupted, it should produce the same hash value which was produced before uploading. Thus, we are able to check the integrity of a file (Fig. 4).

```

Enter the filename to be checked
result.txt
Uploaded file message digest=90b2ae9d3ecf940ac7e4beba8930d62e
Downloaded file message digest=90b2ae9d3ecf940ac7e4beba8930d62e
the data has been verified. No corruption
gautham@gautham-HP-Pavillion-Notebook-15
-bc5xxx:~/Desktop$

```

Fig. 4 Checking the integrity of the file

## 6 Conclusion and Future Work

It is evident that MD5 is insecure as a hashing function but it is not impossible to make it more secure with the aid of simple tasks added to the hashing process such as addition of salt or multiple hashing, and we can drastically improve the hashing function's security.

To improve this hashing procedure further in the future, we can add a variable salt instead of a constant salt. In our program, a constant salt was added before hashing the string in order to avoid complications. However, a variable salt whose length is random makes the hashing function more secure.

The next improvement can be the addition of pepper along with salt. However, it must be noted that the same pepper must be used on both the sender and receiver sides to obtain the same hash value while transferring a file. Further, the hashing function can be extended to a cloud storage platform.

**Acknowledgements** We would like to thank PES University for providing guidance and encouraging us in our work. Special thanks to ISFCR for providing the necessary insights which we needed during this project.

## References

1. L. Jie, Improved collision attack on MD5. J. Comput. Sci. Technol. ACM DL digital library (2007)
2. P. Gauravaram, Security analysis of salt|password hashes, in *2012 International Conference on Advanced Computer Science Applications and Technologies (ACSAT)*, Kuala Lumpur (2012), pp. 25–30. <https://doi.org/10.1109/ACSAT.2012.49>.
3. W. Stallings, *Cryptography and Network Security: Principles and Practice* (Tsinghua University Press, Beijing, 2002).
4. R.L. Rivest, *The MD5 Message Digest Algorithm [EB/OL]* (2005)
5. Secure Hash Standard (SHS), N. I. of Standards and Technology (2012)
6. A. Anand, Breaking Down:SHA-256 algorithm (2019). [Online]. Available <https://medium.com/bugbountywriteup/breaking-down-sha-256-algorithm-2ce61d86f7a3>. Accessed: 29 Jun 2020

# Design and Analysis of a Secure Coded Communication System Using Chaotic Encryption and Turbo Product Code Decoder



S. Khavya, Karthi Balasubramanian, B. Yamuna, and Deepak Mishra

**Abstract** Errors in a transmitted message is unavoidable since noise is inevitable in any communication channel. For reliable transmission of messages, the bit error rate has to be kept at an acceptable rate by the use of proper error control coding schemes. To ensure that the transmission is also secure, data encryption is used as an integral part of the system. This paper deals with the design and analysis of a secure and reliable communication system accomplished using logistic map-based chaotic encryption and turbo product codes. The system is simulated using MATLAB and it is shown that the use of encryption for secure communication does not degrade the system performance. The hardware design of the decoder is also done and verified in Verilog using the same set of vectors as obtained from the system simulation. BER performance was analyzed in all the different scenarios and the correctness of the design was established.

**Keywords** Chaotic encryption · Logistic map · Turbo product codes · Chase-Pyndiah decoding

## 1 Introduction

Combining encryption with error control coding enables us to transmit information in a secure and reliable manner. Chaotic pseudorandom number generation is a technique that uses mathematical equations with a seed value and is iterated multiple times, to generate a key sequence. This randomly generated key sequence is used

---

S. Khavya · K. Balasubramanian (✉) · B. Yamuna  
Department of Electronics and Communication Engineering, Amrita School of Engineering,  
Amrita Vishwa Vidyapeetham, Coimbatore, India  
e-mail: [b\\_karthi@cb.amrita.edu](mailto:b_karthi@cb.amrita.edu)

D. Mishra  
Digital Communication Division (DCD), Space Application Center (SAC),  
ISRO, Ahmedabad, India  
e-mail: [deepakmishra@sac.isro.gov.in](mailto:deepakmishra@sac.isro.gov.in)

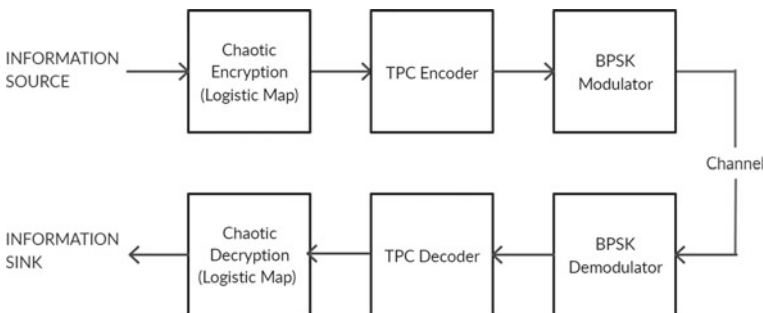
© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2021  
S. M. Thampi et al. (eds.), *Advances in Computing and Network Communications*,  
Lecture Notes in Electrical Engineering 735,  
[https://doi.org/10.1007/978-981-33-6977-1\\_48](https://doi.org/10.1007/978-981-33-6977-1_48)

for encrypting the original information. Among various encryption methods, chaotic encryption that is based on symmetric key cryptography has gained significance due to its dynamic nature [1, 2]. Various maps like logistic map, Arnold cat map, cubic map, Baker's map, and tent map are used for chaotic data generation. Out of these, logistic map has been widely used due to its simplicity and its ability to provide a high level of security [3]. The use of logistic map for symmetric key cryptography for secure communication is brought out in [4] where it is shown that this method is more secure than the commonly used DES algorithm.

Reliable data transmission is facilitated by an efficient error control coding scheme [5, 6]. One such widely used code is turbo product code (TPC) that provides a flexibility in the selection of the component codes. Iterative soft input soft output Chase-Pyndiah decoding algorithm is widely used to decode turbo product codes [5, 7]. Simplified versions of the widely used Chase-Pyndiah decoding algorithm have been investigated and reported in literature [8]. Hardware implementation of the decoder has also been explored in detail by different researchers. Ahn et al. in [9] propose an architecture for high-speed TPC decoding. Krainyk et al. in [10] propose a low complex hardware realization of a TPC decoder. In [11, 12], the implementation of a fully parallel turbo decoding architecture with reduced latency for product codes has been proposed. A high throughput design and its FPGA implementation of a TPC decoder for fiber optics and satellite communication applications is proposed in [13].

The use of chaotic encryption along with TPC for secure and reliable communication is a field that is less explored. Chaware et al. in [14] show the design and MATLAB implementation of a communication system with chaotic encryption using logistic map and TPC encoding. The block diagram of the same is reproduced and shown in Fig. 1.

This work is aimed at a hardware design of the decoder to be used with secure data obtained from the chaotic encryption. The hardware design and simulation of a TPC decoder using the soft input soft output Chase-Pyndiah algorithm is done. A secure coded system is also simulated using chaotic encryption based on logistic map and the performance is analyzed.



**Fig. 1** Block diagram of communication system with chaotic encryption and TPC encoding [14]

The paper is arranged as follows. Section 2 summarizes turbo product code encoding, decoding, and the chaotic encryption techniques. The hardware realization of the TPC decoder is discussed in Sect. 3. Section 4 deals with the results and analysis and the paper concludes in Sect. 5.

## 2 Chaotic Encryption, TPC Encoder, and TPC Decoder

### 2.1 Chaotic Encryption Using Logistic Map

Using the paradigm of a chaotic system for encryption is getting wide acceptance nowadays [15]. Chaos is a subtle behavior displayed by certain nonlinear systems. The behavior looks random but it is not stochastic in nature and results from a deterministic process. One of the key aspects of a chaotic system is its high sensitivity to the initial conditions of the system [16].

Logistic map is a nonlinear system described by  $X_{n+1} = rX_n(1 - X_n)$  where  $r$  is a parameter that decides the nature of the system. Selection of  $r = 3.999$  produces highly chaotic sequences. Starting with an initial value of  $X_n$  and iterating it multiple number of times, random patterns are produced, which can be used as keys for encrypting the information to be transmitted [16, 17]. In this work, an initial value  $X_0$  of 0.6543 and  $r = 3.999$  is chosen as the model parameters.

### 2.2 TPC Encoder

TPC encoding uses two symmetric linear block codes  $C_1(n_1, k_1, d_1)$  and  $C_2(n_2, k_2, d_2)$  where  $k_1$  and  $k_2$  represents the number of information bits,  $n_1$  and  $n_2$  represents the codeword length, and  $d$  represents the minimum hamming distance. The construction of turbo product code  $P = C_1 \oplus C_2$  is shown in Fig. 2. The TPC encoding involves three steps[18]:

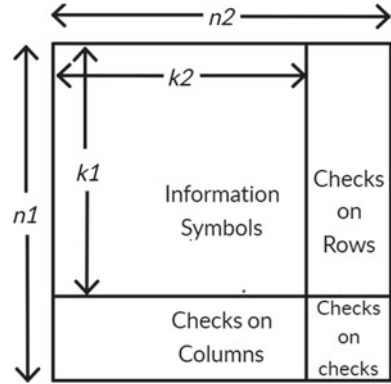
- $k_1 \times k_2$  information bits are placed along  $k_1$  rows and  $k_2$  columns.
- Using code  $C_2(n_2, k_2, d_2)$ ,  $k_1$  rows are encoded.
- Using code  $C_1(n_1, k_1, d_1)$ ,  $k_2$  columns are encoded to obtain the encoded matrix of dimension  $[n_1 \times n_2]$ .

### 2.3 Chase-Pyndiah Decoding Algorithm

$R = (r_1, r_2, \dots, r_n)$  represents the received value at the input of the decoder. The Chase-Pyndiah decoding follows the steps below:



**Fig. 2** Construction of turbo product code  $P$  [18]



1. The positions of  $p = \lfloor d/2 \rfloor$  least reliable bits are determined.
2. The test pattern set  $T^q (q \in [1, 2^p])$  is generated.
3. Test sequence  $Z^q$  is formed by performing modulo-2 addition between the test pattern  $T^q$  and the hard decision sequence  $Y$ .

$$Z^q = Y \oplus T^q \tag{1}$$

4. Syndrome decoding is performed with  $Z^q$  using (2).

$$S = \text{syn}(Z_q) = Z_q * H^T \tag{2}$$

where  $H^T$  represents the transpose of the parity check matrix  $H$ .  $S \neq 0$  represents the presence of errors. With the error vector  $e$  and the corresponding syndrome value  $S$  from the syndrome table, the bit error is corrected as given in (3).

$$C = R - e \tag{3}$$

5. The squared Euclidean distance between  $R$  and  $C_i$  is found using (4).

$$|R - C^i|^2 = \sum_{l=1}^n (r_l - c_l^i)^2 \tag{4}$$

The codeword  $C$  with the minimum squared Euclidean distance forms the candidate codeword  $D$  and is given by (5).

$$D = C^i \text{ if } |R - C^i|^2 \leq |R - C^l|^2 \quad \forall l \in [1, 2^k], l \neq i \tag{5}$$

where  $C^i = (c_1^i, c_2^i, \dots, c_n^i)$  is the  $i$ th codeword of  $C$ .

- The received value  $R$  is updated using the extrinsic information  $w_j$  and the candidate codeword  $D$  as given in (6) and (7) where  $\alpha$  and  $\beta$  represent the weighing and reliability factors, respectively.

$$w_j = (\beta \times D) - R \tag{6}$$

$$R = R + (\alpha \times w_j) \tag{7}$$

### 3 TPC Decoder Architecture

The block diagram of the TPC decoder architecture is given in Fig. 3.

The input data is stored in contiguous memory locations as floating point numbers represented using IEEE 754 standard. The values of each row/column are passed on to the TPC decoder from the memory. The least reliable bit calculation (LRB) module sorts the incoming data and identifies the two least reliable bit positions. These positions are passed to the test pattern generation module for generating  $2^p$  test patterns. Candidate codewords are then generated using the test patterns and the hard values of the original inputs. The candidate codewords are then used by the syndrome decoding module to generate the syndrome. The Euclidean distance between the input hard values and the candidate codewords is calculated and the candidate code word that gives the minimum distance is considered as the decoded code word. The extrinsic information is then updated using the decoded code word, the reliability factor  $\beta$ , and the weighing factor  $\alpha$ , and the memory is updated with the new value.

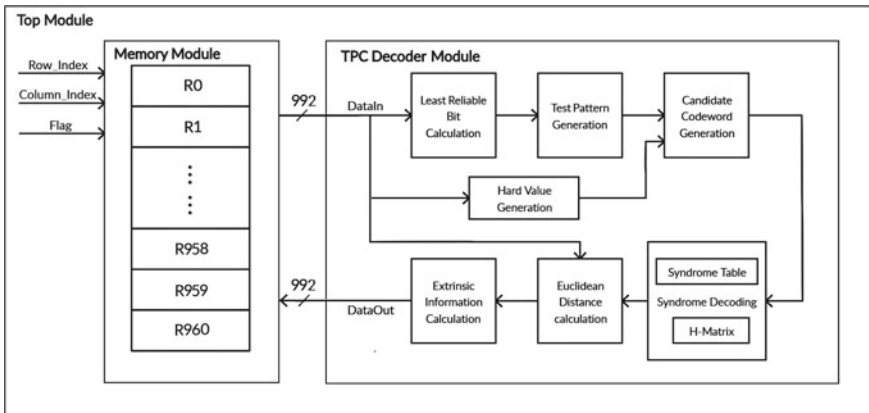


Fig. 3 Block diagram of TPC decoder

### 3.1 Memory Organization

The size of memory used is dependent on the component codes of the TPC. Here,  $BCH(31, 21, 5)^2$  component code is used. The encoded output from the TPC encoder is considered as a matrix of dimension  $[31 \times 31]$  and is stored in memory as shown in Fig. 4.

The number of TPC encoded values is 961 ( $31 * 31 = 961$ ) and hence 961 memory locations are required. Each memory location holds a floating point number represented using 32 bit IEEE 754 format. During each clock cycle, one entire row/column (31 float values) is passed from the memory module to the TPC decoder module to perform decoding operation. With each float value being represented by 32 bits, a total of  $31 * 32 = 992$  bits of data is used by the decoder module for processing; Fig. 5 shows the same. After each row decoding, the decoded values are updated in the corresponding memory locations.

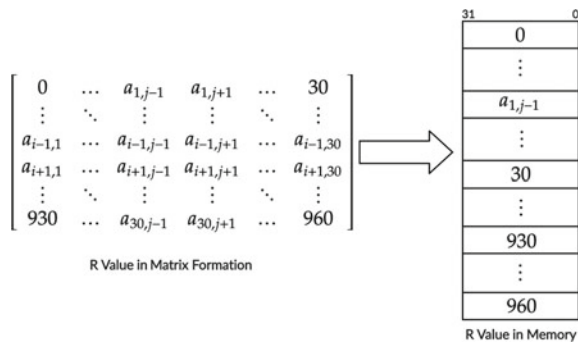


Fig. 4 Memory structure to store the received inputs (R) that are represented using IEEE 754 floating point representation

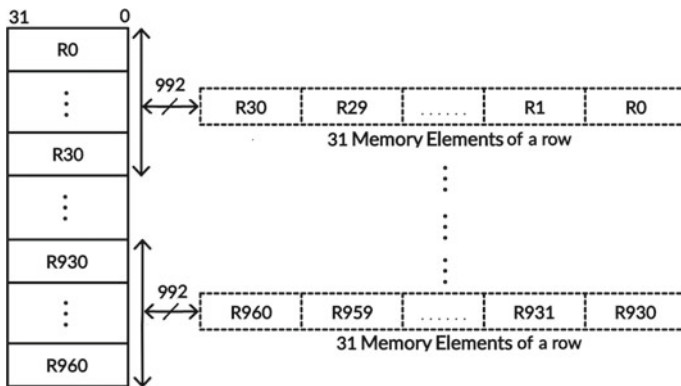
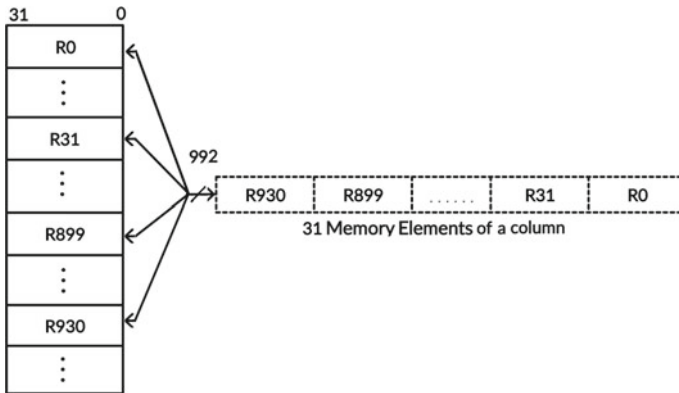


Fig. 5 Accessing and updating memory elements during row decoding



**Fig. 6** Accessing and updating memory elements during column decoding

For performing column decoding operations, the data from each column is chosen. From the memory perspective, it amounts to choosing locations that are separated by 31 float values. Figure 6 shows how 32 bit float data is being fetched from memory locations 0, 31, 62..930 to constitute the 992 bits needed by the decoder module for analysis.

### 4 Simulation Results and Analysis

For chaotic encryption, the number of information bits considered is 441. This is represented in matrix format of dimension  $[21 \times 21]$ . The logistic map equation is iterated twenty one times to generate twenty one logistic map key values. Encryption is done by performing bitwise XOR operation between the original information and the generated logistic map key values. This encrypted data is encoded using  $BCH(31, 21, 5)^2$  TPC. This encoded information is modulated with BPSK modulation and is passed through the AWGN channel. The encoded message which is generated using MATLAB setup is given as input to the simulation setup in Verilog also [19]. Figure 7 shows the comparative performance analysis of the Chase-Pyndiah algorithm with and without the chaotic encrypted data. It can be seen that there is no performance degradation even with encrypted data.

Figure 8 shows the Verilog and MATLAB simulation results of the TPC decoder with chaotic encrypted data. It is found that the BER performance computed with the decrypted data obtained with Verilog simulation coincides with that of the MATLAB performance, thus validating the correctness of the hardware design.

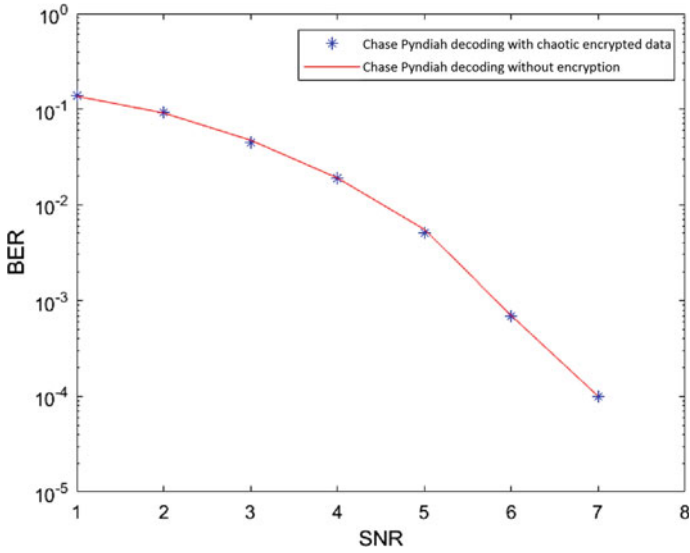


Fig. 7 BER plots of Chase-Pyndiah decoder with and without chaotic encryption showing negligible performance degradation between the two

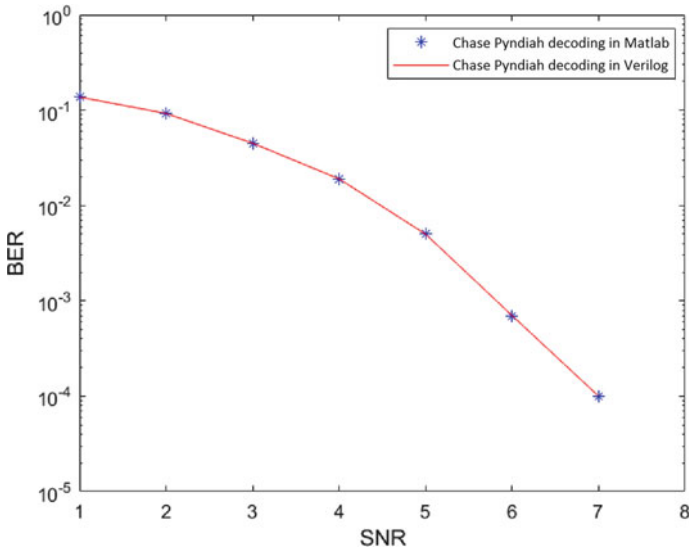


Fig. 8 BER plots of Chase-Pyndiah decoder with chaotic encryption in MATLAB and Verilog showing the correctness of the hardware design

## 5 Conclusions

The design and simulation of a secure coded communication system with turbo product code and logistic map-based chaotic encryption is presented in this work. The hardware design of the decoder is verified using the data generated from the MATLAB simulations. This work can be extended to include the encryption and decryption blocks also in hardware and analyze the performance with a board level implementation.

## References

1. N. Nagaraj, One-time pad as a nonlinear dynamical system. *Commun. Nonlinear Sci. Numer. Simul.* **17**(11), 4029–4036 (2012)
2. X. Wu, H. Hu, B. Zhang, Analyzing and improving a chaotic encryption method. *Chaos Solitons Fractals* **22**(2), 367–373 (2004)
3. T. Yang, A survey of chaotic secure communication systems. *Int. J. Comput. Cogn.* **2**(2), 81–130 (2004)
4. R. Bose, A. Banerjee, Implementing symmetric cryptography using chaos functions, in *Proceedings of the 7th International Conference on Advanced Computing and Communications* (Citeseer, 1999), pp. 318–321
5. R.M. Pyndiah, Near-optimum decoding of product codes: block turbo codes. *IEEE Trans. Commun.* **46**(8), 1003–1010 (1998)
6. K. Eluri, B. Yamuna, K. Balasubramanian, D. Mishra, Low power and area efficient max-log-map decoder, in *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)* (IEEE, 2018), pp. 431–435
7. S.N. Vaniya, N. Kumar, C. Sacchi, Performance of iterative turbo coding with nonlinearly distorted ofdm signal, in *IEEE Annual India Conference (INDICON)* (IEEE, 2016)
8. W. Kuang, R. Zhao, Z. Juan, FPGA implementation of a modified turbo product code decoder, in *2017 IEEE 9th International Conference on Communication Software and Networks (ICCSN)* (IEEE, 2017), pp. 71–74
9. B. Ahn, S. Yoon, J. Heo, Low complexity syndrome-based decoding algorithm applied to block turbo codes. *IEEE Access* **6**, 26 693–26 706 (2018)
10. Y. Krainyk, V. Perov, M. Musiyenko, Y. Davydenko, Hardware-oriented turbo-product codes decoder architecture, in *2017 9th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*, vol. 1 (IEEE, 2017), pp. 151–154
11. C. Jego, P. Adde, C. Leroux, Full-parallel architecture for turbo decoding of product codes. *Electron. Lett.* **42**(18), 1052–1054 (2006)
12. M. El Haroussi, I. Chana, M. Belkasm, VHDL design and FPGA implementation of a fully parallel BCH SISO decoder, in *2010 5th International Symposium on I/V Communications and Mobile Network* (IEEE, 2010), pp. 1–4
13. C. Leroux, C. Jégo, P. Adde, M. Jézéquel, High-throughput block turbo decoding: from full-parallel architecture to FPGA prototyping. *J. Signal Process. Syst.* **57**(3), 349–361 (2009)
14. T.S. Chaware, B. Mishra, Secure communication using TPC and chaotic encryption, in *2015 International Conference on Information Processing (ICIP)* (IEEE, 2015), pp. 615–620
15. L. Kocarev, Chaos-based cryptography: a brief overview. *IEEE Circ. Syst. Mag.* **1**(3), 6–21 (2001)
16. K.T. Alligood, T.D. Sauer, J.A. Yorke, *Chaos* (Springer, Berlin, 1996)

17. M. Ausloos, M. Dirickx, *The logistic map and the route to chaos: From the beginnings to modern applications* (Springer Science & Business Media, Berlin, 2006)
18. P. Mathew, L. Augustine, T. Devis, et al., Hardware implementation of (63, 51) bch encoder and decoder for WBAN using IFSR and BMA
19. A. Ambat, K. Balasubramanian, B. Yamuna, D. Mishra, FPGA implementation of an efficient high speed max-log-map decoder, in *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)* (IEEE, 2018), pp. 747–751

# Digital Image Transmission Using Combination of DWT-DCT Watermarking and AES Technique



Sudhanshu S. Gonge

**Abstract** The Internet technology brings big revolution in twenty-first century. It facilitates communication between man-to-man, man-to-machine, machine-to-machine, and vice versa. There are many applications, viz. (i) vehicle-to-vehicle, (ii) vehicle-to-infrastructure, (iii) drone communication, etc., which transmit and receive data in various formats like image data, audio data, video data, text data, etc. Bank system uses data communication technique for transferring money through debit card, net banking, credit card, demand draft, cheques, RTGS, NEFT, etc. Bank cheques are cleared through CTS system. For clearing the cheque, it is scanned, and image is transferred to cheque clearing house. During cheque image transmission, there is a need of security, confidentiality, integrity, authorization, copyright protection, and indexing services. There are many techniques and algorithm which provide this facility to overcome this issue. To solve this issue, combination of DWT-DCT watermarking along with AES technique is used. Its performance and analysis against various attacks are also explained in this research paper.

**Keywords** DCT · DWT · AES · Digital watermarking · Decryption · Attacks

## 1 Introduction

Data transmission takes place based on networks connected between minimum two systems. During transmission, data is transferred in various formats based on the type of application. There are different processes which are being carried out on data before its transmission and reception, such as encoding and decoding of data [1–5]. There are variety of encoding and decoding schemes which are used to convert data into American Standard Code for Information Interchange (ASCII) for text files.

However, digital images are processed by binding image pixels for its characterization of features. The image encoder has function that are being implemented on image for compression, altering image pixels, i.e. changing the position of pixels, and

---

S. S. Gonge (✉)

Department of Computer Engineering, Vishwakarma Institute of Technology, Pune, India  
e-mail: [sudhanshu1984gonge@rediffmail.com](mailto:sudhanshu1984gonge@rediffmail.com)

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2021  
S. M. Thampi et al. (eds.), *Advances in Computing and Network Communications*,  
Lecture Notes in Electrical Engineering 735,  
[https://doi.org/10.1007/978-981-33-6977-1\\_49](https://doi.org/10.1007/978-981-33-6977-1_49)

667



encryption. There are different classical encryption techniques used for encryption as shown in Fig. 1 [1, 2, 6].

Further, these classifications were enhanced into modern encryption techniques. These techniques were developed based on secret-key and public-key cryptographies. It is shown in Fig. 2.

There are eight strong data encryption techniques, viz. (i) Triple Data Encryption Standard (Triple DES), (ii) Blowfish Encryption Algorithm, (iii) Advanced Encryption Standard, (iv) IDEA Encryption Algorithm, (v) Twofish Encryption Algorithm, (vi) RSA Encryption Algorithm, (vii) HMAC Encryption Algorithm, and (viii) MD-5 Encryption Algorithm. However, there are also various algorithms like cast-128, RC-4, RC-5, elliptical curve encryption algorithms, etc., used for providing security services [1, 2, 6–8]. There are different types of attacks which are being applied by intruders intentionally or unintentionally [9–21]. These encryption techniques and algorithms provide confidentiality, integrity, and authenticity to data, which prevent the illegal access of information. To facilitate the service of copyright protection,

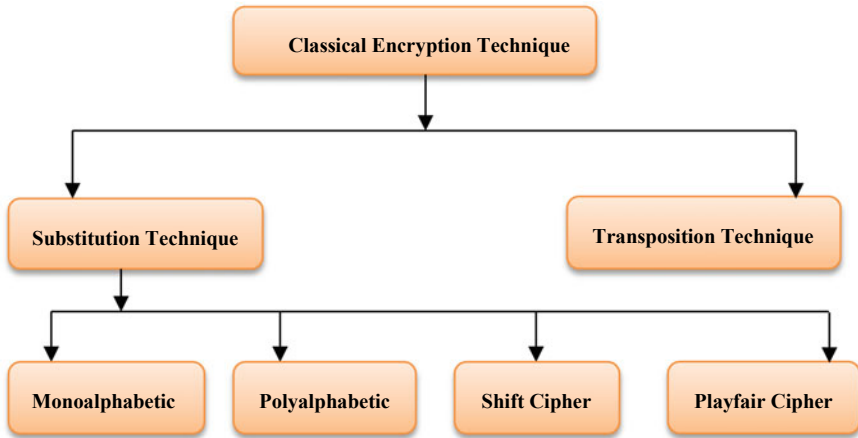


Fig. 1 Classical encryption technique

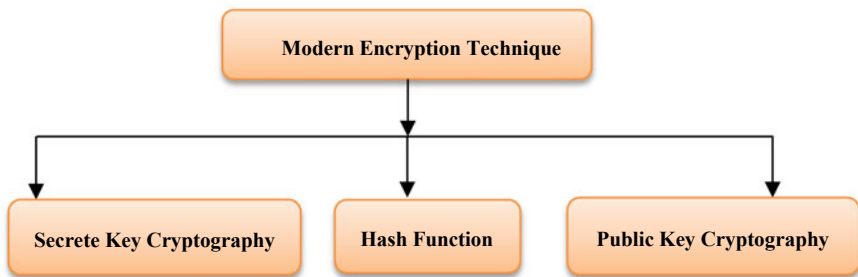


Fig. 2 Classification of modern cryptography

there is requirement of digital watermarking technique [18–23]. It can be implemented using frequency domain spatial domain, computational intelligence technique, and many more. In this research paper, combination of DWT-DCT image watermarking and AES technique are used to provide copyright protection, indexing, confidentiality, integrity, authorization for bank cheque images. The evaluation of this method is done by calculating peak signal-to-noise ratio, robustness of watermark, mean square error, and elapsed time against various types of attacks, viz. (i) cropping attack, (ii) Gaussian blur attack, (iii) JPEG compression attack, (iv) median filtering attack, (v) rotation attack, (vi) salt and pepper noise attack, and (vii) under normal mode attack.

## 2 Literature Survey

The security of digital content is an important factor [1, 2]. The AES is one of the robust encryption techniques, which was invented in 2001 [2]. Cox et al. [3] explain the role of digital watermarking for audio data and video materials by using fingerprint technology. Wang et al. [4] describe the watermarking technique for digital image by quantizing wavelet coefficient in sparse tree using invisible watermarking technique. There are various frequency domain techniques used by many researchers, viz. (i) DCT, (ii) DFT, (iii) K-L transform, (iv) DWT, (v) Gabor transform, etc. Tsai et al. [5] use the subsampling technique for the image watermarking using DCT and DWT in frequency domain. There are variety of application based on data transmission.

The security techniques for the payment through mobile device based on audio watermarking for cheque were being proposed and implemented in 2005. However, the robustness and security were not found good as per the system's requirement [9]. Al-Haj et al. [10] explain the fusion technique by using two different frequency transforms, which can be used for digital image watermarking. Wang et al. [11] proposed image encryption technique for security using DWT and chaos method. There are many applications explained on this method [12]. Based on various combined transform techniques for image copyright protection, robustness and enhancement in the digital image watermarking was explained and applied. Kothari et al. [13] perform an experiment on image watermarking using combined DWT-DCT to show its performance over DWT. During transmission of watermarked image through channel, there are many attacks which may be applied intentionally or unintentionally by user [14].

EI-Mohades et al. [15] explain image protection using hybrid DWT-DCT watermarking for providing relationship between IDEA encryption key and watermark used for encryption and image watermarking. Many researchers have also worked in spatial domain image watermarking. Amini et al. [16] explain the combination of spatial segmentation and wavelet packet frequency division technique used for improving normalized correlation against JPEG and rotation attack. Subramanayam et al. [17] describe the working of compressed and encrypted image watermarking used for storing data in various database management systems. Biometric application

was developed with the help of face recognition and watermarking technique using discrete cosine transform [18].

Mahajan et al. [19] show the comparison between AES, RSA, and DES algorithms, based on the data size and time taken for encryption and decryption. Metkar et al. [20] explain the image watermarking based on rational dither modulation method and encryption using AES and RC4 technique for the improvement of image security. Sirmour et al. [21] describe the working of hybrid DWT-SVD watermarking and its comparison with individual DWT watermarking scheme. Lee et al. [22] deal with visual secret sharing of image with hidden secret image. It also describes cryptography technique implemented using feature extraction process. It also shows the comparison between conventional visual secret sharing, natural image-based visual secret sharing algorithm, and extended visual cryptography scheme [22]. Nambuttee et al. [23] deal with the medical application, which describes about encrypted image technique that is used to store the patient records like X-ray and medicine prescriptions in DBMS. To access these records from the system, scrambling algorithm is used for accessing the image.

However, researcher also explains about DCT-DWT image watermarking for authentication [23]. The barcode system is used to improve the robustness and imperceptibility of the watermarked image [23]. Tayal et al. [7] explain various encryption algorithms in survey. It also deals with visual representation of data, analysis of key space, and its effect on data size with change in key along with correlation coefficient for different encryption algorithms. Kumari et al. [6] deal with encryption comparative study of AES and RC4 along with chaos function used for encryption.

Saikumar et al. [8] describe the working of LSB and DWT technique used for embedding the watermark along with Huffman coding technique for 3-D image. Many researchers explain different schemes and techniques regarding watermarking of image and its security by applying individual algorithms of watermarking or encryption and decryption. But, there is a need to provide both security and copyright protections simultaneously to digital image against attack.

In this paper, a fusion of DWT and DCT is done for digital image watermarking along with 256-bit key AES technique for providing the security to digital bank cheque image against different varieties of attacks discussed in result and discussion section.

### 3 Proposed Algorithm

The algorithm for security and copyright protection used in this work is classified into two sub parts. Part A in Sect. 3.1 describes about digital watermark embedding process and encryption technique, whereas Part B in Sect. 3.2 tells about decryption process carried out on image and watermark extraction.

### 3.1 Part A: Embedding and Encryption Process

- (a) Read bank cheque image.  
 (b) Apply the 2-D DWT on bank cheque image using Eq. 1.

$$\Psi_{a, b_x, b_y}(x, y) = \frac{1}{|a|} \Psi\left(\frac{x - b_x}{a}, \frac{y - b_y}{a}\right) \quad (1)$$

where

$\Psi$  is basis function of wavelet transform,  
 $a$  is scaling factor,  
 $b$  is shifting coefficient,  
 $x$  and  $y$  are image pixels.

- (c) Divide image into four sub-bands, i.e. approximate sub-band, horizontal sub-band, vertical sub-band, and diagonal sub-band.  
 (d) Select approximate sub-band of bank cheque image and apply DCT using Eq. 2.  
 (e) It is divided into  $8 \times 8$  block of selected sub-band, for embedding the watermark pixels in middle frequency coefficient region of DCT applied bank cheque image.

$$C(u, v) = \alpha(u)\alpha(v) \sum_{X=0}^{N-1} f(x, y) \cos\left[\frac{(2x+1)u\pi}{2N}\right] \cos\left[\frac{(2y+1)v\pi}{2N}\right] \quad (2)$$

For  $u, v = 0, 1, 2 \dots N-1$ .

For  $x, y = 0, 1, 2 \dots N-1$ .

where

$$\alpha(u) = 1/\sqrt{2u} = 0.$$

or

$$\alpha(u) = 1 \quad u = 1, 2 \dots, N-1.$$

$$\alpha(v) = 1/\sqrt{2} \quad v = 0.$$

or

$$\alpha(v) = 1 \quad v = 1, 2 \dots, N-1.$$

- (f) Watermark is selected and its formation is created into stream of bit pixel.

- (g) These two pseudo-random numbers of bits pixel of zeros and ones of same length are generated and use for embedding using Eqs. 3.1 and 3.2, respectively. If the watermark bit is '1', then,

$$Iw = Im + (\beta * P_{n\_1}) \quad (3.1)$$

Otherwise, if watermark bit is '0', then,

$$Iw = Im + (\beta * P_{n\_0}) \quad (3.2)$$

where,

- Im* Middle frequency coefficient band of DCT.  
*P<sub>n\_1</sub>* Represents bit '1'.  
*P<sub>n\_0</sub>* Represents bit '0'.  
*B* Gain factor used for embedding the watermark pixel.

- (h) An inverse DCT is applied on approximate sub-band after embedding bits and its modification in middle frequency coefficient using Eq. 4.

$$f(x, y) = \sum_{u=0}^{N-1} \alpha(u)\alpha(v)C(u, v) \cos\left[\frac{(2x+1)u\pi}{2N}\right] \cos\left[\frac{(2y+1)v\pi}{2N}\right] \quad (4)$$

- (i) An inverse DWT is applied to obtain combined DWT-DCT watermarked bank cheque image.  
(j) Further, AES encryption process is carried out on DWT-DCT watermarked bank cheque image using 256-bit key.  
(k) Finally, combined DWT-DCT watermarked and AES encrypted bank cheque is obtained.

The flow diagram of encryption and decryption of bank cheque image of proposed algorithm is as shown in Fig. 3.

After obtaining the DWT-DCT watermarked and AES encrypted bank cheque image, it is transferred through network channel for clearing house of bank. After receiving it, the AES decryption and watermark extraction process are carried out. It is shown in Fig. 4.

### 3.2 Part B: Decryption and Extraction Process

- (a) Read received combined DWT-DCT watermarked and AES encrypted bank cheque image.

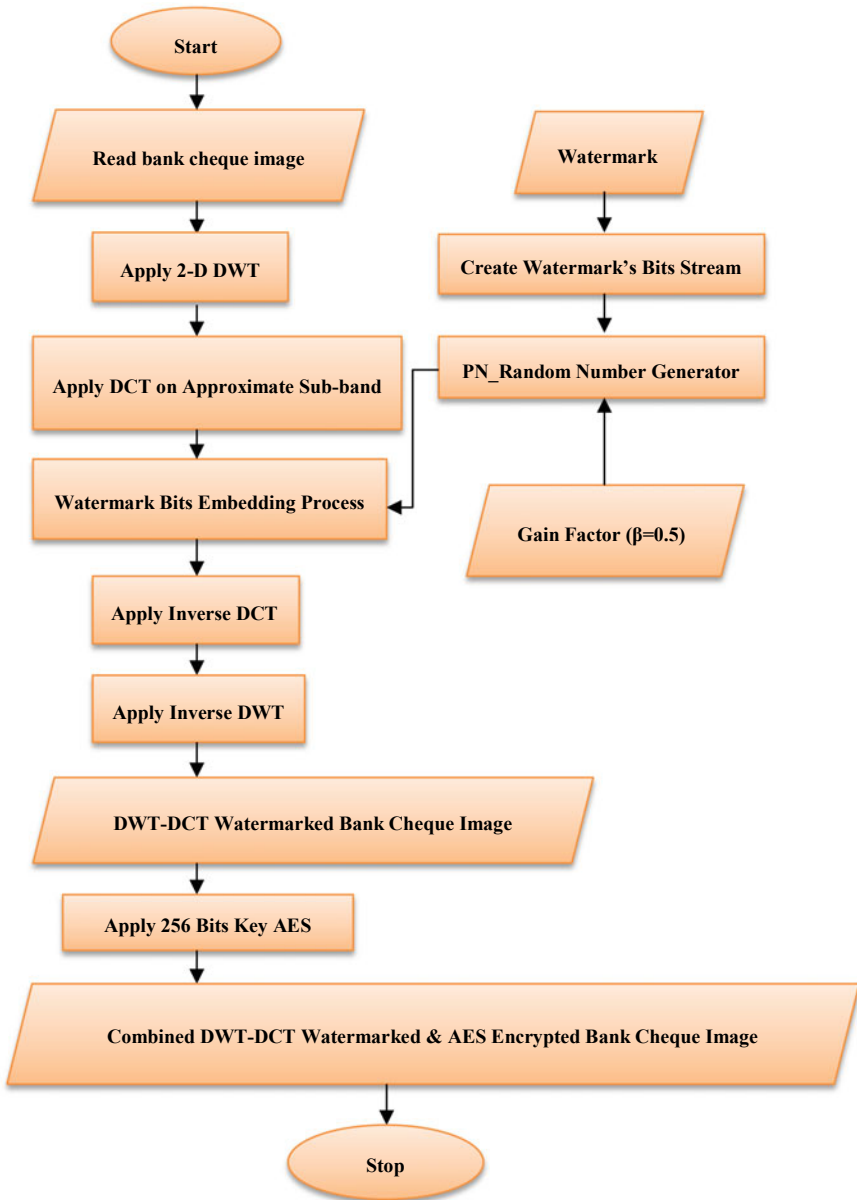
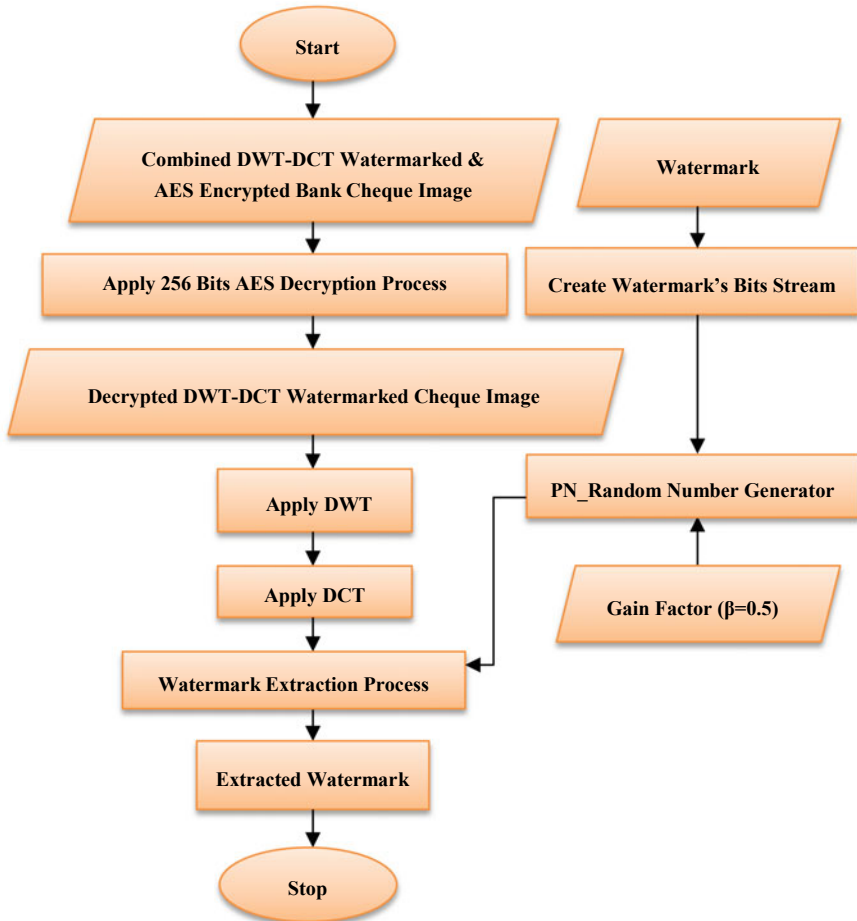


Fig. 3 Watermarking and encryption of bank cheque image



**Fig. 4** AES decryption and watermark extraction of bank cheque image

- (b) AES decryption process is carried out using 256-bit key and decrypted DWT-DCT watermarked bank cheque is obtained.
- (c) Further, DWT is applied to obtain decrypted and watermarked cheque image.
- (d) Select approximate sub-band of DWT-DCT watermarked bank cheque image.
- (e) DCT is applied on processed image and middle frequency coefficient of DCT block for embedded watermark bits pixel extraction.
- (f) The same pseudo-random bit stream of zeros and ones of watermark is used, which were used at the time of embedding process.
- (g) This pseudo-random bit stream is used for calculating the correlation of middle frequency coefficient bits.
- (h) The watermark bit one is extracted if the correlation with PN<sub>1</sub> is greater than PN<sub>0</sub>; otherwise, bit zero is extracted.

- (i) Reconstruction of watermark is done by calculating similarity index between original watermark and extracted watermark.

## 4 Parameter Used for Result Analysis

There are four parameters used for checking performance and doing analysis of this research work. They are: (i) robustness, (ii) mean square error, (iii) imperceptibility, and (iv) time.

### 4.1 Robustness

Robustness terminology is related to the normalized correlation coefficient of extracted watermark and original watermark used for embedding. Mathematically, it is expressed as shown in Eq. 5.

$$\rho(W, W') = \frac{\sum_{i=0}^N W_i * W'_i}{\sqrt{\sum_{i=1}^N (W_i)^2} \sqrt{\sum_{i=1}^N (W'_i)^2}} \tag{5}$$

where

- $W$  is original watermark,
- $W'$  is extracted watermark, and
- $\rho$  is correlation function.

### 4.2 Mean Square Error

Mean square error is terminology used to find the pixel position error between processed image and original image. It calculates in decibels. It is used to find the pixel value function of the difference watermarked bank cheque image and original bank cheque image. Mathematically, it is represented as shown in Eq. 6.

$$\text{MSE} = \left\{ \frac{1}{M \times N} \right\} \sum_{x=1}^N \sum_{y=1}^M [I(x, y) - I'(x, y)]^2 \tag{6}$$

where

- $I(x, y)$  is original bank cheque image,
- $I'(x, y)$  is watermarked bank cheque image,
- $x$  and  $y$  are pixel coordinates of bank cheque image.



### 4.3 Imperceptibility

An imperceptibility term deals with the peak signal to the noise ratio of image, which represents the quality of image. It is used to measure noise in image, when attacks or noise are added in image. Mathematically, it is expressed as shown in Eq. 7.1 and Eq. 7.2.

$$\text{PSNR}_{(\text{dB})} = 10 \log_{10} \left\{ \frac{\text{MAX}_1^2}{\text{MSE}} \right\} \quad (7.1)$$

$$\text{PSNR}_{(\text{dB})} = 20 \log_{10} \left\{ \frac{\text{MAX}_1}{\text{MSE}} \right\} \quad (7.2)$$

where

- Max is maximum image  $x$  and  $y$  pixels of bank cheque image,
- MSE is mean square error value of bank cheque image,
- PSNR is peak signal-to-noise ratio in dB.

### 4.4 Time

Time is an important constraint in every research. There are different times calculated for various process, viz. (i) complete process time, (ii) encryption process time, (iii) digital watermark embedding process time, (iv) digital watermark extraction process time, and (v) decryption process time. It is calculated in seconds. Table 1 shows the time calculated for different process in this research.

From above Table 1, it is observed that the encryption time required for DWT-DCT watermarked bank cheque is 0.452 s, which is same against JPEG compression, median filtering, and under normal mode attack. The encryption time 0.405 s is found against Gaussian noise attack, which is equivalent to salt and pepper noise attack and rotation attack. The comparative study of various times taken by different processes is shown in Fig. 5. The graphical representation is plotted with reference to time shown in Table 1.

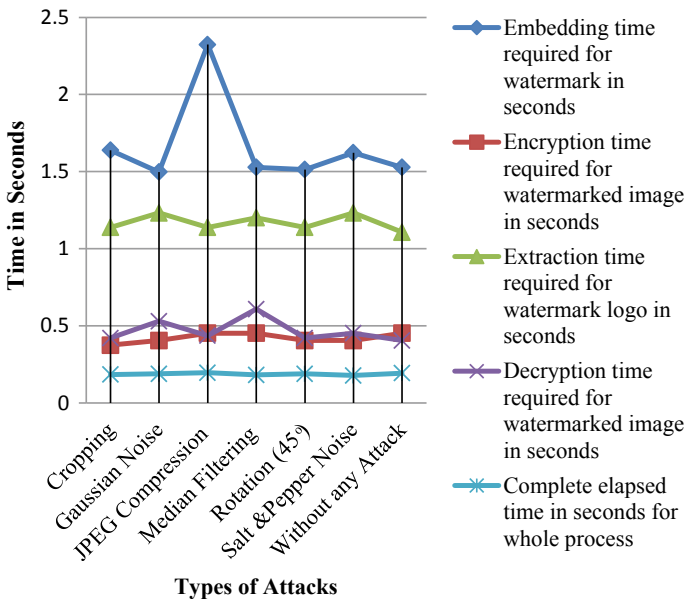
## 5 Results and Discussion

In this research, the operations are performed on bank cheque and watermark logo image having dimensions  $512 \times 512$  with pixel resolution of 96 dpi and bit depth of 24 bits. However, cheque image size is 37.5 KB and watermark logo size is 12.3 KB shown in Figs. 6 and 7, respectively.

The DWT transformed is applied on bank cheque. The approximate sub-band of image is selected. Further, DCT is applied on the selected sub-band of cheque image.

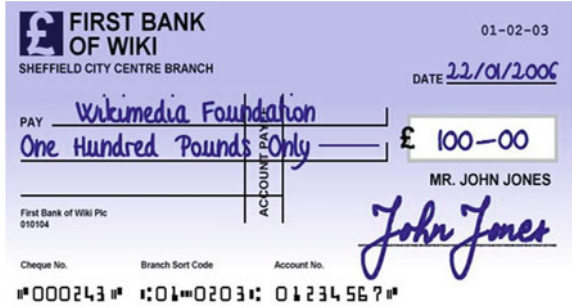
**Table 1** Time taken by different processes against various attacks for gain factor ( $\beta = 0.5$ )

Types of attacks	Embedding time of watermark in seconds	Encryption time of watermarked image in seconds	Extraction time of watermark logo in seconds	Decryption time required for watermarked image in seconds	Complete elapsed time in seconds for whole process
Cropping	1.638	0.374	1.138	0.421	0.184
Gaussian noise	1.497	0.405	1.232	0.530	0.189
JPEG compression	2.324	0.452	1.138	0.436	0.196
Median filtering	1.528	0.452	1.201	0.608	0.181
Rotation (45°)	1.513	0.405	1.138	0.421	0.189
Salt and pepper noise	1.622	0.405	1.232	0.452	0.178
Under normal mode	1.528	0.452	1.107	0.405	0.193



**Fig. 5** Graph of time taken by various processes against different attacks

**Fig. 6** Original bank cheque image



**Fig. 7** Original watermark logo

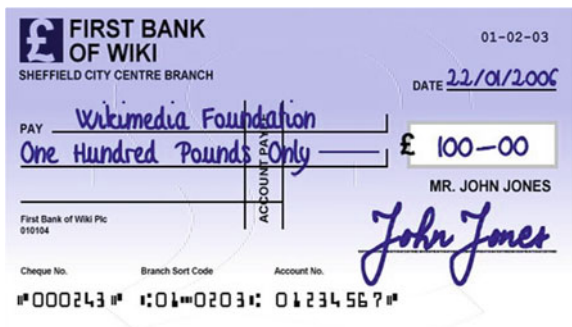


In these operations, the watermark bits are inserted, and DWT-DCT watermarked bank cheque image is obtained as shown in Fig. 8. It is observed that the peak signal-to-noise ratio is quite good after watermarking. However, the watermark logo is slightly visible which affects the quality of cheque image to very small extend.

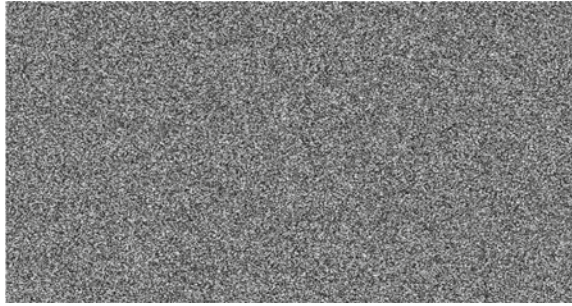
Finally, the 256-bit key AES process is carried out in Fig. 8 to achieve combination of DWT-DCT watermarked and AES encrypted bank cheque image as shown in Fig. 9.

Different attacks are applied on the image shown in Fig. 9, which are considered in this research. It is observed that the attacked image seems same as that of watermarked and encrypted image shown in Fig. 9 except the rotation attacked image. The rotation attacked is applied in Fig. 9 with an angle of 45°. It is appears as shown in Fig. 10.

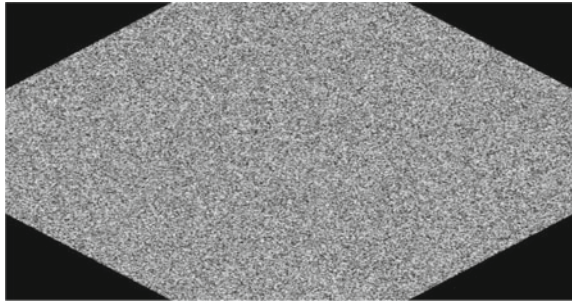
**Fig. 8** DWT-DCT watermarked bank cheque image



**Fig. 9** DWT-DCT watermarked and AES encrypted bank cheque image



**Fig. 10** Rotation attacked DWT-DCT watermarked and AES encrypted bank cheque image

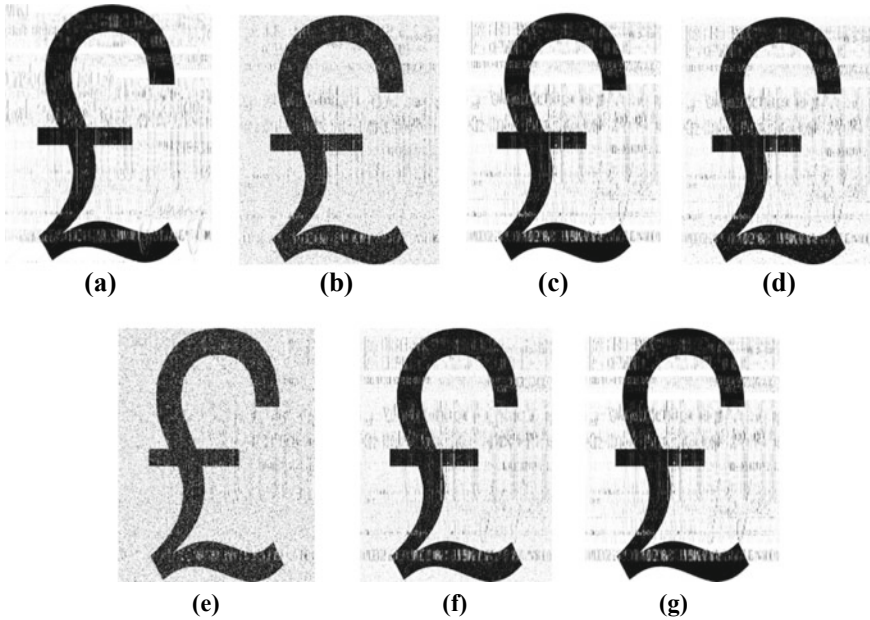


Further, 256-bit key AES decryption process is carried on attacked DWT-DCT watermarked and AES encrypted bank cheque image, to achieve decrypted DWT-DCT watermarked image. Further, watermark extraction process is carried on it to achieve extracted watermark logo as shown in Fig. 11.

The quality, i.e. robustness of watermark, after encryption and watermarking and after decryption and extraction of watermark is calculated against different attacks considered in this work. It is found that the robustness against cropping attack, JPEG compression attack and under normal mode attack is best as compared to remaining attacks after encryption and watermarking process. It is as shown in Table 2.

From Table 2, it is observed that the peak signal-to-noise ratio after DWT-DCT watermarking process is 88.410 dB against all attacks except cropping attack. The PSNR value is found 57.729 dB against cropping attack after watermarking of cheque image. The PSNR value of bank cheque image after extraction is 65.175 dB against JPEG compression attack and under normal mode attack.

It is also found that the mean square error of DWT-DCT watermarked bank cheque image is found 0.000093 dB against all attacks except cropping attack. The MSE value against cropping attack is found 0.1096 dB after watermarking process. The mean square error is 0.0197 dB against JPEG compression and under normal mode attack. It is also found that the PSNR value is found 61.083 dB against rotation attack after watermark extraction which higher than the rest of the attacks considered in this work. The MSE value of bank cheque image after watermark extraction is very high as compared to the mean square error value of watermarked bank cheque image.



**Fig. 11** Extracted watermark against **a** cropping attack, **b** Gaussian blur attack, **c** JPEG compression attack, **d** median filtering attack, **e** rotation attack, **f** salt and pepper noise attack, and **g** under normal mode attack

The graphical representation of PSNR value, MSE value, and NCC value, i.e. robustness of digital watermark logo, is shown in Figs. 12 and 13 respectively.

## 6 Conclusion

In this paper, the combination of DWT-DCT bank watermarking along with 256-bit key AES encryption and decryption process is discussed. It is found that the rotation attack with  $45^\circ$  has more impact as compared to the remaining attack. It also found that this method provides good level for confidentiality, integrity, authentication along with copyright protection for bank cheque image except rotation attack. The work also tells that the quality of bank cheque is maintained well after DWT-DCT cheque image watermarking technique. However, it is observed that robustness of watermark is very good after combined DWT-DCT cheque image watermarking and AES encryption technique as compared to that of quality of watermark obtained from decrypted watermarked cheque image. This method provides security to bank cheque image as well as reduces the physical maintenance of cheque documents. It provides the faster clearance service of bank cheque to customer.

**Table 2** PSNR, MSE, and NCC values of bank cheque image and watermark logo against various attacks for gain factor ( $\beta = 0.5$ )

Types of attacks	PSNR value of watermarked image in dB	PSNR value of image after extraction in dB	MSE value of watermarked image in dB	MSE value of image after watermark extraction in dB	NCC value of watermark image	
					After encryption and watermarked image	After decryption and extraction of watermark logo
Cropping	57.729	66.449	0.1096	0.0147	1	0.9624
Gaussian noise	88.410	62.728	0.000093	0.0346	0.9644	0.9105
JPEG compression	88.410	65.175	0.000093	0.0197	1	0.9479
Median filtering	88.410	64.750	0.000093	0.0217	0.9995	0.9428
Rotation (45°)	88.410	61.083	0.000093	0.0506	0.0079	0.8653
Salt and pepper noise	88.410	64.585	0.000093	0.0226	0.9331	0.9419
Under normal mode	88.410	65.175	0.000093	0.0197	1	0.9479

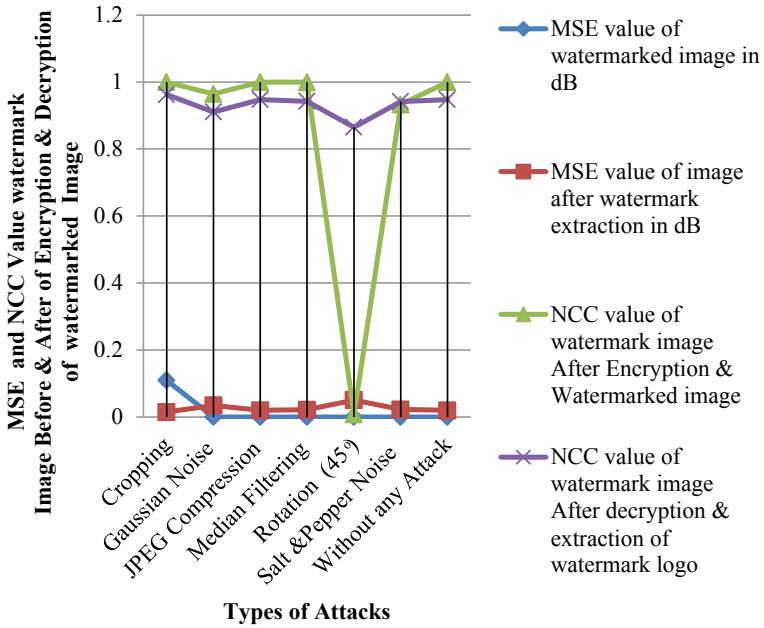


Fig. 12 A comparative representation of MSE value and NCC value watermark image and DWT-DCT watermarked bank cheque against various attacks

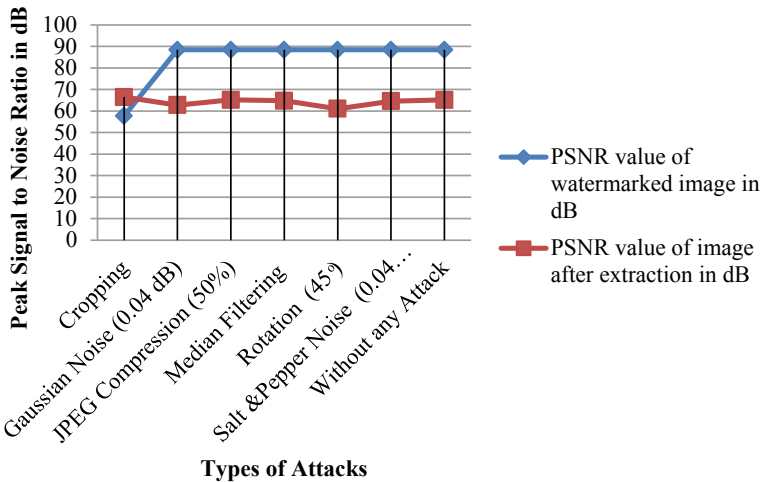


Fig. 13 A comparative representation of PSNR value bank cheque against various attacks

## References

1. W. Stallings, Advance encryption standard, in *Cryptography and Network Security*, 4th ed. (Pearson, India, 2005), pp. 134–165
2. Federal Information Processing Standards Publication 197, in *Announcing the Advanced Encryption Standard (AES)*, 26 Nov 2001
3. I.J. Cox, M.L. Miller, J.A. Bloom, *Digital Watermarking* (Morgan Kaufmann Publishers, 2002)
4. S. Wang, Y. Lin, wavelet tree quantization for copyright protection watermarking. *IEEE Trans. Image Process.* **13**(2), 154–164 (2004)
5. M. Tsai, H. Hung, DCT and DWT based image watermarking using subsampling, in *Proceedings of 4th IEEE International Conference on Machine Learning and Cybernetics*, China (2005), pp. 5308–5313
6. M. Kumari, S. Gupta, P. Sardana, A survey of image encryption algorithms. *3D Res* **8**, Research Article 37 (2017). <https://doi.org/10.1007/s13319-017-0148-5>
7. N. Tayal, R. Bansal, S. Gupta, S. Dhall, Analysis of various cryptography techniques: a survey. *Int. J. Secur. Its Appl.* **10**, 59–92 (2016). <https://doi.org/10.14257/ijjsia.2016.10.8.07>
8. R. Saikumar, D. Napoleon, An efficient watermarking and key generation technique using DWT algorithm in three-dimensional image. *Int. J. Recent Technol. Eng. (IJRTE)* **8**(2) (2019). ISSN: 2277-3878
9. J. Wang, S. Yuan, A novel security mobile payment system based on watermarked voice cheque, in *Proceedings of 2nd International Conference on Mobile Technology, Applications and Systems* (2005)
10. A. Al-Haj, Combined DWT-DCT digital image watermarking. *J. Comput. Sci.* **3**(9), 740–746. ISSN 1549-3636
11. Q. Wang, Q. Ding, Z. Zhang, L. Ding, Digital image encryption research based on DWT and Chaos, in *Proceedings of 2008, Fourth International Conference on Natural Computation*, Jinan (2008), pp. 494–498. <https://doi.org/10.1109/ICNC.2008.105>
12. S.K. Amirgholipour, A.R. Naghsh-Nilchi, Robust digital image watermarking based on joint DWT-DCT. *Int. J. Digital Content Technol. Its Appl.* **3**(2) (2009)
13. A.M. Kothari, A.C. Suthar, R.S.Gajre, Performance analysis of digital image watermarking technique–Combined DWT–DCT over individual DWT. *Int. J. Adv. Eng. Appl.* 177–181, Jan 2010
14. M. Thapa, S.K. Sood, A.P. Meenakshi Sharma, Digital image watermarking technique based on different attacks. *Int. J. Adv. Comput. Sci. Appl. (IJACSA)* **2**(4) (2011)
15. A. El-Mohandes, Hybrid DCT-DWT watermarking and IDEA encryption of internet contents. *Int. J. Comput. Sci. Issues (IJCSI)* **9**(1), 1694–0814. [www.IJCSI.org](http://www.IJCSI.org).
16. M. Amini, H. Sadreazami, Image watermarking through joint spatial segmentation and wavelet packet frequency division, in *Proceedings of 11th IEEE International Conference on Signal Processing* (2012), pp. 632–635
17. A.V. Subramanayam, S. Emmanue, M.S. Kankanhalli, Robust watermarking of compressed and encrypted JPEG-2000 Images. *IEEE Trans. Multimedia* **14**(4):703–716 (2012)
18. M.R. Mohd Isa, S. Aljareh, Biometric image protection based on discrete cosine transform watermarking technique, in *IEEE Proceedings of International Conference on Engineering and Technology (ICET)* (2012), pp. 1–5
19. P. Mahajan, A. Sachdeva, A study of encryption algorithms AES, DES and RSA for security. *Global J. Comput. Sci. Technol. Network Web Secur.* **13**(15), 14–22 (2013). Online ISSN: 0975-4172, Print ISSN: 0975-4350
20. S.P. Metkar, M.V. Lichade, Digital image improvement by integrating watermarking and encryption technique, in *Proceedings of IEEE International Conference on Signal Processing, Computing and Control (ISPCC)*, 26–28 Sept 2013
21. S. Sirmour, A. Tiwari, A hybrid DWT-SVD based digital image watermarking algorithm for copyright protection. *Int. J. P2P Network Trends Technol. (IJPTT)* **6**, 7–10 (2014). ISSN: 2249-2615



22. K.-H. Lee, P.-L. Chiu, Digital image sharing by diverse image media. *IEEE Trans. Inf. Forensics Secur.* **9**(1), 88–98 (2014)
23. A. Nambutdee, S. Airphaiboon, Medical image encryption based on DCT-DWT domain combining 2D-DataMatrix Barcode, in *Proceedings of 2015, 8th Biomedical Engineering International Conference (BMEiCON)*, Pattaya (2015), pp. 1–5. <https://doi.org/10.1109/BMEiCON.2015.7399508>.

# An HTTP DDoS Detection Model Using Machine Learning Techniques for the Cloud Environment



N. Muraleedharan and B. Janet

**Abstract** The cloud computing platform has been evolved as an essential computing paradigm for today's world. As the cloud environment mainly focuses on the service model, to ensure the availability of these services to the intended user is an essential requirement. In this paper, an HTTP DDoS detection model for the cloud environment is presented. The proposed system uses machine learning-based classifiers on network flow data. Four tree-based classifiers, i.e., decision tree, random forest, XGBoost, and AdaBoost are applied to the identified parameters. The CIDDS-001 dataset were used for training and evaluation. Results obtained show that the proposed classifier can achieve 99.99% accuracy using the random forest classifier. Comparing the obtained results with the recent works available in the literature shows the proposed model outperforms it in the classification accuracy.

**Keywords** Denial of service attack · Cloud computing · Flow data · Machine learning · HTTP DDoS

## 1 Introduction

The cloud computing model is defined as a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction [1]. It consists of five essential characteristics, three service models, and four deployment models. The first service model is known as Software as a Service (SaaS), where the cloud user can control the application configurations. The second service model, Platform as a Service (PaaS), allows cloud users to control the hosting environments.

---

N. Muraleedharan (✉)

Center for Development of Advanced Computing (C-DAC), Bengaluru, India

e-mail: [murali@cdac.in](mailto:murali@cdac.in)

B. Janet

National Institute of Technology (NIT), Tiruchirappalli, India

e-mail: [janet@nitt.edu](mailto:janet@nitt.edu)

In the third service model, namely Infrastructure as a Service (IaaS), the cloud user controls everything except the data center infrastructure.

Due to the scalability, maintainability, and ease of management, many businesses, industries, and governments use cloud infrastructure to provide their services to the users. As the cloud environment mainly focuses on the service model, the availability of these services to the intended user is an essential requirement in the cloud environment. In the cloud context, availability refers to the network, software, data resources, hardware, and computing infrastructure. However, adversaries use Distributed Denial of Service (DDoS) attacks to disrupt or delay the service to the users. As per the report published by the cloud security alliance, DDoS is one of the top threats in the cloud computing environment [2].

As the customer data resides on the hardware owned and managed by the third-party, data privacy is one of the major concerns for cloud users [3]. Hence, data privacy needs to be preserved while the collection, monitoring, and analysis of the traffic to detect attacks such as DDoS.

In this paper, a DDoS detection approach using machine learning techniques for the cloud environment is presented. To monitor and analyze the cloud traffic for DDoS detection, we are using the packet header-based flow level data. Since it uses flow data for the analysis, access to the message content or any other part of the packet except header information is not required. Hence, the privacy of the data content has been preserved during the data collection and analysis. The significant contributions of our paper are as follows:

- A flow-based machine learning classifier for DDoS detection
- A privacy aware DDoS monitoring and detection model for cloud environment.

The following are the advantages of our approach

- As it uses flow level data derived from the packet header, the data privacy is ensured during the data collection, analysis, and attack detection. The monitoring and analysis of encrypted data can be done using flow data.
- Compared to the packet level information, the volume of the data will be less.

However, as the flow data is derived from the network gateway, the proposed approach may not be able to detect the DDoS within the virtual environment.

The paper is organized as follows. Section 2 provides the background and related work on DDoS attacks in the cloud environment. Details about the proposed model are explained in Sect. 3. The dataset used, experiment conducted, and results obtained are presented in Sect. 4. Conclusion and future works are shown in Sect. 5.

## 2 Background and Related Work

Denial of Service (DoS) attack is an attack on the availability where the malicious user tries to disrupt or block the services to a genuine user. To deny the services to a user, the attacker may generate several fake requests to the targeted server. Upon receiving

the bogus requests from the attacker, the server may allocate its resources to process the request to give the replay. But as the resources are limited, the overutilization of the resources like memory, CPU, etc., shall affect the server performance. A complicated version of DoS is known as Distributed Denial of Service (DDoS). In DDoS, the number of attackers generates and injects the traffic to the victim machine. Bot-nets of compromised hosts are used to generate a huge volume of DDoS attack traffic [4].

### 2.1 DDoS Attack on Cloud

As the cloud environment uses the traditional network infrastructure for connectivity, the DDoS attacks on the traditional network apply to the cloud environment as well. Multi-tenancy is one of the important characteristics of the cloud computing environment, where multiple cloud users share resources and infrastructure. Due to the multi-tenancy, a DDoS attack on the cloud environment shall affect multiple users and more harmful than the DDoS attack on the traditional computing environment. The DDoS attacks in the cloud can be classified as shown in Fig. 1.

#### Infrastructure Level

Computing resources such as CPU, memory, network bandwidth, and interconnection device are considered as the infrastructure. Traditional volumetric attacks such as SYN flood, UDP flood, and ICMP flood can be targeted to the cloud to deny its operations. In the flooding attacks, the attacker generates a huge number of requests to the server which exceeds the maximum number of connections supported by the server. To generate a huge volume of traffic, amplification attacks generate a higher size response for a small request size [5]. In a reflexive amplification attack, the attacker sends the request by spoofing the request's source IP address as victim IP. Upon receiving this request, the higher-sized response (amplified) is sent to the victim machine and shall be un-responsive [6].

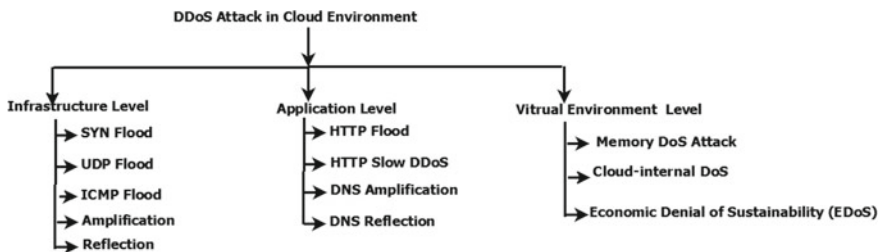


Fig. 1 DDoS classification in cloud environment

## Application Level

Unlike the infrastructure level DDoS, the application layer DDoS attack targets the application layer vulnerabilities. As HTTP is one of the significant application layer protocols used in the cloud, attackers target different types of DDoS. In the HTTP flood attack, the attacker sends a large number of HTTP requests to the server. GET and POST methods in HTTP are used to generate the requests. During the HTTP slow DDOS, upon sending the legitimate request, the attacker tries to prolong the connections by reading the response slowly by advertising a very small TCP receive window size to the server [7].

Domain Name System (DNS) is another application usually the attackers target for DDoS [8]. A huge volume of traffic generated by amplification attack using DNS queries is one of the critical attacks on the cloud.

## Virtual Environment

Virtual machines (VMs) play an essential role in cloud computing as it uses resource isolation, scalability, and multitenancy. Hence, the attackers target VMs to deny or disrupt cloud services. Memory DoS attack and cloud internal DoS attacks are two types of attacks targeted on VMs. In memory DoS attacks, a malicious VM intentionally induces memory resource contention to degrade the performance of co-located victim VMs [9]. The cloud-internal DoS attack is a cloud-specific DoS attack in which the malicious VMs in the same physical host try to attack their host [10]. Due to the on-demand resource allocation in the cloud configuration, DDoS malicious traffic can significantly increase the operational cost to the service provider. The overutilization of the resources due to the DDoS traffic may affect the service provider's economic sustainability and is known as Economics Denial of Sustainability (EDoS).

## 2.2 Related Work

Security and privacy issues in cloud computing environment have been under study by researchers. Zhifeng et al. [3] survey the security and privacy issues in cloud computing based on confidentiality, integrity, availability, accountability, and privacy. Various studies specific to the DoS/DDoS attack on the cloud computing infrastructure are available in the literature. A comprehensive survey on DDoS attack and defense mechanisms in the cloud environment is presented in [11, 12]. Somani et al. [13] observed that, in addition to the victim server or a network, almost all the components and stakeholders of cloud architecture are affected by a DDoS attack. The cloud-specific features like auto-scaling, migration, multi-tenancy have also introduced different attack vectors including DoS/ DDoS.

Idhammad et al. presented an HTTP DDoS attacks detection system for cloud environment using entropy and random forest learning algorithm [15]. They have computed the information theoretic entropy of incoming network packet header using

a time-based sliding window algorithm. The preprocessing and classification of the traffic are carried out further to detect HTTP DDoS. DDoS detection based on fast entropy method using flow-based analysis is presented in [16]. In their approach, the fast entropy of flow count is calculated for each connection.

Zekri et al. [17] proposes a DDoS detection approach for cloud computing environment using a decision tree technique. They experimented using a simulated environment with virtual machines and virtual LAN and obtained more than 98% classification accuracy using the C4.5 algorithm. A deep learning-based DDoS attack classification for cloud environment is presented in [14]. An optimized model using swarm intelligence for DDoS vulnerability analysis and mitigation is presented in [18]. Usage of various nature-inspired algorithms to address DDoS attacks in cloud environment is reviewed in [19].

In this paper, we have used a flow-based HTTP DDoS classification model using machine learning techniques for the cloud environment. The details of the model are explained next.

### 3 DDoS Detection Model

The details of the proposed model for DDoS detection on the cloud environment are described in this section. The model consists of different components such as the dataset, preprocessor, classification models, and model evaluation. Details about each of these components are described below.

#### 3.1 Data Format

Nowadays, flow-level data are used for high-speed traffic monitoring, security analysis, and anomaly detection. The flow can be defined as a sequence of packets traveling from a source to a destination from a specific time [20]. Unlike the packet level monitoring, flow data aggregates the packets based on the flow keys defined in Eq. (1). The incoming packets are grouped into a flow record based on ‘Flow Key’. In addition to the ‘Flow key’ a flow record consists of information like connection duration, number of packets, bytes transferred, TCP flag values, etc. The format of a flow record is defined in Eq. (2) where the ‘Duration’ represents the flow duration. The number of packets and total bytes transferred in the flow is represented as ‘Packets’ and ‘Bytes’, respectively. The flag fields present in the TCP packets are represented as ‘TCP flags’. Other than these fields, a flow record can have more granular level information.

$$\text{Flow Key} = \{\text{SrcIP}, \text{DstIP}, \text{SrcPort}, \text{DstPort}, \text{Protocol}\} \quad (1)$$

$$\text{Flow Records} = \{\text{Flow Key, Duration, Packets, Bytes, TCP flags, \dots}\} \quad (2)$$

### 3.2 Dataset Used

Coburg intrusion detection dataset (CIDDS-001) is a labeled unidirectional flow-based data used for the evaluation of anomaly based network intrusion detection systems [21, 22]. This dataset was generated in a virtual environment that consists of different clients and servers deployed in an open stack environment. The parameters used and their description are tabulated in Table 1.

In the CIDDS-00 dataset, the normal user behavior was emulated using a Python script. To derive this dataset, 92 attacks including DoS, and Brute Force were carried out over four weeks that contains nearly 32 million flows. The DoS attack was carried out during the first and second weeks of the attack period. Hence, we have selected the week1 and week2 dataset for our analysis. This dataset provides labeled flows in CSV (comma separated values) format.

**Table 1** Flow parameter and their description of CIDDS dataset [23]

Flow parameter	Description
Src IP	Source IP address
Dest IP	Destination IP address
Src port	Source port
Dest port	Destination port
Proto	Transport layer protocol
Time duration	Duration of the connection
Date first seen	Start time flow first seen
Duration	Duration of the flow
Bytes	Number of bytes transmitted in the flow
Packets	Number packets transmitted in the flow
Flags	OR concatenation of all TCP Flags
Class	Class label
Attack type	Type of attack
AttackID	All flows which belong to the same attack carry the same id
Attack description	Information about the attack parameters

**Table 2** Summary of flow records selected for the model training and testing

Selected HTTP flows	Benign	DDoS	Training	Testing
1,000,00	66,607	33,393	60,000	40,000

### 3.3 Data Preprocessing

The selected dataset consists of 8,451,520 flows. As our focus is on Web-based DDoS detection, we have filtered the HTTP and HTTPS flows from the TCP flows using the port numbers (80, 443) available in the flow records. Depend on the transfer rate, the flow parameter ‘Bytes’ were represented in ‘Bytes’, ‘Kilo Bytes’, and ‘MegaBytes’. Hence, as the preprocessing step, we have converted the data values into a standard unit (Bytes).

We have filtered out the environment-specific parameters such as ‘Src IP’ and ‘Dst IP’ fields from further analysis in the preprocessing stage. As we were selected only HTTP traffic for our analysis, the ‘Proto’ field was also removed from the training dataset. As we focus on classifying the attack type, the ‘AttackID’ and ‘Attack Description’ fields were also filtered from the dataset in the preprocessing stage. Hence, after the preprocessing stage, the model has nine features in the dataset given for classification.

We randomly selected one lakh flows for our classification model from the HTTP flows, where 66,607 flows are related to benign traffic, and 33,393 flows are derived from DDoS attack traffic. The selected flow records are divided into the training set and test set where the training set is used to train the model and the testing set used to test the classification performance. We have used the ‘train\_test\_split’ function available in Python [24] to split the dataset. The summary of the dataset used for the model training and testing is tabulated in Table 2.

### 3.4 Classification Models Used

The selected flow data consists of nine features including the label field. As it is a labeled dataset, we have used supervised classification techniques. As the tree-based classifiers are efficient and easy to implement, we have used only tree-based classifiers in this experiment. We have used four classification algorithms named as decision tree, random forest, XGBoost, and AdaBoost to compare the performance. Though these classifiers use tree-based structure for classification, they use different classification approaches such as ensemble, bagging, and boosting for their implementation. The details about the classification model used are explained below.

#### Decision Tree

A decision tree (DT) is a common supervised machine learning algorithm used for classification. A decision tree can have zero or more internal nodes and one or more



leaf nodes. The internal node tests the value of an expression and the leaf node represents the classification category. The classification of a sample proceeds from the root node to a suitable leaf node through the internal nodes. We have used the ‘sklearn’ [24] library in Python for the implementation of the decision tree classifier. In this model, we have used the ‘Gini’ impurity as the splitting criteria.

The decision tree may generate biased output for an unbalanced dataset. To improve a single decision tree’s performance, ensemble methods that combine multiple decision trees were introduced. Bootstrap aggregation (Bagging) and boosting are the two common techniques to perform ensemble decision trees. To compare the classification performance, we have used the random forest, XGBoost, and AdaBoost classifiers.

### **Random Forest**

Random forest classifier is bagging-based implementation of an ensemble decision tree where the features are randomly selected in each decision split. It constructs many decision trees and each decision tree node uses a subset of attributes randomly selected from the whole original set of attributes. These trees will be used to classify a new instance by the majority vote. To implement the random forest classifier, we have used ‘sklearn’ [24] library available in Python. The number of estimators is configured as 100 for this classification.

### **XGBoost**

eXtreme Gradient Boosting (XGBoost) is a decision tree-based ensemble machine learning algorithm that combines multiple learners’ predictive power and aggregates the output. XGBoost is derived from the Gradient Boosting where the weak learning models are combined to a stronger model in a sequential iterative manner. In our model, the XGBoost classifier is implemented using the ‘XGBClassifier’ available in the ‘sklearn’ [24] library.

### **AdaBoost**

AdaBoost is also a boosting-based ensemble method that uses weak learners iteratively to form a strong learner. This algorithm uses a method to correct its predecessor is by paying more attention to the under fitted training instances. In our model, the AdaBoost classifier is implemented using the ‘AdaBoostClassifier’ available in the ‘sklearn’ library. In this model, we have used the number of estimators as 50 and the learning rate is fixed as one.

## **4 Results and Discussion**

Details of the metrics used for evaluation, classification results obtained from the model, comparison of the results obtained with a state-of-the-art approach are explained below.

## 4.1 Metrics Used for Evaluation

To evaluate the classification, the performance metrics such as accuracy, precision, recall, and F1 score are calculated from the model. To derive these parameters, we measured the True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN) for each classifier.

- True Positive (TP): Model correctly classified attack data as an attack.
- False Positive (FP): Model classified normal data as an attack.
- True Negative (TN): Model correctly classified normal data as normal.
- False Negative (FN): Model classified attack data as normal.

### Accuracy

To measure performance of the classification algorithm, we calculated the accuracy using Eq. (3).

$$\text{Accuracy} = (TP + TN)/(TP + FP + TN + FN) \quad (3)$$

where ‘TP’ is True Positive, ‘TN’ is True Negative, ‘FP’ is False Positive, and ‘FN’ is False Negative. The accuracy value can be within the range of 0 to 1, where ‘0’ indicates minimum accuracy and ‘1’ shows the maximum accuracy.

### Precision

The predictive power of the model is important for a classification algorithm. A model with high predictive value can be considered as better. To quantify the predictive power of the model, we measured precision as per Eq. (4).

$$\text{Precision} = TP/(TP + FP) \quad (4)$$

### Recall

Recall measures the number of actual positives the model has detected and labeled as Positive. A low recall indicates many False Negatives. The recall can be computed as per Eq. (5).

$$\text{Recall} = TP/(TP + FN) \quad (5)$$

### F1 score

F-score is commonly used for comparing the classification performance of an unbalanced dataset. The equation for calculating the F1 score is shown in Eq. (6). As per the equation, if either precision or recall is low, then the resulting F-measure will be low.

$$F1 \text{ score} = (2 * (\text{precision} * \text{Recall}) / (\text{precision} + \text{Recall})) \tag{6}$$

### 4.2 Results Obtained

The summary of classifiers used, obtained precision, recall, *F1* score, and accuracy are tabulated in Table 3.

From the results obtained, we can observe that the proposed model has achieved satisfactory results with 99.99% accuracy. The precision, recall, and *F1* score of all the selected classifiers have obtained the maximum. By comparing the classification accuracy obtained from the model, we can observe that the random forest classifier achieved the highest accuracy (99.99%) and AdaBoost classifier has the lowest accuracy (99.97%).

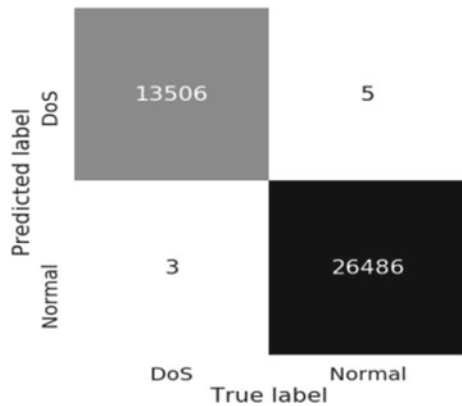
The confusion matrix obtained from different classifiers output is shown in Figs. 2, 3, 4, and 5. The ‘*x*’-axis of the confusion matrix shows the ‘True Label’ and the ‘*y*’-axis shows the predicted labels.

The confusion matrix of the decision tree depicted in Fig. 2 shows that the classifier misclassified eight records and the misclassification rate is 0.002. From Fig. 3, we can derive that the misclassification rate of the random forest classifier is 0.001.

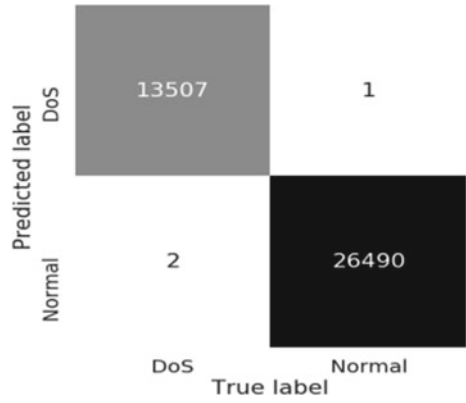
**Table 3** Summary of the classification results

Classifiers used	Precision	Recall	<i>F1</i> score	Accuracy
Decision tree	1.00	1.00	1.00	0.9998
Random forest	1.00	1.00	1.00	0.9999
XGBoost	1.00	1.00	1.00	0.9998
AdaBoost	1.00	1.00	1.00	0.9997

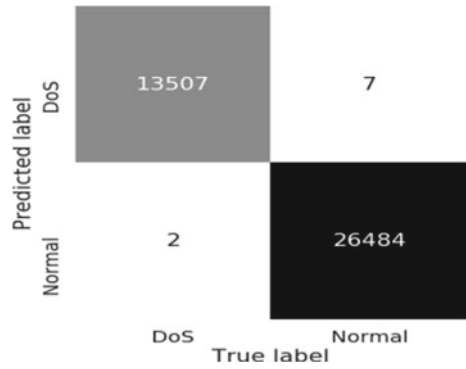
**Fig. 2** Confusion matrix of DT classifier



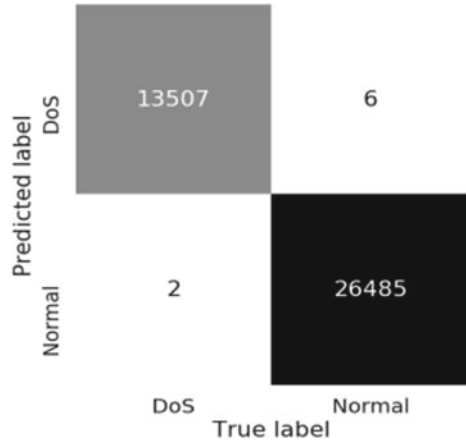
**Fig. 3** Confusion matrix of random forest



**Fig. 4** Confusion matrix of AdaBoost



**Fig. 5** Confusion matrix of XGboost



**Table 4** Comparison of accuracy of the proposed model with the model presented in [15]

Work	Algorithm used	Dataset used	Accuracy
Idhammad et al. [13]	Information theoretic entropy, random forest	CIDDS-001	99.54
Our model	Random forest	CIDDS-001	99.99

Similarly, from Figs. 4 and 5, we can derive the number of misclassification rate of AdaBoost and XGBoost classifiers are 0.003 and 0.002, respectively.

### 4.3 Comparison of the Result with State-Of-the-Art Approach

We have compared the performance of our classifier with one of the state-of-the-art approach described in [15]. The summary of the performance comparison is tabulated in Table 4. In their approach, they have used CIDDS-001 dataset for model evaluation. By comparing the classification accuracy of the proposed model with the state-of-the-art approach reveals that the model has obtained better accuracy compared to the state-of-the-art approach.

## 5 Conclusion and Future Work

In this paper, a DDoS detection model using machine learning techniques for the cloud environment is presented. Flow-level parameters derived from the network traffic are used in this model. Four tree-based classifiers, i.e., decision tree, random forest, XGBoost, and AdaBoost are applied to the identified parameters for detecting HTTP DDoS attacks. The model is trained and evaluated using ‘CIDDS-001’ flow dataset. To quantify the model’s effectiveness, the accuracy, precision, recall, and *F1* score metrics are used. Results obtained show the proposed classifier can achieve higher accuracy of 99.99% using the random forest classifier. Other classifiers are also performed well and obtained accuracy above 99%. A comparison of the obtained results with the recent works available in the literature shows the proposed model outperforms it in the classification accuracy. As a future activity, we plan to extend this work to detect and prevent DDoS and slow DoS attacks in the cloud environment.

## References

1. P. Mell, T. Grance, in *The NIST definition of cloud computing*. Computer Security Division, Information Technology Laboratory, National Institute of Standards and Technology, Sept 2011
2. Cloud Security Alliance, The Treacherous 12—Top Threats to Cloud Computing + Industry Insights. Cloud Security Alliance, 2017. [Online]. Available: [https://downloads.cloudsecurity](https://downloads.cloudsecurityalliance.org/Cloud-Security-Alliance-Treacherous-12-Top-Threats-to-Cloud-Computing-Industry-Insights-2017.pdf)

- [yalliance.org/assets/research/top-threats/traacherous-12-top-threats.pdf](http://yalliance.org/assets/research/top-threats/traacherous-12-top-threats.pdf).
3. Z. Xiao, Y. Xiao, Security and Privacy in Cloud Computing. *IEEE Commun. Surv. Tutor.* **15**(2), 843–859 (2013)
  4. N. Hoque, D.K. Bhattacharyya, J.K. Kalita, Botnet in DDoS Attacks: Trends and Challenges. *IEEE Commun. Surv. Tutor.* **17**(4), 2242–2270 (2015). <https://doi.org/10.1109/COMST.2015.2457491>
  5. C. Rossow, in *Amplification Hell: Revisiting Network Protocols for DDoS Abuse* (2014).
  6. F.J. Ryba, M. Orlinski, M. Wählisch, C. Rossow, T.C. Schmidt, in *Amplification and DRDoS Attack Defense—A survey and New Perspectives*. arXiv preprint [arXiv:1505.07892](https://arxiv.org/abs/1505.07892) (2015)
  7. N. Muraleedharan, B. Janet, Behaviour analysis of HTTP based slow denial of service attack, in *2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)* (2017), pp. 1851–1856
  8. K. Alieyan, M.M. Kadhum, M. Anbar, S.U. Rehman, N.K.A. Alajmi, An overview of DDoS attacks based on DNS, in *2016 International Conference on Information and Communication Technology Convergence (ICTC)*, Jeju, Oct 2016, pp. 276–280. <https://doi.org/10.1109/ICTC.2016.7763485>
  9. T. Zhang, Y. Zhang, R.B. Lee, Dos attacks on your memory in cloud, in *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security* (2017), pp. 253–265
  10. M. Masdari, M. Jalali, A survey and taxonomy of DoS attacks in cloud computing. *Secur. Commun. Networks* **9**(16), 3724–3751 (2016)
  11. G. Somani, M.S. Gaur, D. Sanghi, M. Conti, R. Buyya, DDoS attacks in cloud computing: Issues, taxonomy, and future directions. *Comput. Commun.* **107**, 30–48 (2017)
  12. N. Agrawal, S. Tapaswi, Defense mechanisms against DDoS attacks in a cloud computing environment: State-of-the-art and research challenges. *IEEE Commun. Surv. Tutor.* **21**(4), 3769–3795 (2019)
  13. G. Somani, M.S. Gaur, D. Sanghi, DDoS/EDoS attack in cloud: affecting everyone out there!, in *Proceedings of the 8th International Conference on Security of Information and Networks* (2015), pp. 169–176
  14. S. Velliangiri, P. Karthikeyan, V. Vinoth Kumar, Detection of distributed denial of service attack in cloud computing using the optimization-based deep networks. *J. Experim. Theor. Artif. Intell.* 1–20 (2020). <https://doi.org/10.1080/0952813X.2020.1744196>
  15. M. Idhammad, K. Afdel, M. Belouch, Detection system of HTTP DDoS attacks in a cloud environment based on information theoretic entropy and random forest, in *Security and Communication Networks*, vol. 2018 (2018)
  16. J. David, C. Thomas, DDoS attack detection using fast entropy approach on flow-based network traffic. *Procedia Comput. Sci.* **50**(4), 30–36 (2015)
  17. M. Zekri, S. El Kafhali, N. Aboutabit, Y. Saadi, DDoS attack detection using machine learning techniques in cloud computing environments, in *2017 3rd International Conference of Cloud Computing Technologies and Applications (CloudTech)* (2017), pp. 1–7
  18. N. Mishra, R.K. Singh, DDoS vulnerabilities analysis and mitigation model in cloud computing. *J. Discrete Math. Sci. Cryptogr.* **23**(2), 535–545 (2020). <https://doi.org/10.1080/09720529.2020.1729503>
  19. D.J. Prathyusha, S. Naseera, D.J. Anusha, K. Alisha, A review of biologically inspired algorithms in a cloud environment to combat DDoS attacks, in *Smart Intelligent Computing and Applications*, vol. 160, ed. by S.C. Satapathy, V. Bhateja, J.R. Mohanty, S.K. Udgata (Springer, Singapore, 2020), pp. 59–68
  20. A. Sperotto, G. Schaffrath, R. Sadre, C. Morariu, A. Pras, B. Stiller, An overview of IP flow-based intrusion detection. *IEEE Commun. Surv. Tutor.* **12**(3), 343–356 (2010)
  21. M. Ring, S. Wunderlich, D. Gründl, D. Landes, A. Hotho, Creation of flow-based data sets for intrusion detection. *J. Inform. Warfare* **16**(4), 40–53 (2017)
  22. M. Ring, S. Wunderlich, D. Gründl, D. Landes, A. Hotho, Flow-based benchmark data sets for intrusion detection, in *Proceedings of the 16th European Conference on Cyber Warfare and Security (ECCWS)* (ACPI, 2017), pp. 361–369

23. M. Ring, S. Wunderlich, D. Grüdl, Technical Report CIDDs-001 data set, April 28, 2017. [https://www.hs-coburg.de/fileadmin/hscoburg/Forschung/WISENT\\_cidds\\_Technical\\_Report.pdf](https://www.hs-coburg.de/fileadmin/hscoburg/Forschung/WISENT_cidds_Technical_Report.pdf)
24. F. Pedregosa, et al., Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011)

# IoT Device Authentication and Access Control Through Hyperledger Fabric



Bibin Kurian and Narayanan Subramanian

**Abstract** Internet of Things (IoT) is one of the sizzling technology that connects everything to everyone, everywhere. Security and privacy with confidentiality, integrity, and availability to data are among the most pressing challenge faced by IoT as well as the Internet. Networks are getting more expanded and are becoming more open, and security practices has to be uplifted to ensure protection of this rapidly growing Internet, its users, and data. In this paper, we propose a new authentication and access control mechanism for the IoT devices through a blazing blockchain technology, Hyperledger Fabric, an open-source distributed ledger platform for developing enterprise-grade permissioned blockchains. Blockchain is typically a hash-chain of blocks consisting of a number of (ordered) transactions. Fabric provides a secure and scalable permissioned platform with plug-in components that support data privacy and smartcontracts, rather than a permission-less system where anybody can access and transact data. The authentication and access control of the IoT devices is achieved by making use of newly introduced features in managing channel, chain-codes, policies, Certificate Authority (CA), and others in Hyperledger Fabric version 2.0. Our architecture has the potential to act upon different layers of the IoT in authentication and access control safeguarding the confidentiality, integrity, and availability of data.

**Keywords** Internet of Things · Fog computing · Blockchain

---

B. Kurian (✉) · N. Subramanian  
Center for Cybersecurity Systems and Networks, Amrita Vishwa Vidyapeetham,  
Amritapuri, Kollam, India  
e-mail: [bibin@am.students.amrita.edu](mailto:bibin@am.students.amrita.edu)

N. Subramanian  
e-mail: [nsubramanian@am.amrita.edu](mailto:nsubramanian@am.amrita.edu)

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2021  
S. M. Thampi et al. (eds.), *Advances in Computing and Network Communications*,  
Lecture Notes in Electrical Engineering 735,  
[https://doi.org/10.1007/978-981-33-6977-1\\_51](https://doi.org/10.1007/978-981-33-6977-1_51)

699



# 1 Introduction

A blockchain is a peer-to-peer distributed network with an immutable ledger for recording transactions. Every mutually untrusting peer keeps a copy of the ledger. Peers execute a consensus protocol to validate transactions, it is then grouped into blocks, and a hash chain is build over the blocks. Blockchain originated with Bitcoin, a distributed or permissionless blockchain created in 2008 by an anonymous individual or group of people using the name Satoshi Nakamoto and began in 2009 when its source code was published as open-source software and is widely regarded as a groundbreaking technology for operating trustworthy transactions in the digital world.

In a public or permissionless blockchain, as the name suggests, anyone can participate without a revealing their identity. Anyone who takes part can participate as users, miners, developers, or community members. All the transactions that happen on public blockchains are fully transparent, meaning that anyone can examine the details of any of the transaction happened. They often rely on consensus based on “proof of work” (PoW), “proof of stake” (PoS) and usually involve some form of native cryptocurrency like Bitcoin (BTC), Ethereum (ETH) and provide economic incentives and rewards to participants in the network. On the other hand, on a private or permissioned blockchain, participants need consent to join the networks, and their identities are known. The transactions are kept private and are only available to the participants who have the permission to take part in the network. They come to agreement with practical Byzantine-fault-tolerant (PBFT) consensus or its variants.

We use Hyperledger Fabric or simply fabric in this paper, a blockchain platform for achieving durability, efficiency, scalability, and confidentiality. Crafted as a permissioned blockchain with modular and extensible general purpose, fabric facilitates the execution of distributed applications written in standard programming languages such Go, Java, JavaScript, and Python. We take advantage of the the newly shipped features in version 2.0 of fabric and its flexibility to to operate fabric components even on the resource constrained IoT devices. The IoT devices communicate with the fabric components hosted on the edge/fog gateway node for authentication and access to the functionalities are managed by allocating to definite channels for their operation.

The Internet of Things (IoT) is a network of interrelated physical object called “things” that posses certain computing power by means of embedding sensors, software, and other technologies to add a level of digital intelligence to these devices enhancing them to produce, communicate, or perform on real-time data and to connect and exchange data with other devices and system without involving any human interaction over the Internet. Today, with more than 10 billion IoT devices connected, experts expect this number to rise to 25 billion by 2025. These devices include wearables like smartwatch to household smart objects to the ones used as industrial tools. The adoption of RFID tags and IPv6 along with the increasing availability of Internet and other wireless communication technologies made it possible to solve some of the issues prevailing in network connectivity. IoT technology is finding applica-

tion in varying domain from consumer use including wearable technology, home automation, connected cars, connected healthcare to more sophisticated industrial automation, smart cities, IoT in agriculture and farming, smart retails, energy management, and so on.

Security is one of the most critical issues concerning IoT. In certain cases, sensors gather highly sensitive data which is crucial to customer trust, but the security track record of the IoT has not even grown to a satisfactory level so far. Software bugs are frequently found, but many IoT devices lack the ability to be patched on the go, which means they are constantly at risk. As the cost of manufacturing smart objects becomes marginal, they become more wider and intractable to security flaws.

The rest of the paper is structured as follows: In Sect. 2, we are surveying work related to this; Sect. 3 gives the motivation. Sect. 4 describes the features provided by Hyperledger Fabric v2.0. We are discussing our proposed architecture & analysis of our authentication and access control scheme in Sect. 5, and finally, we conclude with our focus onto future in Sect. 6.

## 2 Related Works

The concept of blockchain integration with IoT is being called as Blockchain for IoT(BIoT) in [1]. In this Tiago M., Paula Fraga-Lamas discuss the practical limitations and identify areas for future research in BIoT scenarios. They also discuss the different types of blockchain, concepts in blockchain, and application domains where BIoT can be applied. Reference [2] defines a lightweight blockchain-based framework named FogBus that offers platform-independent interfaces for execution and interaction of IoT applications and computing instances. This paper discusses achieving data integrity through blockchain built in Java programming language and its side-by-side support in user authentication and data encryption. A credibility achieving mechanism by defining a blockchain system with manage servers (MS) that generates and distributes asymmetric keys to the devices and also manages other MS's in lower layers in identity verification is described in [3].

A thorough analysis of existing authentication mechanisms between the gateway and edge nodes and the development as well as deployment of an efficient key generation and exchange mechanism that results in the authentication with minimal computation on resource-constrained IoT devices is described in [4]. The authors, Shiju and Krishnasree, make the future work vision with the use of Hyperledger technique for authentication in inter and intra nodes cluster. One of the use-case in Ref. [5] describes introducing blockchain to edge nodes with computing power which helps in securing software transactions, downloading of device parameters related to functionality and device management, delivering software update to the nodes periodically which can be achieved through the peer-to-peer blockchain network without involving cloud computing resources, thus introduces the software-defined edge nodes (SDEN).

An insight into the Hyperledger Fabric design, implementation aspects, and performance analysis is discussed in [6]. A re-architect to fabric using a plug-n-play technique by implementing a series of independent optimizations focusing on I/O, caching, parallelism, and efficient data access to scale transaction throughput to 20,000 transactions per second is described in [7]. Reference [8] conducts a performance modeling of practical Byzantine-fault-tolerance (PBFT) consensus process for Hyperledger Fabric permissioned blockchain network. A discussion on the privacy protection mechanisms of Hyperledger Fabric with supply chain finance business scenario is described in [9].

The security threats and possible attacks that can be initiated by the adversaries are discussed in Ref. [10, 11] in security assessment at each layer of IoT architecture. Along with the security threats, a general potential blockchain solution for IoT using the intrinsic features of blockchain is discussed in [12]. The solution of using public blockchain platform 'Ethereum' for IoT access control and authentication management [13, 14] narrows the flexibility and increases performance overhead for a private use-case modeling due to its supported consensus and monetary aspects of Ethereum.

The privacy issues underpinning the use of biometrics for authentication in IoT applications is addressed by a cloud-based lightweight cancelable biometric authentication system in Ref. [15]. A comparison and analysis into the IoT authentication schemes are described in [16, 17]. A blockchain-based authentication and security mechanism discussed in [18] allocates a unique ID for each individual device and records them into the blockchain ledger. While discussing the challenges in adoption of blockchain, Ref. [19] speaks up the benefit of smartcontracts in limiting access to chosen methods only to specific node and functioning of fog node as miners as well in facilitating direct interaction between IoT devices and the blockchain.

The contemporary researches are meant as an enhancement or as to overcome authentication between nodes, gateways, or the cloud. The intention of our proposed architecture is to provide an insight in developing a full-fledged authentication, access control, and management mechanism that can act upon different layers of IoT from edge node, edge, or fog computing to cloud infrastructure by integration with a decentralized private and permissioned blockchain platform, Hyperledger Fabric.

### 3 Motivation

Several security problems exist among the different layers of IoT, in particular the "traditional" protocols in the application layer do not possess sufficient performance and security within IoT and do not have their own international standards. Regarding the application layer security requirements, authentication of device is necessary while protecting user privacy (in terms of data, respectively). Additionally, an information security management system should be in place that includes resource management and physical security specifics.

**Table 1** Security specification and IoT layers

Layer	Security requirement
Perception	Key agreement
	Data confidentiality
	Authentication
	Lightweight encryption
Network	Key management
	Authentication
	Communication security
	Intrusion detection
	Routing security
Application	Information security management
	Authentication
	Privacy protection

Table 1 offers a description of the three-layer protection specifications in the IoT architecture [20]. Authentication is simply a central protection feature that can be extended to various levels. Authentication is required between end devices and gateways (gateways in our proposal hosts components of fabric network). When transferring data to the cloud, the gateway should authenticate itself, and for collecting data for analysis, the application should be authenticated to the cloud. Our proposal has the potential to manage authentication and access control in different scenarios of communication.

The authentication schemes available to IoT varies in accordance to the implementation surface. A taxonomy of this is given below:

- Identity-based authentication scheme uses one (or combinations) of the cryptographic hash, a symmetric algorithm, or asymmetric algorithms.
- Physical and behavioral authentication
- A server created identity token (piece of data) is used in a token-based authentication scheme like in OAuth2 or OpenID protocol while passwords are involved in a non-token authentication whenever data exchange (TLS / DTLS) is needed.
- Single, two-, or three-factor authentication
- Distributed architecture in which the communicating parties use a distributed straight authentication method or use a centralized server to distribute and manage the authentication credentials in a centralized architecture by means of a trusted third party.
- Hardware-based authentication relies on the physical characteristics of some of the dedicated physical devices such as True Random Number Generator (TRNG), Physical Unclonable Function (PUF), Trusted Platform Module (TPM), or a hardware chip to store and process keys used in authentication. Hyperledger Fabric private chaincode (FPC) enables the execution of chaincodes using Intel SGX. It

is an open-source project developed by IBM and Intel to allow for a new form of smart contract that leverages Intel® Software Guard Extensions (Intel® SGX) Trusted Execution Environment (TEE) in achieving confidentiality and integrity in the Hyperledger Fabric blockchains.

The use of PBFT to reach consensus enhances the performance and the optional application of Fabtoken—the cypto-token in fabric can be used in case of payment aspects. The conventional issues of trust in a centralized authority, user and data privacy issues, single point of failure, human interaction errors, and network congestion issues can all be resolved by this approach.

## 4 Hyperledger Fabric v2.0 Features

Hyperledger Fabric was the first greenhouse project from Hyperledger to deliver a v1.0 release in July 2017. The collaborative effort from a growing community of users of fabric has led to several new features and enhancements thereby declaring a long-term support (LTS) with Hyperledger Fabric v1.4 in January 2019 and to the availability of a next stable release Hyperledger Fabric v2.0 in January 2020 with more important features introducing new options for operating nodes, enhanced governance of smartcontracts, support for new application patterns and privacy patterns and more. According to Forbes Blockchain 50 of 2019, 30 companies were using Hyperledger Fabric more than any other framework. This next generation of Hyperledger Fabric can help enterprises build their DLT solutions more efficiently and securely.

Fabric has an architecture that is permissioned, highly modular, and offers a unique approach to pluggable consensus that provides for superior performance and low latency of finality/confirmation (Fig. 1). To meet a broad range of use cases, it provides a number of approaches to privacy to meet the needs of the application. Developers can write smart contracts in widely adopted programming languages such as Go, Java, and JavaScript—it even supports Ethereum’s solidity smart contracts language. Key enhancements and new features of Hyperledger Fabric v2.0 which are utilized in our proposed mechanism include:

### 4.1 *Decentralized Chaincode (Smart Contract) Lifecycle Management*

Decentralized control over the smart contracts is accomplished through a new method of chaincode installation on peers and initiating them on a channel. The new lifecycle of fabric chaincodes enables the various organizations to decide on the chaincode parameters like the chaincode endorsement policy, before it is allowed to engage with the ledger.

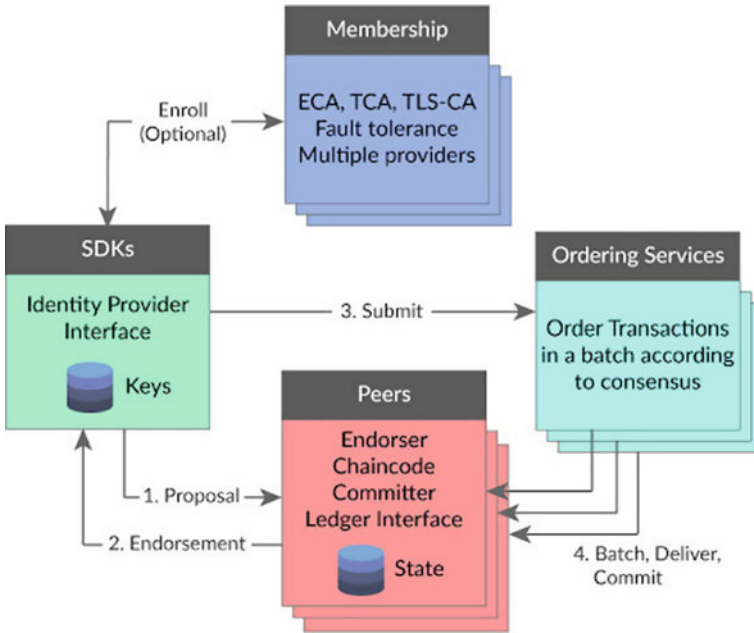


Fig. 1 Hyperledger Fabric components

The latest model provides a variety of lifecycle enhancements over the previous one. The v1.0 release allowed only one organization to set chaincode parameters while other could only decide on whether to install it or not. The new lifecycle model supports both centralized and decentralized trust models. This flexibility requires an agree of adequate number of organizations on an endorsement policy as well as all other detailing before deploying the chaincode on a channel. An upgrade for a chaincode require approval from a sufficient number of organizations.

### 4.2 New chaincode application patterns

To validate additional information before endorsing, a transaction proposal *Automated Checks* can be added by organizations to the chaincode functions to achieve collaboration and consensus and with *Decentralized Agreement* decisions made by the user can be modeled into a chaincode process that could span across multiple transactions.

### ***4.3 Private Data to Ensure Data Privacy***

Private data, also known as private data collection, allows participants of a channel to share private and confidential data without creating a new channel. Private data collection has two parts, *Actual private data* that are IDs that are not endorsed and recorded in the ledger and *hash of this data* that is endorsed and recorded in the ledger. Only participants can view both the private data and the hash in the ledger. Non-participants can view only the hash.

### ***4.4 Chaincodes Can Be Launched Externally***

This new feature allows companies to develop and deploy smart contracts with their existing technology or of their choice. The need of using Docker daemon for deploying smartcontracts has been eliminated, and now an organization may choose an external service like Kubernetes pod to deploy the chaincode to which peers can connect to and utilize for smartcontract execution.

### ***4.5 Improved Performance***

The performance bottleneck of read delays during endorsement and validation phases due to expensive lookups to an external CouchDB state database which has been clearly reduced in the new release with the introduction of a configurable cache on the peer.

### ***4.6 Modified Docker Images***

Hyperledger Fabric v2.0 ships with newly designed Docker images to work with Alpine Linux, a lightweight Linux distribution which takes less disk space on host system, and its minimalist nature reduces the risk of security vulnerabilities. With Alpine Linux operating system, the Docker images takes very less memory and provides quick download and startup times and can even run on resource constrained IoT devices, making it possible for fabric components to operate.

## 5 The Proposed Architecture and Analysis

Our architecture (Fig. 2) is a proposal for developing a flexible and reliable authentication and access control mechanism for the IoT devices and services that can operate in multiple segments or among different layers using Hyperledger Fabric v.20, making use of the newly introduced features (discussed in Sect. 4) in managing channel, chaincodes, policies, CA's, and so forth. The proposed architecture is supported by a proof-of-concept based on theoretical assumptions on the practical implementation.

The IoT devices that enroll to take part in a network are shared credentials from the MSP and MSP can be configured to communicate with the device to deliver updated credentials to the gateway configuration. The user does not have to bother in maintaining factors relating to authentication of the device. With successful authentication, the device is given permission to access the channels allocated to it. Access window is limited with the channel's accessibility to chaincodes hosted on the peer. The access rights are defined through the policies that govern the different components of the network. Hyperledger Fabric maintains a world state that allows the devices to avoid ledger update for all interactions. Querying on the network can be done by invoking smartcontracts and does not require any ordering and updation of the ledger. Information distribution is carried across the network device through a gossip protocol.

The different entities involved with their role in achieving authentication and access control in our proposed system are briefed below:

- **Peers** : A peer on the IoT fabric network represents an actor managing the devices. Peers as a fundamental factor forms the network, host chaincodes, maintains the ledger, and manages the transaction proposals and responses. With regular transaction updates, the peer keeps the ledger up-to-date. The gateway parameters stored in the IoT devices interact with the peers for establishing access to the network. At this point, the peers communicates with the membership service provider (MSP) for credential verification of the device. On successful authentication response, the peer allows the device to access its authorized channels. At this point of successful device authentication, the device can interact with the network to invoke the various smartcontracts for achieving operational functioning in the network.
- **Orderers** : Orderers keep the list of organizations (known as "consortium") which can create channels. This organization considering a smart-home example includes each house administrator actor who has enrolled for the network from the service provider. This list is stored in the configuration called "orderer system channel". This enhances in defining a basic form of access control by limiting read-write access to data and who has permission to configure it, and authority to make modification to elements of channel configuration is susceptible to the policies set by the respective administrators at the time of consortium or channel creation.
- **Policies** : Policies in IoT fabric network acts as the infrastructure management mechanism. Policies reflect how participants decide to approve or deny network changes, a channel, or a smart contract. The consortium members consent on policies when a network is initially configured but may also alter them as the



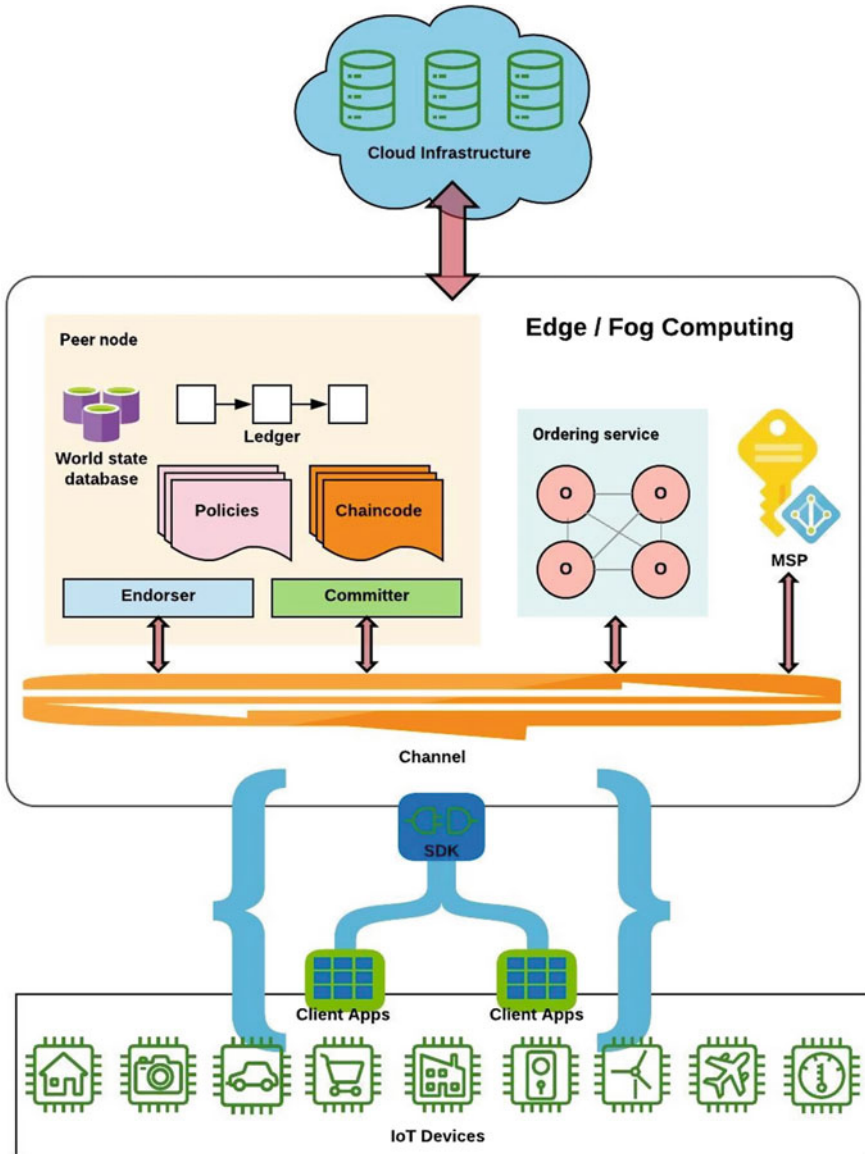


Fig. 2 Proposed architecture

network evolves. They describe, for example, the criteria for adding or removing members from a channel, blocks forming aspects or specify the count of organizations needed for a smart contract endorsement. When written, policies examine the collection of signatures attached with the transactions and proposals, and verify whether the signatures follow the network's acceptance criteria. Each of these activities are defined by a policy that determines who will carry out the action. In simple terms, the policy controls everything you want to do on a fabric network. Strictly speaking, policies in fabric play the major role in the authentication and access control mechanism.

- **Chaincode** : It is common to interchangeably use the words smart contract and chaincode among Hyperledger Fabric users. A smart contract usually defines the logic of transactions that regulates the lifecycle of a business entity embedded in the world state. The IoT devices that got authenticated accesses the functionalities provided to them by making use of these smartcontracts. Smartcontracts can be customized for each organizational need. Multiple related smartcontracts are bundled to form a chaincode that is then distributed to the peers in the network. Smart contracts can be seen as regulating transactions, while chaincode regulates how to package smart contracts for deployment. With the newly introduced features of Fabric v2.0, multiple chaincodes packed as a single package can be deployed on the same channel or on different channels with different names multiple times. Organizations can roll out individual minor fixes to chaincodes for their own use cases, and the changes to endorsement policy or configuration of private data collection can be performed without repackaging or reinstalling the chaincode.
- **MSP** : Membership service provider (MSP) is the component that handles the credentials (keys) in the network. Certificate authorities create the identity-representing certificates, while the MSP includes a list of the approved identities. It offers an abstraction to the membership operations, in particular all the cryptographic mechanisms and protocols behind issuance of certificates, certificate validation, and authentication of the users. The root of trust is achieved by maintaining a list of self-signed (X.509) CA (Certificate Authority) certificates that defines the Root CA, Intermediate CA's, TLS root CA for TLS certificate, and TLS intermediate CA's. One or more MSP's can be assigned in managing the network considering the business logic.

The transactional mechanics gives the details that occur during a standard exchange of assets. The transaction flow involved in achieving this is demonstrated as the following three phases:

Phase 1: A transaction proposal is being initiated by client application to a sub-cluster of peers who then invokes corresponding smartcontract that generates a proposed ledger update and thereby the results are endorsed. No ledger update happens at this point, instead a proposal response is returned to the client application by the endorsing peer.

Phase 2: Orderer do the vital process of collecting proposed transaction updates, do the validation checks, order the transactions, packaged into blocks, consented on, and is made available for distribution to among peers.

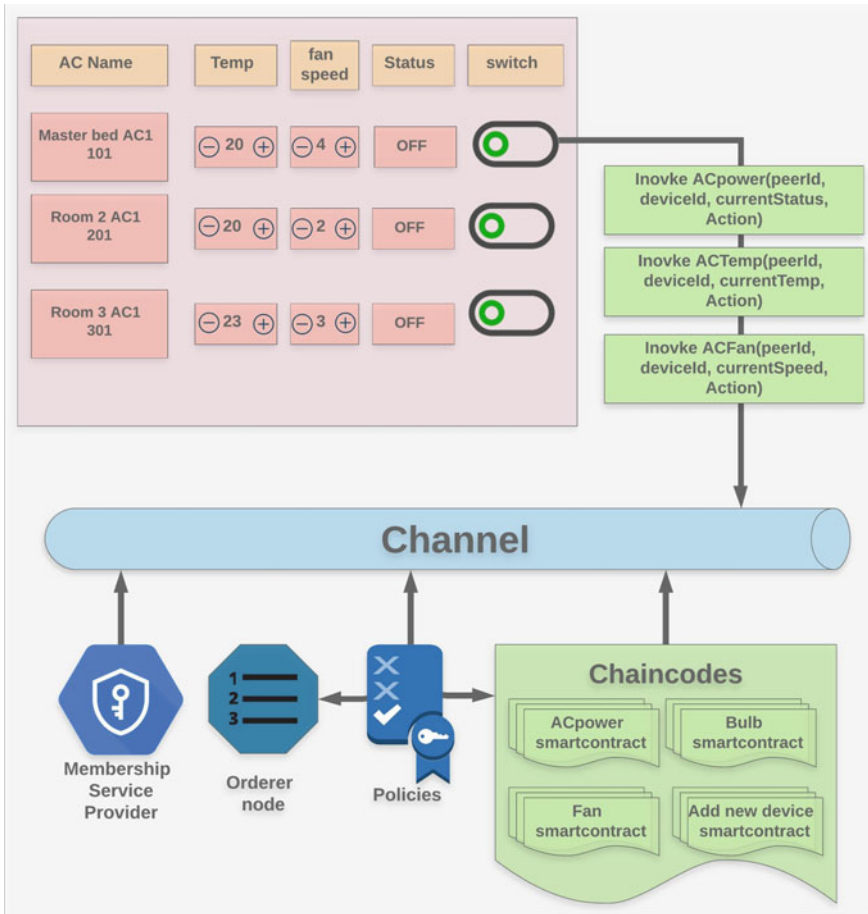


Fig. 3 Chaincode invocation flow

Phase 3: In this final phase, the peers receives the blocks distributed from the orderer, and subsequent validation is performed on these blocks and committed to the ledger. Blocks are ordered strictly which commissions each peer to verify that updates of transactions are implemented uniformly across the blockchain network.

A smartcontract invocation flow for performing different functionalities of an air conditioner (AC) is shown Fig. 3. When the user performs an intended action through the user interface, specific smartcontract function will be invoked for the action to be carried out through the API to the fabric network. The function along with the parameters are sent to the endorsing peers where the smartcontracts get executed depending on the validation system chaincode (VSCC). The different policies described for functioning like Endorsement policies, Channel Configuration policies, Signature and ImplicitMeta policies, and other policies provided by fabric along with the MSP

managing the certificates play the vital role in giving access to the authorized entities on the network and restricting them to perform malicious actions on the network.

The use of Hyperledger Fabric in managing the IoT device authentication and access control has the advantage of functioning of fabric components from the edge/fog nodes making the network isolated gives a decentralized control. The policies can be crafted for multiple actors to agree on an action to be performed. The membership service provider (MSP) manages the factors for authentication and authorization for access control to the network. Only the authorized members for a channels are allowed to interact with specific channel functionalities of the network.

The security requirements of IoT system such as distributed computing and storage, security, integrity, authentication and privacy of data, security and authentication of device, synchronization of software upgrades, key management, access issuance and revocation, identity management, enrolment, authentication, authorization, and privacy of user can be easily established and maintained through the use of Hyperledger Fabric and the various features offered from v2.0 onward. Fabric also provides restricted network and controlled access to user data.

Fabric can meet the performance requirements of fast retrieval of data from state database of couchDB which model assests as JSON makes querying easier which does not need a ledger query or update on the ledger. The distributed isolated clustering of fabric components for each users of the application to a local edge node level enables high throughput with reduces the communication overhead and faster agreeing on transactions. This model also provides scalability supporting network expansion and interaction with the cloud infrastructure.

## 6 Conclusion and Future Work

It is imperative to design and build a stable as well as safe blockchain-based IoT platform that meets the future demands of an autonomous digital world. At present, blockchain can provide IoT a platform for distributing trusted information that defies non-cooperative organizational structures. It is observed from the theoretical analysis and initial implementation, the use of Hyperledger Fabric can support in mitigating security issues among different layers including key agreement, confidentiality of data, authentication and encryption factors in perception layer, key management, authentication, and communication security in the network layer as well as information security management, authentication, and privacy protection in the application layer. The properties and advantages of fabric being discussed possess ability to thwart Man-in-the-Middle Attacks, Password change/Guessing attacks, Denial of Service (DOS) attacks, Sybil attacks, Sinkhole and Wormhole attacks, and Spoofing attacks and ensures secure communication. The newly introduced features of Hyperledger Fabric in v2.0 and the advancement in development of fabric and other blockchain technologies can pave a revolution in enhancing privacy, security, and availability to IoT architecture.

The upcoming IoT system need to be compatible with existing IoT technologies, and our mechanism has the potential in an economically feasible transformation from a conventional centralized model into a self-maintained decentralized structure. In addition, efficiency considerations should be given due consideration, alongside security concerns. Our future work will be solely on developing a performance enhanced implementation of this model as the central focus.

## References

1. T.M. Fernández-Caramés, P. Fraga-Lamas, A review on the use of blockchain for the internet of things. *IEE Access* **6**, 32979–33001 (2018)
2. S. Tuli et al., Fogbus: a blockchain-based lightweight framework for edge and fog computing. *J. Syst. Software* **154**, 22–36 (2019)
3. C. Qu et al. Blockchain based credibility verification method for IoT entities, in *Security and Communication Networks 2018* (2018)
4. S. Sathyadevan et al., Protean authentication scheme A time-bound dynamic KeyGen authentication technique for IoT edge nodes in outdoor deployments. *IEEE Access* **7**, 92419–92435 (2019)
5. J. Zahid, F. Hussain, A. Ferworn, Integrating internet of things and blockchain: use cases, in *Newsletter 2016* (2016)
6. E. Androulaki et al., Hyperledger fabric: a distributed operating system for permissioned blockchains, in *Proceedings of the Thirteenth EuroSys Conference* (2018), pp. 1–15
7. C. Gorenó et al. “Fastfabric: Scaling hyperledger fabric to 20,000 transactions per second”. In: 2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC). IEEE, 2019, pp. 455–463
8. H. Sukhwani, Performance modeling of PBFT consensus process for permissioned blockchain network (hyperledger fabric), in *IEEE 36th Symposium on Reliable Distributed Systems (SRDS)* (IEEE, 2017), pp. 253–255
9. C. Ma et al., The privacy protection mechanism of hyperledger fabric and its application in supply chain finance. *Cybersecurity* **2**(1), 1–9 (2019)
10. I. Ali, S. Sabir, Z. Ullah, Internet of things security, device authentication and access control: a review, in arXiv preprint [arXiv:1901.07309](https://arxiv.org/abs/1901.07309) (2019)
11. T. Yousuf et al., Internet of things (IoT) security: current status, challenges and countermeasures. *Int. J. Inform. Secur. Res. (IJISR)* **5**(4), 608–616 (2015)
12. M.A. Khan, K. Salah, IoT security: review, blockchain solutions, and open challenges. *Future Gener. Comput. Syst.* **82**, 395–411 (2018)
13. A.Z. Ourad, B. Belgacem, K. Salah, *Using blockchain for IOT access control and authentication management*, in *International Conference on Internet of Things* (Springer, Berlin, 2018), pp. 150–164
14. S. Huh, S. Cho, S. Kim, Managing IoT devices using blockchain platform, in *19th international conference on advanced communication technology (ICACT)* (IEEE, 2017), pp. 464–467
15. P. Punithavathi et al., A lightweight machine learning-based authentication framework for smart IoT devices. *Inform. Sci.* **484**, 255–268 (2019)
16. M. El-Hajj, Analysis of authentication techniques in internet of things (IoT), in *1st cyber security in networking conference (CSNet)* (IEEE, 2017), 1–3
17. M. Saadeh, Authentication techniques for the internet of things: a survey, in *cybersecurity and Cyberforensics Conference (CCC)* (IEEE, 2016), 28–34
18. D. Li et al., A blockchain-based authentication and security mechanism for IoT, in *2018 27th International Conference on Computer Communication and Networks (ICCCN)* (IEEE, 2018), pp. 1–6

19. I. Makhdoom et al., Blockchain's adoption in IoT: the challenges, and a way forward. *J. Network Comput. Appl.* **125**, 251–279 (2019)
20. M. El-hajj et al., A survey of internet of things (IoT) authentication schemes. *Sensors* **19**(5), 1141 (2019)

# Author Index

## A

Aanandhi, V. B., 63  
Abinaya, P., 423  
Agarwal, Shyamsundar, 467  
Aishwaraya, H. R., 167  
Alfiya, S., 53  
Amritha, P.P, 633  
Anudeep, J., 319, 437  
Arjun Rathya , R., 405  
Asha, P., 53  
Ashwin, V., 3

## B

Bajaj, Deepali, 247  
Balasubramanian, Karthi, 175, 657  
Balaswamy, Ch., 479  
Basavaraddi, Srushti, 277  
Bharti, Urmil, 247  
Bhatt, Uma Rathore, 91, 119, 389  
Binu Jose, A., 63

## C

Chandrasekar, A., 157  
Chouhan, Nitin, 91  
Cioffi, Vincenzo, 43  
Costanzo, Sandra, 43

## D

Das, Anshida, 63  
Das, Debjit, 491  
Devagopal, A. M., 3  
Devi, Salam Jayachitra, 525

Diana Josephine, D., 423

## G

Gadagi, Priyanka, 541  
Ganti, Sai Sarath Chandra, 347  
Garg, Manika, 217  
Gayathri, G., 3, 405  
Geetha, S., 593  
George, Gemini, 335  
Ghanta, Sandesh, 347  
Goel, Anita, 217, 247  
Gonge, Sudhanshu S., 667  
Gopakumar, G., 347  
Gopinath, Athira, 3, 405  
Gopinath, Greeshma, 53  
Gudi, Revathi, 541  
Gupta, S. C., 247  
Gutam, Bala Gangadhara, 189

## H

Harikumar, Sandhya, 203  
Hegade, Prakash, 231, 277  
Hegde, Vaibhav S., 467  
Hegde, Vibha, 231  
Hiremath, P. S., 467  
Honnvallli, B. Prasad, 619, 643

## I

Iyer, Nalini C., 147

## J

Jadhav, Kiran, 455

Jain, Sourabh, 231  
 Janet, B., 685  
 Jaya Sudha, J. S., 133  
 Jose, Jinesh, 293  
 Joseph, Shilpa, 203  
 Joshi, Rajaram M., 231  
 Justin Gopinath, A., 361

**K**

Kachavimath, Amit V., 605  
 Kalaimagal, G., 105  
 Kalambur, Subramaniam, 305  
 Kale, Tejaswini, 277  
 Keerthan, P. Karan, 643  
 Keerthan, U., 147  
 Khan, Usman, 277  
 Khavya, S., 657  
 Kodi, Charan Ramtej, 491  
 Kowshik, G., 319, 437  
 Krishnendhu, S. P., 557  
 Kulkarni, Akash, 147  
 Kullayappa Naik, K. C., 479  
 Kurian, Bibin, 699

**L**

Lingadhal, Nikhil, 277

**M**

Majjari, Sudhakar, 189  
 Mani, Medha, 75  
 Mary Saira Bhanu, S., 293  
 Mekathoti, Vamsikiran, 373  
 Melchizedek, Melissa Grace, 63  
 Menon, Athul, 3  
 Menon, Vishal, 405  
 Mishra, Deepak, 175, 657  
 Mohandas, Prabu, 557  
 Mulla, Mohammed Moin, 455  
 Muraleedharan, N., 685

**N**

Nair, Advithi, 305  
 Nair, Prashant R., 319, 437  
 Nandakumar, Nandagopal, 133  
 Narayana, Anoop, 643  
 Narayan, D. G., 455, 541, 605  
 Naveenkumar, P. B., 467  
 Nikitha, Padmanabha, 619  
 Nithya, B., 361, 373  
 Nived, P. A., 3

**P**

Patel, Manthan, 633  
 Patil, Prakashgoud, 467  
 Patil, Somashekar, 541  
 Patwari, Ashish, 75  
 Pawar, Manjula K., 467  
 Prameela, P. K., 541  
 Prasad, Abhiram, 405  
 Prathyusha, M., 619  
 Priyadarsan, Nived, 63  
 Punithavathi, P., 593

**Q**

Qureshi, Adil Masoud, 43

**R**

Radhika, S., 157  
 Raikar, Prateeksha, 167  
 Rajalakshmi, K. Ishwarya, 157  
 Rajashree, S., 619  
 Raveendran, Sarath, 133  
 Reddy, Gautham P., 643  
 Reddy, Patil Ramana, 479  
 Rizanov, Stefan, 261  
 Roshan Patnaik, M. P. V., 347

**S**

Sabu, Sheen, 335  
 Sai Shibu, N. B., 3, 405  
 Saleema, A., 573  
 Sam jasper, R., 633  
 Sandeep, Sidharth, 133  
 Sangeerthana, R., 53  
 Sarsodia, Tapes, 389  
 Satheesh Kumar, K., 53  
 Sekhar, Ravi, 15  
 Shah, Pritesh, 15  
 Shaji, Sen, 335  
 Shanmugavel, G., 33  
 Shekar, Shashi, 491  
 Shet, Raghavendra, 147  
 Shetty, Aishwaraya, 167  
 Shivanna, Gautham, 175  
 Siddamal, Saroja V., 167  
 Simon, Alka, 305  
 Singh, Buddha, 525  
 Singh, Sneha, 75  
 Sitaram, Dinkar, 305  
 Sree Harshitha, P., 509  
 Srinivasan, Gokul, 75  
 Sriram, Aiswarya, 305



Stoynova, Anna, [261](#)  
Subhash Chandra Mouli, D., [189](#)  
Subramanian, Narayanan, [699](#)  
Sunny, Minto, [335](#)  
Suriyaprabhaa, S., [53](#)  
Surya Chaitanya, P. V., [347](#)  
Surya Priya, M., [423](#)

**T**

Thampi, Sabu M., [573](#)  
Todorov, Dimitar, [261](#)

**U**

Upadhyay, Raksha, [91](#), [119](#), [389](#)

Uthaman, Udith, [335](#)

**V**

VaraPrasad, Raja, [509](#)  
Vasanthi, M. S., [33](#), [105](#)  
Vasudevan, Shriram K., [319](#), [437](#)  
Venkataraman, Hrishikesh, [509](#)  
Verma, Mausmi, [119](#)  
Vijeth, K. L., [231](#)  
Vineetha, B., [643](#)  
Vineeth, Chennuru, [319](#), [437](#)

**Y**

Yamuna, B., [175](#), [657](#)