

Jyotsna Kumar Mandal
Somnath Mukhopadhyay · Paramartha Dutta
Kousik Dasgupta *Editors*

Methodologies and Application Issues of Contemporary Computing Framework

Methodologies and Application Issues of Contemporary Computing Framework

Jyotsna Kumar Mandal · Somnath Mukhopadhyay
Paramartha Dutta · Kousik Dasgupta
Editors

Methodologies and Application Issues of Contemporary Computing Framework



Springer

Editors

Jyotsna Kumar Mandal
Department of Computer Science
and Engineering
University of Kalyani
Kalyani, West Bengal, India

Somnath Mukhopadhyay
Department of Computer Science
and Engineering
Assam University
Silchar, Assam, India

Paramartha Dutta
Department of Computer
and System Sciences
Visva-Bharati University
Santiniketan, West Bengal, India

Kousik Dasgupta
Department of Computer Science
and Engineering
Kalyani Government Engineering College
Kalyani, West Bengal, India

ISBN 978-981-13-2344-7

ISBN 978-981-13-2345-4 (eBook)

<https://doi.org/10.1007/978-981-13-2345-4>

Library of Congress Control Number: 2018952629

© Springer Nature Singapore Pte Ltd. 2018

This work is subject to copyright. All rights are reserved 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, express 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

Preface

This edited book volume entitled “Methodologies and Application Issues of Contemporary Computing Framework” is a small but sincere effort on the part of the present editors to offer a volume where different aspects of recent computing framework have been reported. The objective of this publication is to enlighten the researchers, scholars, students, and engineers about the state-of-the-art scenario regarding advances in latest computing techniques and associated applications, the latest tools and techniques which is applicable to almost all leading fields of current research. The theme is illustrated in various chapters to encourage researchers to adopt it in multidisciplinary research and engineering design. We hope that promising ideas and outstanding research results of this issue will in still further development of research and will enhance technologies in terms of methodologies and applications of computing paradigms.

This edited book volume entitled “Methodologies and Application Issues of Contemporary Computing Framework” is a collection of 14 chapters, which were initially peer-reviewed by the editorial review board members and reviewers who themselves span over many countries. A brief description of each of the chapters is as follows:

Chapter “[Structural Interactions Between H-NS and IHF with DNA](#)” is an attempt to characterize H-NS and IHF proteins by in silico molecular simulation methods. The models of H-NS and IHF were separately docked on to the corresponding promoter DNA regions of the genes encoding the virulence factors. The docked protein–DNA complexes were subjected to molecular dynamics simulations to identify the mode of binding. IHF was known to displace H-NS bound to the DNA. The molecular dynamics simulations study identified the molecular mechanism behind this action. So far, this is the first report that depicts the mechanistic details of the regulation of virulence gene expression by IHF and H-NS in *Vibrio cholerae*. This work may therefore be useful to illuminate the still obscure molecular mechanism behind the onset of cholera.

Chapter “[A Novel Design of Carry Select Adder \(CSLA\) for Low Power, Low Area, and High-Speed VLSI Applications](#)” presents a new design for carry select adder where the sum is generated and inverted for carry-in 0 and 1, respectively,

and the final selection of the sum is made with the help of multiplexer. The proposed design is extended to build 4-bit, 8-bit, 16-bit, and 32-bit carry select adder. The proposed design of 32-bit carry select adder shows an improvement up to 25.11%, 42.04%, and 24.70% in reduction of average power, propagation delay, and gate count, respectively, as compared to the best existing design of carry select adders available in the literature. Simulation and synthesis of all the circuits have been carried out in Cadence Virtuoso Tool using 45nm technology.

Chapter “[A Novel Objective Function Based Clustering with Optimal Number of Clusters](#)” suggests a novel objective function which is more effective in clustering with the optimal number of clusters detected through cluster validity indices. The proposed objective function-based clustering approach is implemented on different types of images, and the outcomes depict the effective performance in terms of cluster quality based on segmentation entropy and cluster partitioning time.

Chapter “[An Integral Image Based Text Extraction Technique from Document Images by Multilevel Thresholding Using Differential Evolution](#)” presents a multi-level image segmentation technique for extraction of texts from document images at a faster rate. Rectangular sum-table-based concept (known as an integral image) has been used here to find the local threshold value of images. An integral image obtained for a fixed window size is used to calculate threshold value by the proposed probability-based objective function. The achieved local threshold value is further optimized globally by the differential evolution (DE) algorithm to get multiple thresholds. DE provides fast and accurate convergence toward the optimal solution as compared to other well-known optimized algorithms like particle swarm optimization (PSO), genetic algorithm (GA). The proposed technique is applied to different types of degraded document images. The outcomes of the proposed approach are compared both quantitatively and qualitatively with state-of-the-art methods.

Chapter “[Offline Extraction of Indic Regional Language from Natural Scene Image Using Text Segmentation and Deep Convolutional Sequence](#)” presents a novel unified method for tackling the above challenges. The proposed work uses an image correction and segmentation technique on the existing Text Detection Pipeline an Efficient and Accurate Scene Text Detector (EAST). EAST uses standard PVAnet architecture to select features and non-maximal suppression to detect text from the image. Text recognition is done using combined architecture of MaxOut Convolution Neural Network (CNN) and Bidirectional Long Short Term Memory (LSTM) network. After recognizing text using the Deep Learning-based approach, the native languages are translated to English and tokenized using standard Text Tokenizers. The tokens that very likely represent a location are used to find the Global Positioning System (GPS) coordinates of the location, and subsequently, the regional languages spoken in that location are extracted. The proposed method is tested on a self-generated dataset collected from Government of India dataset and experimented on standard dataset to evaluate the performance of the proposed technique. A comparative study with a few state-of-the-art methods on text detection, recognition, and extraction of regional language from images shows that the proposed method outperforms the existing methods.

Chapter “[Understanding Email Interactivity and Predicting User Response to Email](#)” presents a binary classification model to predict an email recipient response, based on certain email meta-data and recipient email usage characteristics. For this task, authors study a 2016 HackerRank contest email dataset. They first identify email interactivity patterns separately for each recipient response and then understand the factors responsible for determining the user response to an email. They then propose a novel feature selection methodology where they perform dataset profiling based on user and sent day characteristics, using k-means clustering. It is observed that the decision tree classifier performs the best with an F1 score of 0.6279 and the fraction of emails opened by the recipient in the past is the most significant feature for the model.

Chapter “[Load Balancing of Unbalanced Matrix Problem of the Sufficient Machines with Min-Min Algorithm](#)” presents an algorithm for load balancing and performance with minimization completion time and throughput. They apply a very famous Hungarian method to balance all loads in distributing computing. Hungarian technique helps them to minimize the cost matrix problem also.

Chapter “[Cohort Assembly: A Load Balancing Grouping Approach for Traditional Wi-Fi Infrastructure Using Edge Cloud](#)” gives a discussion on how to decrease the load by using edge cloud through movable devices for video circulation applications. Here, a group of peoples’ smartphones acts as self-managed cloud edges to interchange video replays without interrupting the intermediate main cloud server or the access points either. They have accomplished the experiment in the T-20 cricket tournament, held in Eden Garden with a group of students using proposed application. This experiment firmly suggests that the load in the access points can be reduced by the edge clouds of Wi-Fi infrastructure and using the grouping technique replay of two or more games can also be conducted simultaneously using the smart devices only.

Chapter “[Design of Fuzzy-PI Controller for Hybrid Distributed Generation System Using Grey Wolf Optimization Algorithm](#)” proposes a fuzzy-PI controller for damping the oscillation of system frequency of distributed energy generation. Solar photovoltaic (PV), wind turbine generator (WTG), diesel engine generator (DEG), fuel cell (FC), aqua-electrolyzer (AE), flywheel energy storage system (FESS), and battery energy storage system (BESS) are employed in the distributed energy generation. The controllers are used before FESS, BESS, and DEG, and the frequency deviation signals with time delay have been considered as feedback to the controller. The gains of controllers have been optimized using gray wolf optimization algorithm (GWOA), and the dynamic responses have been compared with proportional–integral (PI) controllers optimized with GWOA. From the results, it is found that fuzzy-PI controllers perform well in terms of peak deviations and damping of oscillations.

Chapter “[Improved Musical Instrument Classification Using Cepstral Coefficients and Neural Networks](#)” proposes a novel method for an automated classification of musical instruments based on the analysis of the audio signals generated by them. The chapter studies the effectiveness and efficiency of a number of features like Mel frequency cepstral coefficients (MFCC), harmonic pitch class

profile (HPCP), linear predictive coding (LPC) coefficients, spectral centroid, and pitch salience peaks with cepstral coefficients (CC) with multiple machine learning algorithms like artificial neural network (ANN), K-nearest neighbors (K-NN), support vector machine (SVM), and random forest. The analysis finds that CC surpassed all other features to provide maximum accuracy and robustness along with neural networks. Multiple datasets have been used in the experimentations to remove the possibility of bias. The overall accuracy obtained ranged between 90% and 93%.

Chapter “[A Factual Analysis of Improved Python Implementation of Apriori Algorithm](#)” deals with comparative study and critical analysis of various implementations of Apriori algorithm present in different Python packages and implemented another version of the algorithm which is at par with the existing algorithms but without using any existing libraries available in Python. A detailed conclusion has been drawn on their empirical study of Apriori algorithm which will help the researcher to understand and use different packages for data analysis.

Chapter “[User Authentication with Session Key Interchange for Wireless Sensor Network](#)” highlights several limitations of Nam et al.’s protocol such as denial of service, no login phase verification, clock synchronization problem, improper mutual authentication, key-compromised impersonation and man-in-the-middle attack and unverified password update phase. Hence, the authors put forward an efficient two-factor user authentication with session key interchange protocol for WSN (ASKI-WSN) based on elliptic curve cryptography (ECC) to overcome these limitations. The proposed ASKI-WSN scheme is informally verified under several security threats and formally validated using widely accepted automated validation of Internet security protocols and applications (AVISPA) tool. Finally, the performance analysis of ASKI-WSN scheme in terms of security and computation cost shows its efficiency for practical applications as compared to some existing protocols.

Chapter “[Secured Mobile Collaborative Application in Cloud Environments](#)” gives a design of a cryptography-based security model mixed with dynamic early detection technique to manage congestion and provide more secured service. The main aim is to serve the well-secured collaborative application for achieving high-quality collaborative work in the mobile cloud paradigm. In the cryptographic technique, they use the Diffie–Hellman cryptosystem for designing the model, and with the help of dynamic early detection technique, they want to reduce congestion that is created by third party for interrupting mobile collaborative application in case of shared resources.

Chapter “[Investigation of the Effect of Barrier Layer Engineering on DC and RF Performance of Gate-Recessed AlGaN/GaN HEMT](#)” investigates the E-mode AlGaN/GaN high electron mobility transistors over its DC and RF performance. The effect of adding an additional barrier layer of Al_xGa_{1-x}N with a different mole fraction of Al in the two layers has been reported. Simulation results indicate an enhancement in the peak transconductance as well as in the *f_t* and *f_{max}* values as compared to the single-barrier layer E-mode HEMT, thereby indicating an

improvement in the DC and RF characteristics resulting from the barrier layer engineering of the device.

Contributions available in the 14 chapters, after being meticulously reviewed, reflect some of the latest sharing of researchers of the concerned field. The editors want to avail this opportunity to express their sincere gratitude to all the contributors for their efforts in this regard without which this edited volume could have never come to a reality. The editors sincerely feel that the success of such an effort in the form of edited volume can be academically meaningful only when it is capable of drawing significant contributions from good researchers in the relevant field.

The editors also thank the reviewers who are leading researchers in the domain to find their time from their busy schedules to give valuable suggestions to improve the quality of the contributed articles. Last but not least, the editors are inclined to express their sincere thanks to Springer Nature, Singapore, for being the publishing partner. But for their acceptance to publish this volume would never have been possible in the present standard.

Enjoy reading it.

Kalyani, India

Silchar, India

Santiniketan, India

Kalyani, India

September 2018

Jyotsna Kumar Mandal

Somnath Mukhopadhyay

Paramartha Dutta

Kousik Dasgupta

The original version of this book was revised.
Preface has been included in the frontmatter
of this book.

Contents

Structural Interactions Between H-NS and IHF with DNA	1
Angshuman Bagchi	
A Novel Design of Carry Select Adder (CSLA) for Low Power, Low Area, and High-Speed VLSI Applications	13
Nilkantha Rooj, Snehanjali Majumder and Vinay Kumar	
A Novel Objective Function Based Clustering with Optimal Number of Clusters	23
Kuntal Chowdhury, Debasis Chaudhuri and Arup Kumar Pal	
An Integral Image Based Text Extraction Technique from Document Images by Multilevel Thresholding Using Differential Evolution	33
Rupak Chakraborty, Rama Sushil and Madan L. Garg	
Offline Extraction of Indic Regional Language from Natural Scene Image Using Text Segmentation and Deep Convolutional Sequence	49
Sauradip Nag, Pallab Kumar Ganguly, Sumit Roy, Sourab Jha, Krishna Bose, Abhishek Jha and Kousik Dasgupta	
Understanding Email Interactivity and Predicting User Response to Email	69
Soumyadeep Roy, Nibir Pal, Kousik Dasgupta and Binay Gupta	
Load Balancing of Unbalanced Matrix Problem of the Sufficient Machines with Min-Min Algorithm	81
Ranjan Kumar Mondal, Payel Ray, Enakshmi Nandi, Biswajit Biswas, Manas Kumar Sanyal and Debabrata Sarddar	
Cohort Assembly: A Load Balancing Grouping Approach for Traditional Wi-Fi Infrastructure Using Edge Cloud	93
Deepsubhra Guha Roy, Madhurima Das and Debashis De	

Design of Fuzzy-PI Controller for Hybrid Distributed Generation System Using Grey Wolf Optimization Algorithm	109
Santu Giri and Parthasarathi Bera	
Improved Musical Instrument Classification Using Cepstral Coefficients and Neural Networks.	123
Shruti Sarika Chakraborty and Ranjan Parekh	
A Factual Analysis of Improved Python Implementation of Apriori Algorithm	139
Kartick Chandra Mondal, Biswadeep Deb Nandy and Arunima Baidya	
User Authentication with Session Key Interchange for Wireless Sensor Network	153
Prasanta Kumar Roy, Krittibas Parai and Abul Hasnat	
Secured Mobile Collaborative Application in Cloud Environments	167
Enakshmi Nandi, Ranjan Kumar Mondal, Payel Ray and Debabrata Sarddar	
Investigation of the Effect of Barrier Layer Engineering on DC and RF Performance of Gate-Recessed AlGaN/GaN HEMT	177
Shubham Mondal, Sritoma Paul and Angsuman Sarkar	

About the Editors

Prof. Jyotsna Kumar Mandal is Professor at the Department of Computer Science and Engineering and former Dean of the Faculty of Engineering, Technology and Management, Kalyani University, West Bengal. He obtained his M.Sc. in physics from Jadavpur University in 1986 and M.Tech. in computer science from the University of Calcutta, and he was awarded a Ph.D. in computer science and engineering by Jadavpur University in 2000. He has teaching and research experience of nearly 28 years. His areas of research include coding theory, data and network security, remote sensing and GIS-based applications, data compression, error correction, visual cryptography, and steganography.

He has published over 156 papers in various leading international journals and has also edited 29 volumes for various major publishers, including Springer.

Dr. Somnath Mukhopadhyay is currently Assistant Professor at the Department of Computer Science and Engineering, Assam University, Silchar, India. He completed his M.Tech. and Ph.D. degrees in computer science and engineering at the University of Kalyani, India. He has coauthored one book, edited five books, and published over twenty papers in various international journals and conference proceedings, as well as three chapters in edited volumes. His research interests include digital image processing, computational intelligence, and pattern recognition.

He is a member of the IEEE and IEEE Computational Intelligence Society, Kolkata Section; a life member of the Computer Society of India; and currently, the regional student coordinator (RSC) of Region II, Computer Society of India.

Dr. Paramartha Dutta is currently Professor at the Department of Computer and System Sciences, Visva-Bharati University, West Bengal, India. Prior to this, he served at Kalyani Government Engineering College and College of Engineering in West Bengal as a full-time faculty member. He completed his bachelor's and master's degrees in statistics from the Indian Statistical Institute, Calcutta, in 1988 and 1990, respectively. Subsequently, he earned his master of technology in computer science from the same institute in 1993. Later, in service, he completed

his Ph.D. in engineering from the Bengal Engineering and Science University, Shibpur, in 2005.

He has coauthored eight books and edited six books. He has published over 200 papers in various journals and conference proceedings, both international and national, as well as several chapters in books from reputed international publishing houses like Elsevier, Springer-Verlag, CRC Press, and John Wiley. He has also served as an editor of special volumes of several prominent international journals.

Dr. Kousik Dasgupta is working as Assistant Professor in Department of Computer Science and Engineering, Kalyani Government Engineering College, Kalyani, India. He completed his B.E. in Electronics and Power Engineering from Nagpur University in 1993 and M.Tech. in Computer Science and Engineering from the West Bengal University of Technology in 2007, and he was awarded a Ph.D. by the Department of Computer Science and Engineering, Faculty of Engineering, Technology and Management, University of Kalyani, in 2017.

He has presented numerous papers at various conferences and has published book chapters and coauthored books in international publishing houses such as John Wiley, IGI Global etc. He is a member of five scientific and professional societies, including the Association for Computing Machinery (ACM) and Indian Society for Technical Education.

Structural Interactions Between H-NS and IHF with DNA



Angshuman Bagchi

1 Introduction

Pathogenic bacteria *Vibrio cholerae* can increase or decrease the appearance of its genes for virulence by some specific and global transcriptional regulators in response to environmental conditions. It is known that the production of the bacterial virulence factors, the toxin-co-regulated pilus (TCP), and cholera toxin (CT) are responsible for the severe watery diarrhea in *Vibrio cholerae* infected individuals. The expressions of these virulence genes are regulated mainly by a transcription factor called Integration Host Factor (IHF). IHF is a positive regulator of transcription of virulence genes. The transcriptional regulatory activity of IHF is however, dependent on the presence of different other specific and global transcriptional regulators, notably among them are ToxT—a specific regulator and H-NS—a global regulator [1–4]. Earlier studies revealed that H-NS act as a homo-dimer and binds to the DNA with consensus motif TCGATAAATT. On the other hand, IHF is a hetero-dimer having subunits IHF α and IHF β , which after binding to the DNA can bend it by 180° [1, 5, 6]. It has been proposed that the activations of virulence genes by IHF occur in presence of H-NS [1]. H-NS being the repressor of transcription of the virulence genes, and the binding of the H-NS protein to the DNA prevents transcription of the virulence genes. IHF initiates the transcription process by displacing H-NS from the DNA. However, till date, the detail molecular mechanism of this process of virulence gene activation is not reported. In the present scenario, an attempt has been made to analyze the molecular biochemistry of the virulence gene activation process in *Vibrio cholerae* by IHF from a structural point of view. No structures of the IHF α , IHF β , and H-NS

A. Bagchi (✉)

Department of Biochemistry and Biophysics, University of Kalyani, Kalyani, Nadia, India
e-mail: angshu@klyuniv.ac.in; angshumanb@gmail.com

from *Vibrio cholerae* are available till date. So, an attempt has been made to build the homology models of these proteins by structural bioinformatics approach. The models of the proteins have been docked with the corresponding DNA regions and then the docked protein–DNA complexes have been subjected to molecular dynamics simulations (MD). The binding energies of the complexes have been calculated from the MD simulations. The results of the MD studies have been analyzed to predict the probable mode of binding. The analyses could further reveal the details of the DNA-IHF interactions as well as H-NS-DNA interactions and finally DNA-H-NS-IHF interactions reveal the mechanism of displacement of H-NS by IHF from the DNA. This is the first bioinformatics approach towards the understanding of the molecular mechanism of the virulence gene expression by *Vibrio cholerae* from a structural point of view. This study may therefore be useful for future genetic studies to analyze the functionalities of the various amino acid residues from IHF and H-NS in DNA binding and interactions among themselves. The interacting amino acids from these proteins may be targeted to develop new drugs to prevent the spread of *Vibrio cholerae* infections.

2 Materials and Methods

2.1 Sequence Analysis and Homology Modeling of IHF α , IHF β and H-NS

The amino acid sequences of IHF α and H-NS were extracted from GenBank (id: 147673436 for IHF α and 147673645 for H-NS) and from Swissprot (id: A5F6Y4) for IHF β . These sequences were applied for searching the Brookhaven Protein Data Bank (PDB) [7] with the software BLAST [8] to find suitable templates for homology modeling. For IHF α , the BLAST search result picked up the X-ray crystal structure of single chain Integration Host Factor protein (sciIHF2) in complex with DNA from *E. coli* (PDB id: 2IIE, Chain A) with 75% sequence identity. For IHF β , the BLAST search result chose the X-ray crystal structure of mutant IHF (BetaE44A) complexed with the native H' site from *E. coli* (PDB id: 1OWF, Chain B) again with 75% sequence identity. For H-NS, the BLAST result showed 73% sequence similarity with H-NS (DNA BINDING DOMAIN) (PDB id: 1HNR, Chain A). Then homology models of the proteins were built using Modeler [9] using the corresponding templates (2IIE, Chain A for IHF α , 1OWF, Chain B for IHF β and 1HNR, Chain A for H-NS respectively). Since 1HNR is an NMR structure, the structure was first energy minimized using CHARMM [10] force fields before being utilized as the template to build the model of H-NS. The modeled structures of the proteins were then subjected to structural alignments. The root mean squared deviations (RMSD) for the superimpositions were found to be 0.5 Å (for IHF α and IHF β on to their cor-

responding crystal templates) and 0.8 Å for H-NS on 1HNR, Chain A. The models of the proteins were then energy minimized in two steps. In the first step, the modeled structures were minimized without fixing the backbones. In the second step, the energy minimizations were done by fixing the backbones of the modeled proteins to ensure proper interactions. All energy minimizations were done with the help of conjugate gradient (CG) with CHARMM force fields until the structures reached the final derivative of 0.01 kcal/mole.

2.2 Validation of the Models

The main chain properties of the modeled proteins were found to be good having no considerable bad contacts nor C_α tetrahedron distortions nor hydrogen bond (H-bond) energy problems. The side chain properties were also accurately predicted as observed from the results of the side chain torsion angles. The Z-scores calculated with PROSA (<https://prosa.services.came.sbg.ac.at/prosa.php>) would reveal that the predicted structure was a good homology model. The residue profiles of the three-dimensional models were further checked by VERIFY3D [11]. PROCHECK [12] analyses were performed in order to assess the stereo-chemical qualities of the models and Ramachandran plots [13] were drawn. No residues were found to be present in the disallowed regions of the Ramachandran plots of the modeled proteins.

2.3 Building the Model of the Promoter DNA

In order to find the interactions between promoter DNA and the proteins, the nucleotide sequences of the promoter regions from *Vibrio cholerae* were extracted. The nucleotide sequences of the binding regions for IHF and H-NS were taken from [1]. These nucleotide sequences were used separately to build two models of the corresponding DNA regions using the CHARMM software tool and then subjected to energy minimizations. The resulting energy minimized structures were used for docking studies.

2.4 Molecular Docking Simulation of IHF with the Corresponding DNA Region

It was known that IHF binds to DNA as a hetero-dimer of IHF α and IHF β [1]. Thus, a model of IHF hetero-dimer complex was built by docking IHF α with IHF β using the software GRAMM [14]. The hetero-dimeric model of IHF obtained after docking studies was subjected to energy minimization as per the protocol previously

mentioned in Sect. 2.1. In order to elucidate the mode of binding between the DNA and the IHF hetero-dimeric protein, the model of the IHF hetero-dimeric protein and the DNA were docked using the software patchdock [15]. The docked structure of the DNA–protein complexes that yielded the best score was selected and analyzed visually using DS modeling software suite. The docked complex was then energy minimized as per the protocol previously mentioned in Sect. 2.1.

2.5 *Molecular Docking Simulation of H-NS with the Corresponding DNA Region*

H-NS also binds to DNA as a homo-dimer. Therefore, a dimeric model of H-NS was built by docking the individual monomeric units together using the software GRAMM. The homo-dimeric model of H-NS obtained after docking studies was subjected to energy minimization as per the protocol previously mentioned in Sect. 2.1. In order to elucidate the mode of binding between the DNA and the H-NS protein, the homo-dimeric model of the H-NS protein and the DNA were docked using the software patchdock. The docked structure of the DNA–protein complexes that yielded the best score was selected and analyzed visually using DS modeling software suite. The docked complex was then energy minimized as per the protocol previously mentioned in Sect. 2.1.

2.6 *Molecular Dynamics (MD) Simulation of H-NS-DNA Complex*

The MD simulation of the DNA-H-NS protein complex was performed with the CHARMM module of DS modeling software suite. The initial coordinates were extracted from the energy-minimized structure of the DNA-H-NS docked complex. The complex was then placed in an orthorhombic box having dimensions preventing self-interactions. The system was solvated with adequate water molecules at the typical density of water at 298 K and 1.0 atm utilizing single point charge (SPC) model. The whole system was energy minimized keeping the temperature constant to the body temperature of 310 K using NPT dynamics protocol. A 100 ns dynamics run was then performed with the DNA–protein complex. The modes of interactions between H-NS and the corresponding DNA were then analyzed using DS modeling software suite.

2.7 Calculation of Binding Free Energy of the H-NS-DNA Complex

In order to have a quantitative estimation of the interactions between H-NS and DNA, the H-NS-DNA complex was analyzed by the FoldX server [16]. The average solvated coordinate of the H-NS-DNA complex generated from the MD simulation was used as input to FoldX. FoldX calculates the free energy as the sum of the different energetic contributions along with entropy and temperature factors [16].

2.8 Molecular Dynamics (MD) Simulation of IHF-DNA Complex

The MD simulation of the DNA–protein complex was performed using the CHARMM module of DS modeling software suite. The initial coordinates were extracted from the energy-minimized structure of the DNA-IHF docked complex. The complex was then placed in an orthorhombic box having dimensions preventing self-interactions. The system was solvated with adequate water molecules at the typical density of water at 298 K and 1.0 atm utilizing single point charge (SPC) model. The whole system was energy minimized keeping the temperature constant to the body temperature of 310 K using NPT dynamics protocol. A 100 ns dynamics run was then performed for the DNA–protein complex. The modes of interactions between IHF and the corresponding DNA were then analyzed using DS modeling software suite.

2.9 Calculation of Binding Free Energy of the IHF-DNA Complex

For the purpose of comparison of the binding energies of H-NS with DNA and IHF with DNA, the complex of IHF with DNA was analyzed by the FoldX server. The average solvated coordinate of the IHF-DNA complex generated from the MD simulation was used as input to FoldX. To analyze the DNA bending capability of the IHF protein, the cationic residue density (Cpc) and DNA phosphate crowding (Cpp) were determined as in [17].

2.10 Molecular Docking and Dynamics (MD) Simulation of the Docked Complex of H-NS-DNA with IHF

It was proposed that IHF hetero-dimer can displace H-NS bound to DNA [1]. Therefore, the model of the IHF hetero-dimeric complex was used to dock with the DNA-H-NS complex using the software GRAMM as per the protocol mentioned in Sect. 2.4. The docked structure of the DNA–protein complexes that yielded the best score was selected and analyzed visually using DS modeling software suite. The docked complex was then energy minimized as per the protocol previously mentioned in Sect. 2.1. The resulting energy-minimized structure was subjected to MD simulation using the CHARMM module of DS modeling software suite as per the steps mentioned before in Sects. 2.6 and 2.8.

2.11 Calculation of Binding Free Energy of the H-NS DNA Complex with IHF

It was known that IHF hetero-dimer can displace H-NS from H-NS-DNA complex [1]. Thus to quantify the interactions, the average solvated coordinate of the H-NS-DNA-IHF complex generated from the MD simulation was used as input to FoldX. The results of the binding free energies from all the aforementioned complexes were compared to justify the mechanism of the interactions.

3 Results

3.1 Structures of IHF and H-NS Proteins from *Vibrio Cholerae*

The IHF protein is a hetero-dimeric protein having two components IHF α and IHF β . The IHF α and IHF β components have only 35% sequence identity [1]. The sequences of the two proteins were therefore used separately to build their corresponding models. IHF α was built on the template of 2IIE, Chain A. The protein was an alpha beta protein with the N-terminal having two helices followed by a beta sheet having five antiparallel beta strands culminating in another helix (Fig. 1).

The IHF β protein was built on the template 1OWF, Chain B. This protein was also an alpha beta protein with the N-terminal having two helices followed by a beta sheet having five antiparallel beta strands culminating in another helix (Fig. 2). The difference between the structures of IHF α and IHF β was in the presence of typeII

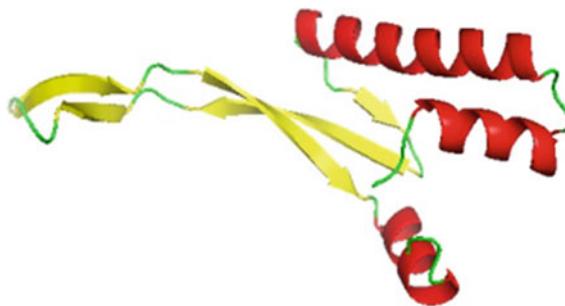


Fig. 1 Cartoon representation of the model of IHF α protein. The helices are shown in red. The sheets are shown in yellow. The remaining portions are coil regions

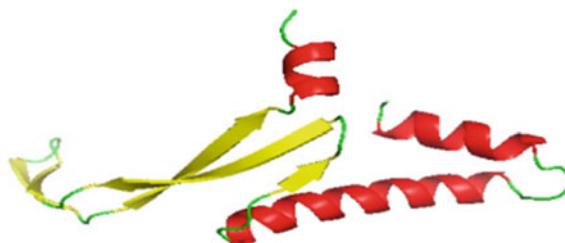


Fig. 2 Cartoon representation of the model of IHF β protein. The helices are shown in red. The sheets are shown in yellow. The remaining portions are coil regions

turn regions. The IHF α had two such turns whereas the IHF β had four such turn regions. However, when the backbone atoms of the models of the IHF α and IHF β were superimposed on to each other, the RMSD was found to be 2.21 Å. This large RMSD difference between the backbone atoms of the IHF α and IHF β proteins implied that though the two proteins had the same sets of secondary structural elements, their structural organizations were different.

The structure of H-NS was built using 1HNR, Chain A as the template. The protein was mainly alpha helical joined by loops. The C-terminal region of the protein had the signature sequence of a global transcriptional regulator. This region had an abundance of charged residues making H-NS capable of binding to the DNA. The surface charge distribution of H-NS protein was presented in Fig. 3.

3.2 Interaction of IHF with DNA

The model of IHF hetero-dimer was docked onto the model of the promoter DNA region. The dimerization of the two components of the IHF protein (i.e., the IHF α and IHF β) led to the formation of a compact globular domain with an extension of the beta sheet region from IHF α . The beta sheet region was found to contact the

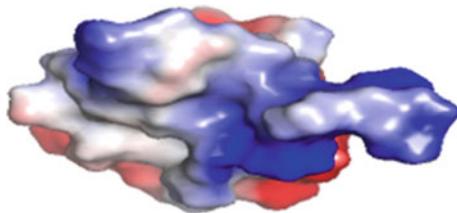


Fig. 3 The surface charge distribution of H-NS. The electrostatically positive surface (blue) of the protein binds to the electrostatically negatively charged DNA

DNA in its minor groove. Analysis of the IHF hetero-dimeric structure revealed that the $\text{IHF}\alpha$ had comparatively less exposed surface area than the $\text{IHF}\beta$ as measured by PDBePISA server; the accessible surface area of $\text{IHF}\beta$ was 8116 \AA^2 whereas that of the $\text{IHF}\alpha$ was 7867 \AA^2 . The docked complex of IHF hetero-dimer and DNA was subjected to MD simulations after energy minimization. The MD simulation results revealed that the $\text{IHF}\beta$ protein was mainly responsible for binding to the DNA and the binding occurred mainly via hydrophobic stacking interactions between the DNA bases and the nonpolar side chains of the $\text{IHF}\beta$ protein. The residues from $\text{IHF}\beta$ protein that were involved in the DNA binding were Leu6, Ile7, Leu17, Leu30, Ile43, Val60, and Val70. However, there were interactions between the phosphate backbone of the DNA as well as the charged amino acid residues of the $\text{IHF}\beta$. The amino acids from $\text{IHF}\beta$ that were involved in ionic interactions as well as hydrogen bonding with the phosphate backbone of the DNA were Lys3, Arg9, Lys20, Lys27, and Lys90. Figure 4 represents the electrostatic surface charge distribution of the protein bound to the DNA. The nonpolar region on the protein surface (colored green) predominantly was bound to the hydrophobic regions on the DNA bases. Interestingly, with the passage of MD run, it was observed that the DNA was getting bent. The bending of the DNA allowed the $\text{IHF}\alpha$ protein to come in contact with the DNA leading to the formation of DNA- $\text{IHF}\alpha$ protein interactions. The side chains of the amino acid residues, viz., Arg21, Lys24, Arg60, Arg76, and Arg77 of $\text{IHF}\alpha$ protein came near the phosphate backbone of the DNA after getting bent. This led to stronger interactions between the DNA and the IHF protein as a whole. The binding energy of the DNA-IHF complex was measured to be 15.732 kcal/mole as measured by FoldX server. The DNA bending capacity of the IHF protein was verified by measuring the correlation between Cpc and Cpp as in [17] and the value of correlation was found to be 0.31(>0.25). This means that IHF indeed can bend the DNA as per [17].

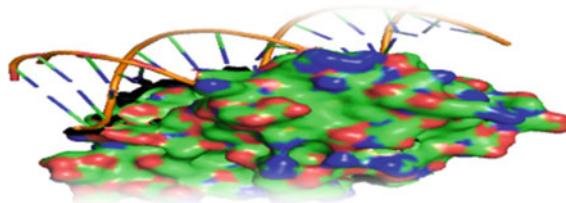


Fig. 4 Interactions between DNA and the IHF protein. The hydrophobic protein surface (colored green) was attached to the hydrophobic DNA bases

3.3 *Interaction of H-NS with DNA*

The model of H-NS-DNA complex was obtained by docking H-NS with DNA followed by MD simulations of the complex. The analysis of the complex obtained after MD simulation revealed that mainly the C-terminal helical region of the H-NS protein was involved in DNA binding. The interactions were mainly phosphate backbone interactions involving the side chains of Lys110, Gln115, Gln123, and Lys130 from H-NS protein. However, no DNA bending was observed in this case. The binding energy of the DNA-IHF complex was measured to be 11.156 kcal/ mole as measured by FoldX server. The binding energy value suggests that IHF interacts more strongly with DNA than H-NS.

3.4 *Interaction of IHF Hetero-Dimer with H-NS-DNA Complex*

It was known that IHF hetero-dimer can displace H-NS from H-NS-DNA complex [1]. In order to study the details of the interactions, the model of IHF hetero-dimer was docked on to the model of H-NS-DNA complex followed by MD simulation. It was revealed that when IHF was allowed to dock with the H-NS-DNA complex, IHF α starts interacting with H-NS in a stronger way than H-NS was doing with DNA. The amino acid residues of the C-terminal helical region of H-NS were found to be getting involved in interaction with IHF α as revealed by the MD simulation run. The amino acid residues Asp104, Lys110, Gln115, Arg116, Lys124, Asp127, Glu128, and Lys130 from H-NS protein were bound to Lys5, Glu10, Lys20, Glu32, and Glu33 of IHF α . This led to displacement of H-NS from its DNA bound state facilitating the binding of IHF β to the DNA as more number of amino acid residues from IHF β started interacting with the DNA (12 residues from IHF β as compared to 4 residues from H-NS). In other words, interactions between IHF α and H-NS became stronger than interactions of H-NS with the DNA and consequently, the interactions between IHF β and DNA became stronger than the corresponding interactions of H-NS with DNA. This way the repressions of virulence genes by H-NS are abolished and the

gene expression ensues. The IHF protein thus acts as an activator of transcription of the virulence genes. The overall binding energy of the H-NS-DNA-IHF complex was measured to be 18.596 kcal/mole as measured by FoldX server. This also signifies that the interactions between H-NS and IHF α are stronger than the interactions between H-NS and DNA. Thereby, IHF can displace H-NS from its DNA bound state.

4 Discussions

In the present work, an attempt has been made to analyze the probable molecular details of the interactions of IHF and H-NS with the corresponding DNA regions to predict the mode of the virulence gene expression. It was known that the expression of the virulence genes by *Vibrio cholerae* occurs by a complex interplay between IHF, H-NS, and the corresponding promoter DNA [1]. Since there were no previous reports regarding the structural aspects of the mode of interactions, molecular simulation approach was employed to do so. The molecular simulation study involving these proteins revealed that the binding of hetero-dimeric IHF to DNA indeed bended the DNA. The interactions were mainly mediated by IHF β with the minor groove of the DNA facilitated by IHF α . Together IHF α and IHF β bended the DNA in such a way that the mutual interactions between the DNA and the IHF protein was enhanced manifold. Further verification of the DNA bending was observed by calculating the correlation between Cpc and Cpp as in [17]. The correlation value being 0.31 clearly indicates DNA bending by IHF protein as per [17]. This phenomenon was again established when the H-NS-DNA complex was docked with the IHF hetero-dimer and the resulting complex was simulated. The resulting complex obtained after MD simulation showed that IHF α formed stronger interactions with H-NS than H-NS does with the DNA. It was also apparent from the simulation results that IHF β formed stronger interactions with DNA than H-NS did. The DNA bending by IHF helped the protein create more interactions with DNA than H-NS as the bending of the DNA made it more accessible to IHF α which initially could not interact with DNA. Calculations of the binding free energy values of these complexes also testified the aforementioned observations. The hetero-dimeric IHF was known to bind to DNA in order to transcribe the virulence genes [1]. The MD simulation results also pointed towards the same. The progress and completion of the MD simulation processes were monitored by plotting a graph of RMSD of the backbone atoms of the docked complexes vs time periods of the MD simulation runs. Since there were no previous reports that dealt with the mechanistic details of the DNA-protein interactions in the expression of virulence genes by *Vibrio cholerae*, the present work may be useful to analyze the hitherto unknown molecular mechanism of the virulence gene expression by *Vibrio cholerae*. The results from this study may be used in future genetic and mutational works to identify the functions of the amino acid residues involved in the interactions. This may further be utilized to develop new drugs against *Vibrio cholerae* infection.

Acknowledgements The authors acknowledge the University of Kalyani, Kalyani (W.B.) India, DBT funded Bioinformatics Infrastructure Facility (BIF) for providing infrastructure facilities and Department of Biotechnology (DBT), Govt. of India for financial support (Grant no. SAN No. 102/IFD/ SAN/1824/2015-2016). The author would like to thank the Indian Council of Medical Research (ICMR)-India (BIC/12(02)/2014) financial support.

References

1. E. Stonehouse, G. Kovacikova, R.K. Taylor, K. Skorupski, Integration host factor positively regulates virulence gene expression in *Vibrio cholera*. *J. Bacteriol.* **190**, 4736–4748 (2008)
2. E. Bouffartigues, M. Buckle, C. Badaut, A. Travers, S. Rimsky, H-NS cooperative binding to high-affinity sites in a regulatory element results in transcriptional silencing. *Nat. Struct. Mol. Biol.* **14**, 441–448 (2007)
3. S.M. McLeod, V. Burrus, M.K. Waldor, Requirement for *Vibrio cholerae* integration host factor in conjugative DNA transfer. *J. Bacteriol.* **188**, 5704–5711 (2006)
4. B.H. Abuaita, J.H. Withey, Bicarbonate Induces *Vibrio cholerae* virulence gene expression by enhancing ToxT activity *Infect. Immun.* **77**, 4111–4200 (2009)
5. T.W. Lynch, E.R. Read, A.N. Mattis, J.F. Gardner, P.A. Rice, Integration host factor: Putting a twist on protein-DNA recognition. *J. Mol. Biol.* **330**, 493–502 (2003)
6. Q. Bao, H. Chen, Y. Liu, J. Yan, P. Droege, C.A. Davey, A divalent metal-mediated switch controlling protein-induced DNA bending. *J. Mol. Biol.* **367**, 731–740 (2007)
7. H.M. Berman, The Protein Data Bank: a historical perspective. *Acta Crystallogr. A* **64**, 88–95 (2008)
8. S.F. Altschul, W. Gish, W. Miller et al., Basic local alignment search tool. *J. Mol. Biol.* **215**, 403–410 (1990)
9. A. Sali, T.L. Blundell, Comparative protein modelling by satisfaction of spatial restraints. *J. Mol. Biol.* **234**, 779–815 (1993)
10. B.R. Brooks, R.E. Brucolieri, B.D. Olafson CHARMM: A program for macromolecular energy, minimization, and dynamics calculations. *J. Comp. Chem.* **4**, 187–217 (1983)
11. D. Eisenberg, R. Lüthy, J.U. Bowie, VERIFY3D: assessment of protein models with three-dimensional profiles. *Methods Enzymol.* **277**, 396–404 (1997)
12. R.A. Laskowski, M.W. MacArthur, D.S. Moss et al., PROCHECK—a program to check the stereo-chemical quality of protein structures. *J. App. Cryst.* **26**, 283–291 (1993)
13. G.N. Ramachandran, C. Ramakrishnan, V. Sasisekharan, Stereochemistry of polypeptide chain configurations. *J. Mol. Biol.* **7**, 95–96 (1968)
14. A. Tovchigrechko, I.A. Vakser, Development and testing of an automated approach to protein docking. *Proteins* **60**, 296–301 (2005)
15. D. Schneidman-Duhovny, Y. Inbar, R. Nussinov et al., PatchDock and SymmDock: servers for rigid and symmetric docking. *Nucleic Acids Res.* **33**, W363–W367 (2005)
16. J. Schymkowitz, J. Borg, F. Stricher, R. Nys, F. Rousseau, L. Serrano, The FoldX web server: an online force field. *Nucleic Acids Res.* **33**, W383–W388 (2005)
17. B.N. Grant, E.M. Dourlain, J.N. Araneda, M.L. Throneberry, L.A. McFail-Isom, DNA phosphate crowding correlates with protein cationic side chain density and helical curvature in protein/DNA crystal structures. *Nucleic Acids Res.* **41**, 1–9 (2013)

A Novel Design of Carry Select Adder (CSLA) for Low Power, Low Area, and High-Speed VLSI Applications



Nilkantha Rooj, Snehanjali Majumder and Vinay Kumar

1 Introduction

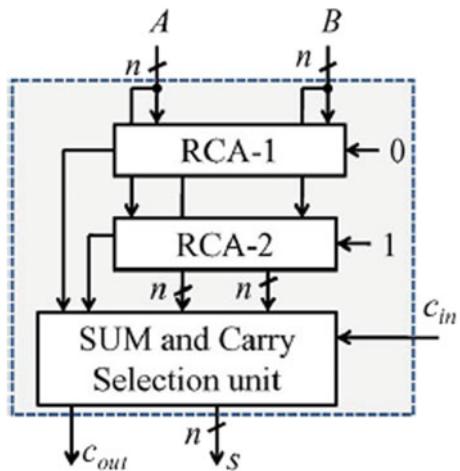
Adders are extensively used in the rudimentary logic blocks in various parts of a processor, microcomputer, digital subtractor, multiplication circuits and in the basic building of on-chip libraries. The perfection of a digital logic segment is dependent on its constituent adders [1–3]. The standard guideline of adder designs includes specifications like propagation delay, fan-outs, and chip area [4]. In binary adders, the rate at which addition is performed is contributed by the time required to generate a carry by the consecutive stages of the adder. The sum for every bit location in a binary adder is evaluated successively only after the former bit spot is calculated and the carry is passed on to the succeeding position. CSLA is used to reduce the complication of delay in carry generation by freely evaluating multiple stages to calculate the carry and then choosing only one carry to produce the sum [5]. However, the CSLA has a larger area than other types of adders because it uses several blocks of ripple carry adders (RCA) to get the half sum and carry by taking into account the carry inputs as 0 and 1 in two steps and then choosing the final sum and carry by the multiplexer [6–10]. The elementary idea of this paper is that only sum is selected by multiplexer and carry is generated by the AND and OR gate and propagated to the next stage.

N. Rooj · S. Majumder
Department of ECE, JGEC, Jalpaiguri 735102, West Bengal, India
e-mail: nilkantharooj48@gmail.com

S. Majumder
e-mail: snehanjali.maj@gmail.com

V. Kumar (✉)
Department of ECE, NIT Meghalaya, Shillong 793003, Meghalaya, India
e-mail: vinay.kumar@nitm.ac.in

Fig. 1 Conventional carry select adder

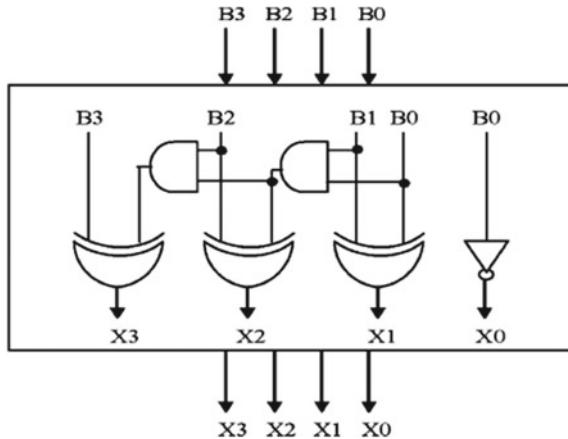
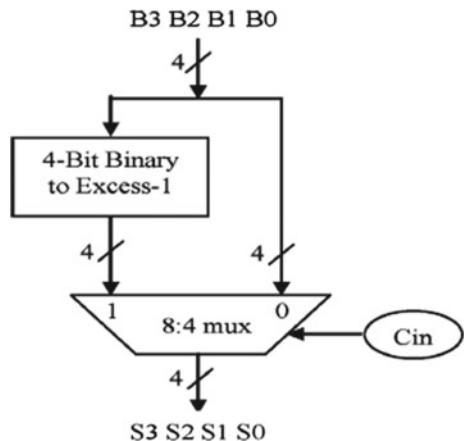


2 Conventional Design of Carry Select Adder

Conventional CSLA uses ripple carry adder as units in parallel in many steps [1–5]. Figure 1 represents the block diagram of a conventional CSLA. First, the sum and carry are generated by considering the carry input as 0 and again subsequently the sum and carry are generated by taking the carry input as 1. At the end, the multiplexer is fed with the actual input carry-in the select line and hence it will select the sum and carry accordingly. Thus in each stage, ripple carry adder blocks are used for the computation [4]. A and B represent the input numbers whereas s and c represent the output sum and carry in the diagram.

3 Existing Designs of Carry Select Adder

Conventional carry select adders suffer large area and high power consumption. Ram Kumar and Kittur proposed binary to excess-1 converter (BEC) for design of CSLA where only a single $(n+1)$ bit binary to excess-1 converter segment can be used as a substitute of two n -bit ripple carry units [6]. A 4-bit BEC logic circuit is given in Fig. 2. Figure 3 illustrates the method by which the fundamental operation of CSLA is performed using a 4-bit BEC and a multiplexer. The multiplexer inputs include any one of B_0, B_1, B_2, B_3 and in the second input enters the output from Binary to excess one converter. Hence, simultaneously two probable outcomes are evaluated. The result is chosen by the multiplexer depending upon the true carry input of the initial stage [6].

Fig. 2 4-bit BEC**Fig. 3** 4-bit BEC with 8:4 MUX

Manju and Sornagopal presented a carry select adder which is efficient in area and based on shared common Boolean logic terms to remove the repeated use of ripple carry adders (RCA) in the conventional CSLA [7]. Mohanty and Patel [10] presented a new approach for the design of CSLA which consists of one unit of half sum generation (HSG), one unit of full sum generation (FSG), one unit of carry generation (CG) and one unit of carry selection (CS) [10]. The block diagram of this CSLA is shown in Fig. 4.

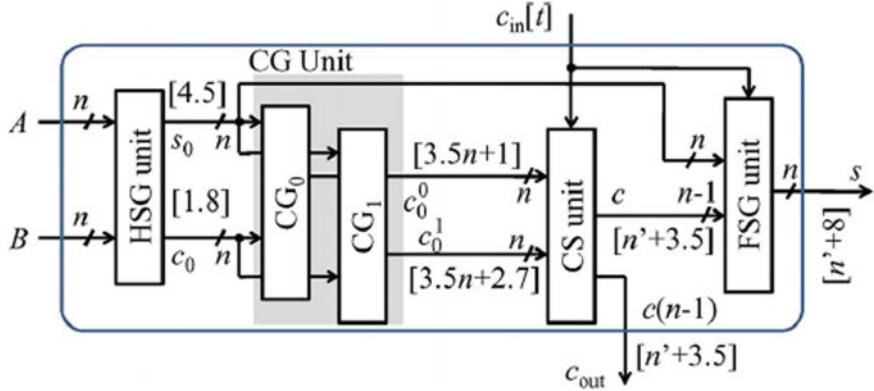


Fig. 4 CSLA based on HSG, FSG, CG, and CS

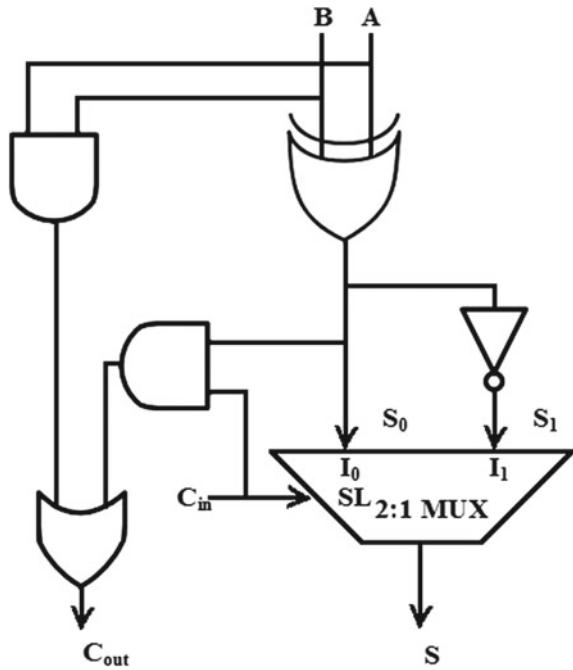
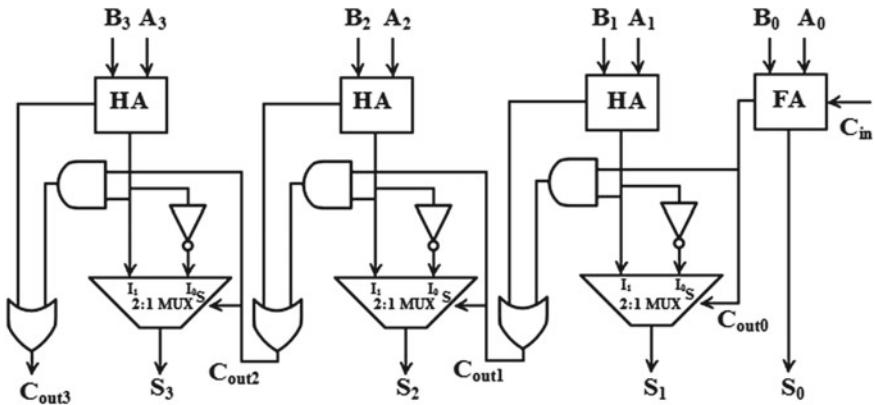
4 Proposed Design of Carry Select Adder

The operation of carry select (CS) in the proposed design of CLSA is scheduled to occur before the evaluation to select the final sum and here it deviates from the regular technique. Bit arrangements of two possible carry words ($c_{in} = 0$ and $c_{in} = 1$) and specified c_{in} bits logically cumulate CS to select the final sum of each stage. In the meantime, the carry for the next stage is generated by the propagate and generate logics. This design can be perfectly implemented in the CSLA and shows improvement in practical and theoretical delay which is much lesser than the conventional design of CSLA and other modified designs of CSLA in literature.

Figure 5 presents a circuit for the proposed logic of CSLA in which carry is propagating as well as select the final sum of each stage as a control signal of 2:1 MUX. Figure 5 illustrates that A and B are the inputs and S_0 is the half adder sum that is the XOR output of two inputs and S_1 is the inverted half adder sum. The 2:1 MUX selects the true sum between S_0 and S_1 by the selection signal that is the previous stage carry-in (C_{in}). Carry generated in half adder in the present stage for the inputs A and B is ORed with the result of 2-input AND gate. AND gate inputs are half adder sum S_0 and previous stage carry-in (C_{in}).

The design of 4-bit CSLA based on proposed logic is shown in Fig 6. In each stage, the addition of two inputs is done through the half adder and the true sum is selected by 2:1 MUX whose two inputs are the sum of the half adder and its inverted counterpart which is demonstrated in Fig. 6 and selection input is previous stage carry. So the carry of present stage is generated by ORing the half adder carry and performed AND operation on half adder sum and previous stage carry, just like carry is propagating through each stage but it takes less time. The generalized Boolean expressions for each stage of proposed carry select adder are as follows:

$$S_n^0 = A_n \oplus B_n \quad (1)$$

**Fig. 5** Proposed logic for CSLA**Fig. 6** 4-bit proposed CSLA

$$S_n^1 = A_n \odot B_n \quad (2)$$

$$C_{outn} = (A_n \cdot B_n) + (A_n \oplus B_n) \cdot C_{outn-1} \quad (3)$$

Figure 7 illustrates the diagram of 16-bit design of CSLA. This design is based on proposed 4-bit CSLA. Similarly, 32-bit CSLA has also been designed.

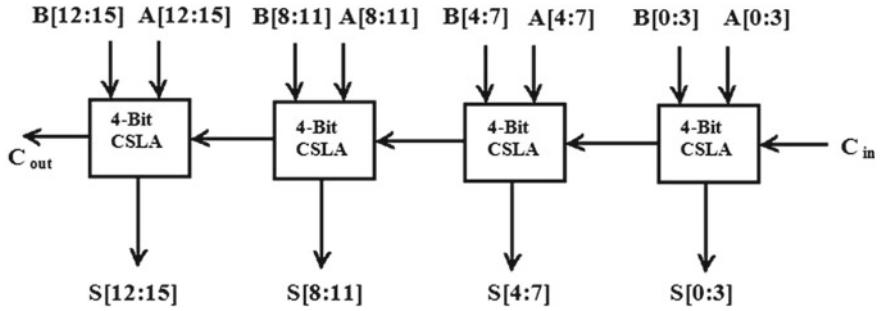


Fig. 7 16-bit proposed CSLA

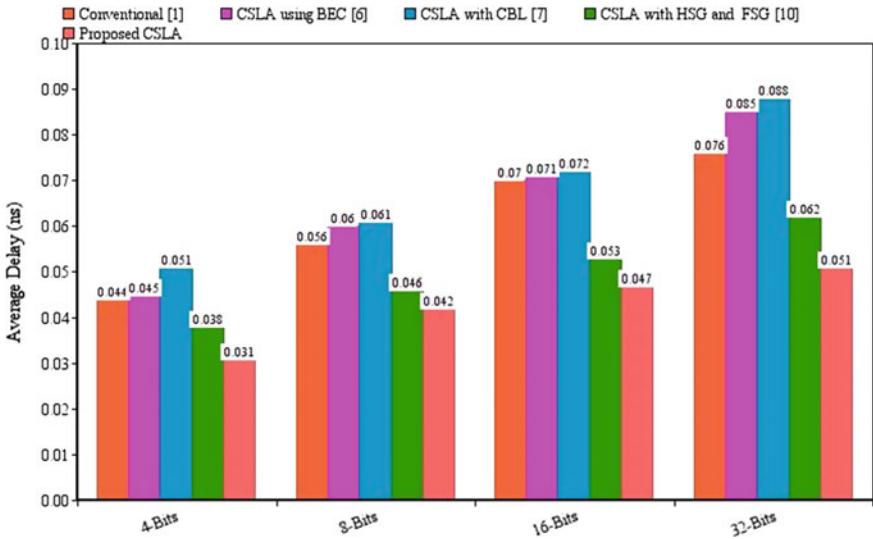


Fig. 8 Comparison between the delay of various topologies of CSLA

5 Comparison Between Proposed and Existing Design of CSLA

The simulation results for the proposed design of carry select adder (CSLA) are shown in Table 1. The circuits have been implemented using CMOS logic style and simulation has been performed on Cadence Virtuoso Tool in 45 nm CMOS technology. It is observed that the area (gate count), power consumption, and delay of the proposed carry select adder (CSLA) for different bits are lesser than that of the conventional CSLA and other existing designs of CSLA. The detailed comparison between various topologies of CSLA on the basis of delay, power, and gate count (area) is shown in Figs. 8, 9 and 10 respectively.

Table 1 Comparison between various topologies of CSLA

Topology	Bits	Average delay (ns)	Power (μW)	Gate count
Conventional [1]	4	0.044	280.2	118
	8	0.056	553.9	233
	16	0.070	976.3	478
	32	0.076	1899.0	938
CSLA using BEC [6]	4	0.045	105.2	97
	8	0.060	215.6	194
	16	0.071	781.0	318
	32	0.085	1054.0	636
CSLA with CBL [7]	4	0.051	182.4	85
	8	0.061	275.0	170
	16	0.072	582.0	340
	32	0.088	899.1	680
CSLA with HSG and FSG [10]	4	0.038	156.1	74
	8	0.046	314.9	148
	16	0.053	605.7	296
	32	0.062	1178.0	592
Proposed CSLA	4	0.031	129.6	64
	8	0.042	221.5	128
	16	0.047	589.9	256
	32	0.051	789.3	512

6 Conclusion

This paper presented a study of the internal architecture of conventional CSLA as well as other existing designs of carry select adder and presented a new design for carry select adder. The operation of carry selection precedes the sum calculation of each stage in this work but with a different approach than conventional carry select adder. Carry out results generated from the different value of carry in ($C_{in} = 0$ and $C_{in} = 1$) in the proposed carry select adder have a certain bit pattern which has been used for logic minimization of the carry select unit. Using these improved logic formulation, an effective new design is proposed for CSLA. The proposed CSLA for 4, 8 and 32-bit shows improvement in terms of power consumption, area (gate count) and delay as compared to conventional CSLA and other existing designs. The presented design of CSLA in this paper will be useful for VLSI circuits which require low power consumption, less area, and high speed.

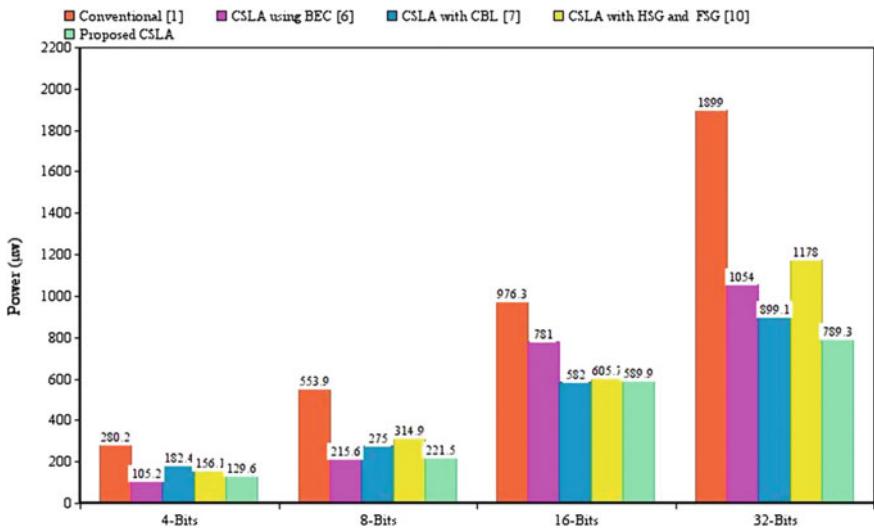


Fig. 9 Comparison between the power of various topologies of CSLA

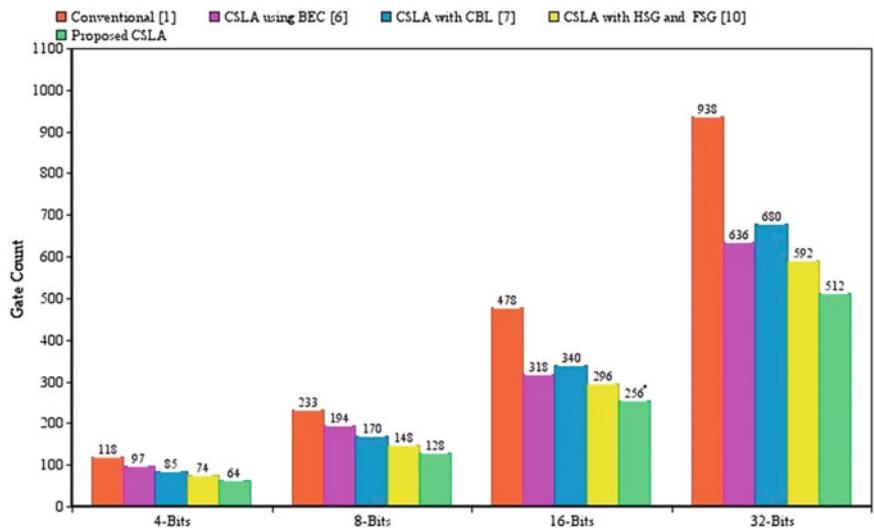


Fig. 10 Comparison between gate count of various topologies of CSLA

References

1. N. Singla, J. Kaur, Analysis of different types of adders: a review, in *Proceedings of International Conference on Advanced Material Technologies (ICAMT)*, Visakhapatnam, India, pp. 1–7 (2016)
2. A. Anand Kumar, *Fundamental of Digital Circuits*, 2nd ed. (PHI Learning Private Limited, 2003)
3. O.J. Bedrij, Carry-select adder. IRE Trans. Electron. Comput. **3**(2), 340–344 (1962)
4. B. Parhami, *Computer Arithmetic: Algorithms and Hardware Designs*, 2nd ed. (Oxford University Press, 2010)
5. T.Y. Ceiang, M.J. Hsiao, Carry-select adder using single ripple carry adder. Electron. Lett. **34**(22), 2101–2103 (1998)
6. B. Ramkumar, H.M. Kittur, Low-power and area-efficient carry select adder. IEEE Trans. Very Large Scale Integr. VLSI Syst. **20**(2), 371–375 (2012)
7. S. Manju, V. Sornagopal, An efficient SQRT architecture of carry select adder design by common boolean logic, in *Proceedings of VLSI ICVENT*, pp. 1–5 (2013)
8. K.B. Sindhuri, Implementation of regular linear carry select adder with binary to excess-1 converter. Int. J. Eng. Res. **4**(7), 346–350 (2015)
9. K.M. Priyadarshini, N.V. Kiran, N. Tejasri, T.C. Anish, Design of area and speed efficient square root carry select adder using fast adders. Int. J. Sci. Technol. Res. **3**(6), 133–138 (2014)
10. B.K. Mohanty, S.K. Patel, Area–delay–power efficient carry-select adder. IEEE Trans. Circuits Syst. II **61**(6), 418–422 (2014)

A Novel Objective Function Based Clustering with Optimal Number of Clusters



Kuntal Chowdhury, Debasis Chaudhuri and Arup Kumar Pal

1 Introduction

Nowadays a huge amount of data are being generated from the variety of fields like medical sciences, earth sciences, due to technological development [12]. These massive unlabeled data can be managed using data mining. Data mining is known as the computational process for the retrieval of useful information from these data [24]. Clustering is used to manage these raw data and retrieving the useful information and futuristic analysis depending on the features. The clustering has been presented in different real-world application areas like computational biology [1], astronomical data [26]. Researchers have given their different views on clustering in different literature. Some of the researcher have taken as a mathematical programming [11], statistical pattern recognition [15]. Cluster analysis is six steps procedure denoted as input data, extraction of features, use of clustering algorithm, cluster validation, result verification, knowledge retrieval. Clustering algorithms are broadly classified into five categories. These are (i) Hierarchical (ii) Partitional (iii) Density-based (iv) Grid-based (v) Model-based. Hierarchical clustering creates the hierarchy of clusters using bottom down or bottom-up approaches. Partitional clustering creates the partition in an iterative manner depending on some criteria function. This clustering can be represented as an optimization problem known as NP-hard [10]. Density-based clustering works on density concept in terms of region of density, and appropriate

K. Chowdhury (✉) A. K. Pal
Indian Institute of Technology (Indian School of Mines), IIT (ISM),
Dhanbad, Jharkhand, India
e-mail: ikuntal09@gmail.com

D. Chaudhuri
DRDO Integration Centre, Panagarh, West Bengal, India
e-mail: deba_chaudhuri@yahoo.co.in

A. K. Pal
e-mail: arupkrpal@gmail.com

for arbitrary shape clusters. This method has a high time complexity. In the case of Grid-based clustering, the data is partitioned into the finite number of cells. The clusters are being represented using the maximum number of density points. It is computationally less expensive. Model-based clustering techniques try to correlate the given data with some mathematical model for optimization. This technique is widely applied to neural networks. Objective function based clustering is the type of clustering which can be taken as an optimization problem depending on some defined objective function [14]. This Objective function based clustering may be the maximization or minimization optimization problem [4, 16, 18]. The application of objective function based clustering with the optimal number of clusters is a challenging task. The clustering of color image segmentation has been done using the maximization of entropy [8].

2 Main Contribution and Organization of Paper

In this paper, an objective function based clustering has been proposed which has been applied with the optimal number of clusters in the dataset. The paper is organized as follows: In Sect. 3, the optimal number of cluster detection (*ONCD*) has been proposed and the mathematical definition of internal cluster validity indices used in this algorithm. In Sect. 4, the objective function and the proposed clustering algorithm have been discussed. The implementation of the proposed algorithm on the images is shown in Sect. 5. In Sect. 6 we conclude our work.

3 Optimal Number of Cluster Detection Algorithm

It is a challenging problem to find out the optimal or exact number of clusters in an unknown data set. The quality of the objective function based clustering is dependent on the optimal number of clusters in the dataset. The comparison can be made of the different clustering algorithms over the same data set using cluster quality measures for the suitability of the clustering model.

3.1 Preliminaries

In the literature, different methodologies are available for the prediction of the number of clusters. Entropy concept has already been used for the determination of an optimal number of clusters. The cluster quality and the prediction of the optimal number of clusters are being done using different internal and external criteria [19]. The basic approaches for the optimal number of cluster detection are By rule of thumb, Elbow method, Information Criterion Approach, Choosing K using the Silhouette,

an Information-Theoretic Approach, and Cross-validation. Other methodologies like rate distortion theory, information criteria, jump statistics, and cross-validation have also been used to predict optimal number of clusters using model selection approach [2, 21, 22, 25]. The optimal number of clusters in the data sets are also calculated using modified cross-validation approach set [23]. The software simulation based scheme has been used to find out the number of clusters in the data set [7]. The researchers have proposed different models to predict the number of clusters in the data set [13, 17]. Literature survey also gives different types of methods for the detection of cluster number [9]. The prediction of the optimal number of clusters methods falls into three main categories. These are Traditional Approaches, Merge-Split based Approaches. Evolutionary Computation based Approaches. Currently, evolutionary-based approaches more suitable than others. Single objective approach and multi-objective approach are the classifications of evolutionary-based approach alternatively automatic clustering approach. The performance of our objective function based clustering mainly depends on the optimal number of the clusters in the dataset by using these two internal cluster validity indices. The notation of the optimal number of clusters for a particular data set as is represented as Opt_C throughout the paper. The mathematical definitions of these indices are represented below.

3.2 Dunn Index

Dunn index [6] is one of the important internal cluster validity indexes to predict the number of clusters by its maximized value in a dataset. It gives good quality clusters by detecting the optimal number of clusters in the dataset. Dunn index is defined as follows:

$$D = \min_{p=1 \dots k} \left[\min_{q=p+1 \dots k} \left(\frac{dis(c_p, c_q)}{\max_{i=1 \dots k} dia(c_i)} \right) \right] \quad (1)$$

where $dis(c_p, c_q) = \min_{x \in c_p, y \in c_q} (dis(x, y))$ and $dia(c_p) = \min_{x, y \in c_p} (dis(x, y))$. The intercluster distance between cluster p th and q th clusters c_p and c_q is being represented as $dis(c_p, c_q)$. The distance between two data elements is denoted as $dis(x, y)$. The number of clusters is represented as k .

3.3 Silhouette Index

It is also a very popular index which uses the combination of both cohesion and separation factors [20]. The dataset is clustered by any clustering technique can be validated by using this index. We have assumed that for each datum x , a_x is the average

distance, between all other points in its cluster. The minimum average distance of x to any other clusters points of which x is not a member of those clusters is taken as b_x . The silhouette coefficient for the x th point is defined

$$S_x = \frac{(b_x - a_x)}{\max(a_x, b_x)} \quad (2)$$

It is important to note that the average value of S_x for all points of a cluster is a measure of compactness among all the points in the cluster. The maximum value of this index denotes the good clustering results.

3.4 Davies-Bouldin Index (DB)

Davies-Bouldin index [5] is another internal index used for the optimal number of clusters prediction. This index integrates both the dispersion measure and dissimilarity measure of the cluster. This index determines the farthest compact clusters using the equation described below. The optimal number of clusters (Opt_C) are represented using the minimum value of this index.

$$DB(c) = \frac{1}{c} \sum_{i=1}^c \max_{j=1, \dots, c, i \neq j} \left\{ \frac{dia(c_i) + dia(c_j)}{\| c_i - c_j \|} \right\} \quad (3)$$

where the number of clusters is denoted as c . The diameter of the cluster is defined

$$dia(c_i) = \left\{ \frac{1}{n_i} \sum_{x \in C_i} \| x - c_i \|^2 \right\}^{\frac{1}{2}} \quad (4)$$

n_i is the total number of points and c_i is the centroid of the cluster C_i . This index gives the minimum intracluster distance.

3.5 Calinski-Harabasz Index (CH)

Calinski-Harabasz index [3] is also an internal validity index to calculate the optimal number of clusters in a dataset. It also produces faster computation of the optimal number of clusters of any data set than other indices. This is also used to maximize the intercluster dispersion and minimize intracluster dispersion. The highest value of this index denotes the optimal number of clusters (Opt_C) in a dataset. The mathematical definition of this index is given below:

$$CH(c) = \frac{trace(B_m)}{trace(W_m)} \times \frac{N - c}{c - 1} \quad (5)$$

Here, the between-cluster scatter matrix and the internal scatter matrix have been represented as B_m and W_m respectively. The number of points and the number of clusters are represented as N and c , respectively.

$$W_m = \sum_{i=1}^c \sum_{x \in C_i} (x - c_i)(x - c_i)^T \quad (6)$$

$$B_m = \sum_i n_i (c_i - k)(c_i - k)^T \quad (7)$$

The center of the cluster C_i is being represented as c_i . The number of points in cluster C_i is represented as n_i . k is the center of the input data set.

Now the optimal number of cluster detection (*ONCD*) algorithm of an unknown data set is being represented using these four internal cluster validity indices below.

Algorithm 1 Optimal Number of Cluster Detection (*ONCD*) Algorithm

Input: Image Data

Output: Optimal Number of Cluster (Opt_C)

- 1: The initial value of number of clusters has been assumed as $C = 2$ considering total number of pixels as P . Calculate the Dunn index value (X_{DU}) and Silhouette index value (X_{SL}), Calinski-Harabasz index (X_{CH}), Davies-Bouldin index (X_{DB}).
 - 2: Perform $C = C + 1$ and Calculate X_{DU} , X_{SL} , X_{CH} , X_{DB} for this value of C .
 - 3: Repeat Step 2 until there is a decreasing tendency for both the values of indices for a certain value of C from the initial stage.
 - 4: Identify the number of clusters C for which the values of all these four indices attain optimal number of cluster. We have assumed that the occurrence of maximum value attain for Dunn index is at $C = c_1$ and the same for Silhouette index is at $C = c_2$, Calinski-Harabasz index is at $C = c_3$. Similarly minimum value attain for Davies-Bouldin index is at $C = c_4$.
 - 5: If $c_1 = c_2 = c_3 = c_4$ then the number of clusters present in the data set is $C = c_1$. Otherwise, find $M_z = \max [X_{DU}, X_{SL}, X_{CH}, X_{DB}]$ and corresponding number of clusters $C = Opt_C = M_z$.
 - 6: Stop.
-

4 Proposed Clustering Algorithm

The proposed clustering algorithm is described in this section which is dependent on the defined objective function. This novel objective function is a minimization problem. The mathematical formulation of proposed objective function based clustering technique is as follows: Assume total number of points of the image I as Q . The objective function has been defined as:

$$Z(k) = \frac{n_{R(k)}E_{R(k)} + n_{S(k)}E_{S(k)}}{QE_L}, 0 \leq k \leq L \quad (8)$$

$$k^1 = \min_{\forall k} [Z(k)] \quad (9)$$

$n_{R(k)}$, $n_{S(k)}$ are the total number of points in $R(k)$ and $S(k)$ regions respectively. $E_{R(k)}$, $E_{S(k)}$ are the entropies of $R(k)$ and $S(k)$ regions respectively. E_L is the total entropy of the image I . $R(k)$ region consists of points having gray values ranging from 0 to k . $S(k)$ region consisting of points having gray values ranging from $(k+1)$ to L . The total number of subregions is equal to Opt_C . $E_{R(k)} = -\sum_{i=0}^k a_i \log a_i$, $E_{S(k)} = -\sum_{i=k+1}^L a_i \log a_i$, $E_L = -\sum_{i=0}^L a_i \log a_i$. a_i is the probability of i th gray value. So we can write $-\sum_{i=0}^L a_i \log a_i = \left(-\sum_{i=0}^k a_i \log a_i\right) + \left(-\sum_{i=k+1}^L a_i \log a_i\right)$. i.e $E_L = (E_{R(k)} + E_{S(k)})$. This objective function gives the partition of the image depending on the minimized value of $Z(k)$. This partition continues recursively until the number of partition equal to the optimal number of clusters (Opt_C).

Algorithm 2 Objective Function based Clustering Algorithm

Input: Image Data and Optimal Number of Cluster (Opt_C)

Output: Clustered Image Data

- 1: Initially the image I can be taken as an input image. Take the value of Opt_C calculated using *ONCD* algorithm as an input. Assign the number of cluster $C_0 = 1$.
 - 2: Calculate the entropy of the current region $R(i)$ as $E_{R(i)}$ of image I .
 - 3: If $(E_{R(i)} \geq E_{Th})$ then calculate the value of k^1 using Eqs. (8) and (9). If $(E_{R(i)} < E_{Th})$ then goto 5. E_{Th} is the automated threshold entropy which is dependent on the optimal number of clusters (Opt_C) in the image data.
 - 4: Split the current region into two sub-regions according to the calculated value by Eqs. (8) and (9). Increase the number of cluster C_0 using $C_0 = C_0 + 1$.
 - 5: Repeat (2 – 4) for the next sub-regions recursively until $(C_0 == Opt_C)$.
 - 6: Stop.
-

5 Experimental Results

The proposed objective function based clustering algorithm has been tested on different types of images like the normal image, satellite image, and medical image. The algorithms have been performed in MATLAB 7.5 with Intel Pentium (R) 2.16 GHz processor and 4 GB RAM. The comparison to classical segmentation algorithm using Kapur's [16] objective function and Chaudhuri's [4] objective function is being shown. This comparison is being represented in a tabular format. In Fig. 1, the first column represents the original images. The second column represents the plotting of

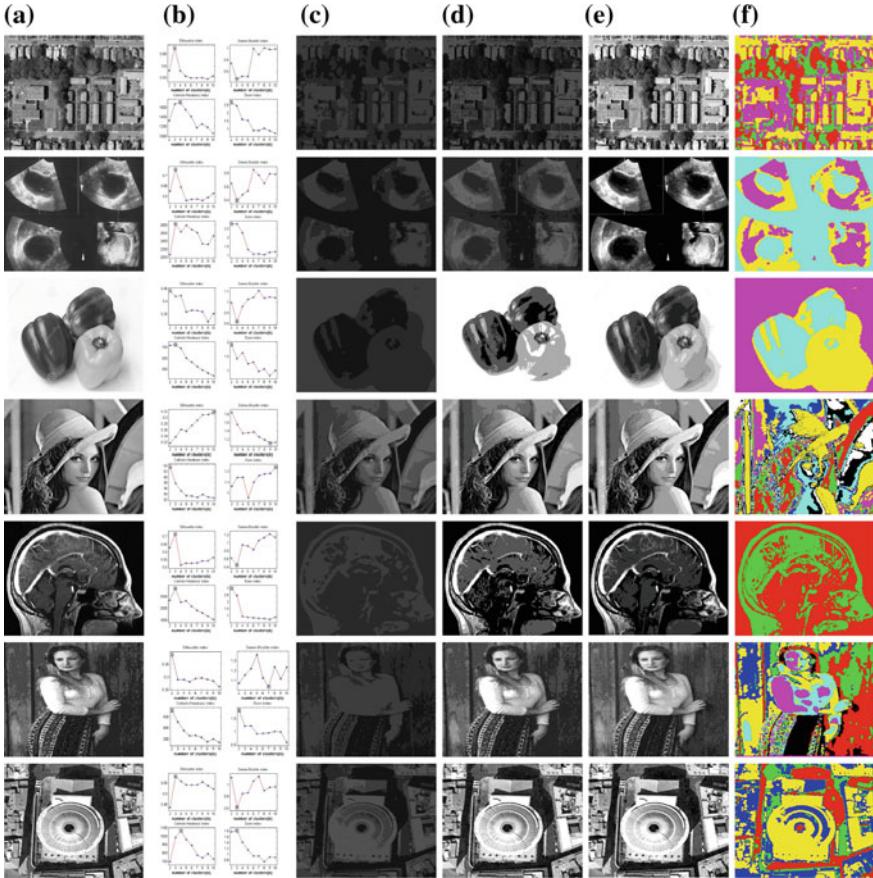


Fig. 1 Different Comparison of Objective Function based Clustering: First Column: **a** Original Image, Second Column: **b** Optimal Number of Clusters, Third Column: **c** Clustering using our proposed Objective function based clustering, Fourth Column: **d** Clustering using Kapur's [16] Objective function, Fifth Column: **e** Clustering using Chaudhuri's [4] objective function, Sixth Column: **f** Color partition using proposed objective function based clustering

the optimal number of clusters using four different internal cluster validity indices. The third column represents clustering by our proposed objective function without giving color. The fourth column represents clustering by Kapur's [16] objective function. The fifth column represents a clustering using Chaudhuri's [4] objective function. The sixth column represents the color partition of the images using our proposed objective function based clustering. In the graphical plot, the rectangle of the corresponding cluster validity index represents the optimal number of clusters. Building image is original IKONOS panchromatic image of size 200×200 . The CT scan of the image of the cysted ovary having dimension 200×200 has been represented as Ovary. Capsicum is a gray scale image having dimension 200×200 . The medi-

Table 1 Comparison of clustering quality using segmentation entropy (SE) of different objective function based clustering

Figure name	$SE_{Chaudhuri}$ [4]	SE_{Kapur} [16]	$SE_{Proposed}$
Building	1.59	1.65	1.54
Ovary	1.51	1.57	1.43
Capsicum	1.78	1.85	1.53
Lena	3.75	3.82	3.26
Medical image	0.98	1.10	0.92
Lady	2.85	2.90	2.71
Qbird	2.41	2.47	1.97

cal image is the original MRI image of brain arteries having dimension 200×200 . Lady image is a gray scale image having dimension 200×200 . The Quickbird is a panchromatic image of having dimension 342×342 . The novelty of our proposed clustering algorithm is shown using graphical results on the basis of the optimal number of clusters. Less segmentation entropy indicates good quality of clustering. The segmentation entropy of the proposed objective function based clustering is less than other methods. We have represented segmentation entropy of Kapur's method [16], Chaudhuri's method [4], proposed method as SE_{Kapur} , $SE_{Chaudhuri}$, $SE_{Proposed}$ respectively. The segmentation entropy for building image using our clustering method is 1.54, for ovary image it is 1.43, for capsicum image it is 1.53. Similarly, for Lena, medical image, Lady, and the Quickbird are 3.26, 0.92, 2.71, and 1.97, respectively. For all these images, our method gives the lowest segmentation entropy. The comparison of segmentation entropy with other objective function based clustering has been represented in Table 1. This comparison of segmented images' entropies proves the superiority of our proposed clustering algorithm. The comparison of cluster partitioning time of different objective function based clustering is shown in Table 2. The result indicates that our proposed objective function based clustering is faster than other clustering algorithms. The representation of cluster partitioning time taken by Chaudhuri's [4] objective function, Kapur's [16] objective function, our proposed objective function has been represented as $CPT_{Chaudhuri}$, CPT_{Kapur} , $CPT_{Proposed}$, respectively. The corresponding optimal number of clusters of building image is 4. This value is being taken as the maximum of all other four values of validity indices. Similarly, for the Ovary image, it is 3. For the Capsicum image, the actual number of cluster is 3. Following the same approach, the actual number of clusters for Lena image, medical image, Lady, and the Quickbird image are 10, 3, 7, and 4, respectively. In Table 3, these values of four different cluster validity indices with the values of the optimal number of clusters of each data set have been represented. Our objective function produces lowest segmentation entropy on the basis of these optimal number of clusters. Finally, the comparison results on the basis of cluster partitioning time, segmentation entropy prove the superiority, and novelty of our proposed clustering algorithm are given.

Table 2 Comparison of cluster partitioning time (CPT) (in Sec) of different objective function based clustering

Figure name	$CPT_{Chaudhuri}$ [4]	CPT_{Kapur} [16]	$CPT_{Proposed}$
Building	30.04	38.70	20.01
Ovary	29.67	38.87	13.89
Capsicum	21.94	29.10	14.91
Lena	90.57	95.67	81.29
Medical image	14.96	29.38	50.32
Lady	59.98	71.86	2.71
Qbird	29.09	39.54	26.13

Table 3 Different data sets with cluster validity indices values and optimal number of clusters (Opt_c)

Figure name	X_{DU} [6]	X_{SI} [20]	X_{CH} [3]	X_{DB} [5]	Opt_c
Building	2	3	4	3	4
Ovary	2	3	3	3	3
Capsicum	2	2	3	3	3
Lena	10	10	2	9	10
Medical image	2	3	3	3	3
Lady	2	2	2	7	7
Qbird	3	3	4	3	4

6 Conclusion

A novel objective function based clustering technique for the clustering of different types of images has been proposed in this research paper. The definition of this objective function is based on Shannon's entropy. This objective function based clustering has been applied to different images on the basis of the optimal number of clusters. The comparison of our clustering results is being made with other classical methods. The analysis of performances of the proposed clustering algorithm has been made on the basis of segmentation entropy with the other existing algorithms. The superiority of the proposed clustering algorithm has been shown through the comparative results.

References

1. A. Ben-Dor, R. Shamir, Z. Yakhini, Clustering gene expression patterns. *J. Comput. Biol.* **6**(3–4), 281–297 (1999)
2. H. Bozdogan, Mixture-model cluster analysis using model selection criteria and a new informational measure of complexity, in *Proceedings of the first US/Japan Conference on the Frontiers of Statistical Modeling: An Informational Approach* (Springer, Berlin, 1994), pp. 69–113

3. T. Caliński, J. Harabasz, A dendrite method for cluster analysis. *Commun. Stat. Theory Methods* **3**(1), 1–27 (1974)
4. D. Chaudhuri, A. Agrawal, Split-and-merge procedure for image segmentation using bimodality detection approach. *Def. Sci. J.* **60**(3), 290 (2010)
5. D.L. Davies, D.W. Bouldin, A cluster separation measure. *IEEE Trans. Pattern Anal. Mach. Intell.* **2**, 224–227 (1979)
6. J.C. Dunn, Well-separated clusters and optimal fuzzy partitions. *J. Cybern.* **4**(1), 95–104 (1974)
7. G. Evanno, S. Regnaut, J. Goudet, Detecting the number of clusters of individuals using the software structure: a simulation study. *Mol. Ecol.* **14**(8), 2611–2620 (2005)
8. S. Haifeng, L. Lanlan, Clustering color image segmentation based on maximum entropy, in *The 2nd International Conference on Computer Applications and System Modeling* (2012), pp. 1466–1468
9. E. Hancer, D. Karaboga, A comprehensive survey of traditional, merge-split and evolutionary approaches proposed for determination of cluster number. *Swarm Evol. Comput.* **32**, 49–67 (2017)
10. E. Hancer, C. Ozturk, D. Karaboga, Artificial bee colony based image clustering method, in *2012 IEEE Congress on Evolutionary Computation (CEC)* (IEEE, 2012), pp. 1–5
11. P. Hansen, B. Jaumard, Cluster analysis and mathematical programming. *Math. Program.* **79**(1–3), 191–215 (1997)
12. J.A. Hartigan, *Clustering Algorithms* (1975)
13. X. Hu, L. Xu, Investigation on several model selection criteria for determining the number of cluster. *Neural Inf. Process. Lett. Rev.* **4**(1), 1–10 (2004)
14. A.K. Jain, R.C. Dubes, *Algorithms for Clustering Data* (1988)
15. A.K. Jain, M.N. Murty, P.J. Flynn, Data clustering: a review. *ACM Comput. Surv. (CSUR)* **31**(3), 264–323 (1999)
16. J.N. Kapur, P.K. Sahoo, A.K. Wong, A new method for gray-level picture thresholding using the entropy of the histogram. *Comput. Vis. Graph. Image Process.* **29**(3), 273–285 (1985)
17. T.M. Kodinariya, P.R. Makwana, Review on determining number of cluster in k-means clustering. *Int. J.* **1**(6), 90–95 (2013)
18. R. Liscano, A. Wong, A Study into Entropy-Based Thresholding for Image Edge Detection (1995)
19. G.W. Milligan, A monte carlo study of thirty internal criterion measures for cluster analysis. *Psychometrika* **46**(2), 187–199 (1981)
20. P.J. Rousseeuw, Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.* **20**, 53–65 (1987)
21. P. Smyth, Clustering using monte carlo cross-validation. *Kdd* **1**, 26–133 (1996)
22. C.A. Sugar, G.M. James, Finding the number of clusters in a dataset: An information-theoretic approach. *J. Am. Stat. Assoc.* **98**(463), 750–763 (2003)
23. J. Wang, Consistent selection of the number of clusters via crossvalidation. *Biometrika* **97**(4), 893–904 (2010)
24. I.H. Witten, E. Frank, M.A. Hall, C.J. Pal, *Data Mining: Practical Machine Learning Tools and Techniques* (Morgan Kaufmann, USA, 2016)
25. L. Xu, Byy harmony learning, structural rpcl, and topological self-organizing on mixture models. *Neural Netw.* **15**(8–9), 1125–1151 (2002)
26. X. Xu, M. Ester, H.P. Kriegel, J. Sander, A distribution-based clustering algorithm for mining in large spatial databases, in *Proceedings of the 14th International Conference on Data Engineering* (IEEE, 1998), pp. 324–331

An Integral Image Based Text Extraction Technique from Document Images by Multilevel Thresholding Using Differential Evolution



Rupak Chakraborty, Rama Sushil and Madan L. Garg

1 Introduction

Text information retrieval technique from document images is used in many interesting applications like optical character recognition and image data compression to analyze and understand the layout of the documents properly. The target of text extraction from document images (which are degraded because of noise and illumination) is mainly used to identify the image regions which contain only text data. To achieve this target, thresholding of the images is the best option. It is the simplest method for segmenting the gray scale document image to get the noise and illusion-free result. The gray levels of pixels belonging to the object are quite different from a gray level of pixels belonging to the background. So bi-level thresholding is an effective technique to separate the objects from the background. The objective of this thresholding is to replace each pixel in an image with a black pixel if an intensity of an image I_{ij} (where i, j varies from 0 to 255) is less than some fixed value TH (i.e., $I_{ij} < TH$), or a white pixel if the image intensity is greater than TH. Literature survey finds some proposed methods to extract text from binary document images [9, 10, 18]. In recent times, multimedia technology is used on document images to decorate it with character style-block in the complex background so that it becomes visually attractive. As background of the images carry various complexities like (variety, change of character size, character brightness, a mixture of textual and background) so, standard binary thresholding methods may face a big challenge to extract the text effectively. Some developed thresholding methods found in the literature [8, 14, 26] which may separate text region from non-text regions

R. Chakraborty (✉) · R. Sushil · M. L. Garg
DIT University, Dehradun 248001, India
e-mail: rupak.chakraborty@dituniversity.edu.in

R. Sushil
e-mail: rama.sushil@dituniversity.edu.in

M. L. Garg
e-mail: dr.ml.garg@dituniversity.edu.in
URL: <http://www.dituniversity.edu.in>

successfully. As these methods depend on binary algorithm also, these techniques are quite sensitive to noise, illusion, distortion, and quality of the image.

Parker [20] proposed a local gray intensity gradient thresholding method which helps to identify text region even in affected illuminated document images effectively. One useful local adaptive binarization method found in the literature where a combination of Soft Decision Method (SDM) and Text Binarization Method (TBM) with the help of Niblack [17] approach was proposed by Sauvola and Pietikäinen [23] in the year 2000, which became very effective to identify text region from images with complex backgrounds on the condition where the contrast of the image is absolutely high. Several color segmentation based techniques for text extraction from the document images are also surveyed in the literature [28, 31]. Gatos et al. [11], in the year 2006, proposed a modified and hybrid approach to extract text from illuminated color images also became popular as a local thresholding approach. After carrying more researches it was found that integral image based technique for segmenting document images demands more popular and accepted approach in computer vision and image processing. Calculation of integral images with a particular window size makes the algorithm to work faster and also the result does not depend on window size. An innovative thresholding technique based on integral images proposed by Bradely and Roth [4] and later Shafait et al. [25] extended that work by proposing a better approach to remove noise from scanned document images for better visibility of the images. Further research continued and found that Shafait et al. [24] and Viola and Jones [29] proposed their approaches which produced better results with less computation time that determined the effectiveness of integral image based techniques. As all these proposed approaches follow bi-level thresholding technique, it can be stated that document image in multilevel segmentation for effective extraction of texts lags in current research interest. Multilevel thresholding of images is popular because it segments the images in different threshold points so that nonhomogeneous regions split properly. Here, multiple clusters are formed for different threshold values. It is observed in multilevel thresholding survey with document images that Chen and Wu [7] proposed a recursive effective multiplane approach for text segmentation of complex images distorted by noise. Vu et al. [30], in the year 2015, proposed an automatic text extraction approach from document images in combination with multilevel thresholding and K-mean clustering algorithm. Here, first, an input image is divided into small blocks, then K-mean clustering is applied to a set of small subblocks where multilevel thresholding method was already applied to the set of subblocks. The separation of text region and non-text regions is done by applying a connected component based filter. But these multilevel approaches may struggle with high-level complexity problem in global search space to generate better and optimal solutions. So to reduce the high-level time complexity generated by multilevel segmentation problem, our research proceeds to find optimal solutions to this problem.

The meta-heuristic approach provides a good solution to an optimization problem with limited computation capacity. In this approach, a sample set of large solution sets can be produced to wide complex problems in search space. Users need not gain deep knowledge to select population of this problem, so this approach gains popularity day

after day. Some popular optimization techniques include Genetic Algorithm (GA) [5, 19], Firefly algorithm (FF) [12, 16], Artificial Bee Colony (ABC) [13, 15], Particle Swarm Optimization (PSO) [2, 6], and Differential Evolution (DE) [1, 3] studied in the literature survey. DE is found as the arguably better optimization techniques and it outperforms other optimizers like PSO, and GA for multilevel segmentation techniques in terms of computation time, as well as faster convergence rate with high convergence values and less standard deviation, found in the literature as well [21, 22]. So, DE has been chosen in our proposed scheme as an optimization technique, motivated by the performance of it.

1.1 Author's Contribution

- In our proposed approach, the integral image is evaluated from the original document image based on a selected fixed window size to reduce the time complexity.
- A local threshold value is calculated from the integral image based on statistical parameters like mean, variance, and probability of pixel values for text extraction.
- To analyze and better extraction of texts from images, the problem is extended to multilevel thresholding.
- To reduce the high-level complexity of the multilevel problem of image segmentation, DE as an optimization technique is applied to retrieve the optimal solutions of multiple threshold values in the search space.
- Finally, the proposed approach is tested on the set of document images and quantitative measurements like objective function values, standard deviations for optimizer is calculated. On the other hand, qualitative measurements of segmented images are evaluated by Peak Signal to Noise Ratio (PSNR), Structural Similarity Index Measurement (SSIM) [16], etc.

Section 2 defines integral image finding from the original image. Section 3 presents the working of Differential Evolution as a fast optimization technique. Section 4 combines integral image and DE to produce the proposed text extraction method. Performance of the proposed scheme is analyzed and discussed in Sect. 5. Finally, Sect. 6 draws the conclusion of the proposed work.

2 Integral Images

An integral image is known as the summed area table used for quickly and efficiently finding the sum of values in a rectangular subset of an area. The integral image at location i, j contains the sum of the pixels above and to the left of i, j , like:

$$I(i, j) = \sum_{i'=0}^i \sum_{j'=0}^j im(i', j') \quad (1)$$

Table 1 Numeric examples of integral image

4	1	2	2	4	5	7	9	$A(i_1, j_1)$	B
0	4	1	3	9	12	17		C	$D(i_2, j_2)$
3	1	0	4	13	16	25			
2	1	3	2	16	22	33			

where $I(i, j)$ is the integral image and $im(i, j)$ is the original image. Now to compute the integral image, cumulative row sum is calculated first using the formula:

$$rs(i, j) = rs(i, j - 1) + im(i, j) \quad (2)$$

where $rs(i, j)$ is the cumulative row sum, and $rs(i, -1) = 0$. Finally, the integral image can be computed in one pass over the original image as follows:

$$I(i, j) = I(i - 1, j) + rs(i, j) \quad (3)$$

where $I(i, j)$ is the expected integral image and $I(-1, j) = 0$. Table 1 illustrates the detailed idea regarding sum-area table calculation. In the first part of the table, some pixel numeric values are given. Second part holds the summation of each pixel of its left and above and the technique is called integral image. This integral image has been considered as the total population of our thresholding problem. As we are focusing on multilevel thresholding in optimal search space, so brief description of popular optimization technique named, Differential Evolution (DE) has been given in immediate section.

3 Differential Evolution (DE)

Differential Evolution was proposed by Storn in 1997 [27] is one of the most popular population-based global optimization algorithms. It contains four basic operations like initialization, mutation, crossover, and, a selection where the last three steps are repeated for subsequent generations. The generations continue until the end of some termination criterion is found like exhaustion of maximum fitness evaluation is found. With the help of fitness evaluation, an optimal solution can be found for a problem. A D-dimensional vector containing a set of N optimization parameters of the i th individual parameter of the population at generation time t can be described as

$$\vec{Z}_i(t) = [Z_{i,1}, Z_{i,2}, Z_{i,3}, \dots, Z_{i,D}] \quad (4)$$

To change the population numbers $\vec{Z}_i(t)$ (say) of each generation, a donor vector $\vec{Y}_i(t)$ is created. In DE/rand/1 scheme (one of the DE version), for creating a donor vector $\vec{Y}_i(t)$ for each i th member, three other parameter vectors (say r1, r2 and r3-th vectors such that $r1, r2, r3 \in [1, NP]$ and $r1 \neq r2 \neq r3 \neq i$) are randomly chosen from the current population. Now to obtain donor vector $\vec{Y}_i(t)$, a scalar number F is

multiplied with the difference of any two of these three. So to get j th component of i th vector, the expression is

$$\vec{Y}_{i,j}(t) = Z_{p1,j}(t) + F \cdot (Z_{p2,j}(t) - Z_{p3,j}(t)) \quad (5)$$

Next binomial crossover scheme is explained as it is used in the proposed method. The crossover is performed on each of the M variables by applying the control parameter Cr , also called crossover rate for DE. Here, the number of parameters inherited from a mutant has almost a binomial distribution [21]. Now for each target vector $\vec{Z}_i(t)$, a trial vector $\vec{R}_i(t)$ is created in the following manner:

$$R_{i,j}(t) = \begin{cases} Y_{i,j}(t), & \text{if } rand_j(0, 1) \leq Cr \cdot j = pn(i) \\ Z_{i,j}(t), & \text{otherwise} \end{cases} \quad (6)$$

where $j=1, 2, \dots, M$ and $rand_j(0, 1) \in [0, 1]$ is the j th evaluation of a random number generator and $pn(i) \in [1, 2, \dots, M]$ is a randomly chosen to ensure that $\vec{R}_i(t)$ gets at least one component from $\vec{Z}_i(t)$. In order to select the resultant vector between target and trial vector which will survive in the next generation at time $t=t+1$, the selection is performed [22]. If trial vector produces better value compared to a target vector, it replaces its target vector in the next generation; otherwise, parent holds its position in the population:

$$\vec{Z}_l(t+1) = \begin{cases} \vec{R}_l(t), & \text{if } f(\vec{R}_l(t)) > f(\vec{Z}_l(t)) \\ \vec{Z}_l(t), & \text{if } f(\vec{R}_l(t)) \leq f(\vec{Z}_l(t)) \end{cases} \quad (7)$$

where $f(\cdot)$ is maximized function.

4 Methodology

The proposed technique works on a $w \times w$ window size to find the integral image from the original image. Let I be the integral image and im be the original image, then the intensity of a pixel position in the integral image can be calculated by the above Eq. 1. Once we get the integral image, local mean for w window size can be calculated with simple addition and subtraction like

$$m(i, j) = \frac{(I(i + \frac{w}{2}, j + \frac{w}{2}) + I(i - \frac{w}{2}, j - \frac{w}{2}) - I(i + \frac{w}{2}, j - \frac{w}{2}) + I(i - \frac{w}{2}, j + \frac{w}{2}))}{w^2} \quad (8)$$

where w is the any squared window size and m represents mean value of an integral image. In a similar way, we can calculate the local variance also for that window

$$s^2(i, j) = \frac{1}{w^2} \sum_{i'=i-\frac{w}{2}}^{i+\frac{w}{2}} \sum_{j'=j-\frac{w}{2}}^{j+\frac{w}{2}} im^2(i', j') - m^2(i, j) \quad (9)$$

After calculation of mean and variance, the probability of gray values of integral images is calculated like $p_i = \frac{fr_i}{M \times N}$ where fr_i is the total frequencies of gray values of image I_i and $M \times N$ is the total number of pixels in the window (w). To get a single threshold value from this window, we can modify the Sauvola's binarization formula [23] in the following manner:

$$th(i, j) = (p_i \times i)[1 + k(\frac{s(i, j)}{R} - 1)] \quad (10)$$

where $th(i, j)$ is threshold value and $s(i, j)$ is standard deviation which can be evaluated by taking square root over variance value calculated previously. R is the maximum value of standard deviation (R = 128 for a gray scale document) and k is a parameter whose value lies between $[0.2 \rightarrow 0.5]$. Objective is to maximize the fitness function like

$$\begin{aligned} f(th) &= O_0 + O_1 && \text{where,} \\ O_0 &= \sum_{i=0}^{th-1} (p_i \times i)[1 + k(\frac{s(i, j)}{R} - 1)] \quad \text{and} \\ O_1 &= \sum_{i=th}^{255} (p_i \times i)[1 + k(\frac{s(i, j)}{R} - 1)] \end{aligned} \quad (11)$$

$$\text{where } 0 \leq th \leq 255$$

The optimum threshold is th which will be used to maximize $f(th)$. This probability-based measure tries to get more centralized distribution for each segmented region of the integral image. This bi-level thresholding problem has been extended to multilevel thresholding problem. Here, given problem is to determine n thresholds for a calculated integral image, i.e., $[th_1, th_2, th_3, \dots, th_n]$. The objective is to maximize the function in search space

$$\begin{aligned} f([th_1, th_2, \dots, th_r]) &= O_0 + O_1 + O_2 + \dots + O_n \quad \text{where,} \\ O_0 &= \sum_{i=0}^{th_1-1} (p_i \times i)[1 + k(\frac{s(i, j)}{R} - 1)] \\ O_1 &= \sum_{i=th_1}^{th_2-1} (p_i \times i)[1 + k(\frac{s(i, j)}{R} - 1)] \\ O_2 &= \sum_{i=th_2}^{th_3-1} (p_i \times i)[1 + k(\frac{s(i, j)}{R} - 1)] \\ &\vdots \\ O_n &= \sum_{i=th_n}^{255} (p_i \times i)[1 + k(\frac{s(i, j)}{R} - 1)] \end{aligned} \quad (12)$$

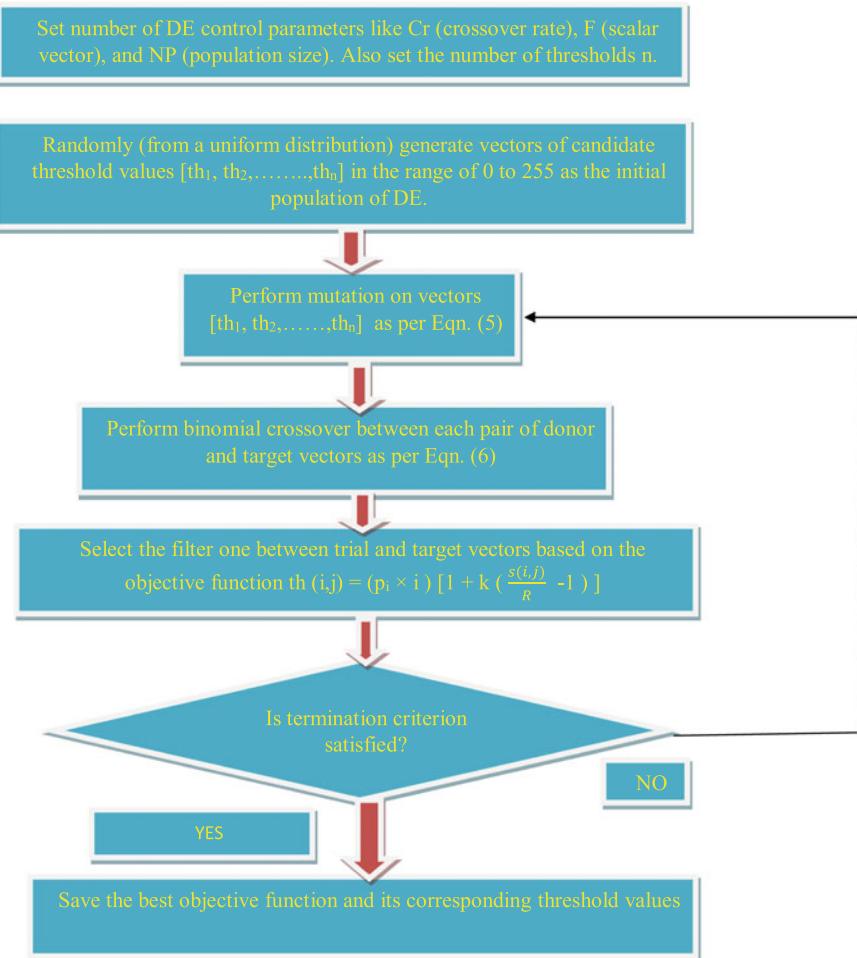


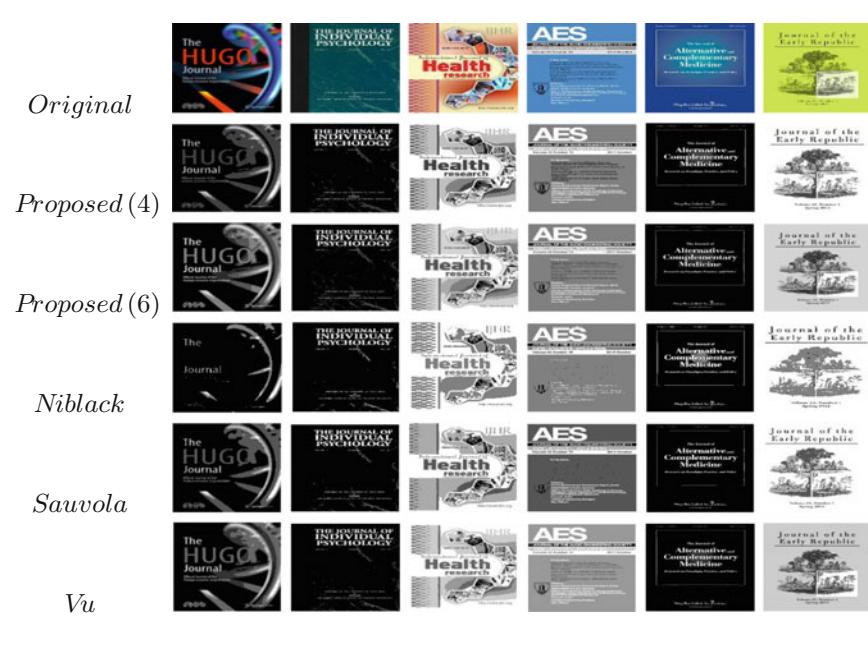
Fig. 1 Flowchart for the selection of multiple threshold values using DE

In the proposed approach, this optimum n -dimensional vector $[th_1, th_2, \dots, th_n]$ can be maximized by the Eq. (13). The procedure of the proposed technique is shown in Fig. 1.

5 Result Analysis and Discussion

The proposed approach is tested on document images which consist of various textual, non-textual, complex background regions, line, table, etc. However, text strings in these images often hold homogeneous pixel values either horizontally or vertically.

Table 2 Qualitative comparison of proposed method with the existing methods of sample images from test dataset



This idea holds the key to segment document images into various classes. Our test database includes almost 100 real-life document images. These images (shown in Table 2) are chosen from various sources such as book cover, advertise banner, a magazine for experimenting with performance evaluation of text extraction. It is noted that textual objects of these images also include various color, font styles, sizes, including adjoint or overlapped text with graphics, table, and background objects. They are resized to 200×200 for effective segmentation and researcher's comfort. The simulations of the proposed method are evaluated in MATLAB R2015a in a workstation with Intel *coreTM* i5 3.2 GHz processor. Three renowned optimization approaches named DE, PSO, and GA are chosen to compare the objective value (f_{obj}), computation time (t), and standard deviation (f_{std}) for the proposed method. All these optimized algorithms tested for 100 independent run where each run continued for the $D \times 1000$ (D denoted the dimension of search space) number of Fitness Evaluations (FE) with the proposed objective function. Segmentation levels are set from 4 to 6 to avoid under segmentation results. Quantitative measurements of the proposed scheme are discussed in next section.

Table 3 Comparison of computational time (t), objective value (f_{obj}) and standard deviation (f_{std}) between DE, PSO, and GA

<i>Im</i>		4-level			6-level		
		<i>DE</i>	<i>PSO</i>	<i>GA</i>	<i>DE</i>	<i>PSO</i>	<i>GA</i>
1	<i>t</i>	3.839	4.231	4.342	5.619	6.195	6.257
	<i>f_{obj}</i>	20.5880	20.5674	20.5543	26.5479	26.3452	26.1256
	<i>f_{std}</i>	0.0712	0.0876	0.0865	0	0.0923	0.0927
2	<i>t</i>	3.765	3.957	4.012	5.628	6.217	6.436
	<i>f_{obj}</i>	19.8559	19.8467	19.8356	25.9787	25.5678	25.4590
	<i>f_{std}</i>	0.0654	0.0723	0.0754	0.0734	0.0844	0.0865
3	<i>t</i>	3.787	3.989	4.056	5.679	5.890	5.945
	<i>f_{obj}</i>	20.9762	20.8590	20.8467	26.9452	26.2045	26.1089
	<i>f_{std}</i>	0.0432	0.0567	0.0578	0.0765	0.0870	0.08745
4	<i>t</i>	3.821	4.083	4.120	5.645	5.650	5.784
	<i>f_{obj}</i>	20.7412	20.7261	20.7015	26.7952	26.5461	26.4573
	<i>f_{std}</i>	0.0547	0.0678	0.0673	0.0850	0.0980	0.07884
5	<i>t</i>	3.868	4.230	4.497	5.542	5.667	5.809
	<i>f_{obj}</i>	20.3854	20.2103	20.1015	26.8442	26.7601	26.5803
	<i>f_{std}</i>	0.0652	0.0709	0.0809	0.0765	0.0880	0.07874
6	<i>t</i>	3.771	3.981	4.050	5.670	5.882	5.938
	<i>f_{obj}</i>	20.9755	20.8584	20.8460	26.9444	26.2040	26.1083
	<i>f_{std}</i>	0.0427	0.0558	0.0570	0.0758	0.0865	0.08739

5.1 Quantitative Measurements of the Proposed Algorithm

To proceed with measurement of statistical parameters, it is found out that the computation time, standard deviation, and objective values produced by Differential Evolution (DE) with comparison to PSO and GA in Table 3 are quite better and effective for the considered levels 4 and 6. High objective values (f_{obj}) indicate better convergence of optimization process whereas less standard deviation values (f_{std}) indicate better stability of the algorithm. Further less computation time (t) taken by DE with the proposed technique proves fast working of this stochastic optimization approach. Comparative convergence plot of the document images between DE, PSO, and GA for level 4 and 6 is drawn in Fig. 2 to show the higher and faster convergence rate of DE with integral image based approach. It is observed that DE for 100 iterations has started to converge near at value 50 for the 4th level and 85 to the 6th level whereas GA and PSO converge near at value 60 for the 4th level and 90 for the 6h level. Objective values produced by DE for both level 4 and 6 also found higher than PSO and GA. After finding DE as a better optimization technique from the above mentioned quantitative comparison, it is applied to find the computation time of different available algorithms including the proposed algorithm. We have compared

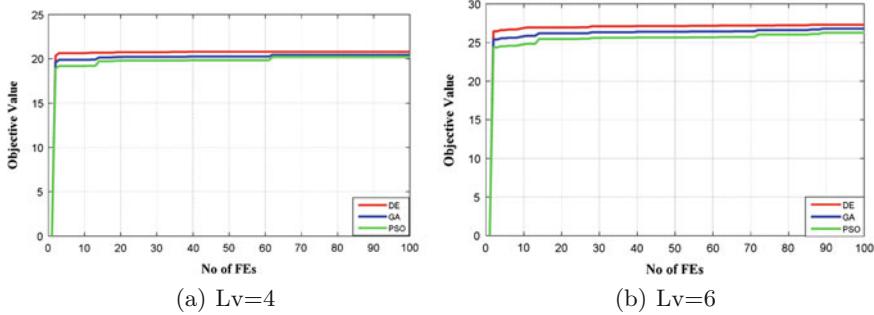


Fig. 2 Convergence plot of GA, PSO, and DE

Table 4 Computational time obtained by DE for 4 and 6 level thresholding

Image	Lv	Computation time (s)			
		Our Algo	Niblack	Sauvola	Vu
<i>Doc1</i>	4	3.839	5.483	5.455	4.418
	6	5.619	7.390	7.345	6.315
<i>Doc2</i>	4	3.765	5.790	5.745	4.711
	6	5.628	7.082	7.045	6.008
<i>Doc3</i>	4	3.787	5.580	5.545	4.512
	6	5.679	7.810	7.765	6.738
<i>Doc4</i>	4	3.821	5.370	5.342	4.313
	6	5.645	7.385	7.339	6.301
<i>Doc5</i>	4	3.868	5.285	5.255	4.214
	6	5.542	7.191	7.162	6.113
<i>Doc6</i>	4	3.771	5.273	5.247	4.207
	6	5.670	7.199	7.167	6.119

our results in Table 4 with other existing thresholding techniques like Niblack [17], Sauvola and Pietikäinen [23], Vu et al. [30] for showing the better effectiveness of computation time of our algorithm. It is clearly observed that integral image based technique for multilevel segmentation of document images produces effective text extraction in very less computation time as compared to existing techniques. Comparative bar graphs are shown for level 4 and 6 in Fig. 3 and the results suggest that proposed method computes result faster over Niblack, Sauvola, and Vu methods for multilevel segmentation for the small window size like 15×15 and as well as size like 40×40 . Now to test the visual clarity of the results, qualitative comparison is carried out in next section.

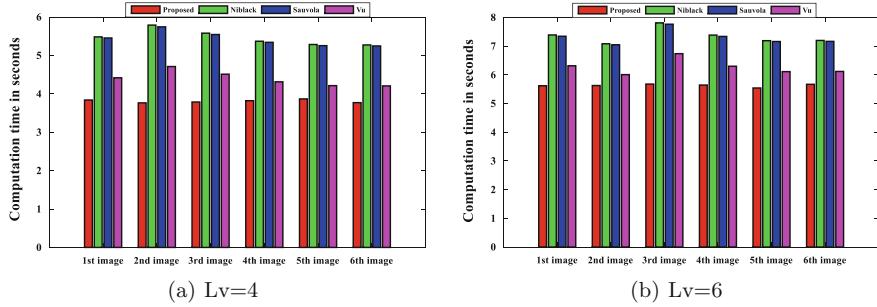
**Fig. 3** Comparison of computation time using DE**Table 5** Comparison of threshold values obtained by different optimization techniques

Image	Level	Threshold values		
		DE	PSO	GA
1	4	49, 99, 163, 215	58, 107, 182, 207	56, 104, 160, 228
	6	35, 71, 102, 155, 193, 227	41, 79, 109, 175, 204, 232	39, 76, 107, 167, 191, 237
2	4	46, 95, 161, 212	54, 106, 180, 205	53, 103, 157, 225
	6	32, 67, 100, 151, 190, 223	40, 77, 106, 172, 201, 227	37, 74, 101, 160, 188, 233
3	4	52, 103, 165, 219	60, 109, 185, 210	59, 107, 162, 230
	6	38, 73, 105, 157, 196, 230	43, 84, 111, 179, 207, 238	40, 77, 109, 169, 194, 239
4	4	48, 100, 163, 216	57, 109, 183, 209	53, 102, 160, 227
	6	37, 73, 107, 155, 191, 228	42, 79, 112, 176, 205, 233	38, 77, 109, 168, 190, 238
5	4	41, 91, 155, 211	51, 100, 180, 201	50, 99, 156, 220
	6	30, 63, 94, 150, 190, 220	33, 76, 105, 170, 201, 224	30, 71, 101, 166, 184, 232
6	4	57, 104, 169, 218	63, 110, 190, 217	61, 100, 169, 232
	6	44, 77, 109, 159, 198, 231	46, 87, 117, 184, 213, 239	44, 87, 120, 175, 199, 239

5.2 Qualitative Measurements of Result

Qualitative measurement checks whether all pixels of the image having same values are properly extracted and assigned to the same class or not. It is the basic criterion for effective segmentation that every region should contain homogeneous pixel values. From Table 5, threshold values for different optimizers can be seen. DE with the objective function produces better thresholds for level 4 as well as in level 6 compared to PSO and GA. Based on these threshold values, segmentations are carried out of different levels for the proposed and above mentioned approaches, clearly shown in Table 2. Human vision with naked eye cannot judge the result properly all the time because judgment varies to individuals. As we know non-textual and textual regions after segmentation hold similar pixel values, so popular segmentation measurement metrics like PSNR and SSIM can easily be applied to hold the strong proof behind our claim.

Table 6 PSNR values obtained from different optimization techniques (6 level)

Im	Proposed method			Sauvola method			Vu method			Niblack method		
	DE	PSO	GA	DE	PSO	GA	DE	PSO	GA	DE	PSO	GA
1	35.37	34.35	34.21	32.56	31.43	31.39	34.21	33.19	33.06	31.89	30.79	30.58
2	35.54	34.45	34.29	32.76	31.54	31.45	34.56	33.35	33.25	31.97	30.86	30.69
3	35.22	34.15	34.02	32.35	31.31	31.17	34.10	33.09	33.01	31.67	30.62	30.46
4	35.79	34.66	34.45	32.78	31.56	31.47	34.52	33.34	33.18	31.99	30.86	30.72
5	35.76	34.61	34.40	32.75	31.63	31.44	34.49	33.16	33.01	31.96	30.74	30.54
6	35.13	34.21	34.07	32.21	31.20	31.09	34.01	33.05	33.01	31.35	30.65	30.41

5.2.1 Peak Signal to Noise Ratio (PSNR)

The dissimilarity between the original and segmented images is measured by evaluating PSNR values in decibel unit. High PSNR values suggest the better segmentation quality of the images. It can be calculated in the following manner:

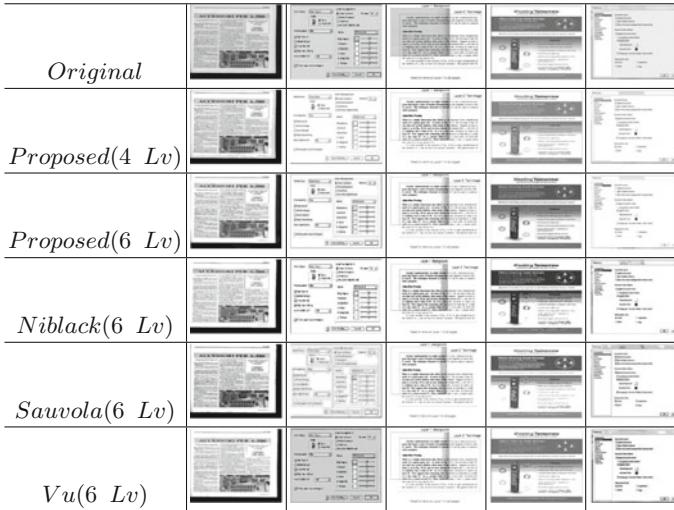
$$PSNR = 10 \times \log_{10}\left(\frac{255^2}{RMSE}\right) \text{ (dB)} \quad (13)$$

where 255 is the maximum gray value of gray scale images and RMSE stands for root mean square error, defined in the following equation

$$RMSE = \frac{1}{r \times c} \sum_i^r \sum_j^c \{im(i, j) - im'(i, j)\}^2 \quad (14)$$

where im and im' are the original and segmented images, respectively. R and C are the row and column size whereas i, j represent the pixel values with size 200×200 of original and segmented images respectively. It can be observed from Table 6 that the proposed method with DE produces high PSNR values in comparison with Sauvola, Niblack, and Vu approaches. It is also seen that DE as an optimizer overcomes the challenges of PSO and GA with all the approaches including our proposed one. PSO and GA almost produce same PSNR values for text extraction whereas Vu method follows our result immediately compared to Sauvola and Niblack methods.

Now we have considered some more document images from our test database to extract the text regions effectively by multilevel thresholding. Original images with their 4 and 6 level thresholding including 6 level thresholding results of Niblak, Sauvola and Vu approaches are shown in Table 7. Next, to analyze the segmented results, measurement metric as Structural Similarity Index Measurement (SSIM) used.

Table 7 Qualitative comparison of proposed method with other existing methods for some more sample images**Table 8** SSIM values obtained from different optimization techniques (6 level)

Im	Proposed method			Sauvola method			Vu method			Niblack method		
	DE	PSO	GA	DE	PSO	GA	DE	PSO	GA	DE	PSO	GA
Doc1	0.89	0.87	0.86	0.86	0.85	0.84	0.88	0.87	0.86	0.85	0.84	0.83
Doc2	0.87	0.85	0.84	0.84	0.82	0.83	0.86	0.85	0.84	0.83	0.82	0.81
Doc3	0.9	0.88	0.87	0.87	0.86	0.85	0.89	0.88	0.87	0.86	0.85	0.84
Doc4	0.88	0.86	0.85	0.85	0.84	0.83	0.87	0.86	0.85	0.87	0.86	0.85
Doc5	0.91	0.89	0.88	0.88	0.87	0.86	0.9	0.89	0.88	0.87	0.86	0.85

5.2.2 Structural Similarity Index Measurement (SSIM)

This measurement technique is also useful to evaluate the visual similarity between the original and segmented images and can be calculated in the following manner:

$$SSIM = \frac{(2\mu_{im}\mu_{im'} + c1)(2\sigma_{imim'} + c2)}{(\mu_{im}^2 + \mu_{im'}^2 - c1)(\sigma_{im}^2 + \sigma_{im'}^2 - c2)} \quad (15)$$

where μ_{im} and $\mu_{im'}$ are the mean values of the original and segmented image, σ_{im} and $\sigma_{im'}$ are the standard deviations of the original Image im and segmented image im', $\sigma_{imim'}$ presents cross-correlation and c1, c2 are constants which may be equal to a default value 0.065. These values generally lie between -1 and 1, where 1 and -1 represent no segmentation and worst segmentation of image respectively. SSIM values nearby 1 for 6th level indicate better visibility and quality of segmentation. It is observed from Table 8 that DE with the proposed approach generates better values

when compared with other techniques. It can be stated from these results that Vu approach strictly follows our results for effective text extraction then Sauvola and Niblack maintain this order.

6 Conclusion

In this paper, integral image based multilevel thresholding is proposed for effective text extraction from document images. To reduce the high-level time complexity of the multilevel problem, a well-known and fast optimization technique Differential Evolution (DE) is used to find the optimal solutions in search space. When the proposed approach is compared with famous recent multilevel solutions of text extraction provided till now, it is found that overall time complexity of multilevel problem has been reduced a lot because of the solution provided by the combination of integral image with DE. The visual results and measurements clearly produce the clarity and effectiveness of the proposed approach for extraction of texts from even complex colored document images contained by the book, journal, a magazine cover, etc. This innovative idea for solving multilevel document image segmentation problem with colored background will reduce computational cost and complexities almost to half as compared to the above-stated techniques.

References

1. S. Abdel-Khalek, A.B. Ishak, O.A. Omer, A.S. Obada, A two-dimensional image segmentation method based on genetic algorithm and entropy. *Optik-Int. J. Light Electron Opt.* **131**, 414–422 (2017)
2. M.N. Alam, Particle swarm optimization: algorithm and its codes in matlab (2016)
3. H.V.H. Ayala, F.M. dos Santos, V.C. Mariani, L. dos Santos Coelho, Image thresholding segmentation based on a novel beta differential evolution approach. *Exp. Syst. Appl.* **42**(4), 2136–2142 (2015)
4. D. Bradley, G. Roth, Adaptive thresholding using the integral image. *J. Graph. Tools* **12**(2), 13–21 (2007)
5. L. Cao, P. Bao, Z. Shi, The strongest schema learning ga and its application to multilevel thresholding. *Image Vis. Comput.* **26**(5), 716–724 (2008)
6. A. Chander, A. Chatterjee, P. Siarry, A new social and momentum component adaptive PSO algorithm for image segmentation. *Exp. Syst. Appl.* **38**(5), 4998–5004 (2011)
7. Y.L. Chen, B.F. Wu, A multi-plane approach for text segmentation of complex document images. *Pattern Recognit.* **42**(7), 1419–1444 (2009)
8. M. Cheriet, J.N. Said, C.Y. Suen, A recursive thresholding technique for image segmentation. *IEEE Trans. Image Process.* **7**(6), 918–921 (1998)
9. J.L. Fisher, S.C. Hinds, D.P. D'Amato, A rule-based system for document image segmentation, in *Proceedings, 10th International Conference on Pattern Recognition*, vol. 1 (IEEE, 1990), pp. 567–572
10. L.A. Fletcher, R. Kasturi, A robust algorithm for text string separation from mixed text/images. *IEEE Trans. Pattern Anal. Mach. Intell.* **10**(6), 910–918 (1988)

11. B. Gatos, I. Pratikakis, S.J. Perantonis, Adaptive degraded document image binarization. *Pattern Recognit.* **39**(3), 317–327 (2006)
12. M.H. Horng, R.J. Liou, Multilevel minimum cross entropy threshold selection based on the firefly algorithm. *Exp. Syst. Appl.* **38**(12), 14805–14811 (2011)
13. D. Karaboga, B. Gorkemli, C. Ozturk, N. Karaboga, A comprehensive survey: artificial bee colony (abc) algorithm and applications. *Artif. Intell. Rev.* **42**(1), 21–57 (2014)
14. Y. Liu, S.N. Srihari, Document image binarization based on texture features. *IEEE Trans. Pattern Anal. Mach. Intell.* **19**(5), 540–544 (1997)
15. M. Ma, J. Liang, M. Guo, Y. Fan, Y. Yin, Sar image segmentation based on artificial bee colony algorithm. *Appl. Soft Comput.* **11**(8), 5205–5214 (2011)
16. M. Naidu, P.R. Kumar, K. Chiranjeevi, Shannon and fuzzy entropy based evolutionary image thresholding for image segmentation. *Alex. Eng. J.* (2017)
17. W. Niblack, An introduction to digital image processing. Strandberg Publishing Company (1985)
18. L. O'Gorman, R. Kasturi, Document image analysis, vol. 39 (IEEE Computer Society Press Los Alamitos, 1995)
19. S. Pare, A.K. Bhandari, A. Kumar, G.K. Singh, S. Khare, Satellite image segmentation based on different objective functions using genetic algorithm: a comparative study, in *2015 IEEE International Conference on Digital Signal Processing (DSP)* (IEEE, 2015), pp. 730–734
20. J. Parker, Gray level thresholding in badly illuminated images. *IEEE Trans. Pattern Anal. Mach. Intell.* **13**(8), 813–819 (1991)
21. S. Sarkar, S. Das, S.S. Chaudhuri, A multilevel color image thresholding scheme based on minimum cross entropy and differential evolution. *Pattern Recognit. Lett.* **54**, 27–35 (2015)
22. S. Sarkar, S. Paul, R. Burman, S. Das, S.S. Chaudhuri, A fuzzy entropy based multi-level image thresholding using differential evolution, in *International Conference on Swarm, Evolutionary, and Memetic Computing* (Springer, 2014), pp. 386–395
23. J. Sauvola, M. Pietikäinen, Adaptive document image binarization. *Pattern Recognit.* **33**(2), 225–236 (2000)
24. F. Shafait, D. Keysers, T.M. Breuel, Efficient implementation of local adaptive thresholding techniques using integral images. *DRR* **6815**, 681510 (2008)
25. F. Shafait, J. Van Beusekom, D. Keysers, T.M. Breuel, Page frame detection for marginal noise removal from scanned documents, in *Scandinavian Conference on Image Analysis* (Springer, 2007), pp. 651–660
26. K. Sobottka, H. Kronenberg, T. Perroud, H. Bunke, Text extraction from colored book and journal covers. *Int. J. Doc. Anal. Recognit.* **2**(4), 163–176 (2000)
27. R. Storn, K. Price, Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces. *J. Global Opt.* **11**(4), 341–359 (1997)
28. C.M. Tsai, H.J. Lee, Binarization of color document images via luminance and saturation color features. *IEEE Trans. Image Process.* **11**(4), 434–451 (2002)
29. P. Viola, M.J. Jones, Robust real-time face detection. *Int. J. Comput. Vis.* **57**(2), 137–154 (2004)
30. H.N. Vu, T.A. Tran, I.S. Na, S.H. Kim, Automatic extraction of text regions from document images by multilevel thresholding and k-means clustering, in *2015 IEEE/ACIS 14th International Conference on Computer and Information Science (ICIS)* (IEEE, 2015), pp. 329–334
31. Y. Zhong, K. Karu, A.K. Jain, Locating text in complex color images. *Pattern Recognit.* **28**(10), 1523–1535 (1995)

Offline Extraction of Indic Regional Language from Natural Scene Image Using Text Segmentation and Deep Convolutional Sequence



Sauradip Nag, Pallab Kumar Ganguly, Sumit Roy, Sourab Jha, Krishna Bose, Abhishek Jha and Kousik Dasgupta

1 Introduction

As Internet and use of social media increase, the rise of easily accessible multimedia data, such as video, images, and text created unprecedented information overflow. In contrast to this, information extraction from Natural Scene is still a daunting task and currently active area of research mainly because of the complexities faced during extraction. India being a vast country has a wide span of uncovered regions and various colloquial languages. Currently, India has 26 different regional languages, many of these languages are native languages of mainly rural areas of India. So, extraction of these regional language from rural area is itself a challenging task, since the rural areas generally do not contain well-defined address. Recent studies explored

S. Nag (✉) · P. K. Ganguly (✉) · S. Roy (✉) · S. Jha (✉) · K. Bose (✉) · A. Jha (✉)
Kalyani Government Engineering College, Kalyani, Nadia 741235, India
e-mail: sauradipnag95@gmail.com

P. K. Ganguly
e-mail: pallabkumarganguly@gmail.com

S. Roy
e-mail: sroy8091@gmail.com

S. Jha
e-mail: sourab.jha@kgec.edu.in

K. Bose
e-mail: krishna.bose02@gmail.com

A. Jha
e-mail: j7.abhi@gmail.com

K. Dasgupta (✉)
Faculty of Computer Science and Engineering, Kalyani Government
Engineering College, Kalyani, Nadia 741235, India
e-mail: kousik.dasgupta@gmail.com



Fig. 1 Represents sample images in different languages having address information

the extraction of Devanagari Scripts from text [1] but these do not give information about the specific regions where these languages are spoken, or the approximate location of the image. Modern applications focus on using metadata of an image to find the ground location of the image and from there find the accurate regional language. However, if images are captured using digital single lens reflex (DSLR) or Digital Camera then the images may not be geotagged, so in such cases, it will fail to get the location. Therefore, it may be concluded that it is very much essential to extract location information from natural scene image offline, as it will be novel in its kind. Thus, the work in this paper focusses on the extraction of location from natural scene image having address information offline and subsequently finding regional language of that region. This is interesting since we are using a text candidate segmentation, which outperforms the existing methods in performance in both low and high resolution. We are doing the entire process offline by training the proposed text recognition model for different regional languages. A sample collection of images has been generated from the Government of India dataset [2] for experimental study. Some sample images are shown in Fig. 1. The main contribution of the proposed work is

- Using image correction for low-quality images and using existing deep learning method to improve performance.
- Using deep convolutional sequence to detect Indic languages as shown in Fig. 1.
- Detecting location from an image having Address Information in native languages.
- Detecting the regional languages from its address information.

The rest of the paper is laid out as follows: In Sect. 2, we discuss some related works regarding detection of text and text recognition using both traditional and Deep Learning based methods and finally extraction of language from it. In Sect. 3, the proposed text segmentation and connectionist text proposal network for Indic languages have been described alongside regional language extraction from location information. In Sect. 4, we discuss the details of dataset, the proposed experimental setup, the comparison of proposed and the existing methods and results in details.

Section 5 points out the efficacy and drawbacks of the proposed approach, some direction towards future work, concluding with suggestive applications.

2 Related Work

Since natural scene image is occluded by various lightning condition, text orientation, inappropriate text information, so extraction of text candidate region from the natural scene image is very important in this literature. Tian et al. [3] proposed detecting text in natural image with connectionist text proposal network. The method detects a text line in a sequence of fine-scale text proposal directly convolutional feature maps. This method explores context information of the image for successful text detection. The method is reliable on multi-scale, multi-language text. Ami et al. [4] used face, torso information to detect text from marathon images in challenging conditions. Shivakumara et al. [5] used snooper text detector for detection of text from the natural scene. The classical methods used several techniques like Lee and Kim [6] proposed. Raj and Ghosh [1] proposed the detection of Devanagari Scripts from the natural scene using morphological operations on connected components and passing it to OCR. Similarly, Wu et al. [7] proposed a mechanism using contour component's gradient angle and Fourier phase angles. However, it has been seen in the survey of Zhang et al. [8], the accuracy of such classical text detection and recognition methods using image processing does not reach the desired accuracy. Shi et al. [9] proposed detecting oriented text in natural images by linking segments. The idea here is that the text is decomposed into segments and links. A segment is an oriented text box covering part of text line while links connects two adjacent segments, indicating that they belong to the same text line. To achieve this, the method explore deep learning concept. Zhou et al. [10] proposed an efficient and accurate scene text detector. The method predicts the text line without detecting words or characters based on deep learning concepts, which involves designing the loss of functions and neural network architecture. The method works well for arbitrary orientations. He et al. [11] proposed Deep Text Recurrent Network (DTRN), which generates ordered high-level sequence from a word image using Maxed out CNN and then uses Bi-LSTM to detect the generated CNN sequence. These works on text detection on natural scene images consider high-resolution, high-contrast images. These methods directly or indirectly explore connected component analysis and characteristics at character level for successful text detection of arbitrary orientation, multi-language, font face, font size, etc. However, the performance of the methods degrades when the image contains both high and low contrast text. To overcome the problem of low contrast and resolution, the proposed method is effective as the Image with poor quality is fixed using image correction techniques and then uses EAST [10] to detect text boxes. It also uses convex hull to generate orientation wise correct text segmentation. The work in this paper presents a new method of text candidate region selection and segmentation by exploring. This work explores the problem of regional language detection from both low quality and high-quality

images. Since Deep Learning based approaches perform better than conventional approaches, Neural Network based Architecture is used for both Text Detection and Text Recognition which is elaborated in details in the upcoming sections.

3 The Proposed Methodology

In this work, an RGB image having at least 1 line of address Information is taken as input. This input image is then preprocessed for image correction using techniques like applying Gaussian Blur to reduce noise and applying Gamma Filter for enhancing contrast for night images. Then the Meta information is checked for GPS coordinates, if it is present then we proceed to find Regional Language. If not present we pass this corrected Image into EAST [10] Text Detection Pipeline which uses standard PVAnet and Non-Maximum Suppression Layer as a part of its architecture. This returns the Bounding Box over probable Text Region. We binarize the image and perform periodic growing of the boxes on both sides until neighboring boxes overlap. In this way, we make sure that the Adjacent Information is not lost. Still due to orientation some important information may be lost, so for that, we apply Polygonal Convex Hull to correct the Segmentation and subsequently mask out the Color Image. After Segmenting the Text ROI from the Image, we use DTRN [11] since Address information can be in any font, we chose a method that is not dependent on Nature of Text. Motivated by the work of He et al. [11] we use MaxOut CNN to get High-Level Features for Input Text instead of grueling task of Character Segmentation from input since it is not Robust and Trustable. After this step as given in Fig. 2, these high-level features are passed into Recurrent Neural Network architecture. Since we need address information which is quite sensitive to the context of a line, it may suffer from vanishing gradient problem. So Bi-LSTM is used since it solves the vanishing gradient problem and it can learn meaningful long-range interdependencies which is useful in our context especially when our main task is to retrieve Address Information from Text Sequence. A Connectionist Temporal Classification (CTC) Layer has been added to the output of LSTM Layers for sequential labeling. We have trained this architecture for Hindi and Telugu which is our contribution. We pass the same input image for all the three-language architecture (English, Hindi, Telugu). The softmax Layer which is added in the End of MaxOut CNN Layers returns confidence scores for detection of High-Level Features. We pass high-level features of that architecture whose softmax score is above some threshold. After recognition, we detect the language using Google’s Offline Language Detector Module [12] and subsequently translate to English for better recognition rate.

The proposed Approach then uses Moses Tokenizer [13] to tokenize the text as this tokenizer tokenizes the words without punctuation from the raw translated text. Each of these tokens/words is queried individually on post office Database collected from Government Of India to obtain the GPS coordinates of all the common post office of the locations found using Tokenization. After obtaining (Pincode, Longitude, and Latitude) for a particular image, we use reverse geocoding [14] and database RLDB

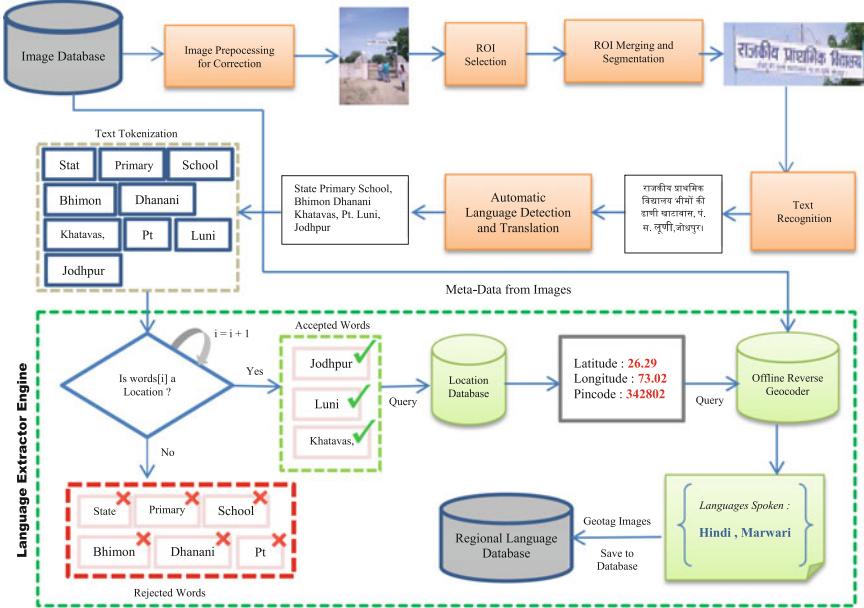


Fig. 2 Framework of the proposed method

created by collecting Regional Language Information found in [15]. From this, we obtain the Regional Language Spoken for the particular location found in Input Image and finally, we Geotag the images and save them. The complete framework of the proposed method can be seen in Fig. 2.

3.1 Image Correction

Any picture from the natural scene can contain certain amount of noise, blur and varying lightning condition, which may affect any method for text candidate selection. So extra care must be taken to ensure that the performance must not drop irrespective of the quality of picture if it has at least one address information. So, to correct the image before text detection, the input image is preprocessed. Bright or Dark Image is identified using the Histogram of Intensity. If the input Image is night image (dark) then a Gamma Filter with Gamma Value 2.5 is applied. The value 2.5 is empirical and is obtained after proper experimentation. So, this gamma value is used to correct it as shown in Fig. 3b. Noise is removed from color image by using nonlocal de-noising method [16] and unsupervised Weiner Filter [17] is used to de-blur the de-noised image as shown in Fig. 3c.

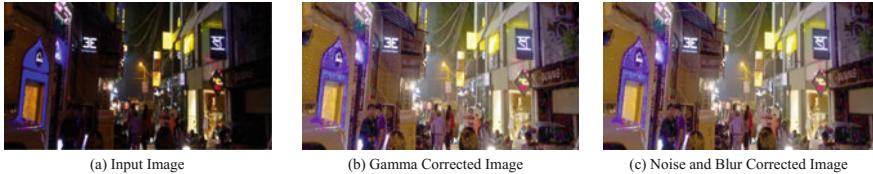


Fig. 3 Depicts how input image is preprocessed before text detection

3.2 Text Detection and Candidate Segmentation

Text Detection in Natural Scene has been an area of research for some years in the Recent Past. Conventional Approaches Rely on manually designed features like Stroke Width Transform (SWT) and MSER based methods which identify character candidates using Edge Detection or extremal region Extraction or contour features. Deep Learning Methods outperform these methods as these methods fall behind in terms of robustness and accuracy.

Text Detection: Since Zhou et al. [10] proposed EAST which predicts text line without detecting words or characters based on deep learning concepts, this approach is suitable for Extracting Information like Text in Multiple Fonts. EAST [10] also implemented Geometry Map Generation which can Detect Text in Arbitrary Orientation without losing Text information with the help of rotating box (RBOX) and Quadrangle (QUAD). But EAST [10] fails for Low-Quality Images since Trained Model used High-Quality Images from Standard Datasets. As mentioned in Sect. 3.1, the Low-Quality images are improved and fed to the Text Detection Pipeline. The pipeline proposed by Zhou et al. [10] consists of only two stages namely Fully Convolutional Neural Network (FCN) and Non-Maximum Suppression (NMS). The FCN internally uses a Standard Neural Network Architecture called PVAnet which consist of 3 internal Layers: Feature Extraction Layer, Feature Merging Layer, and Geometry Map Generation for Bounding Box which is after the Output Layer as depicted in Fig. 4. The input image to this step is the Uncorrected Low/High-Quality Images. The input image is then resized and fed to the Neural Network Pipeline where it extracts the feature relevant to text in first layer, and in second layer it merges the features and extracts possible text region mapping using Geometry Map Generation losses. After this step, the NMS layer selects the maps which are only for text and marks a Bounded Box around it as shown in Fig. 4c.

Text Segmentation: Since the Architecture mentioned above selects individual boxes, it may be difficult to extract location information. Since each location connected by “-”(dash) may represent a location and EAST alone will only detect the two connected words individually which may create ambiguity in location extraction. Hence, we need to process the Image and segment out the Text Candidate Region. The Proposed Pipeline thus contains three key steps: Image Correction, Text Detection, Merging and Segmentation as shown in Fig. 5. Input Image is corrected by maintaining Intensity, Blur Correction, and De-noising. This image is then passed

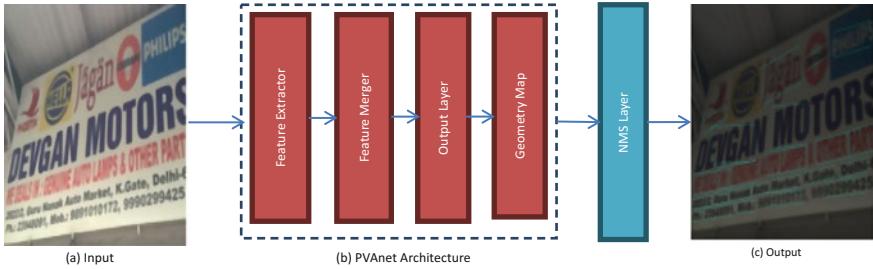


Fig. 4 Architecture of text detection pipeline of EAST [10]

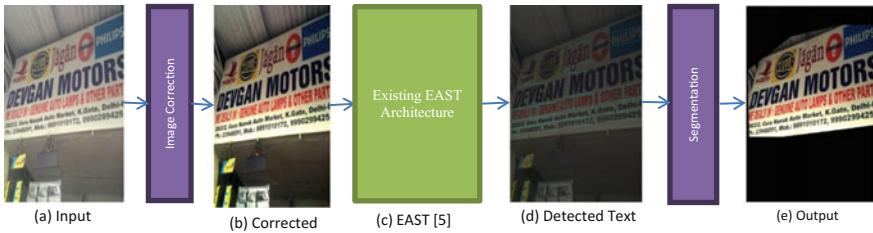


Fig. 5 Architecture of proposed text detection pipeline

onto the Existing EAST [10] Pipeline as mentioned in Fig. 4. This FCN takes corrected images as input and outputs dense per pixels prediction of Text Lines. This eliminates the time consuming Intermediate Steps which makes it a fast text detector. The postprocessing steps include thresholding and NMS on predicted Geometric Shapes. NMS is a postprocessing algorithm responsible for merging all detections that belong to the same object. The output of this stage is bounded box on the text regions as shown in Fig. 5d.

After the formation of bounding boxes over the image as shown in Fig. 6c the box sizes are increased in a periodic manner on both the Sides until it overlaps with neighboring bounding box. In this way, we make sure that the Neighboring Textual Information is not lost. This is represented in Fig. 6d. After this operation, we observe that the segmented image is not uniform (in Fig. 6d) and has lost Information which may affect the overall results since address information is interlinked and order of text matters. So the proposed approach first binarizes the incorrect segmentation and then uses Polygon Convex Hull as in Fig. 6f to correct the Image and the text information which was lost due to orientation, and box merging are recovered. Finally, we masked out the region from Color Input Image as shown in Fig. 6g and 5e. Now this will be the input to the Text Recognition Stage.

Now the Text Recognition Method will not search the entire Image for text instead it will search on a smaller region which makes this method quite Fast and Accurate. The whole process of Execution takes less than 15 s for images having size of 2000 × 1500 on a Intel Core i5-5200U CPU @ 2.20 GHz Processor with 8 GB Ram. Hence,

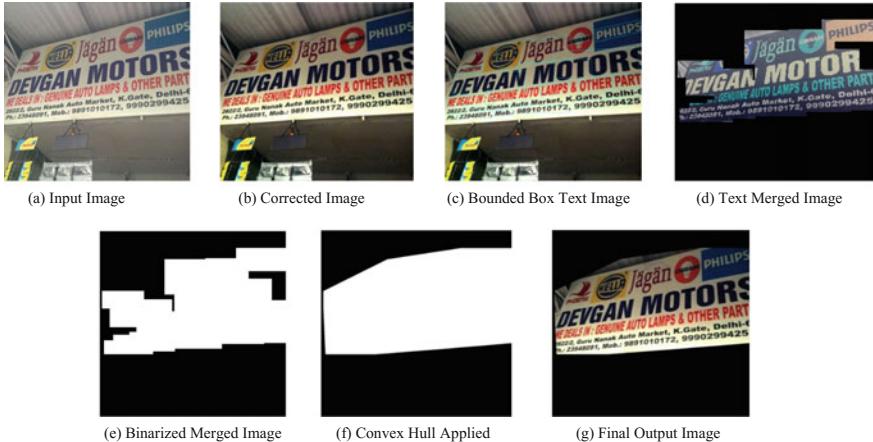


Fig. 6 Represents step by step process of detecting text from input image and segmenting it for text recognition step

it segments out the text in a time which is almost equivalent to the time taken by EAST [10] to process an image individually.

3.3 Text Recognition and Keyword Extraction

Detecting Text from Natural Scene is a daunting task due to the changing background in different images and other external factors. Several works have been done on detecting text from Image using both traditional methods and more recently with Deep learning methods. But in most cases, the Deep Learning methods outperform the traditional methods due to the complexity and orientation. However, it must be noted that the images used for training and testing the Deep Learning Based Methods require a high DPI, so images must not be of low resolution. The modern approaches use Deep Learning Based Text Detection and Text Recognition Pipeline [10]. However, these methods fail to detect a candidate region in low-resolution images. But the proposed approach can detect text candidate in both high and low-resolution images better than other existing approaches and for text recognition purpose we have used Deep Learning Based Text Recognition Method as proposed by Tian et al. [3]. Since the method with connectionist text proposal network detects a text line in a sequence of fine-scale text proposal directly convolutional feature maps. Unlike the problem of OCR, scene text recognition required more robust features to yield results comparable to the transcription-based solutions for OCR. A novel approach combining the robust convolutional features and transcription abilities of recurrent neural networks as introduced by [37]. DTRN [11] uses a special CNN called MaxOut CNN to create High-Level Features and RNN is used to decode the Features into word Strings.

Since maxout CNN is best suited for character classification, we use this on entire word image instead of whole image approach. We pass the bounding box text information of the boxes which lie in the Segmented Text Region shown in Fig. 6g to this MaxoutCNN for extracting High-Level Features for Extraction. The features are in the form of sequence $\{x_1, x_2, x_3, \dots, x_n\}$. These CNN features $\{x_1, x_2, x_3, \dots, x_n\}$ are fed to the bidirectional LSTM in each time sequence and produce the output $p = \{p_1, p_2, p_3, \dots, p_n\}$. Now the length of input sequence X and length of output sequence p is different, hence a CTC Layer is used which follows the equation:

$$S_w \approx \beta(\arg \max_{\pi} P(\pi|p)) \quad (1)$$

where β is a projection which removes repeated labels and noncharacter labels and π is the approximate optimized path with maximum probability among LSTM outputs. For example, $\beta(--ddd--ee--l--hh--i--) = delhi$, hence it produces sequences $S_w = \{s_1, s_2, s_3, \dots, s_n\}$ which are of the same length as Ground Truth Input. India being a vast Country has a diverse spread of Language [15], since training data collection for these languages are challenging, for the present, we have focused to use English, Hindi, and Telugu Text Recognition in this work. For training the CNN part of DTRN [11], we collected individual character images from various datasets [2, 18–20] and we trained the Maxout CNN individually for English, Hindi, and Telugu. For training samples, we collected individual character from [18–20] for English Texts, Hindi, and Tamil were obtained from [2, 20] and publicly available images in Google. In total, we trained characters of multiple fonts, each class has 10 different sample of varying width size and style which were collected from around 2000 Images for English, 1100 for Telugu and 1800 for Hindi Scripts. For training RNN part, we took the images from the training samples [2, 18–20] and cropped out word images individually and collected 3700 words for English, 1900 for Telugu and 2900 for Hindi Language. Although He et al. [11] proposed DTRN for word recognition, we have extended this to sentence recognition with good accuracy. The hyperparameters used are: Learning Rate fixed to 0.0001 and momentum of 0.9. The Loss Function used in this Recurrent Layers is defined in Eq. (2) as shown below:

$$L(X, S_w) \cong - \sum_{t=1}^T \log(P(\pi_t|X)) \quad (2)$$

where $\pi \in \beta^{-1}(S_w)$. Then, this approximate error is backpropagated to update parameters. The Raw Text is then Translated from Native Language to English Language using [21]. After this step, the proposed approach uses Moses Tokenizer [13] which is especially good in terms of Tokenizing phrases/words without Punctuation than other existing Tokenizers. Each Token is then Passed to Google Location Offline Database to check if it is a location, if it is location, we categorize such tokens as keyword which are most likely to generate Location Information else we reject the token. The reason behind Tokenizing text is to remove ambiguity for the name of place which can exist in two or more different states whose regional language is different. For example, “Raniganj Bazar” is a place in the State Uttar Pradesh, India

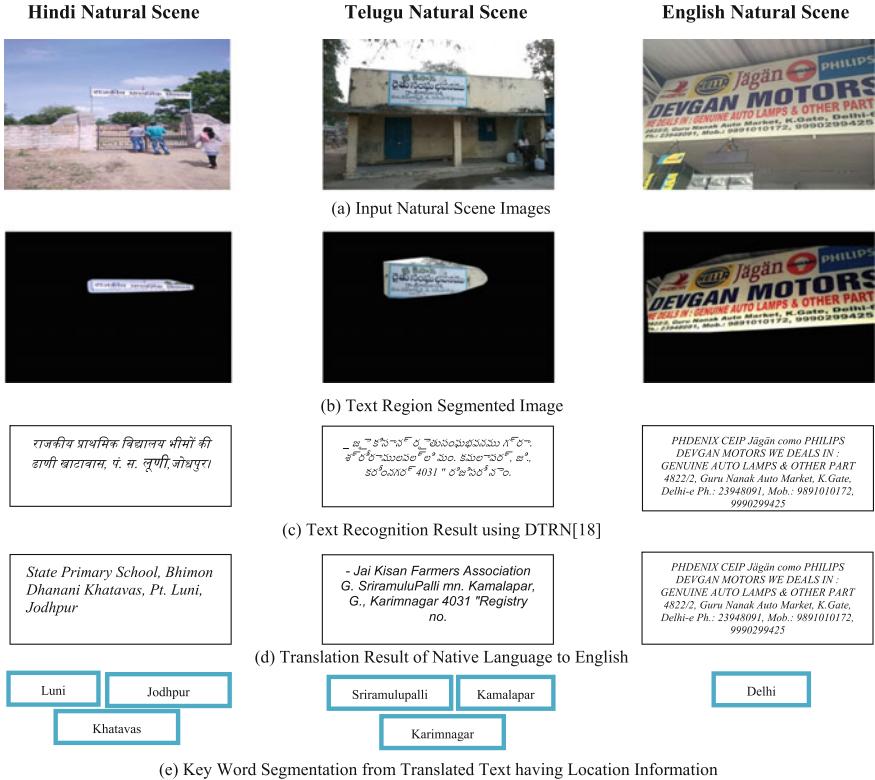


Fig. 7 Representation of how key location word is extracted from segmented text using DTRN

but “Raniganj” is also a Place in state West Bengal, India where Regional Languages are Hindi and Bengali respectively. In such cases, we look for secondary information which is discussed later in Sect. 3.4.

The illustration in Fig. 7 explains the importance of Text Recognition in keyword extraction from the Segmented Image. So segmentation and text recognition plays an important role in detection of Regional Language. Moses Tokenization is particularly Important in Conversion of Fig. 7d, e since any wrong keyword may result in extraction of Different Regional Language. The step-by-step process from the bounding box information of the Input Image to keyword Extraction including the intermediate steps are clearly illustrated in Fig. 8.

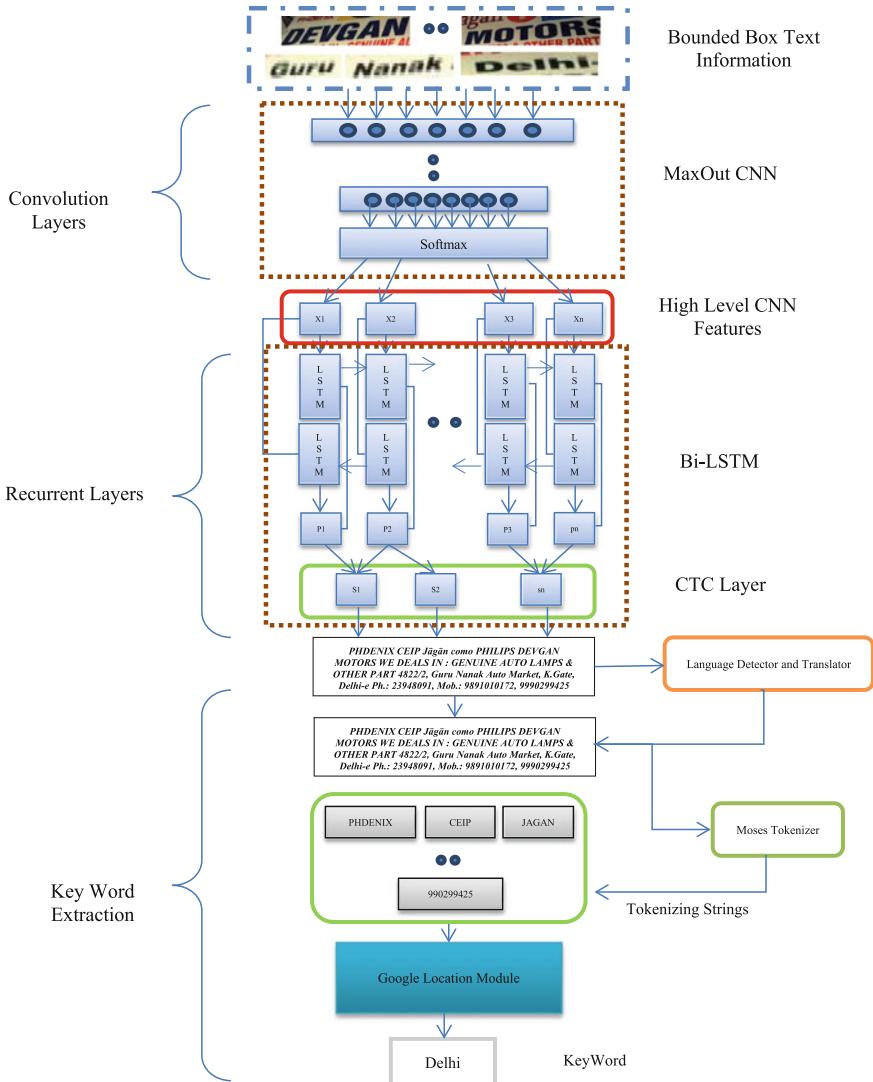


Fig. 8 Pictorial representation of text recognition and keyword extraction pipeline architecture

3.4 Regional Language Extraction from Location Information

The keywords obtained from the previous section is used for detecting regional language. For each token, we query the database collected from post office database from Government of India. Our choice of database is post office because post office

Table 1 Representing the post office database [22] CSDB with its schema and internal structure

Div_Name	Pincode	Taluk	Circle	Region	District	State
A-N Islands	744112	Port Blair	West Bengal	Calcutta HQ	South Andaman	Andaman and Nicobar Islands
Saharsa	852201	Kahara	Bihar	Bhagalpur	Saharsa	Bihar

Table 2 Representing the GPS database [23] LLDB with its schema and internal structure

City Id	City_Name	Latitude	Longitude	State
1	Port Blair	11.67 N'	92.76'	Andaman and Nicobar Islands
255	Saharsa	25.88 N'	86.59 E'	Bihar

is present in every corner of India and since we are targeting regional language we are covering entire Country. The keywords are queried in the Database where the Primary Key is Composite Key of (Division Name, Taluk, Pin code), Taluk is generally a Administrative center controlling several villages, hence along with Pin code, it determines a region uniquely. This database is referred to as CSDB (City and States Database) [22] as shown in Table 1. Since we need location information, so we also need another database which contains GPS information corresponding to such regions. For that, we consider the Database [23] which contains City ID, City, State, Longitude and Latitude as attributes and City Id as Primary Key. This database is referred to as LLDB (Latitude and Longitude Database) as shown in Table 2.

Using a Nested Query as shown below in Fig. 9, we will fetch all possible tuples of combination (Latitude, Longitude, Pin Code) from the keyword token individually for a particular Image and then we take the common tuple among all keywords of the image to get the approximate location. In cases where we have ambiguity in Location as mentioned in previous Sect. 3.3, we first observe the common tuple among keywords, if not found then we conclude that there is ambiguity and we are missing some information. To solve this issue we propose a method to check secondary information like the detected language of Text Recognition, even if we cannot separate after this step then we combine adjacent tokens pairwise from left to right and re run the same queries again to check common tuple. If we fail after this step, we conclude that Location cannot be found and some more operations or information needs to obtained. After this operation we perform reverse Geocoding [14] on (Longitude, Latitude) as input to obtain the approximate location. Subsequently, we query the location from Language Database (RLDB), which is created by us with data collected from Wikipedia. Finally, we obtain the Regional Language set which is obtained for a particular input image. We Geotag the Image with the Language Information and store it in a Database.

```

for i in keywords :
    tuples[j++] = sql.execute(" select t1.latitude,t1.longitude,t2.pincode
                                from LLDB t1,CSDB t2 on t2.Division_Name = t1.City_Name
                                where t2.Taluk = '" + i + "'; ")
end
com_loc=common(tuples);
if (com_loc.size() > 0 )
    place = reverse_geocode(com_loc.latitude,com_loc.longitude)
else
    Solve Location Ambiguity
end

```

Fig. 9 Illustrates pseudo code for extraction of location from keywords

4 Experimental Results

For evaluating the proposed Method, we created a Complex Dataset comprising of Standard Datasets like ICDAR-15, KAIST, NEOCR [2, 18–20]. The input images contain languages of three classes namely English, Hindi, and Telegu. Each class has images ranging from daylight image to night image. Images were reportedly taken from good quality cameras, but to make the dataset diverse, images captured from smartphone cameras of both high and low resolutions are added. The total size of datasets is 8500 images out of which 30% images are low resolution ($<400 \times 400$ pixels) and rest are high resolution. Out of these, 3700 words for English, 1900 for Telegu and 2900 for the Hindi Language. From this dataset, well shuffled 70% data are used for training and the remaining 30% are used for testing the proposed CNN and RNN Models. To measure the performance of Text recognition, we use standard measures namely Recall (R) and Precision (P) as defined in [24]. To evaluate the performance of text detection, we use standard measures as defined in [25, 26]. We also compare location accuracy with Google Maps to justify our offline location searching. To prove the usefulness of method in text detection, we consider standard Deep Learning Methods, EAST [10], Tian et al. [3] and Shi et al. [9]. As per our knowledge, these methods work best for daylight, good resolution images. For the effectiveness of our approach and fair comparative Study, we pass the input images from the database as it is including night images, low-resolution, high-resolution images. For text recognition, we will compare our method with Traditional Tesseract OCR [25] and OCropus [26]. We pass the input images directly to OCR [25, 26] for a fair comparison. So text detection and text recognition are the key steps of this method. We also evaluate the accuracy of location found out using the proposed method.

The following Figure represents the sample database considered for our literature for testing the efficacy of the proposed method (Fig. 10).



Fig. 10 Sample images of the databases

4.1 Evaluating Text Detection

Qualitative results of the proposed and existing methods for text detection in Natural Scene are shown in Fig. 11. It is observed that the proposed method performs well in comparison to the existing methods [9, 10]. Although Deep Learning solves many complex problems like occlusion, font size, it requires a good number of training samples, high-quality images. However, it performs badly in low light images as depicted in Fig. 11a, however proposed approach corrects any Input Image using Gamma Filter as explained earlier, hence detection rate is high. The advantage of the proposed method over the existing method of text detection lies in the fact that it performs exceedingly well in low-resolution and low light images.

Quantitative results of the proposed and existing methods for three databases are reported in Table 3 where it is noted that the proposed method is best at precision, recall, F-measure in comparison to the all the existing methods. It is obvious since, the Image Database (ICDAR-15, KAIST, NEOCR [2, 18–20], Mobile Camera Images) contains images ranging from low resolution to high resolution of various fonts and occlusion, less text information, Deep Learning based approaches sometimes fails to detect text regions but with correction these images produce good results. EAST



Fig. 11 Represents text detection on image database for proposed approach and two existing approaches

Table 3 Performance of the proposed and existing methods for text detection

Methods	Technique	Precision	Recall	F-Score
Proposed method	Deep learning	0.81	0.78	0.80
Seglink [9]	Deep learning	0.73	0.74	0.73
EAST [10]	Deep learning	0.77	0.75	0.76
Yin et al. [31]	Image processing	0.68	0.86	0.71

[10] scores second best among the three deep learning based methods and Seglink [9] performs poorly.

4.2 Evaluating Text Recognition

Text Recognition in Natural Scene is a very interesting work. Due to orientation, sometimes feature extraction becomes quite difficult. Traditional approaches like MSER features work well in images having good lightning, high-resolution images

Table 4 Performance of the proposed and existing methods for text recognition

Methods	Precision	Recall
DTRN [11]	0.77	0.81
Tess-OCR [25]	0.32	0.46
OCRopus [26]	0.58	0.73
Gordo et al. [29]	0.67	0.71
Zhao et al. [30]	0.73	0.66

but fail significantly in low resolution and night time images. Preprocessing improves the accuracy but not to desired standards. Since we are dealing with location extraction from text, recognition of text is very important in this context, so partial text recognition is not reliable. Huang et al. [27] proposed MSER based text recognition but it detects simple text orientations, complex text orientation does not give good results using MSER features. Roy et al. [28] proposed conditional random field model defined on potential character locations and the interactions between them. This works well for individual words and not on region-based word recognition, where words are interlinked, so missing on word leads to lossy information. Although many works used Tesseract OCR [25] for text recognition since it was originally developed for Document Based Extraction, performance significantly drops in Natural Scene due to complex background. Similar scenario with OCRopus [26], it is also developed for document-based extraction but it performs better than Tesseract OCR [25] as shown in Table 4 and Fig. 12, since it uses preprocessing like Deskew, DeNoise, Demask before performing Text recognition. Gordo et al. [29] proposed a fast text recognition technique using a combination of label embedding and attribute learning. Zhao et al. [30] proposed a text recognition pipeline where he proposed a modified VGG-Net architecture for character recognition.

So Deep Learning based method performs significantly better in such scenarios; although DTRN [11] works well for individual words, according to our experiments, it performed well in region-based word recognition. DTRN [11] uses MaxOut CNN to extract High-Level Features from the Region of Interest and then uses Recurrent Layers to connect all information sequentially. Hence it performs better than existing Approaches. The metrics used for performance calculation are Precision and Recall as mentioned earlier.

Location Accuracy is also important since we are investigating Regional Language in this literature. Regional language in India, however, does not change within a few kilometers distance until and unless the location is close to an International Border or State Border, we consider those as exceptions. For Measuring Location, we consider the keywords as locations and search for their GPS coordinates, since this approach is offline there is a high chance of error. So as a ground truth, we search Google Online Maps with the keywords and compare with the Proposed Offline Version to calculate the Distance(d) between the Proposed Point and Ground Truth Point. We run each image individually on 8500 images and calculate “ d ” and finally we take the average, so we found out the average distance of proposed and ground truth point as

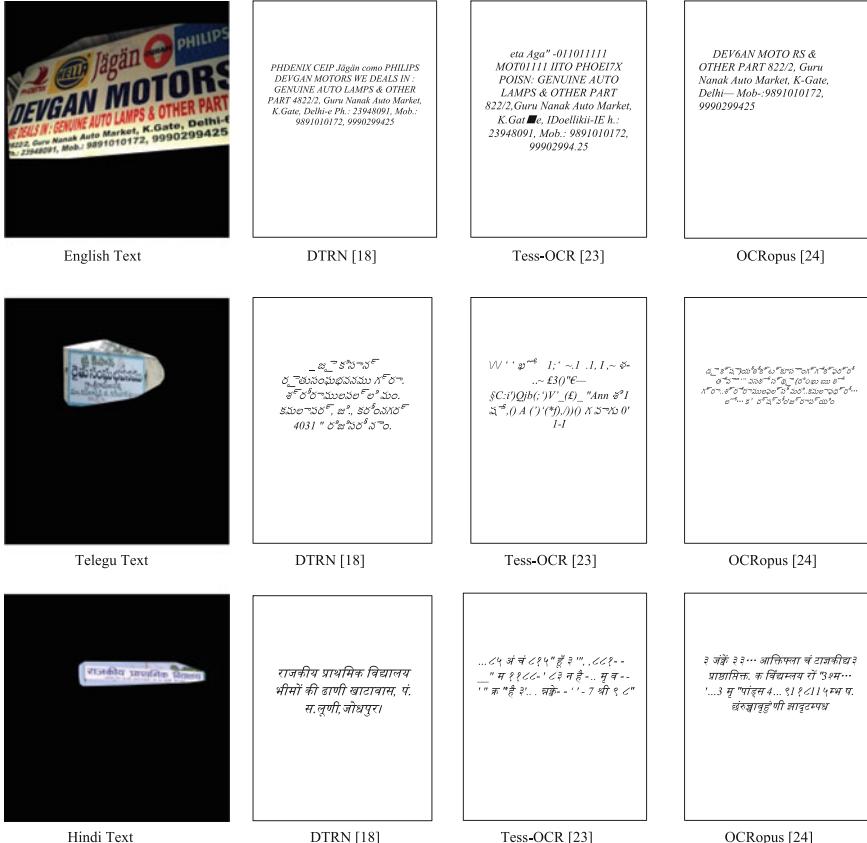


Fig. 12 Represents text recognition results for indic scripts using both proposed and two existing methods

a metric to justify the Accuracy. In our Experiments, the Average Distance between Proposed Point and Ground Truth Point comes out as 11.87 km as shown in Fig. 13.

5 Conclusion and Future Work

This work highlights a novel method in extracting regional language from an image containing any address information. This is a novel work of its kind where regional language is extracted from an Image without Internet (offline). The proposed method also defines a unique text detection approach where Images of low resolution and varying font size can be handled easily and text can be extracted. However, it is noted that the proposed method is not robust to occlusion, severe blur, and too small font as shown in Fig. 14c while capturing image from long distance camera hence

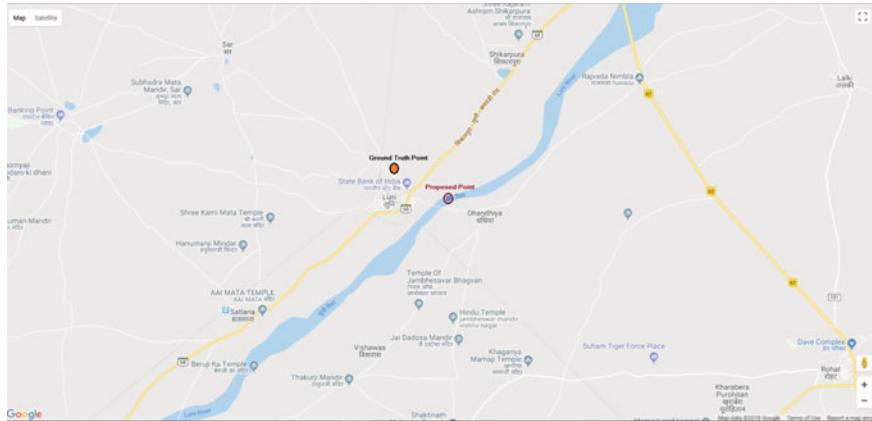


Fig. 13 Illustrates the proposed location obtained empirically from LLDB and ground truth point obtained from Google street on Indian map for location Luni, Jodhpur



Fig. 14 Represents the drawbacks of proposed method

address cannot be extracted. The proposed method also fails in scenarios where advertisements which contains location of office not in the current state as represented in Fig. 14a, here picture location is Kolkata but the proposed method also captures New Delhi but the regional Language of New Delhi and Kolkata are different. It also fails in scenarios like Fig. 14b where the first word is a country which has different regional language. Currently, text recognition can detect English, Hindi, and Telegu Scripts, in future the proposed work can extend to 26 Officially accepted languages in India for unearthing the rarest regional languages which is unknown to mankind. The work needs to further address the location ambiguity using other information to detect accurate languages. The query for selecting the post office can also be optimized based on Ground truth data. The application of this literature lies in the field of identification of unknown places in India and also can be used in computer forensics.

References

1. H. Raj, R. Ghosh, Devanagari text extraction from natural scene images. Int. Conf. Adv. Comput. Informat. 513–517 (2014)
2. <https://nrsc.gov.in/hackathon2018/>
3. Z. Tian, W. Huang, T. He, P. He, Y. Qiao, Detecting text in natural image with connectionist text proposal network, in *Proceedings of ECCV* (2016), pp. 56–72
4. I.B. Ami, T. Basha, S. Avidan, Racing bib number recognition, In Proc. BMVC (2012)
5. P. Shivakumara, R. Raghavendra, L. Qin, K.B. Raja, T. Lu, U. Pal, A new multi-modal approach to bib/text detection and recognition in Marathon images. Pattern Recogn. Voil. **61**, 479–491 (2017)
6. H. Lee, C. Kim, Blurred image region detection and segmentation, in *Proceedings of ICIP* (2014), pp. 4427–4431
7. Y. Wu, P. Shivakumara, T. Lu, C.L. Tan, M. Blumenstein, G.H. Kumar, Contour restoration of text components for recognition in video/scene images, IEEE Trans. IP 5622–5634 (2016)
8. H. Zhang, K. Zhao, Y.Z. Song, J. Guo, Text extraction from natural scene image: a survey. Neurocomputing **122**, 310–323 (2013)
9. B. Shi, X. Bai, S. Belongie, Detecting oriented text in natural images by linking segments, in *Proceedings of CVPR* (2017), pp. 3482–3490
10. X. Zhou, C. Yao, H. Wen, Y. Wang, S. Zhou, W. He, EAST: an efficient and accurate scene text detector, in *Proceedings of CVPR* (2017), pp. 2645–2651
11. P. He, W. Huang, Y. Qiao, C.C. Loy, X. Tang, *Reading Scene Text in Deep Convolutional Sequences* (2015)
12. <https://pypi.org/project/goslate/>
13. <https://github.com/moses-smt/mosesdecoder/blob/master/scripts/tokenizer/tokenizer.perl>
14. https://pypi.org/project/reverse_geocode
15. Regional Language List (https://en.wikipedia.org/wiki/Regional_language)
16. A. Buades, B. Coll, J.-M. Morel, Non-local means denoising. Image Process. Line **1**, 208–212 (2011)
17. J. Chen, J. Benesty, Y.A. Huang, S. Doclo, New insights into the noise reduction wiener filter. IEEE Trans. Audio. Speech. Lang. Process. **14**(4), 1218–1234 (2006)
18. http://www.iapr-tc11.org/mediawiki/index.php?title=KAIST_Scene_Text_Database
19. http://www.iapr-tc11.org/mediawiki/index.php?title=NEOCR:_Natural_Environment_OCR_Dataset
20. N. Sharma, R. Mandal, R. Sharma, U. Pal, M. Blumenstein, ICDAR2015 competition on video script identification (CVSI 2015), in *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*, vol. 2015, no. Cvsi (2015), pp. 1196–1200
21. <https://github.com/libindic/indic-trans>
22. <https://data.gov.in/catalog/all-india-pincode-directory>
23. <https://gist.github.com/gsivaprabu/5336570>
24. A. Kumar, An efficient approach for text extraction in images and video frames using gabor filter. Int. J. Comput. Electr. Eng. **6**(4), 316–320 (2014)
25. <https://github.com/tesseract-ocr/tesseract>
26. <https://github.com/tmbdev/ocropy>
27. X. Huang, T. Shen, R. Wang, C. Gao, Text detection and recognition in natural scene images, in *2015 International Conference on Estimation, Detection and Information Fusion (ICEDIF)* (2015), pp. 44–49
28. U. Roy, A. Mishra, K. Alahari, C.V. Jawahar, Scene text recognition and retrieval for large lexicons, in *Accv2014* (2014), pp. 7–10

29. A. Gordo, A. Forn, E. Valveny, J. Almaz, *Word Spotting and Recognition with Embedded Attributes*, vol. 36, no. 12, pp. 2552–2566 (2014)
30. H. Zhao, Y. Hu, J. Zhang, *Character Recognition via a Compact Convolutional Neural Network* (2017)
31. X. Yin, X. Yin, K. Huang, H. Hao, *Robust Text Detection in Natural Scene Images*, vol. 36, no. 5, pp. 970–983 (2014)

Understanding Email Interactivity and Predicting User Response to Email



Soumyadeep Roy, Nibir Pal, Kousik Dasgupta and Binay Gupta

1 Introduction

Organizations face difficulty in addressing email overload due to the growing customer base and the increased volume of communications both within the organization and outside. Understanding how users interact with email helps organizations to develop targeted strategies by understanding the customer base and also helps in improving the organizational productivity. Controlling email overload at work either by adopting software that is designed for making email easier to use or by adopting effective tactics for using email as a communication medium significantly improves the coordination at work [5]. The average time taken by the employees to recover from an email interrupt and return to the same work rate at which they left it is 64 seconds [2]. As desktop search, machine learning and text processing techniques improve, and we utilize them in order to tackle the problems in task management [11] and in email overload. Intelligent agent-based systems and different information extraction techniques are used to extract content information from email and also generate meaningful user summaries based on the task at hand.

Most of the existing studies have worked on small samples of data or have performed surveys on employees of a given organization. The collected feedback contain details regarding recipient actions like delete, leave in inbox, file, reply, or plan immediately or plan to reply later. Some studies [4, 12] have large data samples. The work in [10] is extended by [3] where they perform analysis on intelligently categorizing the messages and determining its importance to the user. Previous research studies

S. Roy (✉)

Indian Institute of Technology Kharagpur, Kharagpur 721302, India
e-mail: soumyadeep.roy9@iitkgp.ac.in

N. Pal

Dynamic Digital Technology, Kolkata 700091, India

K. Dasgupta

Kalyani Government Engineering College, Kalyani, Nadia 741235, India

B. Gupta

dishq, Bengaluru 560102, India

typically involve small user samples and do not provide insightful models which will help to better understand the recipient action with email [3].

Email rhythms are understood more through relationships than through isolated messages. Email expectation is influenced by the recipient, urgency of topic conveyed by using markers in the subject field or priority flag and also combining it with voice mail and based on the time shift between the sender and the recipient [10]. Emails are also varied based on the content and the recipient type. The expectation and breakdown points are not the same for every email, or even for every recipient. Mackay categorizes the recipients into two categories—prioritizers, who actively maintain good control over their inbox and address the messages as they arrive and archivers, who preserve the email messages for future use and actively ensure that no important messages are missed [9]. The current backlog of emails of a recipient is a significant factor since if one has unusual number of unread messages, one may not be able to maintain the usual responsiveness [10]. This formed the second focus of our study, for which we took into consideration the user preferences extracted from its own email communication history. Certain message content particularly of task management and delegation, scheduling, information exchange, and social communication also changes the importance of a message. A significant proportion of email does not even require any response since many messages are broadcasted to users for informational purposes or send to an additional recipient who is not the primary recipient. The messages that require responses from the recipients are the ones that require user attention and cognitive effort [10]. Sender characteristics accounted for an additional 15% of the total variance in the likelihood of responding to an email [3].

In this paper, we study email interactivity patterns on a very recent dataset from the year 2016 and we observe certain similar patterns to [7] and [6]. Throughout the paper, we use the term mail and email interchangeably implying the same meaning. We determine the feature importance based on information gain. We then use a novel methodology of developing user and send day profiles using k-means clustering as a feature space reduction step, since our training dataset is very large. We then use these features to train three distinct binary classifiers using Logistic Regression, Naive Bayes, and Decision Tree, to predict an email recipient response. Specifically, we predict whether the email recipient will open the email or not.

The paper is organized as follows. In Sect. 2, we discuss the related works in this domain. Section 3 presents the dataset details. In Sect. 4, we derive the feature importance based on information gain and explain how the final feature set is prepared after undergoing the stages of feature engineering and dataset profiling. In Sect. 5, we provide the results obtained by understanding the interactivity of emails with respect to the profiling of the dataset using k-means clustering. We also explain how the features are derived with respect to our classification models. We then compare the performance of our classifiers based on Logistic Regression, Decision Tree, and Naive Bayes. In Sect. 2, we provide the important results.

2 Related Works

Survey information is used to predict recipient action on specific messages as a function of message characteristics like message importance and relationship to the sender and content [3]. Interviews are also a good way to elicit recipient perceptions on email usage but they are unable to verify to what extent these perceptions are actually substantiated in reality at work [10]. Existing works use machine learning and information extraction techniques for addressing email overload. Logs of how quickly people respond to particular email senders over time can be analyzed to develop a response time prediction for a message. We can also determine a threshold to estimate the time instance after which breakdown has probably occurred.

One study [4] uses both local(individual user-specific) and global(sender-specific) features to predict the actions—read, reply, delete, and delete-without-read, instead of determining the importance of the email to the user. In another study [12], both dyadic(one-to-one) and one-to-many email communication in an enterprise email setting is considered. They consider factors like email content and metadata, historical interactions, and temporal features for characterizing the email reply behavior prediction. A large-scale log analysis is carried out in [1] in order to understand differences in query formulation, refinding patterns and intent of search between email search and web search. Another study [7] analyzes the factors like stage as well as history of conversation, email load, day of week, time message is received, user demographics and use of portable devices, for determining the time taken to reply and length of reply. Contrary to previous works, they focus on quantitative measures of overload and its corresponding effect on the user behavior. A study [8] explores the impact of the frequency of checking emails on subjective well-being.

3 Dataset

Our dataset contains metadata of emails sent from HackerRank to the users who have opened their profiles in the HackerRank platform. They are referred to as hacker in the dataset documentation. It is taken from a HackerRank contest¹ held from August 29 to September 4, 2016. It has missing values in hacker timezone and mail category and has inconsistent values in hacker created at, last online, opened, clicked, unsubscribed. Both continuous and categorical attributes are present in the dataset. The training dataset consists of 400 k email records across 54 features, which are sent between February 12 and May 7, 2016. The test dataset covers emails sent between May 8 and May 17, 2016, totaling to 86,048 email records across 48 features. We remove the user reaction information, mentioned in Sect. 3.3 during the testing phase.

¹<https://www.hackerrank.com/machine-learning-codesprint>.

3.1 Contest Information

The features are contest login count, contest participation count, submission count, and contest submission count and their values are provided in terms of last 1, 7, 30, 365 days and from the time of joining HackerRank.

3.2 Account Information

The features are hacker created at, forum comments count, forum count, forum expert count, forum question count, hacker confirmation, user id, hacker timezone, in-product notifications (ipn) count, ipn read, last online, mail category, mail id, mail type, and sent time. These features provide details about the recipient and also include its participation history details in the HackerRank community.

3.3 User Reaction Information

The features are click time, clicked, open time, opened, unsubscribe time, and unsubscribed. These features are associated with the user reactions. They do not appear in the test dataset.

4 Predicting User Response to Email

In this section, we first preprocess and normalize the dataset separately for the different binary classification models. We aggregate the user metadata into user profiles and the sent mail metadata into sent day based profiles by using k-means clustering. We perform feature engineering in stages, where first we derive the feature importance and then reduce the feature set from 62 features to 6 features, through dataset profiling, which we will discuss in detail later in this section. The derived set of features is used for training the binary classification models, Logistic Regression, Decision Tree, and Naive Bayes. Then, we compare the performance of proposed binary classification models and evaluate it with accuracy, precision, recall, and F1 score.

4.1 Preprocessing

We first clean the dataset by removing the observations with missing mail category values and logically inconsistent data conditioned as: hacker created at > last online; opened = false and clicked = true; opened = false and unsubscribed = true. The

missing values of hacker timezone is filled with its median value. Since all the observations start on and after February 12, 2016, we perform normalization on sent time, hacker created at and last online, by subtracting the minimum value of the epoch from the time-stamped attributes. This modification does not affect the relative temporal ordering. In order to address the multicollinearity problem, we compute pairwise feature correlation values. For each feature pair having high correlation values between 0.8 to 1.0 and -0.8 to -1.0 , we only keep only one feature out of them. The mail id and user id features are removed. Decision Tree and Naive Bayes classifier can work with categorical variable. In order to work with Logistic Regression, the categorical variables need to be first converted to binary-valued vectors using one-hot encoding.

We have data from the different timezones and it causes difficulty in understanding the email usage pattern in terms of hour of the day, when mails are more frequently sent and opened, corresponding to each timezone separately, as shown in Fig. 1. For the purpose of consistency, we only consider the data points belonging to the timezone 18,000, which has the highest number of observations among all the other timezones.

4.2 *Dataset Profiling*

After the initial cleaning of the dataset, in this part, we first perform feature engineering, where we derive new features like total mid open, total high open, max open time gap. We then perform feature selection, where we use k-means clustering to perform dataset profiling. This step results in our feature set to be reduced to only six features.

4.2.1 User Profile

Each email recipient identifier is mapped to its user number. We have a total of 28,509 unique users across the entire dataset. More than one data point may be associated with a single email recipient. We assign each user in the user profile dataset to their most recent profile data. We assign the cluster labels obtained after k-means clustering based on user characteristics, as a feature named user cluster number. We now have 73 attributes. We obtain the optimal set of parameter values, after trying different combinations of: k (number of clusters), nstart (the number of random initial cluster sets) and iter.max (maximum number of iterations allowed).

4.2.2 Sent Day Profile

We perform k-means clustering over the sent day profile dataset. We add the features like age of recipient in weeks and percentage of mails opened among all the mails

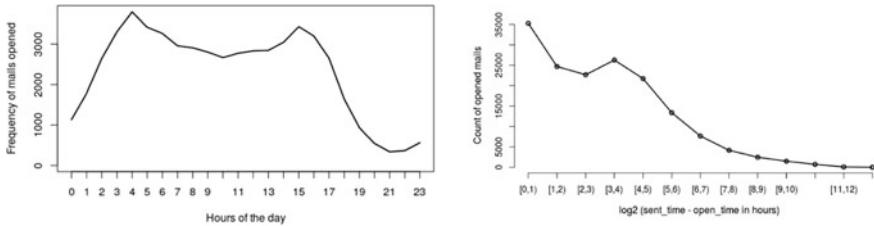


Fig. 1 (left) Variation of mails opened count over each hour of the day in timezone 18,000. (right) Mails opened count distribution against time gap between mail sent and opened in \log_2 hours

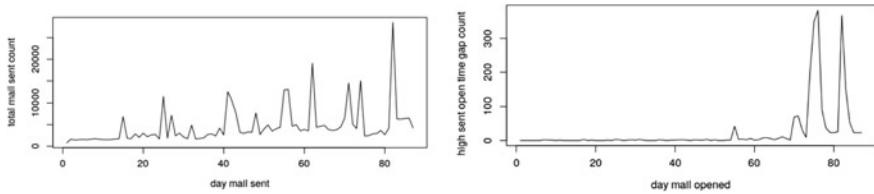


Fig. 2 (left) Variation of total mail sent by HackerRank over all 87 sent days. (right) Variation of high sent open time gap count over all 87 sent days

that are sent on that day and mails that are directly open that day and their temporal behavior is shown in Fig. 2. We also use the user activity related to clicked and unsubscribed actions on emails, as features. We add new features like total mid open, total high open, and max open time gap and analyze their effect on each other. We further preprocess them using min-max normalization.

4.3 Feature Importance

Here, we determine the variable importance of the derived feature set, which is then used for performing feature selection. The features are ranked based on the information gain associated with a Decision Tree classifier. Given the current feature set, we observe that the decision tree classifier cannot linearly split the data, when trying to predict the opened attribute. Upon further inspection based on the top five features based on higher information gain, we cannot differentiate between the mails that are opened and those not opened. We observe similar value distribution as shown in the corresponding box plots in Figs. 3 and 4.

We then focus on dataset profiling in terms of the recipient type and the day it is sent. This significantly improve the model performance and may be because we are using the recipient response history. This characteristic is captured by the opened percent attribute, which also shows a very high variable importance in the classification models developed.

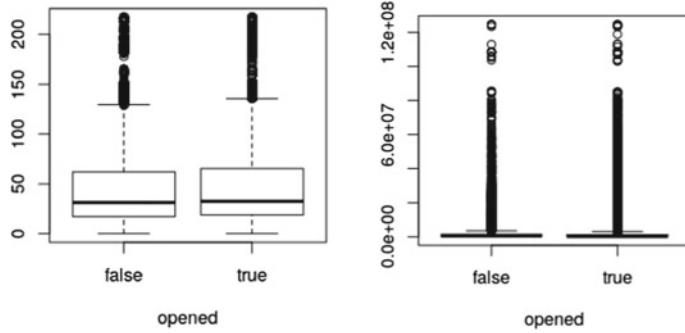


Fig. 3 (left) User profile age in weeks. (right) Difference between sent time and last online diff

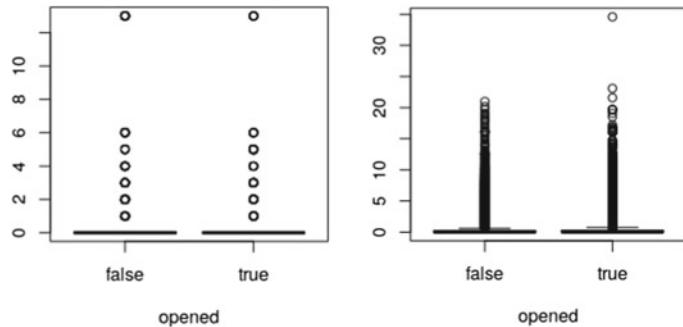


Fig. 4 (left) Forum expert count. (right) Ipn read aggregate

4.4 User Response Prediction

Importance of a message plays a significant role in users action on a message [3] and deduces that the mail content, sender characteristics, and user details that are available from the dataset are crucial for deriving new insights and usage patterns. Since it is proving difficult to construct one-model-for-all approach, we approach it by developing personalized models, which we capture in this study through dataset profiling. After dataset profiling, we finally have the final training dataset, which is now used for predicting the response of an email recipient. We now compare the performance of three distinct binary classification models—Logistic Regression, Decision Tree, and Naive Bayes, over the evaluation metrics namely, accuracy, precision, recall, and F1 score.

4.4.1 Logistic Regression

Logistic regression cannot directly work with categorical variables. In our study, where no ordinal relationship exists in the categorical variables, we use one-hot encoding to convert the categorical data to numerical data as a preprocessing step. The training set now contain 83 features. The `glm` function in R is used to train with the arguments: *family = binomial*. We use the `predict` function for testing purposes and use the arguments: *type = response*. This indicates that the output will be in the form of predicted probabilities. We then determine an optimal threshold to assign a predicted class, given a prediction probability.

4.4.2 Decision Tree

Decision Tree can handle both categorical as well as continuous attributes and therefore the dataset require no further transformation. The training dataset now has 15 features. The model is trained using the `rpart` R package with default arguments. The test output is predicted using `predict.glm` function with the arguments: *type = class*

4.4.3 Naive Bayes

We require the same preprocessing as required for the Decision Tree classifier, since Naive Bayes classifier can also handle categorical and continuous features. The training dataset has six features. We use the `naiveBayes` R package of `e1071` library to train the model using the argument: *type = class*. The test output is generated using its `predict` function.

5 Results

We observe comparable patterns with the existing works from the email interactivity experiments. We then experimentally derive that the dataset for the task of user response prediction is not linearly separable, given the current feature set. We observe it by comparing the value distribution among the different features for each recipient response status. We then address this problem by performing user profiling along with sent day profiling. We can now train three binary classification models using Logistic regression, Decision Tree and Naive Bayes and achieve good performance for the email recipient response prediction task.

5.1 Email Interactivity

We observe a trend in the recipient and sender interaction, in terms of the volume over days, time of day and day of the week, versus the number of mails sent from HackerRank that are opened by the recipients in Fig. 2. The plots obtained give the intuition to develop separate profiles in terms of recipient preferences (uniquely identified by user id) and sent day (derived from sent time), in terms of sudden spikes and behavior specific to the email dataset and in terms of the quantity of mails sent and distinct behavior among the users as shown in Fig. 1. The mail records in the dataset are opened over a duration of 149 days. We also observe certain sudden spikes of high magnitude. We find an inconsistent trend in terms of percent of positive user response over the days the mails were opened, when plotted against the number of mails that crossed a high threshold of time gap between mail sent and when it was opened. The overall email interactivity in terms of the day of the week, time gap between mails sent and opened, opened and clicked, opened, and unsubscribed is also studied.

5.2 Dataset Profiling

We perform k-means clustering over the user profile dataset with argument: $k = 55$ and $nstart = 20$, $iter.max = 50$. The optimum value of (Between_SS/Total_SS) is 0.892.

After performing k-means clustering over the sent day profile dataset, with argument: $k = 5$ and $nstart = 20$. The optimum value of (Between_SS/Total_SS) is 0.853. We try to visualize the clusters using the pair of features having high variance with respect to the sent day profile dataset, which are: gap norm and mid open norm.

5.3 User Response Prediction Comparison

After performing feature selection, we have six attributes and this is the final dataset on which we will apply the three binary classification algorithms, Logistic Regression, Decision Tree, and Naive Bayes. We find that the opened percent attribute has a very high variable importance, which masks the effect of a significant feature—profile age, which is calculated in terms of the number of weeks covered starting from when the hacker profile is created. The features in accordance to their decreasing order of importance are: open percent, user cluster number, sent time gap, hacker timezone, hacker confirm, and profile age in weeks.

We compare the performance of three binary classification models of Logistic Regression, Decision Tree, and Naive Bayes in Table 1. For the current dataset, I

Table 1 Comparison of performance for different models

Approach	Accuracy	Precision	Recall	F1
Logistic regression	0.7941	0.6895	0.5052	0.5832
Decision tree	0.7588	0.5605	0.7139	0.6279
Naive Bayes	0.7588	0.6	0.6529	0.6253

strongly feel recall and F1 score are most appropriate as they give strong preference to the true positives, which are crucial for this dataset.

We find that the Decision Tree performs significantly better due to its recall value of 0.7139 and a higher F1 score of 0.6279. Naive Bayes almost has a similar performance as the Decision Tree but has a more balanced precision and recall score. This is particularly significant for our case, as we are mainly interested to understand what makes a recipient open a mail and the factors responsible for it, thus adding more weightage in correctly identifying the positive examples. All the codes used in this study are available at Github.²

6 Conclusion

The analysis of email-related behavior as a function of message and user characteristics is important for understanding the computer-mediated communication technology as well as for the development of automated tools to help people manage their email. We perform a detailed email interactivity study and observe certain similar patterns as mentioned in the existing studies. The recency of the dataset adds to the effectiveness and usefulness of this study. Then, we propose a novel methodology of feature selection where we perform dataset profiling using k-means clustering, based on the characteristics of the recipient as well as the day the mail is sent. We train three distinct binary classification models using Logistic Regression, Naive Bayes, and Decision Tree to predict whether an email recipient would open the email or not. We also ranked the features of each model in terms of its variable importance, calculated from its corresponding information gain. We compare the performance of the classifiers over accuracy, precision, recall, and F1 score. Decision Tree performs the best with a F1 score of 0.6279 and observe that the most significant feature to this model is the fraction of emails opened by the user in the past. We also observe that the training models achieve high performance, without applying any kind of boosting or ensemble methods.

²<https://github.com/roysoumya/email-interactivity>.

References

1. Q. Ai, S.T. Dumais, N. Craswell, D. Liebling, Characterizing email search using large-scale behavioral logs and surveys, in *Proceedings of the 26th International Conference on World Wide Web. WWW '17, International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland* (2017), pp. 1511–1520. <https://doi.org/10.1145/3038912.3052615>
2. L.A. Dabbish, R.E. Kraut, Email overload at work: an analysis of factors associated with email strain, in *Proceedings of the 2006 20th Anniversary Conference on Computer Supported Cooperative Work, CSCW '06* (ACM, New York, NY, USA, 2006), pp. 431–440. <https://doi.org/10.1145/1180875.1180941>
3. L.A. Dabbish, R.E. Kraut, S. Fussell, S. Kiesler, Understanding email use: predicting action on a message, in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. CHI '05* (ACM, New York, NY, USA, 2005), pp. 691–700. <https://doi.org/10.1145/1054972.1055068>
4. D. Di Castro, Z. Karnin, L. Lewin-Eytan, Y. Maarek, You've got mail, and here is what you could do with it!: analyzing and predicting actions on email messages, in *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining, WSDM '16* (ACM, New York, NY, USA, 2016), pp. 307–316. <https://doi.org/10.1145/2835776.2835811>
5. T.W. Jackson, A. Burgess, J. Edwards, A simple approach to improving email communication. *Commun. ACM* **49**(6), 107–109 (2006). <https://doi.org/10.1145/1132469.1132493>. Jun
6. T.W. Jackson, R. Dawson, D. Wilson, Understanding email interaction increases organizational productivity. *Commun. ACM* **46**(8), 80–84 (2003). <https://doi.org/10.1145/859670.859673>. Aug
7. F. Kooti, L.M. Aiello, M. Grbovic, K. Lerman, A. Mantrach, Evolution of conversations in the age of email overload, in *Proceedings of the 24th International Conference on World Wide Web. WWW '15, International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland* (2015), pp. 603–613. <https://doi.org/10.1145/2736277.2741130>
8. K. Kushlev, E.W. Dunn, Checking email less frequently reduces stress. *Comput. Human Beh.* **43**, 220–228 (2015). <https://doi.org/10.1016/j.chb.2014.11.005>
9. W.E. Mackay, Diversity in the use of electronic mail: a preliminary inquiry. *ACM Trans. Inf. Syst.* **6**(4), 380–397 (1988). <https://doi.org/10.1145/58566.58567>. Oct
10. J.R. Tyler, J.C. Tang, When can i expect an email response? a study of rhythms in email usage, in *ECSCW 2003*, ed. by K. Kuutti, E.H. Karsten, G. Fitzpatrick, P. Dourish, K. Schmidt (Springer, Netherlands, Dordrecht, 2003), pp. 239–258
11. S. Whittaker, V. Bellotti, J. Gwizdka, Email in personal information management. *Commun. ACM* **49**(1), 68–73 (2006). <https://doi.org/10.1145/1107458.1107494>. Jan
12. L. Yang, S.T. Dumais, P.N. Bennett, A.H. Awadallah, Characterizing and predicting enterprise email reply behavior, in *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '17* (ACM, New York, NY, USA, 2017), pp. 235–244. <https://doi.org/10.1145/3077136.3080782>

Load Balancing of Unbalanced Matrix Problem of the Sufficient Machines with Min-Min Algorithm



**Ranjan Kumar Mondal, Payel Ray, Enakshmi Nandi,
Biswajit Biswas, Manas Kumar Sanyal and Debabrata Sarddar**

1 Introduction

We know Cloud Computing is an online-based facility on the Internet environment [1]. It transfers the application and information away from the physical machine into large information centers. Even this type of computing has also changed Information Technology companies utilized to intend software [2]. As it is in its developing phase, therefore there are a lot of complications stay on in its environment [3]. Example:

- To ensure capable access control.
- Migration for the necessity of minimum cost.
- For information move stage and resting stage supply right safety.
- Information Availability.
- Information marsh and transitive cases.
- Information security, the disclosure of susceptible information is feasible, and the vital difficulty in the environment is load balancing. In the event of load balancing,

R. K. Mondal (✉) · P. Ray · E. Nandi · D. Sarddar

Department of Computer Science and Engineering, University of Kalyani, Kalyani, India
e-mail: ranjan@klyuniv.ac.in

P. Ray
e-mail: payelraycse18@klyuniv.ac.in

E. Nandi
e-mail: enakshmicse18@klyuniv.ac.in

D. Sarddar
e-mail: dsarddar1@gmail.com

B. Biswas · M. K. Sanyal
Department of Business Administration, University of Kalyani, Kalyani, India
e-mail: biswajit.biswas0012@gmail.com

M. K. Sanyal
e-mail: manassanyal123@gmail.com

it has altered varieties of information, for example, a number of jobs those are waiting in the queue, and processing time at each processor and neighboring systems also might be exchanged amongst the systems to get the proficient and improved total work. So, many algorithms have proposed for Load Balancing purpose. In this article, we will say an innovative algorithm for proper load balancing to find the developed processing time for issues of improved superiority performance.

Task scheduling process can meet up clients' necessities, and get better the uses of resource, thereby developing the total performance of the cloud. However, the scheduling in a cloud environment is frequently about the static job requirements, and use of the resources rate is small. As a novel characteristic of the cloud, this article presented a two stages task scheduling technique based on load balancing [4].

Task scheduling is a multi-objective combinatorial optimization example in the web, among tasks is spanning and nodes balancing are vital features. Load balancing get-togethers service suppliers' condition. Scheduling determination inevitably reduces the job spanning. That is not interrelated with job finishing time. This article gives a scheduling approach that does not work merely out of a scheduling series with little task runtime and average task makespan but convinces load balancing also [5].

2 Task Scheduling Optimization

Task scheduling is associated with the competence in cloud computing services and is of dominant significance. Scheduling in distributed methods typically has the objective of distributing the loads and makes the best use of their operation, while reducing the overall runtime [6]. Task scheduling has an essential role in developing elastic and dependable distributed systems. Its central principle is to schedule the tasks over resources with dynamism, taking into account the resources existing for implementation involving the innovation of an appropriate [7]. In a cloud environment, the number of tasks of the workflow can modify extremely rapidly, particularly when resources are allocated. The use of the heuristic algorithm makes sure a satisfactory runtime of the scheduling because it considerably decreases the difficulty of the search space. This offers a negotiation among the jobs to processors at execution time and the minimization of the assignment [8]. In prior works, heuristic optimization, GA, fuzzy-genetic algorithm, multi-objective GA, swarm optimization, and regular best-oriented ant colony have been applied to minimize task scheduling, with two primary objectives: (1) to decrease execution time, and (2) to decrease the result of tasks in cloud surroundings [9].

3 Cloud Computing

Now we see what cloud computing is? It is the most demand for web computing and a delivery mechanism nowadays. Still, it is not a new idea, but rather than today's advanced age, it has turned out to be universal because of the growth of the web vastly, broadband, and versatility requirements for users are it consumers, endeavors. The concentration is on the obvious slant of lesser scale and little ventures on the way to cloud computing and the benefits obtained by them. The investigation applies few components impacting the cloud exploit by this commerce group, whose requirements and industry fundamentals are in sum dissimilar from generous endeavors.

Cloud is simple to users and information; they are able inbuilt in different ways marked items, preventive open source, tools or programming. As a rule, they are standing on groups of servers to Open Source programming attached with in-house applications with framework programming. The actual difficulty that is anticipating cloud related to a standard phase is security and displaying cost [10]. Cloud computing has one of a type pay-as-you-go gain express; throughout which relationships pay for operation [11].

3.1 *Virtual Machine Migration*

Since cloud computing is an expressed structure, while the overall load is enlarged in an exact server farm, VM movement evades implementation corruption of the structure. The price of distribution and testing a sound cloud requires a bundle of exertion including apparatus assets and in the lengthy execution cost for its moved frameworks arrangement [10]. Be that as it may, this disadvantage can be destroyed with the support of Cloud Simulator. It displays the evaluation of the cloud computing. At this point, the structure, phase, and the resources are come into view [12].

3.2 *Cloud Service Models*

The model is throughout of some essential characteristics, three administration models, and some sending models.

A. SaaS

It is a service where an application is assisted by an administration offered to users. This type of service mitigates a load of encoding maintenance yet consumers give up manage over encoding forms and fundamentals. This evades disbursements of the concern, maintenance, patches.

B. PaaS

It provides a computing phase and a reply stack as supervision. The purchaser builds the product utilizing instruments with libraries from the dealer. The client likewise manages encoding business and agreement settings. The dealer provides the systems, servers, and diverse administrations. It presents a phase for constructing applications.

C. IaaS

It dealers present virtual machines, and document-based capacity, firewalls, virtual neighborhood, and encoding packs, etc. Collections of hypervisors can balance profits all over as signified by user' differing requirements. All structures are rearranged on demand. This supplier delivers basic system. They withdraw from purchasing machines, and evaluating benefit require [13].

4 Scenario

We outline the pictures for virtual machine load balancing algorithms in the various cloud. Under various pictures, the algorithms possibly will dissimilar restrictions.

Public Cloud: This cloud demotes to while a Cloud is accessible in a pay-per style [14]. Some main profit to facility contributors is suggested by this cloud, including no primary investment in communications and shifting of threats to communications suppliers. Though, this clouds lack fine-grained manage over information, network, and security settings, which gets in the way their efficiency in a lot of business developments [15]. Due to the lack of consistency, a variety and regularly altering APIs compose it hard to imprison all the virtual machines and hosts applications in this picture. Additionally, inconsistent loads is another face for virtual machine load balancing algorithms. So, a number of researchers have approved historic information to forecast the prospect load to conquer this challenge [16, 17].

Private Cloud: This cloud word denotes to inside information centers of a business not made obtainable to the common public. Though a public Cloud has the advantage of reduced assets to deal and improved operation speed, private Clouds are even further admired among activities along with a survey by IDG in [18]. The revision exposed that companies have a tendency to minimize active infrastructure with the performance of a private cloud which effects in an entire minor cost of possession. In some research, the entity clouds with small volume are executed to estimate virtual machine load balancing presentation. Within a cloud, the intra-information center system frequently has moderately unique properties compared to the inter-information center system [19]. So, dealing with the virtual machine load balancing trouble in a private cloud, the presentation like throughput would be regarded as a limitation.

Hybrid clouds: This cloud is a mixture of other two cloud models that make an effort to attend to the drawbacks of all move towards. In this cloud, a division of the facility communications runs in this clouds when the left behind portion runs in

public clouds. These hybrid clouds deliver additional elasticity than other clouds. Exclusively, they supply control and security over information compared to public clouds, while still assisting on-demand facility. The drawback, planning a hybrid cloud demands suspiciously determining the paramount dividing among other both cloud components [20]. Under this circumstance, the messaging cost will be the main limitation for virtual machine load balancing algorithms. Such as, in a distributed cloud, asks for may have the restriction that these demands are needed to be assigned to an official information center. As well, in a multi-cloud involving more than one clouds [21], the immigrations operations may be associated to load immigration from a cloud to another cloud (private to public) [22].

5 Load Balancing Definition

Load balancing is the first tenure utilizes to give out substantial progression loads to slighter progression systems to improve the tremendous work of the system [23]. In a system, this is a procedure of distributing total loads among different systems of a dispersed system to progress operation and performance time. A model load balancing algorithm should keep away from any exact system [24].

In cloud background, the choice of algorithm is not effortless; this is because it occupies additional constraints like security, reliability, throughput, etc. So, the most important objective of an algorithm in this environment is to develop the response time by sharing a whole load of a system. The algorithm should additionally make sure it is not overloading any particular system [25].

Load balancers efforts in two different ways: one is cooperative, and another is noncooperative. In a collaborative approach, the systems work at the same time in order to attain the general objective to minimize the total response time. And another mode, the tasks execute separately to develop the response time of limited functions [26].

Algorithms of load balancing are categorized as static algorithm and dynamic algorithm. The static algorithms are typically appropriate for uniform and steady surroundings and create great consequences in this atmosphere. Though, they are typically not elastic and could not equivalent the dynamic changes to the quality at some point in the implementation time. Dynamic algorithms are further bendable and get into concern dissimilar categories of attributes in the system together prior to and during execution time [22].

These algorithms can get used to modify and give superior results in various and dynamic settings. On the other hand, the distribution attributes become more composite and dynamic. Consequently, some algorithms could become useless and cause more overhead than compulsory resulting in substantial degradation of the presentation of the service [27].

A. The Basic Idea of Algorithms:

Load balancing makes a decision the task that how to decide the subsequent system and to transport a new demand to relocate the load from the process to under full process. It extends the incoming requesting load among the available systems to develop the performance considerably by the Cloud manager. There are two different kinds of an algorithm based on their functioning process; they are static and dynamic.

- B. **Static LB Algorithm:** It does not depend on the current circumstances. It makes a decision in the host; the demand would be implemented before setting up the demand.
- C. **Dynamic LB Algorithm:** The load balancer analyzes the present condition of load approaches at all obtainable host and performs the demand at the suitable host.

RR, WRR (Weighted Round Robin) algorithm and so on are an illustration of load balancing algorithm, but among them, RR is the most simple algorithm, utilized in the case when all the machines in the cluster have the similar processing facility. First-Come-First-Served, Throttled, Honey Foraging Algorithm and so on are the example of dynamic scheduling algorithm are superior to a static algorithm and appropriate for a lot of demands, which can carry the various workload, which would be unable to predict.

6 Related Works

Cloud computing offers a range of services to the user example resource sharing, online tools, and online storage [28]. In surroundings, each system performs a task or a subtask [29]. The Minimum Completion Time algorithm allocates tasks to the systems having the expected lowest amount of execution time of this job over other systems [25]. The Min-Min assumes the similar work as the MCT [30] algorithm to allocate a task and the scheme to finish this task with least amount execution time over other systems [31]. The LBMM [32] accepts Min-Min and load balancing strategy. It evades the unnecessarily copied assignment. Load Balancing with Job Switching [33] minimize loads of heavy loaded machine to under loaded machine by switching a particular job. Load balancing of Unbalanced Matrix with Hungarian Method [34] uses the Hungarian algorithm where jobs are greater than all nodes. Load balancing of the unbalanced cost matrix [35] is the same as the previous algorithm.

6.1 Criteria for Performance

Our proposed load balancing algorithm is considered to get together all scheduling criterion example greatest CPU operation, lowest turnaround time, utmost through-

put, smallest waiting time, and context switches. We discuss some definitions as follows: [36].

Arrival Time:	Entrance time is the time of entrance of the process in the memory.
Response Time:	This means the time of arrival minus initial reaction time by the processor.
Burst Time:	This time is how long the process holds the processor.
Turnaround Time:	This means the time of arrival minus the time of execution of the task.
Waiting Time:	This is that how long of a time, the process waiting in the ready queue.
Throughput:	This is a number of the process executed per unit period

7 Proposed Work

Whenever the matrix of an assignment problem is not a square matrix, that is, while the figure of sources system is not identical to the figure of destinations system, the assignment problem is called an unbalanced assignment problem. This kind of problems, replica rows (or columns) are added to the matrix to complete it to shape a square matrix. The replica rows or columns will contain all elements as zeroes. The Hungarian method [37] could be used to solve this type of crisis.

8 A. Algorithm

To provide an algorithmic demonstration of the technique, consider an example which consists of ‘i’ systems $i = \{N_1, N_2, \dots, N_i\}$. And ‘j’ jobs $j = \{J_1, J_2, \dots, J_j\}$ is considered to be allocated for performance on ‘m’ available computing systems and the performance value C_{mn} , where $m = 1, 2, \dots, i$ and $n = 1, 2, \dots, i$, where $j > i$, i.e., the numeral of jobs is greater than a numeral of computing systems.

The development of the proposed algorithm is presented as follows

Step 1: At first, we evaluate the average run time of each machine for all jobs, in that order.

Step 2: Next, it is to discover the job having the highest average run time.

Step 3: Remove those machine one by one with the highest average run time until a quantity of machines is equal to the quantity of jobs.

Step 4: Now, it is to discover the unallocated job having the least runtime minimum than the highest average run time for the job chosen in Step 3. Then, this job is removed to the chosen job for execution.

Step 5: If there is no unallocated job can be chosen in Step 2, all machines that include unallocated and allocated machines should be reestimated. The least run time of an allocated machine is the summation of the least runtime for an allocated job on this machine and the least run time of the present job. The least run time of an unallocated machine is the present least runtime for the job. It is to locate the unallocated node or allocated machine that has the least runtime minimum than the highest average run time for the job chosen in Step 2. After that, this job is removed to the chosen machine for execution.

Step 6: Do again Step 4 to Step 5, until everyone jobs have been executed thoroughly.

Example: Suppose an IT company has six machines that are used for four jobs. All jobs can be allocated to one system at the same time.

The cost of every job on every machine is given in the Table 1.

Assignment Problem

See Table 1.

Final Result

See Table 2.

Table 1 The execution time for each task in various systems

Jobs/Machines	M ₁₁	M ₁₂	M ₁₃	M ₁₄	M ₁₅	M ₁₆
J ₁₁	18	19	18	15	36	18
J ₁₂	22	18	17	17	42	23
J ₁₃	13	20	12	14	42	22
J ₁₄	22	24	19	18	38	22
J ₁₅	14	17	10	19	24	21

Table 2 Assigned each task at each computing systems respectively

Jobs/Machines	M ₁₁	M ₁₂	M ₁₃	M ₁₄	M ₁₆
J ₁₁	18	19	18	15	I8
J ₁₂	22	18	19	I7	23
J ₁₃	I3	20	12	14	22
J ₁₄	22	24	I9	18	23
J ₁₅	14	I7	10	19	21

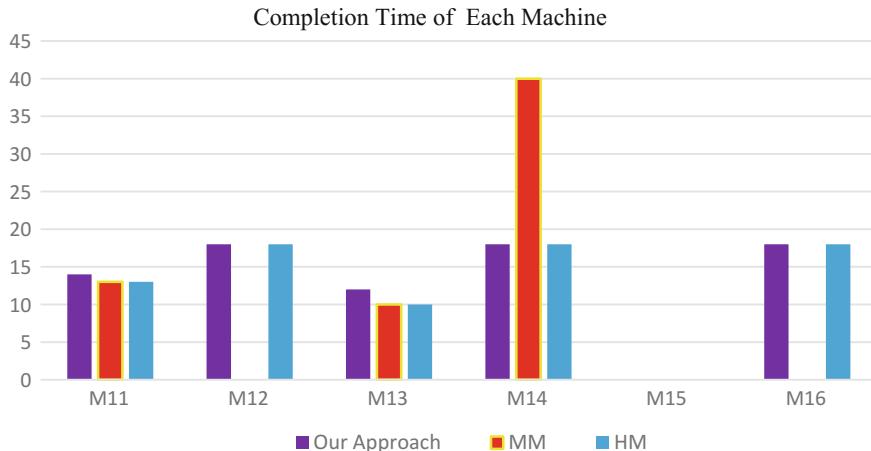


Fig. 1 The judgment of run time (in ms) of all tasks at systems

9 Case Study

Table 1 demonstrates the execution time for all tasks at systems. The threshold is the average execution time of the task t_i in all executing systems. For calculating the performance of our proposed algorithm, our approach is compared with MM and HM shown in Fig. 1. Figure 1 displays the judgment execution time of all executing system among our approach, MM and HM. The execution times for executing all tasks by using our approach, HM and MM are 18, 18, and 42 ms, correspondingly. Our proposed work gets the smallest execution time and greater load balancing than another algorithm, example MM and HM in this environment.

10 Comparison

Our proposed load balancing of unbalanced cost matrix can get improved balancing and execution than other algorithms, example LBMM and MM from the Fig. 1.

11 Conclusion

In this present article, we have proposed a proficient work for the computing network to allocate tasks to computer systems in accordance with their resource potential correspondingly. Our proposed work can complete improved balancing and presentation than other algorithms, example HM and MM from the demonstration.

References

1. B. Hayes, Cloud computing. Commun. ACM **51**(7), 9–11 (2008)
2. S. Marston, Z. Li, S. Bandyopadhyay, J. Zhang, A. Ghalsasi, Cloud computing—the business perspective. Decis. Support Syst. **51**(1), 176–189 (2011)
3. B.P. Rimal, A. Jukan, D. Katsaros, Y. Goeleven, Architectural requirements for cloud computing systems: an enterprise cloud approach. J. Grid Comput. **9**(1), 3–26 (2011)
4. Y. Fang, F. Wang, J. Ge, A task scheduling algorithm based on load balancing in cloud computing, in *International Conference on Web Information Systems and Mining* (Springer, Berlin, Heidelberg, 2010), pp. 271–277
5. T. Wang, Z. Liu, Y. Chen, Y. Xu, X. Dai, Load balancing task scheduling based on genetic algorithm in cloud computing, in *2014 IEEE 12th International Conference on Dependable, Autonomic and Secure Computing (DASC)* (IEEE, 2014), pp. 146–152
6. A.Y. Zomaya, T. Yee-Hwei, Observations on using genetic algorithms for dynamic load-balancing. IEEE Trans. Parallel Distrib. Syst. **12**(9), 899–911 (2001)
7. C. Zhao, S. Zhang, Q. Liu, J. Xie, J. Hu, Independent tasks scheduling based on genetic algorithm in cloud computing, in *5th International Conference on Wireless Communications, Networking, and Mobile Computing* (2009), pp. 1–4
8. E. Juhnke, T. Dörnemann, D. Böck, B. Freiselen, Multi-objective scheduling of BPEL workflows in geographically distributed clouds, in *4th IEEE International Conference on Cloud Computing* (2011), pp. 412–419
9. F. Ramezani, J. Lu, F.K. Hussain, Task-based system load balancing in cloud computing using particle swarm optimization. Int. J. Parallel Program. **42**(5), 739–754 (2014)
10. Y. Wang, J. Li, H.H. Wang, Cluster and cloud computing framework for scientific metrology in flow control. Cluster Comput. 1–10 (2017)
11. A. Shawish, M. Salama, Cloud computing: paradigms and technologies, in *Inter-cooperative Collective Intelligence: Techniques and Applications* (Springer, Berlin, Heidelberg, 2014), pp. 39–67
12. A. Shawish, M. Salama, Cloud computing: paradigms and technologies, in *Inter-cooperative Collective Intelligence: Techniques and Applications* (Springer, Berlin, Heidelberg, 2014), pp. 39–67
13. Y. Wang, J. Li, H.H. Wang, Cluster and cloud computing framework for scientific metrology in flow control. Cluster Comput. 1–10 (2017)
14. M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R.H. Katz, A. Konwinski, G. Lee, D.A. Patterson, A. Rabkin, I. Stoica, et al., Above the clouds: a Berkeley view of cloud computing (2009)
15. L. Zhao, S. Sakr, A. Liu, A. Bouguettaya, *Cloud Data Management* (Springer, 2014)
16. J. Hu, J. Gu, G. Sun, T. Zhao, A scheduling strategy on load balancing of virtual machine resources in cloud computing environment, in *2010 3rd International Symposium on Parallel Architectures, Algorithms, and Programming* (IEEE, 2010), pp. 89–96
17. W.T. Wen, C.D. Wang, D.S. Wu, Y.Y. Xie, An aco-based scheduling strategy on load balancing in cloud computing environment, in *2015 Ninth International Conference on Frontier of Computer Science and Technology* (IEEE, 2015), pp. 364–369
18. G. Roos, Enterprise prefer private cloud: Survey 2013, <http://www.eweek.com/cloud/enterprises-prefer-private-clouds-survey/>
19. A. Li, X. Yang, S. Kandula, M. Zhang, Cloudcmp: comparing public cloud providers, in *Proceedings of the 10th ACM SIGCOMM Conference on Internet Measurement* (ACM, 2010), pp. 1–14
20. Q. Zhang, L. Cheng, R. Boutaba, Cloud computing: state-of-the-art and research challenges. J. Internet Serv. Appl. **1**(1), 7–18 (2010)
21. D. Petcu, Multi-cloud: expectations and current approaches, in *Proceedings of the 2013 international workshop on Multi-cloud applications and federated clouds* (ACM, 2013), pp. 1–6
22. M. Xu, W. Tian, R. Buyya, A survey on load balancing algorithms for virtual machines placement in cloud computing. Concurr. Comput. Pract. Exp. **29**(12) (2017)

23. S. Song, T. Lv, X. Chen, Load balancing for future internet: an approach based on game theory. *J. Appl. Math.* (2014)
24. A.A. Neghabi, N.J. Navimipour, M. Hosseinzadeh, A. Rezaee, Load balancing mechanisms in the software defined networks: a systematic and comprehensive review of the literature. *IEEE Access* **6**, 14159–14178 (2018)
25. H. Mehta, P. Kanungo, M. Chandwani, Decentralized content aware load balancing algorithm for distributed computing environments, in *Proceedings of the International Conference & Workshop on Emerging Trends in Technology* (ACM, 2011), pp. 370–375
26. A. B. Singh, S. Bhat, R. Raju, R. D’Souza, Survey on various load balancing techniques in cloud computing. *Adv. Comput.* **7**(2), 28–34 (2017)
27. K. Al Nuaimi, N. Mohamed, M. Al Nuaimi, J. Al-Jaroodi, A survey of load balancing in cloud computing: Challenges and algorithms, in *2012 Second Symposium on Network Cloud Computing and Applications (NCCA)* (IEEE, 2012), pp. 137–142
28. Y.-T. Wang, Load sharing in distributed systems. *IEEE Trans. Comput.* **100**(3), 204–217 (1985)
29. M. Antoine, L. Pellegrino, F. Huet, F. Baude, A generic API for load balancing in distributed systems for big data management. *Concurr. Comput. Pract. Exp.* **28**(8), 2440–2456 (2016)
30. D. Grosu, A.T. Chronopoulos, M.-Y. Leung, Load balancing in distributed systems: an approach using cooperative games, in *Parallel and Distributed Processing Symposium., Proceedings International, IPDPS 2002, Abstracts and CD-ROM* (IEEE, 2001), pp. 10
31. Robert Fox, Library in the clouds. *OCLC Syst. Serv. Int. Digital Library Persp.* **25**(3), 156–161 (2009)
32. S.-C. Wang, K.-Q. Yan, W.-P. Liao, S.-S. Wang, Towards a load balancing in a three-level cloud computing network, in *2010 3rd IEEE International Conference on Computer Science and information technology (ICCSIT)*, vol. 1 (IEEE, 2010), pp. 108–113
33. G. Ritchie, J. Levine, A fast, effective local search for scheduling independent jobs in heterogeneous computing environments. *J. Comput. Appl.* **25**, 1190–1192 (2005)
34. T.D. Braun, H.J. Siegel, N. Beck, L.L. Bölöni, M. Maheswaran, A.I. Reuther, J.P. Robertson, M.D. Theys, B. Yao, D. Hensgen, R.F. Freund, A comparison of eleven static heuristics for mapping a class of independent tasks onto heterogeneous distributed computing systems. *J. Parallel Distrib. Comput.* **61**, 810–837 (2001)
35. S.C. Wang, K.Q. Yan, W.P. Liao, S.S. Wang, Towards a load balancing in a three-level cloud computing network, in *CSIT* (2010), pp. 108–113
36. C.-L. Hung, H.-H. Wang, Y.-C. Hu, Efficient load balancing algorithm for cloud computing network, in *International Conference on Information Science and Technology (IST 2012)*, April 2012, pp. 28–30
37. H.W. Kuhn, The Hungarian method for the assignment problem. *Naval Res. Logist. (NRL)* **2**(1–2), 83–97 (1955)

Cohort Assembly: A Load Balancing Grouping Approach for Traditional Wi-Fi Infrastructure Using Edge Cloud



Deepsubhra Guha Roy, Madhurima Das and Debashis De

1 Introduction

At the present time, approximately 4.77 billion of users have smartphone and it supposed to have 82.7% of data traffic [1]. From medical care to e-learning, e-commerce and daily life-related mobile applications have an impact in our regular life. Mobile phones are now known as smartphones as they are multi-tusker. Also, it has multiuser functionalities, multiplayer gaming, etc. To make it successful, the internal resources are integrated with cloud and sensing with the private/public cloud.

Generally, we have a concept regarding mobile cloud computing that data is moved, processed, and stored in cloud datacenters from the smart devices. But here, the users become helpful when the data, task, and computation offloading reduce battery consumption in the smartphones by pretending to access unlimited computational and storage resources. But distance is a factor which detached a device from cloud at the edge of the network. In crowded environment like sports or any type of event, natural calamity scenarios where bandwidth is limited or Wi-Fi infrastructure is unavailable, mobile cloud computing cannot be directly put into these types of cases. To fix out these problems, different solutions like mobile edge clouds [2] are applicable where the conventional infrastructure is combined with the resources presented in the smart devices permitting proximity-aware applications. Nowadays, smart devices are not only a “client” but also a “server” by performing several com-

D. G. Roy (✉) · M. Das · D. De

Department of Computer Science and Engineering, West Bengal University of Technology,
Presently, Maulana Abul Kalam Azad University of Technology, WB B.F.-142, Sector-I, Salt
Lake City, Kolkata 700064, West Bengal, India
e-mail: roysubhraguha@gmail.com

M. Das
e-mail: madhurimadas64@gmail.com

D. De
e-mail: dr.debashis.de@gmail.com

Table 1 Performance analysis I

Time	Edge group	Connected device	Edge users	4G user
18:30	0.08	2.5	0.5	0.01
18:40	2.18	6.2	4.8	0.01
18:50	9.05	20.3	15.1	0.01
19:00	8.12	28.9	13.2	1.1
19:10	10.09	28.5	10.1	2.08
19:20	10.51	31.1	17.3	3.85
19:30	10.01	30.2	22.3	4.48
19:40	7.85	25	20.2	4.48
19:50	10.03	26.9	17.3	4.48
20:00	9.36	25.1	16.6	2.08
20:10	10.03	30	24.4	2.07
20:20	10	32.1	25.7	2.07
20:30	9.89	32.5	20.9	2.12
20:40	6.97	12.1	8.8	0.99
20:50	4.5	5	5	0.99

Table 2 Performance analysis II

Time	Scope	Group owner	Member of group
18:50	0.742	0.59	0.401
19:00	0.409	0.392	0.591
19:10	0.613	0.439	0.409
19:20	0.702	0.581	0.421
19:30	0.692	0.489	0.493
19:40	0.618	0.499	0.501
19:50	0.643	0.529	0.422
20:00	0.621	0.541	0.429
20:10	0.779	0.398	0.589
20:20	0.81	0.387	0.61
20:30	0.652	0.439	0.579

putational tasks. The devices which are adjacent to one another are accumulated to compose an edge cloud and work jointly to configure a cache of computing resources with limited operations accessed to crowdsourced information. There is no need of offloading of computational tasks in traditional cloud infrastructure. Applications regarding image and video processing, data mining, video streaming, applying reliability, and validation techniques are used by edge cloud [3]. Still there are various challenges related to edge cloud infrastructure:

Table 3 Performance analysis III

Time	Edge downloads	Server downloads (AP)	Server downloads (4G)
18:30	0	0.99	0
18:40	0	0	0
18:50	0.05	3.25	0
19:00	0	0	0.02
19:10	1.71	3.39	1.69
19:20	0.01	2.22	1.01
19:30	0.01	0	0.02
19:40	0.01	0	0
19:50	2.09	0.03	0
20:00	18.9	4.19	1.85
20:10	1.95	0	1.91
20:20	0	0	0
20:30	0	0.1	0.01
20:40	0	0.78	0.04
20:50	0	0	0

- (i) Maintaining miscellaneous smart devices.
- (ii) Smart devices entering, moving, and leaving edge cloud in a dynamic way. But we have to copy the total movement of these devices.
- (iii) Resolve when to control one wireless technology over another like peer-to-peer or client-server, etc.

Currently, in real-life existence, circulation of video contents, as for example replaying of any match or event is to associate the audiences in a cricket or football match by offloading in the edge cloud calls a chunk of the Wi-Fi congestion. There are so many apps in the market by which the users are aware of the recent statistics of data and multimedia contents that replays for an “out”, “goals”, or interesting events. If a user wants to repeat a video (replay), the intermediate main servers help in downloading the content through the Wi-Fi or cellular access points installed in the stadium or any open space and played it in the device. If the place is crowded, all the requests emphasized the infrastructure. To resolve this, mobile edge clouds can be used where the user’s smart device and his neighbors can generate a native pool of the quantity of the servers and share the services among the users in several ways. As for example, if there is already any request for the same video before by any other mobile user for the same video replay, then for the second or further demands the application will ask those smart devices which are already accessed the video. All the mobile smart devices should be placed within the edge cloud network and if a duplication of replay exists, the user can fetch it directly from the adjacent user’s phone. Scenarios related to privacy and security issues focused on sandbox infrastructure (security model for Java Virtual Machine) for privacy purpose

and for security purpose a group of people envision the videos submitted and digital signature is applied before giving permission to allow in cascading the video through the application. In this paper, there are four major contributions.

- (i) A prototype model of application for Android devices is built and implemented that dynamically switches between the central server named as Wi-Fi infrastructure and an edge cloud (containing cached copies) to circulate video contents.
- (ii) We performed a real-life experiment in a T-20 cricket match and used the application. The analysis from the records suggests that by reducing load in the access point using edge cloud at crowded venue, quality of experience is improved.
- (iii) A new grouping approach is proposed to form the connection establishment to the edge cloud.
- (iv) If more than one match is running, then the user will get an option of having a glimpse to any of the matches or events replay and according to that, grouping of similar viewers will be obtained.

2 Related Work

A content delivery network (CDN) is an arrangement of distributed servers that deliver pages and other Web content based on the geographic locations of the user merged by enhancing the dissemination of web content to the users by increasing network bandwidth, reducing latency, and supplying high data availability. The fundamental research on CDNs emphasizes on traditional networks. But here, these infrastructures are going to be supplied for mobile edge clouds purpose. If there is some gain in performance, there will be a decision-making regarding the application to this situation.

Edge cloud in the context of mobility is a collection of smart devices which helps in collaborating the computational resources and storing the data in a locally formed network like ad hoc, Bluetooth, or Wi-Fi. Though in the edge cloud individual smart device would not be with unlimited resource, the collection of a pool of smart devices is significant for computational resources.

At the beginning, computational purposes like Honeybee [4], mobile message passing interface [5], and Hyrax [6] were used for mobile devices. Mobile message passing interface (MMPI) was built based on Bluetooth to provide the communication infrastructure like interconnection between a wireless headset or a fully interconnected mesh network and a mobile phone to accelerate a two-way communication between each node with the network. With message passing interface, tasks can connect to each other in a peer-to-peer trend or communicational method calls. But this MMPI was applied in Java machine and does not allow turbulence. Hyrax is a platform obtained from Hadoop which helps in supporting *cloud computing* on Android smart devices which were used to schedule computational tasks using smart heterogeneous devices or assorted system of smart devices and servers on Wi-Fi net-

works. This system was able to fulfill churn handling using Hadoop fault tolerance algorithm. Another system named as Honeybee is able to use both Bluetooth and Wi-Fi to form a network locally and supplies distributed computing. This skeleton works on minor networks and applies techniques to handle the churns partially by checking the availability of other nodes, rescheduling failed jobs, etc. mCloud is also a similar approach like Honeybee, but unfortunately, no implementation was applied for this system [7].

The concept of FemtoClouds is very much same as an edge cloud. In this system, a group of users created a network using smart devices and was efficient at accepting tasks and then scheduling them with the help of the network. For this purpose, a special device named “control device” was proposed which was responsible for guiding the smart devices [8, 9].

For storing data and to perform traditional computations of resource processing and storage of devices, the aforesaid edge cloud applications are used. Our focus will be on caching and cascading of data contents like videos using the edge cloud instead of offloading the total video from the main cloud datacenter and removing weight in the access points and the servers to enhance user. Several groups will be formed when two or events are going to be replayed at the same time.

3 Architecture of Edge Cloud System

We have considered a scenario in this experiment that replays are generated from a distance without any physical contact and stored in a cloud or centralized server. Also, “Wi-Fi Direct” is used as adjacent devices for peer-to-peer communication. It is an authorized application of scientific knowledge. This edge cloud consists of several Wi-Fi direct groups, and each group has an owner named as Group Owner (GO) who represents itself as an Access Point (AP) for other subscribed devices in that group. One device at a time cannot be able to establish connection with two Wi-Fi APs at a time. For that issue, none other than GO is unable to establish a connection with the basic physical and organizational structure AP and to the application program server by installing a VPN or a proxy in the particular GO. The important point to be noted is that the devices are able to continue Internet using with the discussed configuration (Fig. 1).

3.1 Metadata

User can only determine whether or not the replay download is interesting for him or not. Distribution of metadata is one of the most important elements of edge cloud architecture. A replay of a video is considered of video content and metadata consisting of title, duration, and thumbnail. By examining the metadata, downloading the replay would be valuable and the user can take the decision. The structure must

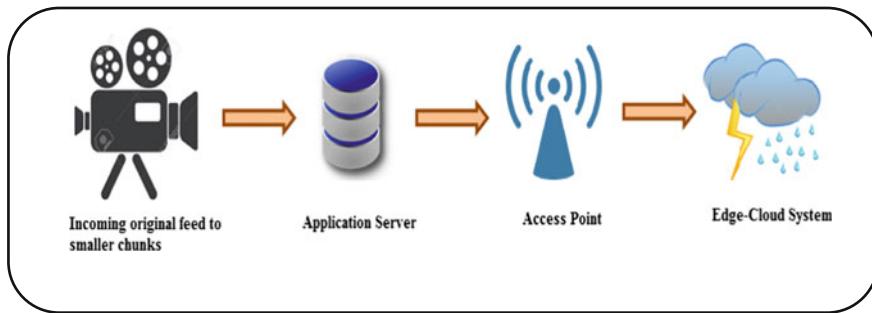


Fig. 1 Experimental flow of data through various service layers

frame the process by which a device can acquire metadata of video by giving a group of data telecommunication links.

Before being a part of edge cloud, a device must be awarded with the details about the subsistence and accessibility of an adjacent edge cloud and resolve the role in the system.

3.2 Wi-Fi Direct Groups

Groups are created by the smart devices. Those devices become group owners and are accountable when the metadata of the group is registered on the application server. The information related to existing devices of the same group is comprised of the Group Owner (GO) to find the Wi-Fi network of the existing group and connect with the network, and they communicate with the GO. There exist a metadata version number which is used to avoid clashing views by handling it in a straightforward fashion. Moreover, all these information are collected by the IPs of the group owner, the SSID of the network, password of ssid, group size, number of members of the group, and the user listening to the TCP port. This number is issued and modified by the GO only, and also increased when a member wants to join or leave the group an update to metadata is required. GO is eventually responsible for tracing the members of its group and the connection-related information. Correlatively when a smart device joins to a group, it instantly sends “HI” message to the GO and also information related with connection and record of its self-video replays.

3.3 Centralized Server

Retrieve video metadata from time to time. Metadata of groups and videos resides on the server containing the replays also. If the device is GO, using an HTTP request, it fetches metadata from the server else, and using TCP socket it fetches it from the GO.

3.4 Replay Requests

While transferring the replays, in-group transfers get preference over available infrastructure access points. When GO is the requesting smart device, then local replay owner will be searched else GO will be asked for content holders by the devices (Figs. 2 and 3).

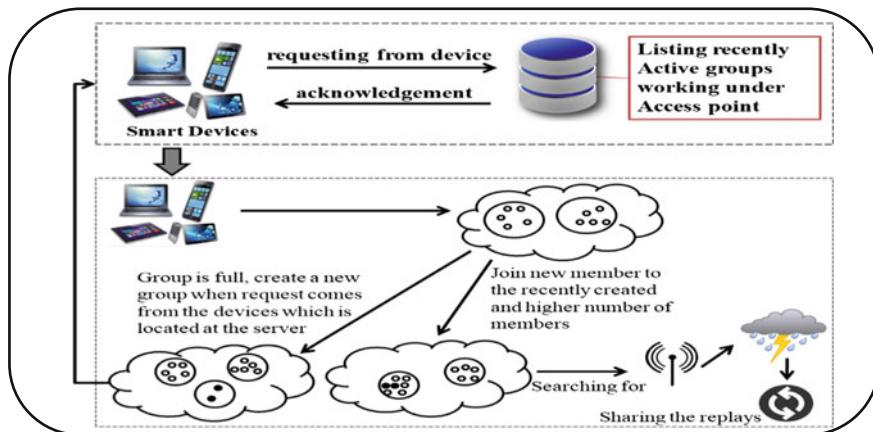


Fig. 2 Process of getting connection

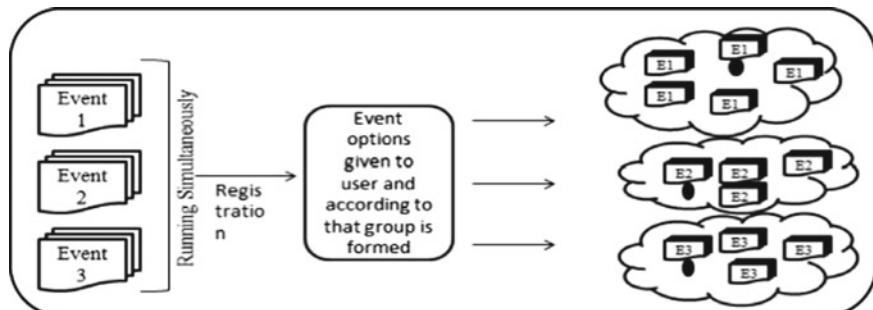


Fig. 3 Sorting the events into similar groups according to the type of event

4 Methodology

Each smart device has an activity on the edge-cloud. As the smart devices from the server have to acquire the metadata of edge cloud, connection with the server is needed to be established first. When the connection is established with the server, request comes from the device which is located at the server to figure out the working groups which are performing under the access point to which it was connected recently. If at least one group is available and not reached to its extreme range, then the smart device would join the group. Else a new group is created. Each group must have one group owner.

We have four smart devices connected to a group, and each device can try to attempt to make connection only once for the given purposes: First, we have trialed with different smart devices performing as GO and the devices were comfortable to work with 5–6 connections at a time. These limitations are fixed by the network drivers and hardware in the Android devices. For that reason, 4 is a safe position to decrease the connection-related errors. Second, we also restricted the size of the group to avoid straining the GO during the event which would be an outcome of congestion and deconstruction of the group. Third, each device can try to connect with a group only once because it takes significant amount of time to connect with a Wi-Fi AP. In our proposed Algorithm I, we have described that each and every device has to work, to join a group. As soon as the program begins, the algorithm is executed to have a decision of taking responsibility of activity for each device in the edge cloud. As edge cloud is dynamic in nature, it is difficult to pre-decide the position of each of the devices on the edge; depending upon the count of retries, the algorithm waits for an exponential back out time and after that the algorithm is again started over. To keep the connection with the server, as soon as the device is on the exponential awaiting queue, it attempts to reestablish connection with the utmost connected Wi-Fi network to observe the updates on metadata and to receive the videos and replays when needed. At the time when a disconnection of groups occur, the device attempts to reestablish the connection to the lastly connected Wi-Fi and reoperate the algorithm. But if a group is created recently, the GO will verify whether it consists of fresh members after uniform interval of time. The same process will be followed for an empty group also. If there is no new member included in the group, that group will be destroyed and the algorithm will seek to reduce the number of groups again.

Algorithm I: Group Formation Algorithm

```

1. Procedure Forming_Of_Groups (attempts, gssid) /* Group forming when a de-
   vice wants to join Group */
2. Decision of playing role in the edge-cloud

Start

3. if numberOfEntries^numberOfGroupMembers then /* the number of retries
   the algorithm waits an exponential back out time*/
4.     restart()
5. else
6.     groups<-- received_groups(gssid) /*group ssid is received to the
   groups parameter */
7.     group_conv<-- receive_available_group(groups) /* groups parameter
   is passed to the available group*/
8.     if NumberOfGroups=0 || group_conv=False then
9.         restart()
10.    else
11.        Connect_to_Group(groups) /* establishing connection between a
   new device and a group*/
12.    end if
13. end if
14. if restart then
15.     timeout<-- receiving_waittime(attempts)
16.     restart the total algorithm process
17. end if

```

End procedure

Following are the components accumulating into similar group:

- Information matrix,
- Data selection or extraction,
- Pattern representation,
- Inter-domain similarity observation, and
- Grouping.

As resemblance is vital for group, resemblance measurement from the same characteristics must be drawn between two groups [10]. If we can calculate the contrast based on the distance between two groups, we can collect the data which are in between a range (Fig. 4).

$$d_p(g_i, g_j) = \left(\sum_{k=1}^d (g_{i,k} - g_{j,k})^p \right)^{1/p} = \|g_i - g_j\|_p \quad (1)$$

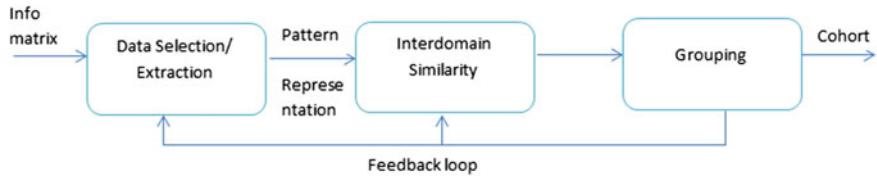


Fig. 4 Stages of cohort selection

The Euclidean distance [10] is used for calculating the nearness of groups in 2-D or 3-D space.

It can work soundly if the dataset has dense grouping. Here, g is a single group member using in grouping algorithm.

If $p = 2$, then it is a special case of Minkowski metric, i.e.,

$$d_2(g_i, g_j) = \left(\sum_{k=1}^d (g_{i,k} - g_{j,k})^2 \right)^{1/2} = \|g_i - g_j\|_2 \quad (2)$$

The problem while using Minkowski metrics directly is that it has the propensity of largest-scaled quality to rule over others.

Linear association among quality may also deform distance measurements.

$$d_M(g_i, g_j) = (g_i - g_j) \sum^{-1} (g_i, g_j)^T \quad (3)$$

In Algorithm 2, we have discussed that when two or more matches and events are simultaneously running, group will be formed according to the choice of the users. At the time of registering, options will be given to the user and according to his/her selection of the given options, group generation will be dependent on that option. Each group has a group owner and is dynamic in nature but for device issues, a maximum number of users can only connect to a group, suppose here it is 10. When a same group of users already created a group and there is no space in that group, a new group will be formed. But if in any of the group has a less number of users suppose only three members are present in one group out of 10, then the remained members will be allocated to the rest of the same types of group dynamically.

Algorithm 2: Cohort Group Formation Algorithm

```

1. Start
2. Event-wise registration
3. Formation_Of_similarGroups() /* According equation */
4.  $G_s \leftarrow$  Group Size /* total members in all the groups */
5.  $n \leftarrow$  number of users in each of group /* Each group has a number of
smart devices n*/
6.  $N_{max} \leftarrow$  maximum number of users in a group /* here  $N_{max} = 10$  */
7.  $G_c \leftarrow$  number of groups in edge-cloud /* Suppose  $G_c = 7$  */
8. If group_members =  $G_c(N_{max})$  then
9.     create_new_group()
10.    else
11.        n=n+1
12.        /*device connected to the same
group is increased by 1 */
13.    break
14. End if
15.  $p \leftarrow G_s \bmod N_{max}$  /* p is a variable where the remainder is stored */
16. If  $p \leq N_{max}/2$  then
17. Distribute p into other existing groups
18. End if
19. End

```

5 Experimental Results and Discussion

A real-life experiment was performed to check whether the proposed application we made is acceptable or not. We invited our friends to watch the cricket match openly telecasted over television. The applications developed by our members were installed in the audiences' smartphone. The application would allow them the accessing procedure to replay lists originated during the match.

5.1 Experimental Setup

The replays of video were performed by our members presented in two individual rooms. As the match was being channeled via DVB-T2 and operated on Linux setup, ffmpeg software tool (produces libraries and programs of multimedia data) reduces the original feed into smaller section of data that are accumulated for figuring short

videos of the cricket match replays. Ultimately, the application server received those videos with the help of HTTP.

In our friends' circle, the application running in the smart device continuously requesting for the list of replays if they are the part of the group. Then, they were requesting via the GO, whereas if the device itself is a GO then it will request through the server. Also, there was an access point by which an access to the Internet is provided. The minimum Android version supported was Jellybean or Android 4.1. This application also enlisted the activity of users and devices on a database as character list. Whenever connection with the server was available, the application moved the database contents to the server. The following events occurred while registering:

- a. Establishing successful connection/disconnection from/to the Wi-Fi infrastructure or to the edge cloud.
- b. Creation/deletion of a Wi-Fi-Direct group.
- c. Download finishing of video replay.

Each device was registered along with suitable and proper information with the time of occurrence and user id. This information may help us on finding back the action of every smart device belonging to the edge cloud where the download occurs.

Using the registered timestamp, we cannot fix the global timeline of events because each of the devices has its own clock, that is why a generalized clock is set which is the application server here. While starting the application, a calculation was performed which evaluated the divergence Δ in between the timestamps of the devices and server. HTTP request was forwarded from the smart device to the server. Δ was evaluated as the contrast also for the reciprocated timestamp of the server and the mean value from the sent and received synchronized packet timestamps.

6 Log Analysis

The trial was held on April 8, 2018, from 8 pm IST, during the IPL T-20 Cricket match between KKR and RCB. The record analyzes that 39 different users involved with the application and on the server 3859 individual entries were recorded. Out of the total number of recorded messages, Wi-Fi connection or disconnection number was 1231, group creation/ deletion was 1496, and download attempt was 1189 where 839 was successful and 350 was unsuccessful.

6.1 General Deliberations

Each device has a median and depending on those median of Δ s registration, device timestamps were corrected and all the occurrences were categorized by timestamp and assessed in a logical order. Suppose O is a type of given occurrence which was

assigned as smart device S, we can categorize O's state between T and T + 1. Hence, O is accessing as a connection-oriented device if $S_{(T-1)}$ is an authenticated connection to a Wi-Fi network. When O is already attached and $S_{(T-1)}$ is first compared and then selected for creating a group occurrence, that smart device is also group owner. Likely, the devices are filed as connectionless when disconnection occurred.

Collecting all these information, we had to calculate the area of the edge cloud at T as the accumulation of summation of the connection-based GOs and group members of that GO. That area was increased by one whenever a group is created or a group is connected to edge cloud and decreased when any type of disconnection like event, group, or application elimination took place.

The range of the edge cloud is represented by the correlation of the area of edge cloud and the collection of connection-based users and depth is represented as the ratio of group members and group owners. The range focuses on all adjacent connection-based devices that are dynamically joined over the edge cloud. A depth relationship can be most favorable if it indicates toward the highest number of smart devices which were allowed in each of the groups where edge cloud operates at its utmost range.

This resembles only if the number of connected and disconnected devices are equal and it cannot have different numbers of connections and disconnections.

Thus, all these properties must be followed before moving with the computations. While discovering, the missing events and placing them by applying semi-automatic approach were caused as result.

This graph portrays that the movement of edge cloud is over the period of time. In this graph, activity is represented by number of active groups which is labeled as "Edge Group" and smart devices related with the edge cloud as "Edge users". These points increased gently when the match began and when the first half of the match was near about, those values were steady. The graph decreased when there was a halftime. In the second half, also the same type of behavior was observed till the match was over. Users who were connected through Wi-Fi access point are marked as "Connected device" and some users were using 4G and connected with the server is labeled as "4G users".

There was a defect on the AP in the first half of the match and that is why major number of downloads were failed. The connections were lost on GO for those access points and that is why the groups were demolished. The edge cloud reformed itself using the Algorithm 2 has described (Fig. 5).

In Fig. 6, we are viewing the range over the period of time of the match. We presumed that all users were engaged over the edge cloud but 0.80 was near about perceived fraction because those devices cannot connect to a Wi-Fi direct group but kept the connection with the AP only. Also, the par magnitude for edge cloud was 1.20 which is equivalent to 30.00% of the highest group capacity. It also views the fraction of devices that are GO and the member of groups.

In Fig. 7, the graph provides us with the successful downloading via server and edge cloud, i.e., 4G and AP, respectively. New replays are marked with vertical lines. We also can observe that whenever a new replay is formed, a spike is formed through the server.

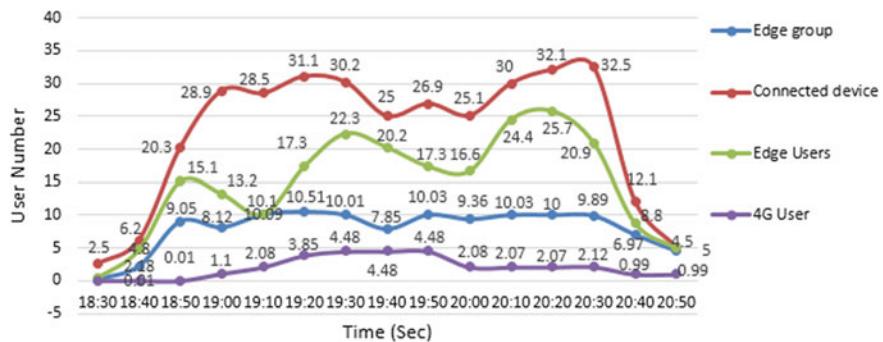


Fig. 5 Graph 1 for Table 1 related to number of users of edge cloud, 4G, and Wi-Fi connection

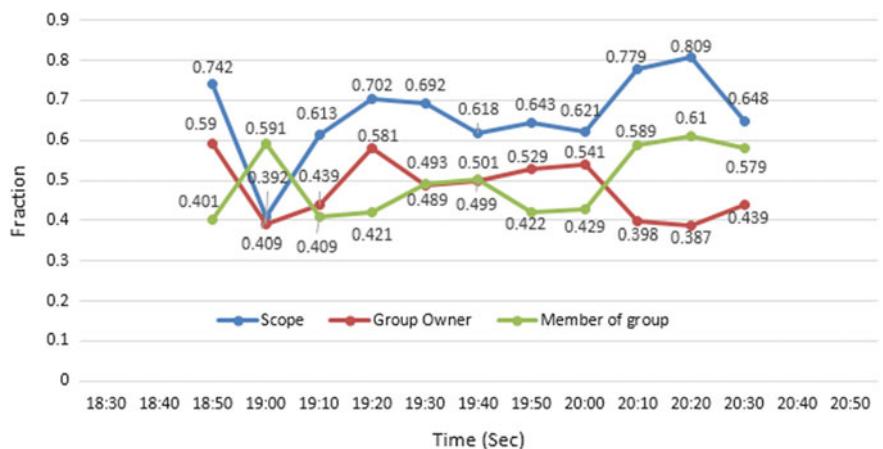


Fig. 6 Graph 2 for Table 2 related to the scope of edge cloud with the average percentage of owner

Download characteristics: While studying the activities when downloading, we focused on the reduction of traffic from access point. It happened only based on the replays the edge cloud generated in the presence and absence of the employment of access point. Downloads were categorized based on providing and requesting area:

- Edge–Edge (EE): Edge cloud requested for downloading and allocation also done with the help of the edge cloud only.
- Edge Server (ES): Edge cloud requested for downloading and allocation also done with the help of the server through access point.
- Remote Server (RS): From outside the edge cloud, request was made for downloading and allocation also done with the help of the server through access point.

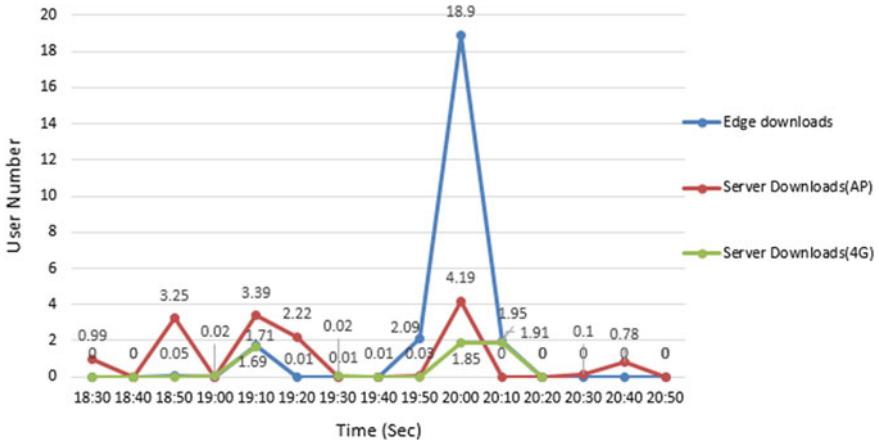


Fig. 7 Graph 3 for Table 3 related to the efficient downloads

7 Conclusion

In this twenty-first century, mobile communication is not only limited to transfer data but also a large and huge enhancement is being done on the cloud, edge cloud, and other latest technologies like VoLTE, cloudlet, and so on. As far we know, earlier there were no applications which were related to real-time scenarios about consuming data from the events and storing it in the cloud edges to replay the videos from the connected devices only by grouping them event-wise. Though in our experiment there are some limitations like limited devices connections, limited groups related to the wireless ad hoc network, we were able to apply a new cohort technique for an entertainment purpose. This experiment is a step toward the expansion of edge cloud.

References

1. <https://www.statista.com/statistics/274774/forecast-of-mobile-phone-users-worldwide/>
2. D.G. Roy et al., Application-aware cloudlet selection for computation offloading in multi-cloudlet environment. *J. Supercomput.* **73**(4), 1672–1690 (2017)
3. D.G. Roy et al., Multi-cloud scenario based QoS enhancing virtual resource broker, in *2016 3rd International Conference on Recent Advances in Information Technology (RAIT)* (IEEE, 2016)
4. https://link.springer.com/chapter/10.1007/978-3-642-40238-8_19
5. https://www.researchgate.net/publication/221593136_MMPI_a_message_passing_interface_for_the_mobile_environment
6. <https://pdfs.semanticscholar.org/8cd2/11cc816952f036ed65a7022adba063486008.pdf>
7. A. Mukherjee, D. De, D.G. Roy, A power and latency aware cloudlet selection strategy for multi-cloudlet environment. *IEEE Trans. Cloud Comput.*

8. D. De et al., Architecture of green sensor mobile cloud computing. *IET Wirel. Sens. Syst.* **6**(4), 109–120 (2016)
9. P.M.P. Silva et al., Using edge-clouds to reduce load on traditional wifi infrastructures and improve quality of experience, in *2017 IEEE 1st International Conference on Fog and Edge Computing (ICFEC)* (IEEE, 2017)
10. A.K. Jain, M.N. Murty, P.J. Flynn, Data clustering: a review. *ACM Comput. Surv.* **31**(3) (1999)

Design of Fuzzy-PI Controller for Hybrid Distributed Generation System Using Grey Wolf Optimization Algorithm



Santu Giri and Parthasarathi Bera

Nomenclature

P_S	Generation of total average power
P_W	Wind turbines mechanical power output
P_{PV}	Photovoltaic system output power
P_L	Load demand
$H(t)$	Heaviside step function
ρ	Air density
A_r	Swept area of the blades
Φ	Solar irradiation (KW/m^2)
K	Gain
T	Time constant
M	Inertia constant
D	Damping constant

1 Introduction

The demand for electrical energy is increasing continuously due to highly increase industry number and population growth. The distributed way of power generation has become more popular than the conventional way of power generation because of the gradual depletion of fossils fuel, the deregulation of electrical energy market,

S. Giri · P. Bera (✉)

Department of Electrical Engineering, Kalyani Government
Engineering College, Kalyani 741235, India
e-mail: parthabera1977@gmail.com

S. Giri
e-mail: santugiri1994@gmail.com

high cost of transmission and distribution equipment material. The distributed way of power generation is eco-friendly, has reliability, and is cost-effective over the conventional way. The distribution generation system with a combination of renewable resources like solar and wind along with fuel cell (FC) is used to produce electrical energy [1]. The renewable resources like wind and solar energy power generation are growing fast because of its easy availability. But the wind and solar energy generation depending on an intensity of wind speed and sunlight and the output of electrical energy from wind turbine or photovoltaic highly fluctuate in nature for an isolated island. To minimize the output energy fluctuation and control supply and demand power balance, integration of aqua electrolyser (AE), fuel cell (FC), flywheel energy storage system (FESS), battery energy storage system (BESS) and diesel engine generator (DEG) with wind turbine generator (WTG) and photovoltaic (PV).

Fuel cells (FC) has been used in the hybrid power system to generate electrical energy converted from hydrogen chemical energy. The fuel hydrogen chemical energy is produced by aqua electrolyser (AE) and supply to the FC [2]. Energy storage devices have been employ in hybrid system to maintain power balance with the variation of renewable resources. A controllable BESS can be used to improve the overall dynamic responses of load frequency control (LFC) of a grid-connected power system [3, 4]. FESS is used to store energy with the advantage of high efficiency, longer durability and high power exchange [5]. In the hybrid distributed system with wind-diesel to performed load frequency control and blade pitch control mechanism of the wind turbine [6, 7]. A controller has been designed and communication has been establishing between controllable and uncontrollable sources for improving frequency regulation in an isolated microgrid [8]. In a hybrid distributed generation system, the frequency fluctuation occurs when the power system is connected to large number. To diminish this problem a LFC based on disturbance observer has been designed in [9]. The damping oscillation of system frequency may occur when the system is not controlled properly. A new controller has been designed using the alternative energy facilities for the hybrid distributed generation system to improve frequency regulation in [10]. Small signal stability analysis has been performed for an autonomous hybrid system comprising of aqua electrolyser (AE), wind turbine generator (WTG), photovoltaic cell (PV), flywheel energy storage system (FESS), fuel cell (FC), battery energy storage system (BESS), and diesel engine generator (DEG) in [11]. The proportional-integral (PI) controller [12] and fractional order PI controller [13] have been designed for hybrid distributed generation system to control the system frequency regulation and maintain power balance.

Fuzzy logic based PI controllers have been applied in the different field of power system such as automatic generation control [14–17] to improve the dynamic responses. Further, the grey wolf optimizer (GWO) [18] has been successfully applied for optimizing the control parameters for LFC problems [19–22]. But GWO has not been applied in the hybrid distributed generation system and needs further investigation.

In view of the above, in the present work, fuzzy-PI controllers have been applied for controlling distributed energy resources and the grid frequency is taken as input

to the controller with time delay. Further, the scaling factors of fuzzy PI controller have been optimized based on GWOA.

2 Modelling of Hybrid Distributed Generation System

The proposed model [13] of the hybrid system consists with WTG, PV, FC, AE, BESS, FESS and DEG are shown in Fig. 1 and the parameters of the various components are shown in Appendix.

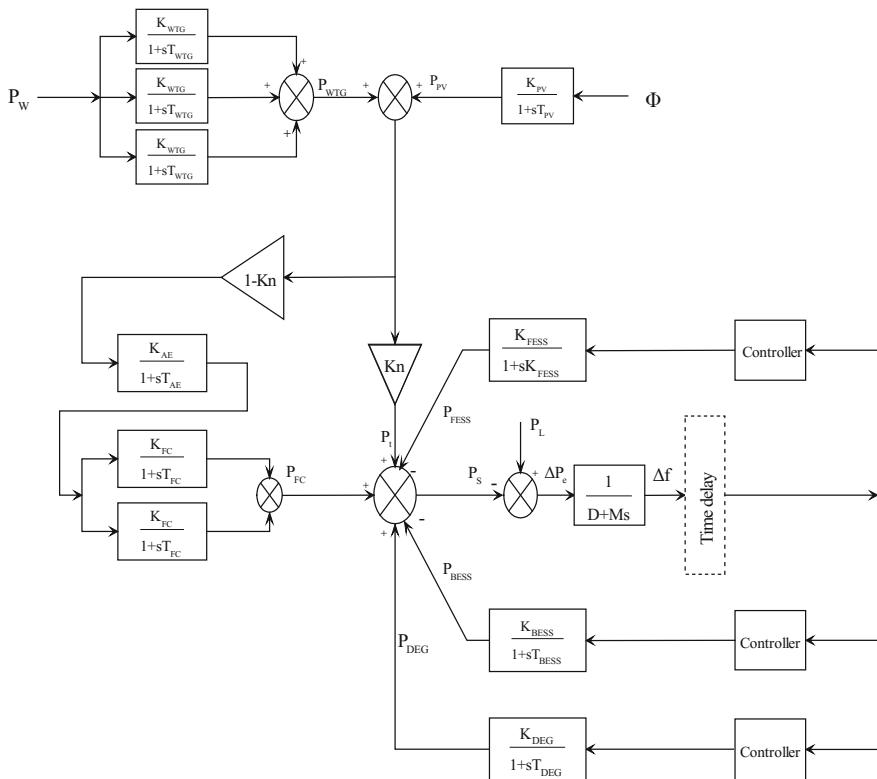


Fig. 1 Block diagram of the proposed hybrid distributed generation system

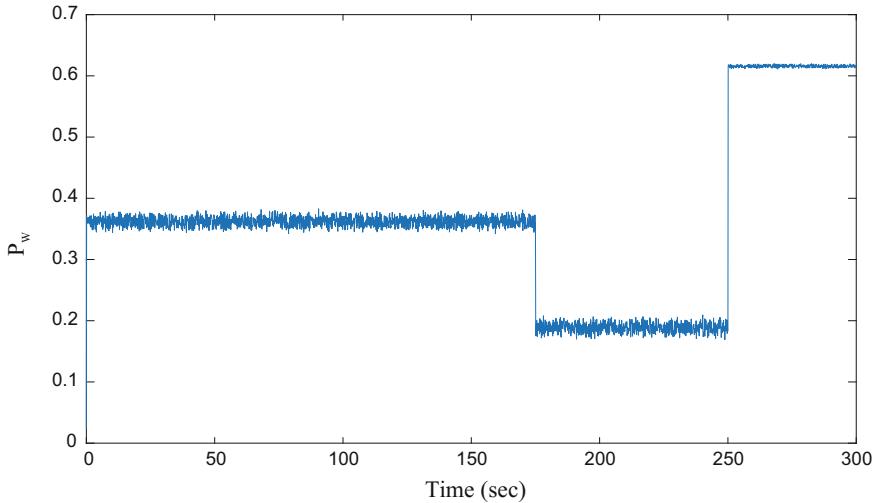


Fig. 2 Stochastic variation of wind power output

2.1 Model of Wind Power Generation

The output power of WTG totally depends on the wind speed. The speed of wind (V_W) [13] has been considered as

$$V_W = 10H(t) - 2.5H(t - 175) + 8H(t - 250) \quad (1)$$

The wind turbine's mechanical output power is

$$P_W = \frac{1}{2} \rho A_r C_p V_W^3 \quad (2)$$

where, $\rho = 1.25 \text{ kg/m}^3$, $A_r = 1735 \text{ m}^2$ and C_p = power coefficient. The stochastic variation of power output of wind turbine generations is depicted in Fig. 2.

2.2 Photovoltaic Output Power Model

The power output of photovoltaic (PV) system is varied with solar radiation (Φ) and the power output of the PV system is determined by

$$P_{PV} = \eta S \Phi [1 - 0.005(T_a + 25)] \quad (3)$$

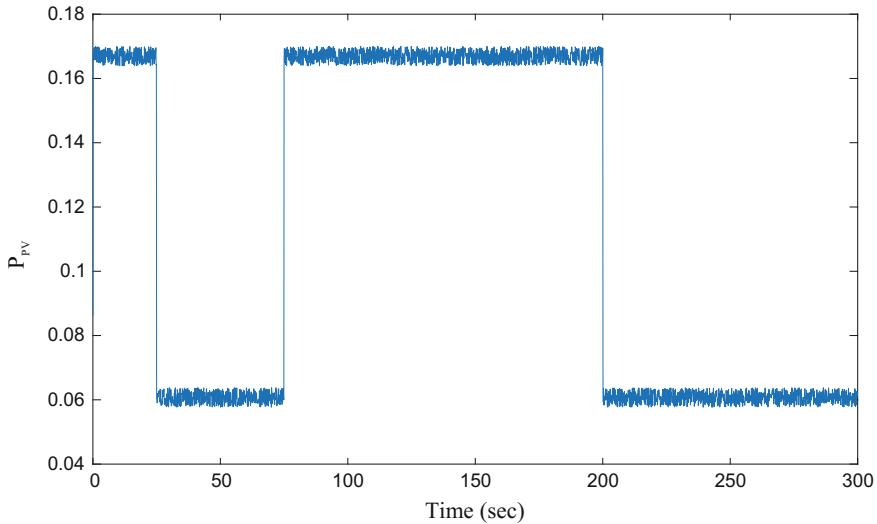


Fig. 3 Stochastic variation of PV system output power

where the conversion efficiency of the PV cell, $\eta = 10\%$, measured area of PV array, $S = 4084 \text{ m}^2$ and $T_a = 25^\circ\text{C}$ is the ambient temperature. Figure 3 the stochastic variation of the power output of PV system. The solar radiation (Φ) has been considered as

$$\Phi = 0.55H(t) - 0.35H(t - 25) + 0.35H(t - 75) - 0.35H(t - 200) + \Phi_n(t) \quad (4)$$

where $\Phi_n(t) = U(-0.1, 0.1)$.

2.3 Model of Load Demand

The load demand model [13] has been considered as

$$\begin{aligned} P_L = & H(t) - 0.3H(t - 50) - 0.15H(t - 100) + 0.2H(t - 150) \\ & + 0.1H(t - 200) + 0.1H(t - 250) + N_L \end{aligned} \quad (5)$$

where, $N_L = U(-0.05, 0.05)$. The stochastic variation of load demand is shown in Fig. 4.

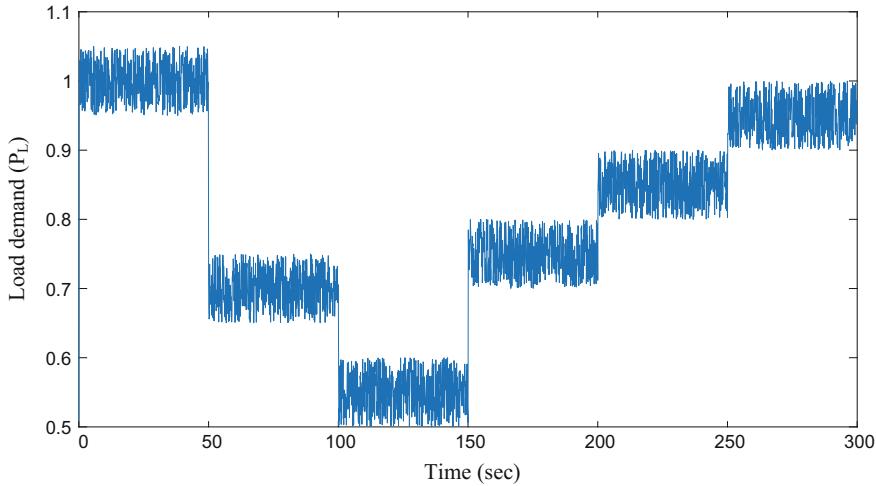
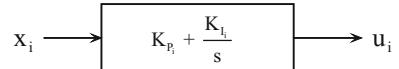


Fig. 4 Stochastic variation of load demand

Fig. 5 Block diagram of PI controller



3 PI Controller Structure

In this present work, proportional-integral (PI) controllers are used for controlling the DEG, FESS and BESS so as to maintain power balance and improved the frequency regulation. The block diagram of this is shown in Fig. 5, where x_i and u_i represent input and output of the controller respectively.

4 Fuzzy-PI Controller Structure

In this work, frequency deviation (Δf) and derivative of frequency deviation ($\dot{\Delta f}$) are used as the input signals to fuzzy-PI controllers and Mamdani fuzzy inference engine has been chosen. The identical rules of the proportional and integral gains of fuzzy-PI controller have been selected for the controllers of FESS, BESS and DEG and the scaling factors for input and output are tuned using GWOA. The triangular membership functions are utilized with the fuzzy linguistic variables for inputs and the output are presented in Fig. 6 and the rule base is given in Table 1.

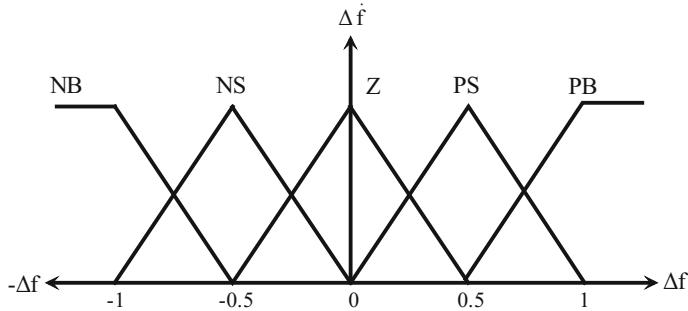


Fig. 6 Membership functions for the proposed controller

Table 1 Fuzzy rules for the fuzzy-PI controllers

Δf	$\Delta \dot{f}$					
		NB	NS	Z	PS	PB
NB	PB	PB	PB	NS	Z	
NS	PS	PS	PS	NS	NS	
Z	PS	PS	PS	NB	NS	
PS	Z	PS	Z	NB	NB	
PB	Z	Z	NS	NB	NB	

5 Objective Function

The gains of PI controllers and scaling factors of fuzzy-PI controllers for FESS, BESS and DEG are optimized by using the following objective function based on the integral square error criterion [23, 24] for GWOA.

$$J = \int_0^{\infty} (\Delta f)^2 dt \quad (6)$$

6 Grey Wolf Optimization Algorithm (GWOA)

The grey wolf optimizer (GWO) [18] is a new meta-heuristic optimization technique, which mimics the hunting behaviour of the grey wolf species. Grey wolves live in sets where each set consists of 5–12 wolves and there are four types of wolves in each set such as alpha (α), beta (β), delta (δ), and omega (ω) wolves. Searching for prey, encircling the prey and attacking the prey are main steps of hunting process respectively.

The grey wolf optimization algorithm is as follows:

I. Social hierarchy

In the grey wolf optimization algorithm (GWOA), the most dominant one is alpha (α) wolf, the second is beta (β) and third dominant one is delta (δ) respectively. The rest of wolves are omega (ω) and they follow the higher order dominated wolves.

II. Encirclement of the prey

The mathematically model of encirclement of the prey is represented as [18]:

$$S = |C W_p(t) - W(t)| \quad (7)$$

$$W(t+1) = W_p(t) - A S \quad (8)$$

where S is position vector distance between the prey position vector (W_p) and a grey wolf position vector (W), t is the current iteration, A and C are coefficient vectors and are expressed as:

$$A = 2 b r_1 - b \quad (9)$$

$$C = 2 r_2 \quad (10)$$

where the value of b vector is changed from 2 to 0 linearly with iterations and r_1, r_2 are random vectors in $[0, 1]$.

III. Hunting

In hunting behaviour of grey wolves, it is assumed that alpha, beta and delta wolves have better knowledge of prey position. To find the optimum position, the best first three solutions are recorded and utilized to update the wolf positions based on the best search position agents. Mathematically equations are shown below [18]:

$$S_\alpha = |C_1 W_\alpha - W|, \quad S_\beta = |C_2 W_\beta - W|, \quad S_\delta = |C_3 W_\delta - W| \quad (11)$$

$$W_1 = W_\alpha - A_1 S_\alpha, \quad W_2 = W_\beta - A_2 S_\beta, \quad W_3 = W_\delta - A_3 S_\delta \quad (12)$$

$$W(t+1) = \frac{W_1 + W_2 + W_3}{3} \quad (13)$$

IV. Attacking the prey

The hunting process of grey wolves is completed by attacking the prey when it ends up moving. The mathematically model is done by decreasing the value of b vector from 2 to 0 over the no of iterations. It is found that $|A| < 1$ compels the wolves to attack the prey.

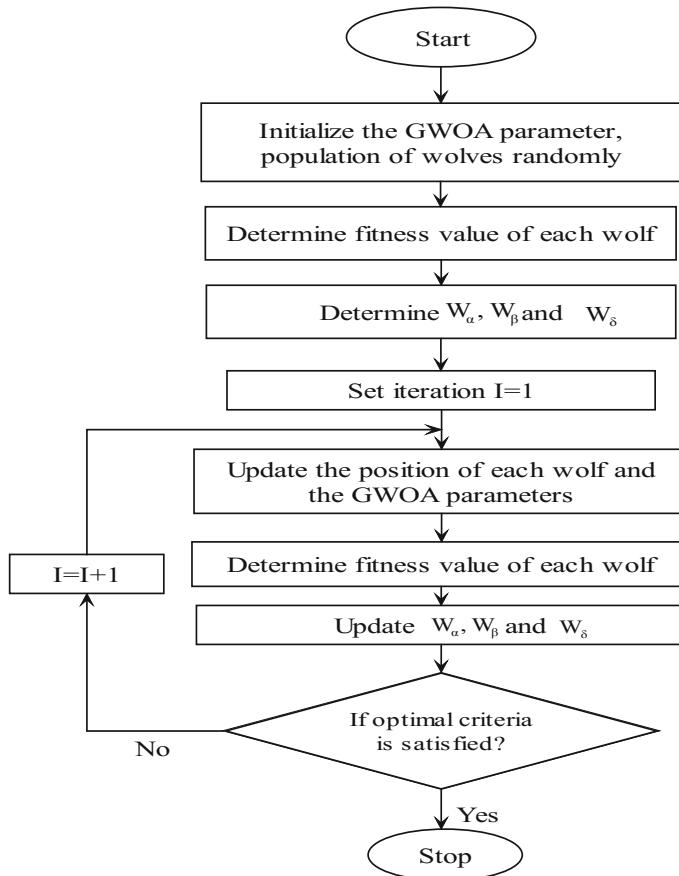


Fig. 7 Flowchart for grey wolf optimization algorithm

V. Searching for prey

According to the dominance of α , β and δ wolves, the GWOA is used to search the prey. The wolves diverge from each other to search the prey and converge when attack to prey.

The flowchart of the Grey Wolf Optimization Algorithm is shown in Fig. 7.

7 Results and Discussions

In this work, the gains of PI controllers for FESS, BESS and DEG have been optimized by using GWO algorithm according to the objective function as given in Eq. (6). For optimizing using GWOA, 50 population size and 300 numbers of iterations

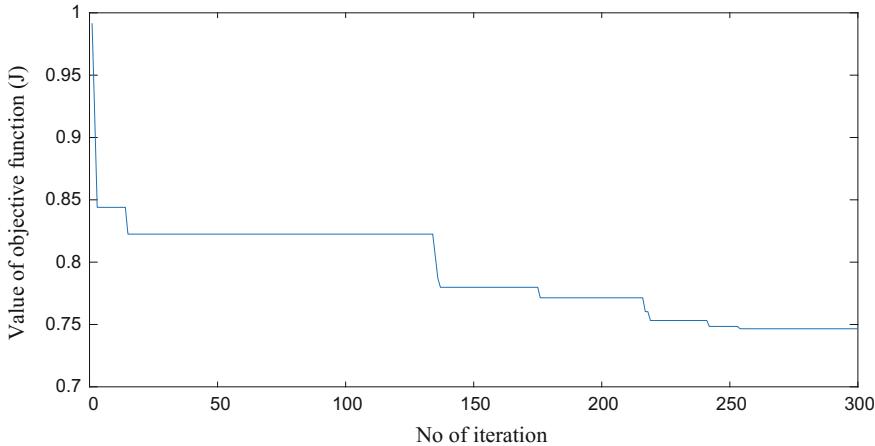


Fig. 8 Convergence characteristic of the objective function

Table 2 Gains and scaling factors for PI and fuzzy-PI controllers optimized by GWOA

Gains and scaling factors	Controllers for FESS	Controllers for BESS	Controllers for DEG
K_P	3.95	4.15	4.35
K_I	3.98	4.22	4.46
α_1	2.05	2.11	2.98
α_2	2.06	2.12	2.32

are taken and the convergence characteristic of the objective function is shown in Fig. 8. Table 2 shows the gains of GWOA optimized PI controllers and the scaling factors α_1 and α_2 , optimized by GWOA for fuzzy-PI controllers for FESS, BESS and DEG optimized based on objective function given in Eq. (6). A time delay of 0.1 s with the system grid frequency signal as input to the controller has been considered. The dynamic response of frequency regulation has been compared with and without PI controller and time delay are shown in Fig. 9. From this figure, it is seen that with the PI controller, the dynamic response is better than the without controller. Figure 10, shows the comparison of dynamic responses of frequency with PI and fuzzy-PI controllers and fuzzy-PI controllers reduce peak deviations and damp out oscillations significantly compared to PI controllers. Figures 11, 12 and 13 are shown in the dynamic response of the individual powers of BESS, FESS and DEG.

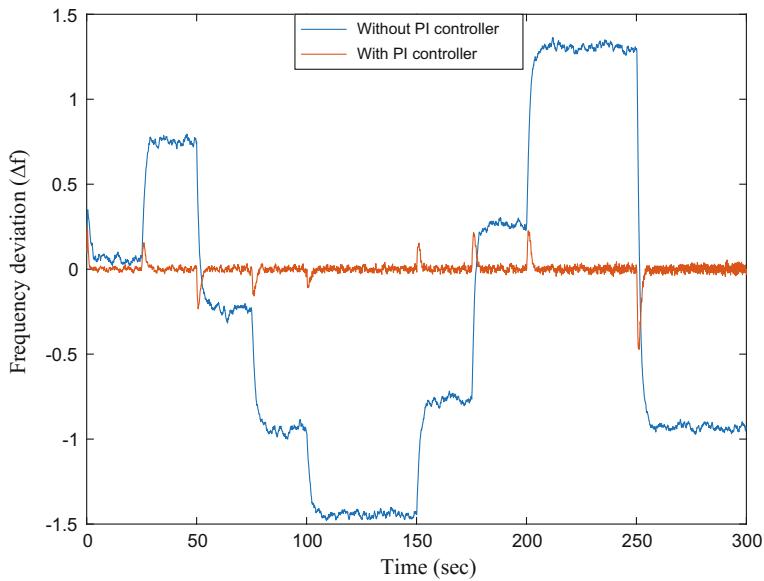


Fig. 9 Dynamic response of frequency deviation (Δf) with and without PI controller

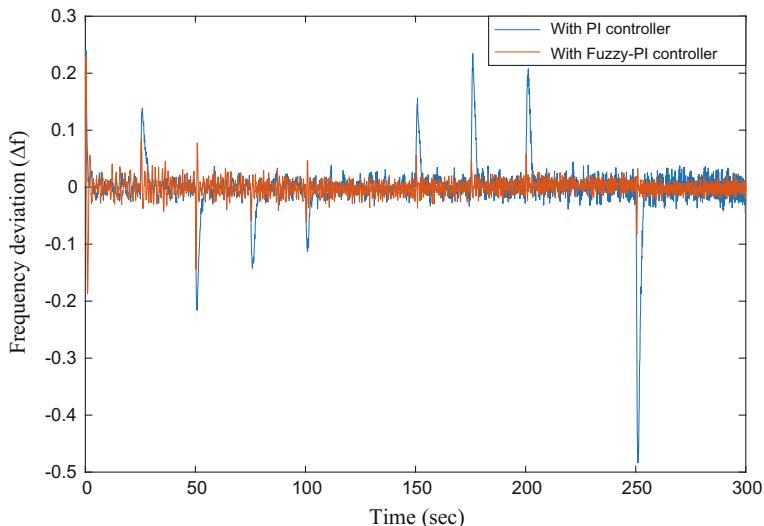


Fig. 10 Dynamic response of frequency deviation (Δf) with PI and fuzzy-PI controller

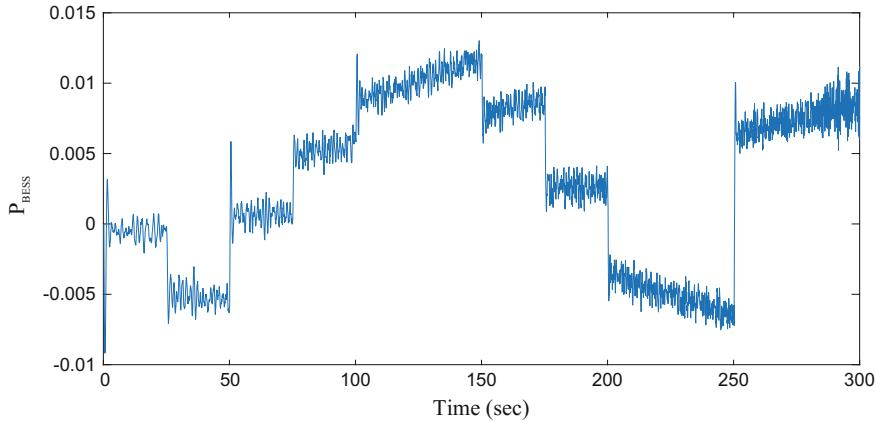


Fig. 11 Dynamic response of P_{BESS} with fuzzy-PI controller

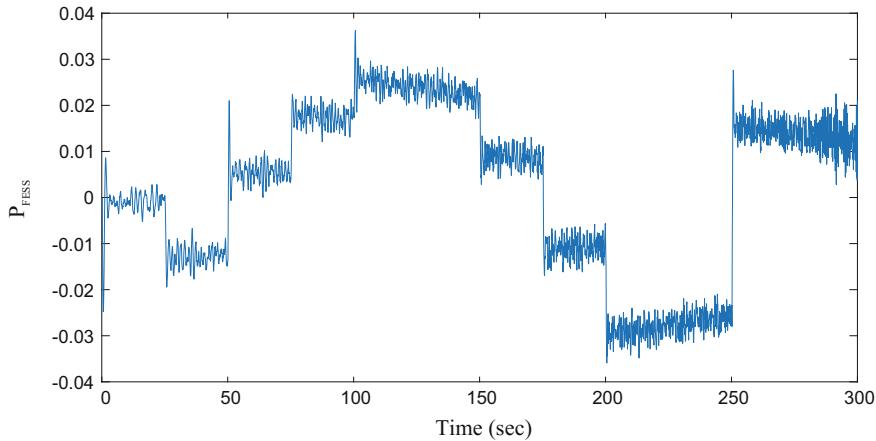


Fig. 12 Dynamic response of P_{FESS} with fuzzy-PI controller

8 Conclusions

In the present work, fuzzy-PI controllers are used before FESS, BESS and DEG and frequency signal with delay of 0.1 s have been considered as the feedback signal to the controller. The scaling factors of fuzzy-PI controllers have been optimized using grey wolf optimization algorithm (GWOA) and the dynamic responses of frequency regulation have been compared with the GWOA optimized PI controllers. From the results, it is found that the fuzzy-PI controller performs well to reduce the peak deviations and gives better dynamic response compared to the PI controllers.

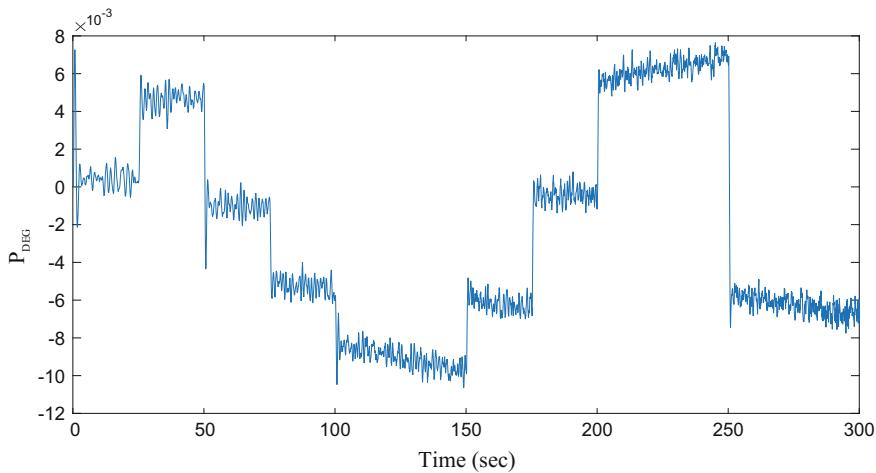


Fig. 13 Dynamic response of P_{DEG} with fuzzy-PI controller

Appendix

Energy resource/storage system	Gain constant (K)	Time constant (T)	Other parameters
Wind turbine generator (WTG)	$K_{WTG} = 1$	$T_{WTG} = 1.5$	$M = 0.4$ $D = 0.03$
Solar photovoltaic cell (PV)	$K_{PV} = 1$	$T_{PV} = 1.5$	
Aqua electrolyser (AE)	$K_{AE} = 0.002$	$T_{AE} = 0.5$	
Fuel cell (FC)	$K_{FC} = 0.01$	$T_{FC} = 4$	
Flywheel energy storage system (FESS)	$K_{FESS} = -0.01$	$T_{FESS} = 0.1$	
Battery energy storage system (BESS)	$K_{BESS} = -0.003$	$T_{BESS} = 0.1$	
Diesel engine generator (DEG)	$K_{DEG} = 0.003$	$T_{DEG} = 2$	

References

1. H. Fathabadi, Novel standalone hybrid solar/wind/fuel cell power generation system for remote areas. *Solar Energy* **146**, 30–43 (2017)
2. A. Rajabi-Ghahnavieh, S.A. Nowdeh, Optimal PV–FC hybrid system operation considering reliability. *Electr. Power Energy Syst.* **60**, 325–333 (2014)
3. S.K. Aditya, D. Das, Battery energy storage for load frequency control of an interconnected power system. *Electr. Power Syst. Res.* **58**(3), 179–185 (2001)
4. E. Reihani, S. Sepasi, L.R. Roose, M. Matsuura, Energy management at the distribution grid using a battery stroge storage system (BESS). *Electr. Power Energy Syst.* **77**, 337–344 (2016)
5. G.S.M. Mousavi, F. Faraji, A. Majazi, K. Al-Haddad, A comprehensive review of Flywheel Energy Storage System technology. *Renew. Sustain. Energy Rev.* **67**, 477–490 (2017)
6. I. Ngamroo, Robust frequency control of wind–diesel hybrid power system using superconducting magnetic energy storage. *Int. J. Emerging Electric Power Syst.* **10**(2) (2009)
7. T.S. Bhatti, A.A.F. Al-Ademi, N.K. Bansal, Load-frequency control of isolated wind diesel hybrid power systems. *Energy Convers. Manag.* **38**(3), 829–37 (1997)
8. S. Mishra, G. Mallesham, A.N. Jha, Design of controller and communication for frequency regulation of a smart microgrid. *IET Renew. Power Gener.* **6**(4), 248–258 (2012)
9. K. Abe, S. Ohba, S. Iwamoto, New load frequency control method suitable for large penetration of wind power generations, in *IEEE Power Engineering Society General Meeting* (2006), <https://doi.org/10.1109/pes.2006.1709177>
10. T. Senju, T. Nakaji, K. Uezato, T. Funabashi, A hybrid power system using alternative energy facilities in isolated island. *IEEE Trans. Energy Convers.* **20**(2), 406–414 (2005)
11. D.-J. Lee, L. Wang, Small-signal stability analysis of an autonomous hybrid renewable energy power generation/energy storage system, part I: Time-domain simulations. *IEEE Trans. Energy Convers.* **23**(1), 311–320 (2008)
12. P.K. Ray, S.R. Mohanty, N. Kishor, Proportional–integral controller based small-signal analysis of hybrid distributed generation systems. *Energy Convers. Manage.* **52**(4), 1943–1954 (2011)
13. I. Pan, S. Das, Fractional order AGC for distributed energy resources using robust optimization. *IEEE Trans. Smart Grid* **7**(5), 2175–2186 (2016)
14. Z.A. Obaid, L.M. Cipcigan, M.T. Muhsin, Fuzzy hierachal approach-based optimal frequency control in the Great Britain power system. *Electr. Power Syst. Res.* **141**, 529–537 (2016)
15. J.S. Saini, V. Jain, A genetic algorithm optimized fuzzy logic controller for automatic generation control for single area system. *J. Inst. Eng. India Ser. B* **96**(1), 1–8 (2015)
16. C.S. Chang, W. Fu, Area load frequency control using Fuzzy gain scheduling of PI controllers. *Electr. Power Syst. Res.* **42**, 145–152 (1997)
17. R.K. Sahu, S. Panda, G.T. Chandra Sekhar, A novel hybrid PSO-PS optimized fuzzy PI controller for AGC in multi area interconnected power systems. *Electr. Power Energy Syst.* **64**, 880–893 (2015)
18. S. Mirjalili, S.M. Mirjalili, A. Lewis, Grey wolf optimizer. *Adv. Eng. Softw.* **69**(3), 46–61 (2014)
19. V. Kumar, D. Kumar, An astrophysics-inspired grey wolf algorithm for numerical optimization and its application to engineering design problems. *Adv. Eng. Softw.* 1–24 (2017)
20. D. Guha, P.K. Roy, S. Banerjee, Load frequency control of interconnected power system using grey wolf optimization. *Swarm Evol. Comput.* (2015)
21. Y. Sharma, L.C. Saikia, Automatic generation control of a multi-area ST—thermal power system using grey wolf optimizer algorithm based classical controllers. *Electr. Power Energy Syst.* **73**, 853–882 (2015)
22. D.K. Lal, A.K. Barisal, M. Tripathy, Grey wolf optimizer algorithm based Fuzzy PID controller for AGC of multi-area power system with TCPS. *Proc. Comput. Sci.* **92**, 99–105 (2016)
23. W.C. Schultz, V.C. Rideout, Control system performance measures: past, present and future. *IRE Trans. Autom. Control AC-6*(22), 22–35 (1961)
24. K. Ogata, *Modern Control Engineering: A Book* (Prentice-Hall, Englewood cliffs, NJ, 1970), pp. 293–313

Improved Musical Instrument Classification Using Cepstral Coefficients and Neural Networks



Shruti Sarika Chakraborty and Ranjan Parekh

1 Introduction

A recognition problem deals with the identification of a test data, comparing it against a set of train data stored in a training database, with which the system has been modeled. Classification is the categorization of objects (in this case, audio test samples) based on their similarities. Musical instrument identification is the process where a test audio signal derived from a musical instrument is matched to one of the sets of predefined classes of musical instruments which are trained with their respective sounds during a training phase. For robustness, many train samples encompassing variations in sound are used during the training phase. If there exist a number of test samples, derived from multiple instruments of different families, the system classifies each of them to their respective families based on what the machine has learnt during the training phase. This is known as musical instrument classification [1]. As the domain of musical instruments is very wide and expanding, manual cataloging is difficult and prone to errors. Automated musical instrument classification aids in extraction of melody from musical sound, identification and separation of sound sources in polyphonic audio, identification of solo musical instruments, automatic music transcription, beat tracking, musical information retrieval, and much more application of similar types [2]. The focus of this article is to improve on existing techniques available for musical instrument classification using multiple features and multiple classifiers and observing which produces optimal results. The organization of this paper is as follows: Sect. 2 highlights previous approaches, Sect. 3 outlines the various features and classifiers used in the proposed approach, Sect. 4 tabulates the experimental procedures and results obtained, Sect. 5 provides a comparative analysis of the present approach vis-à-vis other approaches, and Sect. 6 brings up the conclusions and future scopes of work.

S. S. Chakraborty (✉) · R. Parekh

School of Education Technology, Jadavpur University, Kolkata, India

e-mail: shrutisarikachakraborty@gmail.com

© Springer Nature Singapore Pte Ltd. 2018

J. K. Mandal et al. (eds.), *Methodologies and Application Issues of Contemporary Computing Framework*, https://doi.org/10.1007/978-981-13-2345-4_10

2 Previous Work

There is a number of works related to musical instrument classification. MFCC features are proposed for identification tasks along with delta MFCC features which are obtained by taking the time derivative of MFCCs. SVM algorithm is popularly used for classification [2]. In other works, also, MFCC is used to extract the features of audio signals arising from musical instruments and are paired with K-NN classifier for classification into five classes which are Cello, Piano, Trumpet, Flute, and Violin. The overall accuracy was more than 80% with 90 samples in train set and 60 for the test set. The sound samples were obtained from Electronic Music Studio, University of Iowa, and accuracy is prone to decrease with increase in the number of instruments and instruments in polyphonic recordings will not be distinguished by this method [1]. An algorithm proposed for classification task involved SVM, MLP, and AdaBoost where AdaBoost gave the best result with an accuracy of more than 90% [3]. The use of MFCC- and Timbral-related audio descriptors for the identification of musical instrument is very common. K-nearest neighbor, support vector machine, and binary tree are used for classification purpose. The accuracy was found to decrease from 90 to 75% with an increase in the number of instruments from 5 to 15 [4]. The use of MFCCs for the musical instrument identification is also overviewed in signal processing techniques for music analysis [5]. Musical instrument classification using wavelet dependent timescale features has been proposed. In this, at first, the continuous wavelet transform of the signal frame is taken and then features related to temporal variation and bandwidth are considered for feature extraction [6]. A preliminary work on ontology designed for musical instruments has been proposed. The paper also provided the investigation of heterogeneity and limitations in existing instrument classification schemes [7]. A method utilizing convolutional neural networks has been proposed. It produced high performances in their confusion matrix. The experiment used 11 instruments including cello and clarinet. The accuracy obtained was better than previous approaches [8]. Mel scale cepstral coefficient (MFCC)-based features coupled with a multilayer perceptron (MLP)-based classifier has been used for categorization of 2716 clips from seven different instruments obtaining an average accuracy of 98.38% accuracy [9]. Recent journals mostly dealt with instrument identification from polyphonic sources which is beyond the scope of this study. However, extraction of the timbre of an instrument utilizing Bayesian network achieved an accuracy of over 95%. The timbre was extracted using a set of features. The feature is the amplitude of the frequency peak with the highest magnitude within each window. The FFT of the signal was partitioned into 10 exponentially increasing windows [10]. A method was proposed for musical instrument identification using short-term energy and ZCR. ZCR is the count of how many times signal changes the sign [11].

3 Proposed Approach

The classification of musical instruments has many challenges due to their multidimensional characteristics. The musical instruments also vary in shape, sizes, types, geographical locations, cultures, and playing style all of which makes it extremely difficult for a common person to categorize them reliably. As the machine learns from its train data, hence, it is required to input a large number of sound samples as train samples ranging in all variety for each instrument to the system in order to attain maximum accuracy for major problem faced in this case that is low accuracy as number of musical instrument increases in train set, the accuracy decreases. Musical instrument classification is a very important task for musical information retrieval system, audio source separation, automatic music transcription, and genre classification as the domain of musical instrument is very wide. The goal of this paper is to maximize the accuracy of musical instrument classification.

For the musical instrument classification task, in the initial stage after preprocessing, the feature vectors or acoustic vectors of all six audio descriptors are extracted. MFCC, LPC, and spectral centroids are spectral descriptors, while pitch salience, HPCP, is tonal descriptors. The CC is categorized into spectral descriptors according to its characteristic nature. Then, all the features are individually inputted to four different classifiers which are K-NN or K-nearest neighbors (number of neighbors ranging between 1 and 5 and metric Euclidian distance), support vector machine (SVM) (linear), artificial neural network (ANN) (with Softmax and Rectifier activation function), and random forest (number of decision trees equal to 10,000). The performance of six features against all four classifiers is studied, and the set of best feature and classifier is chosen to be cepstral coefficients and ANN as the accuracy of cepstral coefficient as more than all others in maximum cases. The reason behind the choice of the best classifier is discussed later. Three sets of experiments were performed with varied conditions for each case and the maximum overall accuracy ranged between 90 and 93%. The overall block diagram of two processes is given in Fig. 1.

3.1 Preprocessing

The sound samples used in musical instrument classification are sampled at 44,000 kHz. They are noise free. Ten instruments are used which belongs to five different families. The instruments are organ, French horn, cello, clarinet, tambourine drum flute, trumpet, violin, and piano. The train samples are of duration 5–10 s each while the test samples range from 10 to 20 s. Fourteen to fifteen train samples are used in train set per instrument or class and three test samples are used per instrument in the test set in each of three experiments.

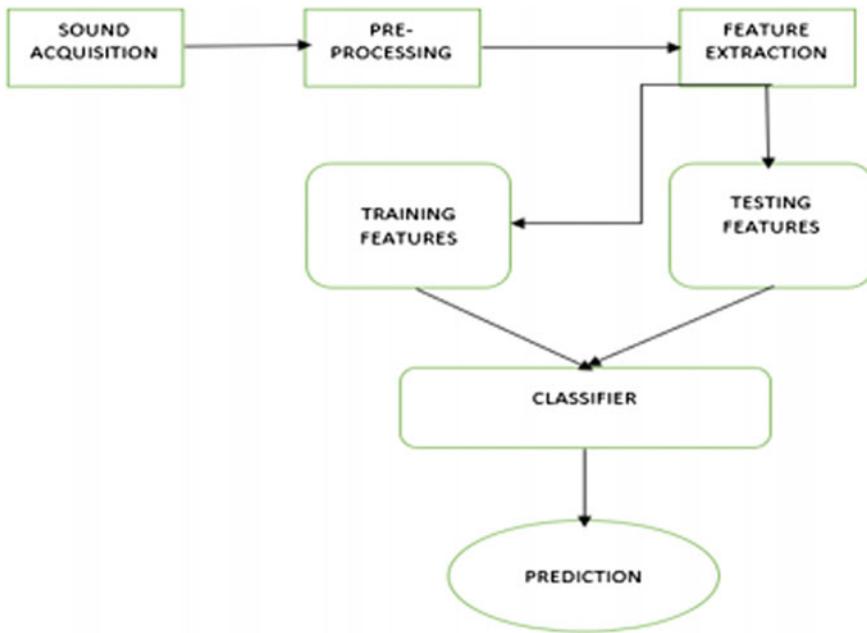


Fig. 1 Overall block diagram of overall process

3.2 Feature Extraction

The features extracted for musical instrument classification are harmonic pitch class profile (HPCP), linear predictive coding (LPC) coefficients, Mel frequency cepstral coefficients (MFCC), spectral centroids, pitch salience peaks, and cepstral coefficients (CC). A brief description of their characteristics is listed below.

Mel frequency cepstral coefficients (MFCC): MFCCs are useful in identifying the linguistic content and timbre of the sound discarding the background noise, emotion, etc. MFCCs are commonly used in speech recognition and are finding increased use in music information recognition and genre classification systems and also speaker recognition. It is evident from psychological studies that the human perception of the contents of the frequency of sound for speech signals does not follow a linear scale. For each tone with an actual frequency f measured in Hz, a subjective pitch on a scale being measured is called the “Mel” scale. This Mel frequency scale M follows a linear frequency spacing below 1000 Hz and a logarithmic spacing above 1000 Hz [1]. See Eq. (1).

$$M = 2595 \cdot \log_{10} \left(1 + \frac{f}{700} \right) \quad (1)$$

(Linear Predictive Coding- LPC) coefficients: A filter that approximates spectral characteristics of a sound is the filter which is approximated by LPC coefficients. The LPC coefficients give a set of filtered coefficients (a_k), and the frequency response of the resulting filter approximates spectrum of the input sound. So, a signal can be approximated by LPC model as the linear combination of past samples which is basically the expression of IR filter (infinite response filter) that is a linear combination of previous samples. The goal of LPC is to find the coefficients that best approximates the signal in question.

Spectral Centroids: The centroid, of the frequency spectrum, is the human perception of “brightness”. It is derived by multiplying the value of each frequency by its magnitude in the spectrum, then taking the sum of all these and again dividing the whole numerator by the magnitude in the spectrum of the signal. The centroid is the descriptor feature that helps in the characterization of the spectral shape of a particular sound. See Eq. (2) where k = frequency, $X[k]$ = magnitude in the spectrum.

$$\text{Centroid} = \frac{\sum_{k=0}^{N/2} k |X[k]|}{\sum_{k=0}^{N/2} |X[k]|} \quad (2)$$

Pitch Salience Peaks: The salience function is given by Eq. (3) where $S[b]$ = salience at bin frequency b , $e()$ = magnitude threshold function, $g()$ = weighting function applied to peak p , β = magnitude compression value, A_p = amplitude of peak, P = number of peaks, and H = number of harmonics. The spectral peaks which are extracted from the spectrum of the signal are used to construct a salience function which is a representation of pitch salience over timescale. The function contains peaks which are F0 candidates for the main melody. The idea of peak salience relates to how much of a peak is present at a particular frame of the sound sample.

$$S[b] = \sum_{h=1}^H \sum_{p=1}^P e(A_p) g(b, h, f_p) (A_p)^\beta \quad (3)$$

Harmonic Pitch Class Profile(HPCP): HPCP is a vector which represents the intensities of the twelve semitone pitch classes (corresponding to notes from A to G#). It is a group of features which is extracted from a sound signal, based on a pitch class profile. HPCP are features that are pitch distributed and are sequences of feature vectors which describes tonality and measures the relative intensity of each of the 12 pitch classes of the equal-tempered scale within the frame under analysis. They are often referred to as chroma. This is explained using Eq. (4) where A_p = amplitude of spectral peak p , P = total number of peaks, $w(k, f_p)$ = weight of the peak frequency f_p for bin k , and k = spectral bin locations of the chosen HPCP frequencies. HPCP is the sum of the weighted square of amplitudes of peaks along all peaks. Twelve HPCP filters are taken to represent a sound sample.

$$\text{HPCP}[k] = \sum_{p=1}^P w(k, f_p) A_p^2 \quad (4)$$

Cepstral Coefficients(CC): These features are not used extensively but only in few research articles [12]. These features are extracted with a view to extracting the overall spectral characteristics of the signal which unlike MFCC does not obstruct spectral regions and in fact this is an improved version of MFCC for overall sound recognition problems as MFCC was made especially for speech recognition problems as the phonemes uttered by human are better captured by application of Mel scale which gives more emphasis to areas of lower frequencies. The CC represents the timbre of the sound by the envelope of the spectrum. Again, the short-time power spectrum enhances the process of timber extraction from the sound. The timbre is the property which works independently of tone or pitch to identify sounds. Hence, correct representation of it will lead to better identification. The cepstral coefficients are simply DCT of the log of short-time power spectrum given by Eq. (5).

$$\text{CC} = \text{DCT}\left(\log\left(\frac{1}{N}|X_k|^2\right)\right) \text{ where } X_k = \sum_{i=0}^{N-1} x_i \cdot e^{-j \cdot \frac{2\pi k i}{N}} \quad (5)$$

Here, X_k is the FFT of a hamming windowed signal applied on frame size 25 ms overlapped by 10 ms. In a typical case, the signal in a frame, denoted by (n) , where $n = \{0, \dots, N - 1\}$, and after windowing the signal is given by $s(n) * t(n)$, where $t(n)$ is the representation of Hamming window which is given by Eq. (6)

$$t(n) = 0.54 - 0.46 \cos\left(\frac{2\pi n}{N-1}\right); 0 \leq n \leq N-1 \quad (6)$$

3.3 Classification

This is a vital step in any recognition problem. In this step, the features from the test data and train data are compared against each other to measure similarity in order to achieve classification. There are many classifiers present but the choice of most effective one for a specific problem helps in achieving the goal of the system and has better accuracy compared to others while also preventing overfitting. In this study, four major and most prevalent machine learning classifiers are used which are K-NN, SVM, ANN, and random forest. After observing the effectiveness, efficiency, robustness, and reproducibility, the choice of the best classifier is made.

K-nearest neighbors (K-NN)

The K-nearest neighbors algorithm (k-NN) is a popular nonparametric algorithm used for solving classification as well as regression problems. Here, k-NN is used as a classifier. The input to k-NN is two vectors each deriving from test set and train

set. For CC, each vector is of size $[1 \times 2560]$. As 40 coefficients are taken and 64 centroids are taken in Vector quantization step for codebook formation, $[40 \times 64 = 2560]$ is the size. The output of a k-NN classifier is a class membership. An object is classified by a majority vote of its surrounding neighbors. That means an object is assigned to the class that consists of the highest number of common elements among its nearest neighbors. The number of neighbors of K-NN was between 1 and 5 with metric as the Euclidian distance in all cases. For two n -dimensional vectors $P = \{p_1, p_2, \dots, p_n\}$ and $Q = \{q_1, q_2, \dots, q_n\}$, Euclidean distance is defined as

$$d(P, Q) = \sqrt{\sum_{i=1}^n (p_i - q_i)^2} \quad (7)$$

Support Vector Machine (SVM)

Support vector machine (SVM) is a supervised machine learning algorithm which can be used for both classification and regression problems. However, it is mostly used in solving classification tasks. Each data item is plotted as a point in an n -dimensional space (where n is the number of features) with the value of each feature being the value of a particular coordinate. Classification is performed by finding the hyperplane that differentiates the two classes with the maximum margin. The sum of distances between two boundary points of two classes has to be maximum in order to choose that hyperplane as an optimal hyperplane. One versus rest method is used for multi-class classification. In this case, we choose the class which classifies the test datum with greatest margin. The SVM used here was linear.

Random Forest

Random forests or random decision forests are an ensemble learning method for classification, regression, and other tasks, which operate by constructing many decision trees at training time and outputting the class based on maximum votes achieved from the number of decision trees used. The number of trees chosen in this case is 10,000.

Artificial Neural Networks (ANN)

An ANN is based on a collection of connected units or nodes called artificial neurons. Each connection between artificial neurons is capable of transmitting a signal from one to another. The artificial neuron that receives the signal processes it and then it signals artificial neurons which are connected to it. The input nodes connected to the layer of neurons connect again to the next hidden layer of neurons (if the number of hidden layers > 1) and after connecting to n hidden layers (specified by the user) at last the network connects to the output layer. Now there is weight associated to each connection or synapse and that gets modified due to backpropagation in order to minimize error. So, in this study, multilayer perceptron (MLP) is used. A multilayer perceptron (MLP) consists of more than two layers of nodes and also utilizes a supervised learning technique known as backpropagation for training purposes. It is distinguished from linear perceptron by its multiple layers and nonlinear activation function. In this study, four hidden layers are used apart from input and output layer.

The rectifier activation function is used for input and hidden layers while Softmax activation function which can be vaguely said as the categorical version of sigmoid function where sigmoid is used for binary classification and Softmax function is used for multi-class classification is used at the output layer. The loss function used here is categorical cross entropy and Adam optimizer is used. The loss in case of CC and MFCC for all three cases was in the order of e^{-6} . The rectifier function is given by Eq. (8), where x is the independent variable of input from input/hidden layers.

$$\Phi_x = \max(x, 0) \quad (8)$$

The Sigmoid function is given by

$$\Phi_x = \frac{1}{1 + e^{-x}} \quad (9)$$

The Softmax function is given by Eq. (10):

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}} \text{ for } j = 1, \dots, K \quad (10)$$

Here

$$z = w_0 x_0 + w_1 x_1 + \dots + w_m x_m \quad (11)$$

where w is the weights of the input variable and $j = 1, 2, \dots, K$ are the categories or classes. Softmax function calculates the distribution of probability of the event over “n” different events. Later, the calculated probabilities serve helpful for the purpose of determining the target class for the given inputs.

The accuracy obtained with the Cepstral coefficients (40 coefficients are taken, as it provides best accuracy and reproducibility) along with ANN surpassed the accuracy of previous approaches conducted with MFCCs and that is discussed in next section. The decrease of accuracy with an increase in a number of train data is found to be less and the feature is also robust for it provided good accuracy in all cases. ANN and K-NN both provided very good accuracy but ANN is chosen over K-NN for it can be tuned and accuracy can be improved further.

4 Experiments and Results

This section is a major part of this study as it consists of all the results and plots that has led us to the conclusion. It is this part, which has helped us in choosing appropriate feature and classifier for the system. The experiments are mostly done with Python code in Spyder environment. The dataset for musical instrument classification is derived from Philharmonia Orchestra instruments [13] and Freesound

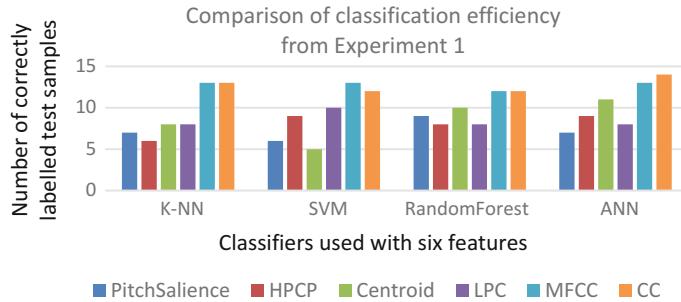


Fig. 2 Comparison of classification efficiency from experiment 1

music database [14]. The pitch of the instruments ranged from A#3 to G5, and the dynamic was of two types, Forte and Piano. The articulation was normal. Dynamics investigates how loud or quiet the sound is. In music, articulation is the performance technique which affects the transition or continuity on a single note or between multiple notes or sounds. Pitch may be quantified as a frequency. From the whole set, experimentation is done only with ten instruments. They are organ, French horn, cello, clarinet, tambourine drum, flute, trumpet, violin, and piano. They belong to five families of instruments. Three experiments have been carried out varying the characteristics of sounds. Two experiments are carried out with the same set of five instruments with varying conditions to study the effect of change of pitch and dynamics in the classification process and the third experiment is made with 10 instruments. The instruments belong to the following families.

Keyboards and Harp:	Organ, Piano
Woodwind Family:	Flute, Clarinet
Brass Family:	Trumpet, French Horn
String Family:	Violin, Cello
Percussion Family:	Drum, Tambourine

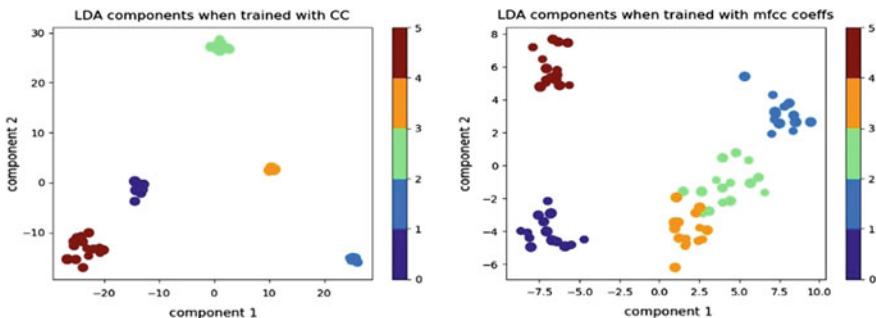
Experiment 1:

The first experiment was carried out with drum, flute, trumpet, violin, and piano which belongs to five different families of musical instrument mentioned earlier. For each instrument or class, 15 samples were taken during training and 3 samples were taken for testing, so a total of 75 samples were taken during train phase and 15 samples were used in the test phase to check the accuracy and robustness of the system. The samples belonged to random pitch and dynamics. The experiment was carried out with six features and four classifiers as specified in the previous section. The features are MFCC, CC, LPC coefficients, Centroid, HPCP, and Pitch salience peaks. The classifiers are K-NN, ANN, random forest, and SVM. The result obtained showed maximum accuracy with CC. The result is graphically plotted in Fig. 2.

The results show that the accuracy is quite low for Pitch salience peaks, HPCP, LPC coefficients, and centroid. It can be observed that maximum accuracy is obtained

Table 1 The confusion matrix obtained with CC and ANN for experiment 1

Actual	Predicted				
	Drum	Flute	Trumpet	Violin	Piano
Drum	3	0	0	0	0
Flute	0	2	1	0	0
Trumpet	0	0	3	0	0
Violin	0	0	0	3	0
Piano	0	0	0	0	3

**Fig. 3** LDA plots depicting clustering and overlapping of train samples for CC and MFCC

against ANN followed by K-NN for CC and MFCC. The highest number of the correctly labeled test sample is 14 with CC, while it is 13 for MFCC (Table 1).

Since the accuracy obtained for MFCC and CC is very close, hence, the linear discriminant analysis (LDA) plots are used to decide the best classifier to observe the degree of uniqueness imparted by each during the training phase (Fig. 3).

The above plots clearly show that the cluster formed by CC is tighter and farther apart from each other than MFCC. There is some overlapping in case of MFCC but there is no overlapping in case of CC.

Experiment 2:

The second experiment was carried out with the same instruments but each musical instrument had random dynamics but a fixed pitch with six features and four classifiers: Drum—Unpitched, Flute—B4 and B5, Trumpet—C5 and C6, Violin—A3 and A4, and Piano—D4. The pitch information of each instrument is different. The number of train samples used here is 14 per class and 3 test samples per class are used. This was done deliberately to observe the capability of classification of HPCP and pitch salience especially. If CC and MFCC give good accuracy under this condition also, it can be inferred that they are robust features as the train samples and test samples are different from the previous case. The result obtained showed maximum accuracy with CC. The result is graphically plotted in Fig. 4.

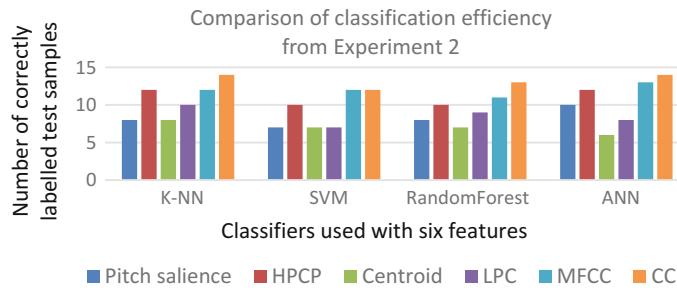


Fig. 4 Comparison of classification efficiency from experiment 2

Table 2 The confusion matrix obtained with CC and ANN for experiment 2

Actual	Predicted				
	Drum	Flute	Trumpet	Violin	Piano
Drum	3	0	0	0	0
Flute	0	3	0	0	0
Trumpet	0	0	3	0	0
Violin	0	0	0	2	1
Piano	0	0	0	0	3

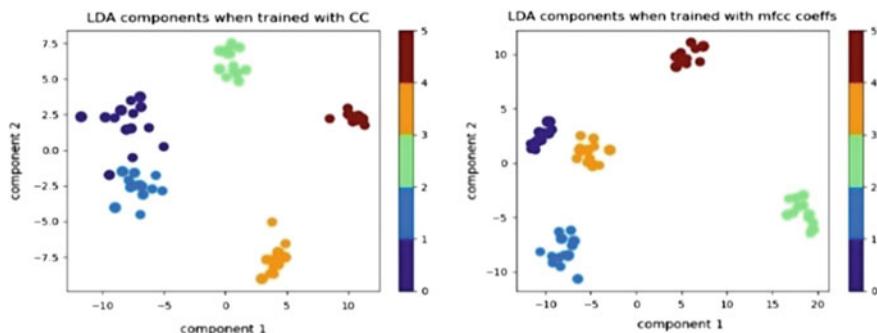


Fig. 5 LDA plots depicting clustering and overlapping of train samples for CC and MFCC

As observed from this plot, the maximum accuracy is obtained for CC with ANN. The highest number of the correctly labeled test sample is 14 with CC, while it is 13 for MFCC. The other features have much lesser accuracy (Table 2).

Since the accuracy obtained for MFCC and CC are very close, hence, the LDA plots are used to decide the best classifier to observe the degree of uniqueness imparted by each during the training phase (Fig. 5).

The above plot clearly shows that the cluster formed by CC is tight and far apart from each other and so does MFCC. In this experiment due to varying pitch characteristics, the classification is even better for MFCC. In order to come to a conclusion

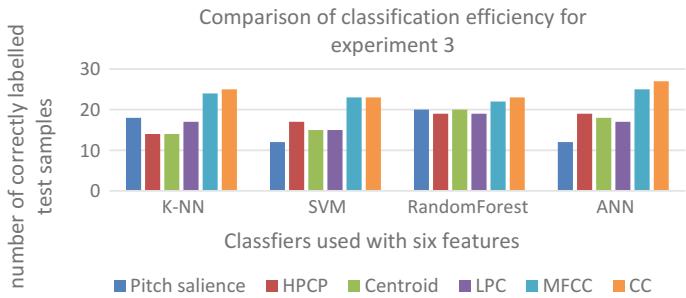


Fig. 6 Comparison of classification efficiency from experiment 3

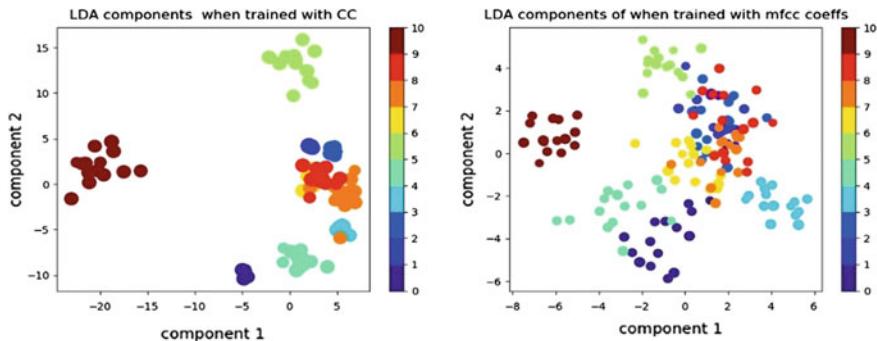


Fig. 7 LDA plots depicting clustering and overlapping of train samples for CC and MFCC

of the best feature in next experiment, number of the instrument has been increased to ten in order to observe differences.

Experiment 3:

The third experiment is carried out with ten instruments which are organ, French horn, cello clarinet, tambourine drum flute, trumpet, violin, and piano. For each instrument or class, 15 samples were taken during training and 3 samples were taken for testing, so a total of 150 samples were taken during train phase and 30 samples were used in the test phase to check the accuracy and robustness of the system. The pitch and dynamics are random. The result is graphically plotted in Fig. 6.

As observed from this plot, the maximum accuracy is obtained for CC with ANN. The highest number of the correctly labeled test sample is 27 with CC, while it is 25 for MFCC. The other features have much lesser accuracy. This shows that the accuracy was maximum in case of CC which is followed by MFCC when compared with ANN and K-NN. The LDA plots used for further study are given in Fig. 7.

Some overlapping is observed for MFCC, and the cluster boundary is not well defined so there are chances of false mismatch. Classes 9, 1, 2, and 8 are overlapping quite a lot (in case of MFCC) and hence with an increase in a number of instruments accuracy will fall much steeply. Clearly, CC is the best feature as the overlapping is

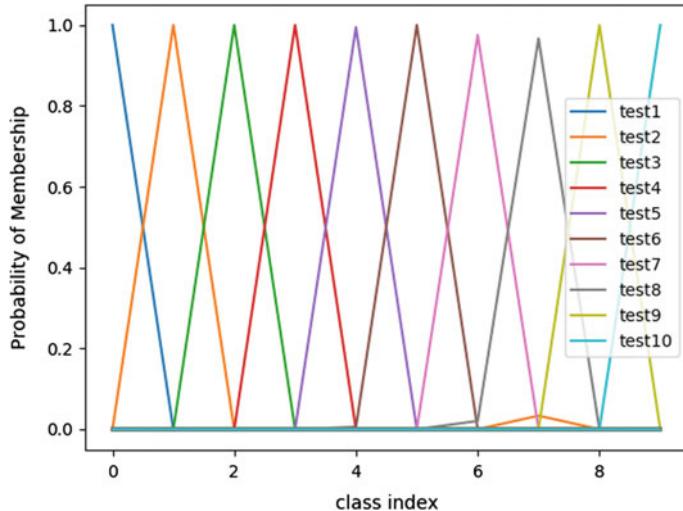


Fig. 8 Classification plot for test samples

much less compared to MFCC and the clusters are tight and well defined. Only classes 9 and 8 have some overlapping; otherwise, other classes are well distinguished. So, the accuracy will fall much less steeply when compared to MFCC and it has the potential for better results. Hence, observing the above results CC is chosen as the final feature and ANN as the final classifier for this set provided best accuracy and the feature is robust as can be observed from LDA plots and accuracy with an increase in train data decreases less steeply than MFCC which provided second best result in terms of accuracy. The test samples are classified into the class with which it received the highest probability of likeliness. The plot below shows how each of ten test samples belonging to ten different classes are correctly classified based on the maximum probability obtained when cepstral coefficients are used as features and ANN is used as a classifier (Fig. 8).

It can be observed that the test samples are matched to the class with which it has maximum probability. In the above case, the accuracy obtained is 100% and hence each test class is classified into its respective train classes.

5 Analysis

The proposed approach uses cepstral coefficients as features and ANN as the classifier for the musical instrument classification task. The overall accuracy obtained in experiment 1 and 2 is 14/15 as 14 test samples from a total of 15 samples are correctly classified. Hence, the accuracy is given by Eq. (12)

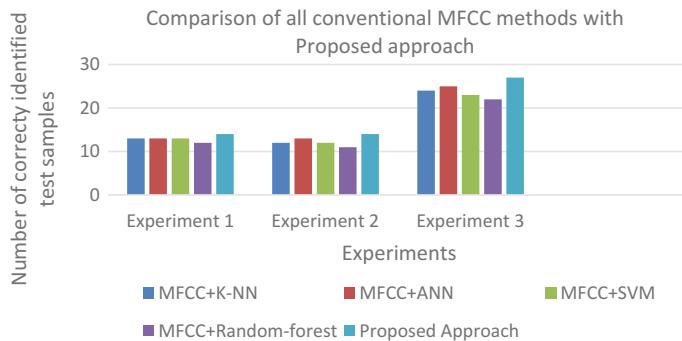
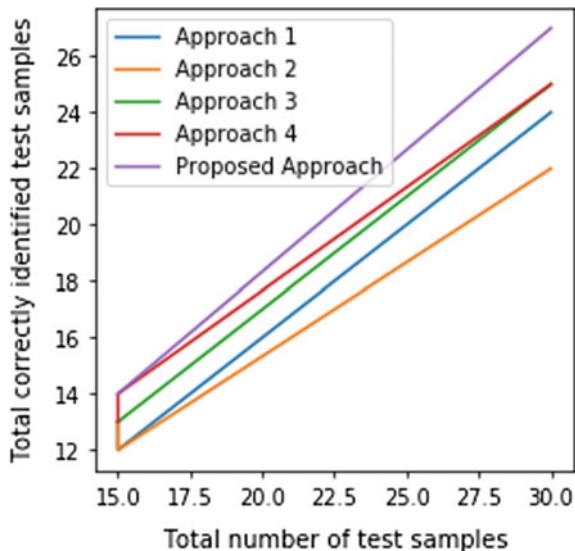


Fig. 9 Comparison of all conventional MFCC methods with proposed approach

Fig. 10 Comparison of earlier approaches with proposed approach



$$\text{Accuracy} = \frac{\text{Number of correctly classified test samples}}{\text{Total number of test samples}} \times 100 \quad (12)$$

Hence, in the first and second case, overall accuracy is 93% (14/15) while in third case 27 out of 30 samples are correctly classified, and hence overall accuracy is 90% for the proposed approach. For MFCC, in first two cases, overall accuracy is 86% as 13 out of 15 test samples are correctly classified, while in third case it is 83% as 25 out of 30 samples were correctly classified by it. The accuracy of proposed approach is larger than all other combinations of MFCC (Fig. 10).

The above plot shows the comparison of earlier approaches [1, 4, 9, 10] with the proposed approach for three experiments. The proposed method obtained with CC and ANN has the following advantages over other methods.

1. The accuracy obtained with MFCC across all classifiers used in related papers and timbre related other feature used in [10] was less than the accuracy obtained with the proposed approach [1, 4, 9] (Fig. 9).
2. ANN is chosen as final classifier as with better tuning, improvement of accuracy is possible. K-NN can be used as an alternative to avoid tuning.
3. The rate of decrease of accuracy with CC is much less compared to MFCC, as observed from graph and LDA plots. Hence, this limitation is somewhat reduced [1, 4].
4. The feature is robust as across three experiments in varied conditions, it provided least overlapping of train samples and maximum accuracy of test samples.
5. For CNN or convolutional neural network [8], the number of train and test data required is very high. The method will fail in situations where there is a limited number of data. This is not the problem with the proposed approach.
6. CC is the best classifier for musical instrument classification among LPC, HPCP, MFCC, pitch salience, and centroid as observed from experimental results and LDA plots.

6 Conclusions and Future Scopes

It is hence concluded that the proposed approach is not only capable of classifying ten aforementioned musical instruments of five different musical families better than conventional approaches dealing with MFCC but also better than other features like LPC coefficients, HPCP, pitch salience peaks, and spectral centroids. It can work in situations where data is limited unlike CNN-based approaches and the decrease of accuracy with an increase in the number of instruments is less than all other features mentioned above. The approach is robust to changes in pitch and dynamics of musical instruments. The limitations of this method are that it cannot distinguish instruments playing in polyphonic audio which will be dealt with in future.

References

1. M.S. Nagawade, V.R. Ratnaparkhe, Musical instrument identification using MFCC, in *2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, Bangalore (2017), pp. 2198–2202
2. S. Essid, G. Richard, B. David, Instrument recognition in polyphonic music based on automatic taxonomies. *IEEE Trans. Audio Speech Lang. Process.* **14**(1), 68–80 (2006)
3. S.D. Patil, P.S. Sanjekar, Musical instrument identification using SVM, MLP & AdaBoost with formal concept analysis, in *2017 1st International Conference on Intelligent Systems and Information Management (ICISIM)*, Aurangabad (2017), pp. 105–109
4. P.S. Jadhav, Classification of musical instruments sounds by using MFCC and timbral audio descriptors. *Int. J. Recent Innov. Trends Comput. Commun. (IJRITCC)* **3**(7), 5001–5006 (2015)
5. M. Muller, D.P.W. Ellis, A. Klapuri, G. Richard, Signal processing for music analysis. *IEEE J. Sel. Top. Signal Process.* **5**(6), 1088–1110 (2011)

6. F.H. Foomany, K. Umapathy, Classification of music instruments using wavelet-based time-scale features, in *2013 IEEE International Conference on Multimedia and Expo Workshops (ICMEW)*, San Jose, CA (2013), pp. 1–4
7. M. Abulaish, *Ontology Engineering for Imprecise Knowledge Management* (Lambert Academic Publishing, Saarbrucken, Germany, 2009)
8. Y. Han, J. Kim, K. Lee, Deep convolutional neural networks for predominant instrument recognition in polyphonic music. *IEEE/ACM Trans. Audio Speech Lang. Process.* **25**(1), 208–221 (2017)
9. H. Mukherjee, S.M. Obaidullah, S. Phadikar, K. Roy, SMIL—a musical instrument identification system, in *Computational Intelligence, Communications, and Business Analytics. CICBA 2017. Communications in Computer and Information Science*, ed. by J. Mandal, P. Dutta, S. Mukhopadhyay, vol. 775 (Springer, Singapore, 2017)
10. P.J. Donnelly, J.W. Sheppard, Classification of musical timbre using Bayesian networks. *Comput. Music J.* **37**(4), 70–86 (2013)
11. S.K. Banchhor, A. Khan, Musical instrument recognition using zero crossing rate and short-time energy. *Int. J. Appl. Inf. Syst. (IJAIS)* **1**(3), 16–19 (2012)
12. J.L.C. Loong, K.S. Subari, M.K. Abdullah, N.N. Ahmad, R. Besar, Comparison of MFCC and cepstral coefficients as a feature set for PCG biometric systems. *World Acad. Sci. Eng. Technol. Int. J. Biomed. Biol. Eng.* **4**(8) (2010)
13. Philharmonia Orchestra Sound Samples, www.philharmonia.co.uk/explore/sound_samples
14. Freesound, <http://freesound.org/>

A Factual Analysis of Improved Python Implementation of Apriori Algorithm



Kartick Chandra Mondal, Biswadeep Deb Nandy and Arunima Baidya

1 Introduction

Data mining, also known as Knowledge Discovery in Databases (KDD), includes the task to find anomalies, correlations, patterns, and trends to predict outcomes [1, 2]. Association rule mining is one of the most prominent data mining tasks along with classification and clustering is gaining much importance in recent years many application domains. In general, the KDD is a sequence of processes stated [3] as follows:

- Data cleaning which includes the removal of noise and inconsistency from the data.
- Data integration, where multiple data sources are combined and integrated into one.
- Data selection, where data relevant to analysis task are retrieved.
- Data transformation, in which data is transformed into forms appropriate for mining by performing several aggregation operations.
- Data mining, which includes intelligent methods to extract various patterns in data.
- Pattern evaluation, where the various patterns are evaluated and the ones truly representing the knowledge are identified.
- Knowledge representation, including techniques to represent the knowledge.

Association Rule Mining (ARM) is a prominent and a well-explored method for determining relations among variables in large databases. ARM is a very popular

K. C. Mondal (✉) · B. D. Nandy · A. Baidya
Department of Information Technology, Jadavpur University, Kolkata, India
e-mail: kartickjgec@gmail.com

B. D. Nandy
e-mail: biswadeep1997@gmail.com

A. Baidya
e-mail: arunimabaidya@gmail.com

and important, but expensive, task in data mining [4]. An association rule can be represented in the form of Antecedent \Rightarrow Consequent [support, confidence], where support and confidence are the interestingness measure of the rules. The technique can be used in many data types like sequential data, spatial data, and multimedia data along with transactional data. In today's world, ARM is used in various fields like Telecommunication, Weather Analysis, Banking Sectors, etc.

The Apriori algorithm is the first attempt for mining association rule by finding frequent itemsets from the dataset. The main idea behind this algorithm is to prune the exponential search space using anti-monotonicity. As the name suggests, the algorithm uses prior knowledge (of the support properties) to prune the search space. It is devised to operate on a database containing a lot of transactions. It uses a “bottom-up” approach where frequent subsets are extended by one item at a time. It is generally used to operate on transactions containing data [5]. This algorithm has several real-world applications such as [6] market analysis to help the customers in purchasing their items with more ease which increases the sales of the markets or the field of health care helps in detecting Adverse Drug Reactions (ADRs). It produces association rules that indicate all combinations of medications and patient characteristics lead to ADRs.

In this paper, we have presented a comparison and analyzed some implementations of Apriori algorithm in Python as well as our own implementation of the same. The rest of the paper is organized as follows. Some brief terminologies are described in Sect. 2. In Sect. 3, we have presented a very brief explanation of the Apriori algorithm. Section 4 gives a brief study about different Python packages implementing the algorithm. This section also presents the detail of our own implementation of the algorithm. Section 5 includes the various parameters on which we have compared different python packages. The experimental results are shown in Sect. 5, while Sect. 6 contains the concluding remarks.

2 Terminologies

Definition 1 (*Database*) A database D is defined as (I, T) where I refers to the transaction ids and T refers to the set of transactions.

Definition 2 (*Transaction*) A transaction T is defined as a set of items (known as itemset), where each item is denoted by i.

Definition 3 (*Itemset*) A non-empty finite set of items I where $I \subseteq T$ in D.

Definition 4 (*Support*) The support of an itemset Q, denoted by $sup(Q)$, is defined as the frequency of occurrence of Q in D.

$$sup(Q) = (\{q \in T; X \subseteq q\}) / |T|$$

Definition 5 (Frequent Itemset) An itemset Q with support equal to or more than the user-defined threshold of support (minsup) is called a frequent itemset, i.e.,

$$Q \subseteq T \text{ is frequent iff } \text{sup}(Q) \geq \text{minsup}.$$

Definition 6 (Association Rule) The relationship between two itemsets I1 and I2, $I1 \rightarrow I2$, where $I1, I2 \subset L$ and $I1 \cap I2 = \emptyset$ is called an association rule. I1 is known as antecedent of the rule, while I2 is known as consequent of the rule.

Definition 7 (Confidence) The confidence value of a rule $I1 \rightarrow I2$ is the proportion of transactions T that contains I1 as well as I2. It is denoted by $\text{conf}(I1 \rightarrow I2)$ where

$$\text{conf}(I1 \rightarrow I2) = \text{supp}(I1 \cup I2) / \text{supp}(I1)$$

3 Apriori Algorithm

Apriori algorithm was proposed by Agrawal and Srikant in 1993 [7]. It is used originally for mining frequent itemsets in Boolean association rules from transaction dataset [8]. The Apriori algorithm helps in pruning the number of itemsets that are infrequent which needs to be examined because, as a principle, if an itemset is infrequent then all its subsets must also be infrequent.

The algorithm is comprised of the following steps [6]:

- Step 0: Take the minsup and the itemsets from the user. Let us consider that the minsup is 60% and the transaction database as it is shown in Table 1.
- Step 1: Calculate the support of all items. A frequency table of all the items that occur in the transactions is made as shown in Table 2. In this table, each itemset is annotated with its support. The support of an itemset is basically its frequency, i.e., how many times the itemset appears in the transaction database. Here, the itemset {9, 40} has support value 4 because it appears in transactions i2, i3, i4, and i5.
- Step 2: We know that only those elements are significant for which the support is greater than or equal to the threshold support (minsup). So, we discard those items whose support is less than minsup. The final result with the significant itemsets and their support is denoted in Table 3.

Table 1 Data containing all transactions

Transaction id	Items
i1	7, 14, 18
i2	9, 14, 40
i3	7, 9, 14, 40
i4	9, 40

Table 2 The intermediate table containing all itemsets and their respective supports

Itemsets	Support	Itemsets	Support
7	3	9, 40	4
14	4	14, 40	3
18	1	9, 14, 40	3
7, 14	3	7, 9	2
7, 18	1	7, 40	2
14, 18	1	7, 9, 14	2
7, 14, 18	1	7, 9, 40	2
9	4	7, 14, 40	2
40	4	7, 9, 14, 40	2
9, 14	3		

Table 3 Final result after running Apriori algorithm on the dataset

Itemsets	Support
7	3
14	4
7, 14	3
9	4
40	4

The diagrammatic representation of the functional flow of the algorithm [9] is provided in Fig. 1.

4 Python Implementations of Apriori Algorithm

In this section, we have provided an outline of various packages that are implemented and widely used in Python which have implemented the Apriori algorithm. At the end of this section, we have also explained our implementation of the Apriori algorithm.

4.1 Explanation of Existing Packages

1. **akapriori 0.1.0:** It works for both Python 2.x and 3.x. It does not support file input. Input is given by a program consisting of a list of transactions, and this input is passed to the package. The resultant data is contained in a frozen set corresponding to their support and confidence.
2. **apriori-1.0.0:** It imports numpy package for faster computation. Input data is set and returned from function loadDataSet(). Mapping is done between frozen set and Ck, candidate k-tuples. It returns candidate generation with support in list

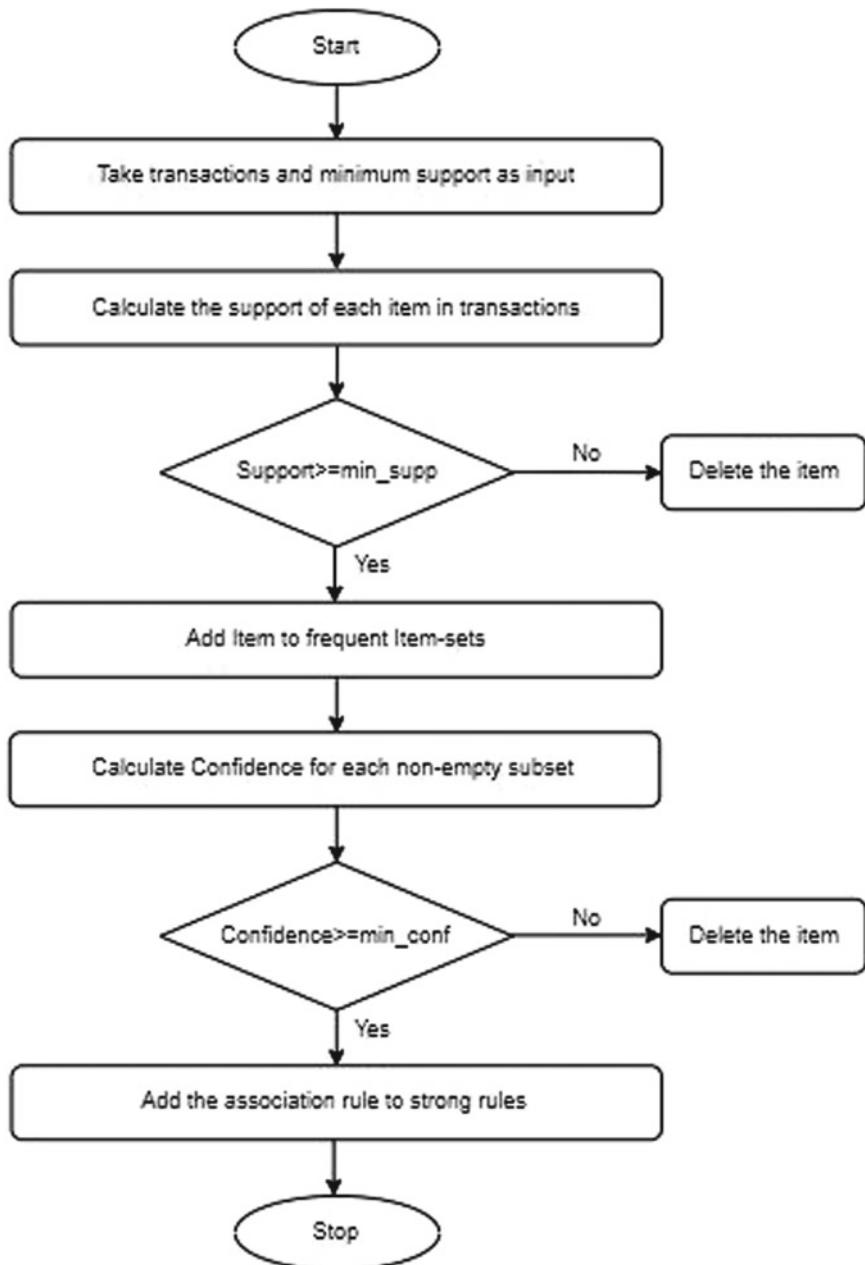


Fig. 1 Flowchart of the Apriori algorithm

Lk, and association rules are also generated by calculating the confidence and pruning of rule is done by comparing with the minimum confidence.

3. **apyori-1.1.1:** It imports json and csv packages. It is an implementation of Apriori algorithm with Python 2.7 and 3.3–3.5, provided both as APIs and as command line interfaces. Input to this implementation is a .tsv file (transaction values are tab separated). Output can be generated in.tsv or .json file format.

Basic usage: `python apyori.py < test.tsv`

TSV Output: `python apyori.py -f tsv < test.tsv`

Specify details: `python apyori.py -s 0.5 -c 0.5 < test.tsv`

4. **myapriori-0.0.22:** It imports apriori functionalities from the apyori package. In addition to this, it requires the support of numpy as well as pandas library. Transaction data can be passed as a pandas DataFrame or list of lists.

4.2 *Explanation of Our Package*

The originality of our implementation is that no external libraries have been imported or used in it. Input data is passed in the form of a file consisting space-separated values as transaction. We have done our implementation in Python 3.6.5. Command line input pattern for running our implementation is `python 3 apriori.py -f "path/filename" -s "minsupport" -l "number of transactions"`.

Eg: `python 3 apriori.py -f datasets/set1.dat -s 0.2 -l 100`

Default value of minsupport is set to 0.25, and the default value of the number of transactions is set to –1. We have created a class Apriori, and we have defined the following functions in it:

- `createC1(self)`: This function creates C1 candidate k-tuples.
- `filterCk(self, Ck)`: This function filters Ck candidate tuples and returns Lk and its supports.
- `createCk(self, Lk, k)`: This function creates a set of Ck candidate k-tuples.

The output generated is a list of frequent itemsets. Figure 2 shows the activity diagram of our implementation.

4.3 *Comparison Between Packages*

Following are the characteristics which we have used for comparison on various packages:

1. **Imports:** These depict the various Python libraries used by the apriori packages.
2. **Input pattern:** This denotes the type of input data that the program requires.
3. **Data Structures:** This signifies the various data structures used in different packages.

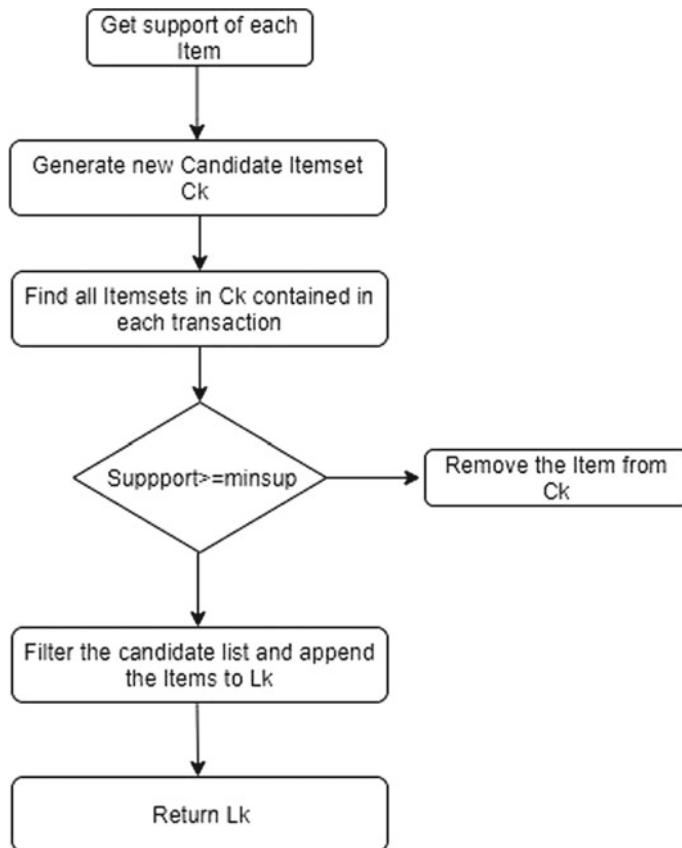


Fig. 2 Activity diagram of our Apriori implementation

4. **Time Complexity:** It refers to the implementation of complexity that estimates the time taken for an algorithm to run for that particular implementation.
5. **Space Complexity:** Space complexity of an algorithm is total space taken by the implementation with respect to the input size. Space complexity includes both auxiliary space and space used by input.

For the time and space complexities, we have studied different implementations and used Big O notation to show the analysis of the same. Table 4 given below shows the comparison between the different packages, according to their features mentioned above.

Table 4 Comparison of various packages on the basis of features where d = unique items in entire set of transactions

Packages	Imports	Input pattern	Data structures	Time	Space
akapriori 0.1.0	Collection	List of sets	Defaultdict, frozenset, set, list	$O(n^2)$	$O(n^2)$
apriori-1.0.0	numpy	List of lists	List, dictionary	$O(n^3(\log n))$	$O(n^2)$
apyori-1.1.1	Collection	Dictionary	List, dictionary, set, frozenset, namedtuple	$O(n^d)$	$O(n^2)$
myapriori- 0.0.22	apyori	Type flexible	List	$O(n^d)$	$O(n^2)$
Our imple- mentation	Collection	List of lists	List	$O(n^2)$	$O(n^2)$

5 Experiments and Critical Analysis

In this section, we have described our experimental setup, the kinds of dataset we have used as well as the statistical and experimental results of our experiment.

5.1 Experimental Setup

This sort of experimental study can be very sensitive to hardware configuration and the possibility of other programs or tasks running at the same time in a multiprocessing and multiprogramming environment. So, we have used a single machine for carrying out all the experiments, whose configuration is stated as follows:

- **Processor:** 6th Generation Intel Core i5-6 300HQ Quad Core (6M Cache, up to 3.2 GHz)
- **Memory:** 8 GB 1 DIMM (1×8 GB) DDR3L 1600 Mhz
- **Disk/Hard Drive:** 1 TB (5400 rpm) Hybrid HDD with 8 GB Flash
- **System Type:** 64 bit Operating System, $\times 64$ -based Processor
- **Operating System:** Windows 10 Home Single Language

5.2 Dataset Preparation

Conceptually, there are two types of databases, namely, dense and sparse. Sparse databases refer to the databases where most of the cells are filled with null values [10].

Table 5 Table signifying the features of datasets used for experiments

Set no.	Minimum # of items in a transaction	Maximum # of items in a transaction	Total # of distinct items
Set 1	2	4	5
Set 2	2	7	19
Set 3	2	10	101
Set 4	1	9	869

Dense databases, on the other hand, have fewer null values and are filled with very useful and significant information. According to many studies done in literature, the Apriori algorithm is applied to a dense dataset. Its performance declines because of the long patterns that emerge. When the parameters are low and the database is dense, it searches through all possible itemset combination and becomes computationally infeasible. So, for our experiments, we have considered sparse databases [6, 11, 12].

We have divided our data into four sets with respect to the number of transactions. Set 1, Set 2, Set 3, and Set 4 are comprised of a set of 5, 10, 100, and 1000 transactions, respectively. Additional features about the datasets are given in Table 5. To check the performances of the packages, we have considered three different minsup values which are 20%, 40%, and 60%, respectively.

5.3 Result Generation

We have generated the results by investigating the total runtime and the memory consumed by the different implementations where *Runtime* denotes the total time for which a program runs and *Memory consumption* signifies the amount of memory used by the program while running. Now, we have considered these two parameters for our experimental study as these two are vital benchmarks for stating the efficiency of any algorithm. The complete experimental result sets are shown in Tables 6 and 7.

5.4 Result Analysis

In this paper, we deal with comparison between the different packages of Apriori algorithm. The analysis of results generated from above experiments for different packages is given below on different parameters.

Runtime: Figure 3 showcases the runtime (in seconds) for four different datasets of data (Set 1 (Fig. 3a), Set 2 (Fig. 3b), Set 3 (Fig. 3c), and Set 4 (Fig. 3d), respectively) for different minsup values (20%, 40%, and 60%, respectively). The data

Table 6 Runtime (in seconds) for four different datasets for different minsup values (20%, 40%, and 60%, respectively)

Packages datasets	Minsup (in %)	akapriori-0.1.0	apriori-1.0.0	apyori-1.1.1	myapriori-0.0.22	Our imple-mentation
Set 1	20	0.001505	0.003071	0.015994	0.019281	0.006554
	40	0.001503	0.002618	0.012031	0.018418	0.006017
	60	0.002023	0.000789	0.011034	0.017692	0.007064
Set 2	20	0.002007	0.001162	0.017602	0.026016	0.006023
	40	0.002006	0.000202	0.013995	0.025351	0.006016
	60	0.002005	0.000187	0.011017	0.019815	0.005982
Set 3	20	0.004018	0.002013	0.018584	0.032692	0.013034
	40	0.003973	0.001904	0.015036	0.028407	0.013579
	60	0.003008	0.001919	0.013035	0.025493	0.011539
Set 4	20	0.297486	0.128429	0.321865	0.475892	0.466642
	40	0.180798	0.125685	0.315902	0.454569	0.448878
	60	0.180479	0.131158	0.202961	0.382356	0.342496

Table 7 Memory requirement (in bytes) for four different datasets for different minsup values (20%, 40%, and 60%, respectively)

Packages datasets	Minsup (in %)	akapriori-0.1.0	apriori-1.0.0	apyori-1.1.1	myapriori-0.0.22	Our imple-mentation
Set 1	20	14,454,784	15,425,536	15,454,208	15,734,314	15,114,240
	40	14,401,536	15,560,704	15,417,344	15,685,931	15,081,472
	60	14,352,384	15,589,376	15,433,728	15,689,892	15,101,952
Set 2	20	14,430,208	15,491,072	15,470,592	15,891,652	15,110,144
	40	14,508,032	15,560,704	15,470,592	15,886,073	15,044,608
	60	14,393,344	15,429,632	15,482,880	15,797,435	15,077,376
Set 3	20	14,587,552	15,474,688	15,566,496	15,725,689	15,159,296
	40	14,528,512	15,572,992	15,540,224	15,725,645	15,134,720
	60	14,516,224	15,388,672	15,450,112	15,656,892	15,073,280
Set 4	20	15,609,856	17,137,664	15,765,504	18,561,245	16,764,928
	40	15,577,088	16,879,616	15,536,128	17,959,871	16,633,856
	60	15,458,304	16,637,952	15,716,352	17,936,814	16,478,208

corresponding to minsup value = 60% is marked with gray, minsup value = 40% is marked with dark gray, and the one with minsup value = 20% is marked with black.

Memory Consumption: Figure 4 showcases the memory consumption (in bytes) for four different datasets of data (Set 1 (Fig. 4a), Set 2 (Fig. 4b), Set 3 (Fig. 4c), and Set 4 (Fig. 4d), respectively) for different minsup values (20%, 40%, and 60%, respectively). The data corresponding to minsup value 60% is marked with gray,

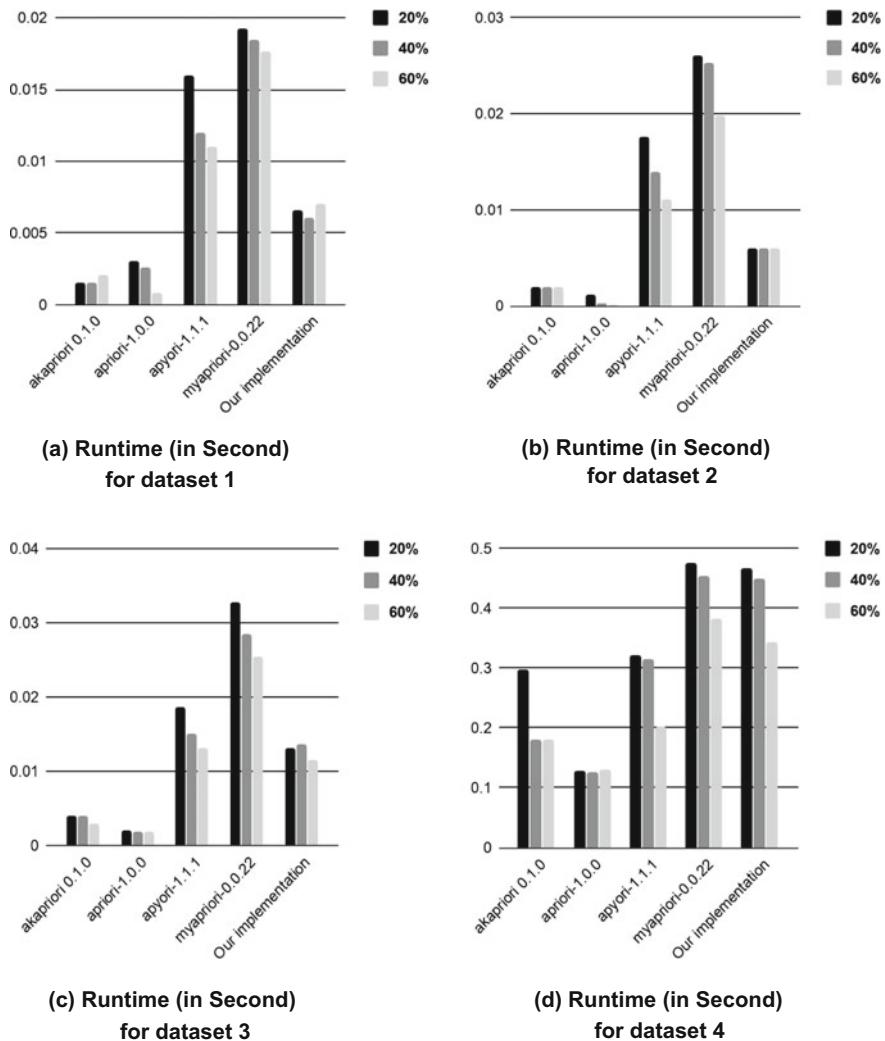


Fig. 3 Runtime analysis of different implementations of Apriori Algorithm

minsup value 40% is marked with dark gray, and the one with minsup value 20% is marked with black.

Akapriori implements candidate item pruning using the help of numpy package. From the resultant graphs, it is clear that for large datasets, packages apriori and akapriori are almost at par.

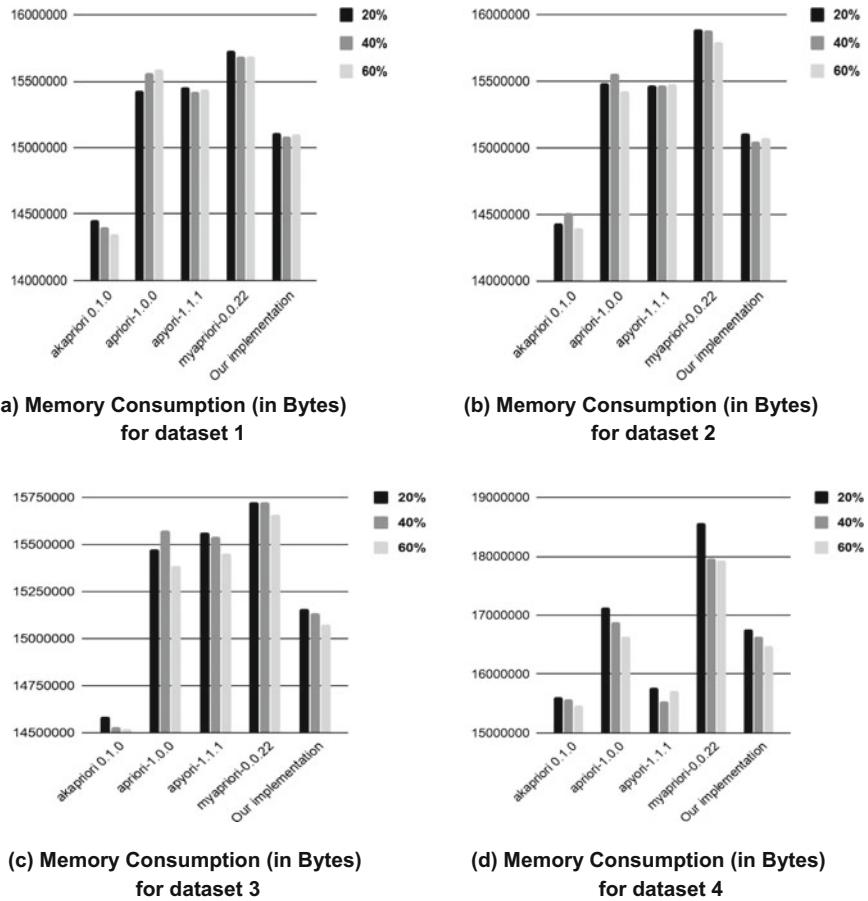


Fig. 4 Memory consumption analysis of different implementations of Apriori algorithms

6 Conclusion

Apriori algorithm is the first try to generate association rule mining. Different versions of Apriori algorithm have been developed but no comparison is available with different well-known implementations of the same algorithm from performance perspective. We have done an empirical analysis with different algorithmic implementations of the Apriori algorithm done in Python language. This study will help the researchers working in this domain to choose appropriate version of the algorithm for experiment. With respect to our experimental evaluation as stated in this paper, we conclude that the same algorithm might have different performances for different datasets and minimum supports. The lower the minimum support, the higher is the time and memory consumption. We see clearly how different implementations

perform with respect to time and memory consumption. From our study, *akapriori* turns out to be the best package in terms of performance, with our implementation coming as a close second in terms of memory and third in terms of time. In addition to this, we have also found that the performance of the Apriori algorithm is mainly dependent on the data structures used. We would like to add that our package has not used any present Python libraries, so there is much scope for improvement in future using techniques like data pruning and dynamic itemset counting.

References

1. J. Han, M. Kamber, *Data Mining: Concepts and Techniques* (Morgan Kauffman, San Francisco, 2000)
2. S. Jain, R. Raghuvanshi, M. Ilyas, A survey paper on overview of basic data mining tasks. *Int. J. Innov. Adv. Comput. Sci.* **6**(9) (2017)
3. A. Silberschatz, A. Tuzhilin, What makes patterns interesting in knowledge discovery systems. *IEEE Trans. Knowl. Data Eng.* **8**(6), 970–974 (1996)
4. J. Hipp, U. Güntzer, G. Nakhaeizadeh, Algorithms for association rule mining—a general survey and comparison. *SIGKDD Explor. Newsl.* **2**(1), 58–64 (2000)
5. C. Borgelt, R. Kruse, Induction of association rules: apriori implementation, in *Compstat. Physica*, ed. by W. Härdle, B. Rönz (Heidelberg, 2002)
6. M. Hegland, The apriori algorithm—a tutorial. *Math. Comput. Imaging Sci. Inf. Process.* **11**, 209–262 (2007). World Scientific Publishing Co.
7. R. Agrawal, R. Srikant, Fast algorithms for mining association rules, in *Proceedings of the 20th VLDB Conference* (1994), pp. 487–499
8. <https://www.philippe-fournier-viger.com/spmf/Apriori.php>
9. M. Mittal, *Efficient Ordering Policy for Imperfect Quality Items Using Association Rule Mining*, vol. 14 (2014)
10. F. Ye, J. Wang, B. Shao, New algorithm for mining frequent itemsets in sparse database, in *International Conference on Machine Learning and Cybernetics*, vol. 3 (Guangzhou, China, 2005), pp. 1554–1558
11. <https://wiki.csc.calpoly.edu/datasets/wiki/apriori>
12. <http://fimi.ua.ac.be/data/>

User Authentication with Session Key Interchange for Wireless Sensor Network



Prasanta Kumar Roy, Krittibas Parai and Abul Hasnat

1 Introduction

Due to the ubiquitous WSN, a wide range of low-power real-time applications such as monitoring and controlling health care, vehicular tracking, mobile roaming services, anonymous web browsing, etc. are feasible. However, users can be benefitted with these services by means of wireless media. Hence, achieving a secure authentication scheme is a major requirement to deal with [1]. Typically, three different parties are involved in this regard: User (*USR*), sensor node (*SR*), and gateway node (*GTW*). Initially, each new *USR* makes a registration to *GTW* and obtains a smart card (*SC*) containing the login information. Then, mutual authentication with session key agreement will take place between *USR* and *SR* via *GTW*.

Recently, several user authentication schemes for WSN have been suggested by many researchers. Unfortunately, some of them still suffer from several security drawbacks. Wong et al. [2], in 2006, proposed their scheme based on dynamic strong password. However, in 2009, Das [3] claimed that Wong et al. protocol cannot resist replay attack, stolen verifier attack, and forgery attack. They also proposed an enhanced two-factor authentication scheme to provide strong authentication and proper session key agreement. Chen and Shih [4], in 2010, found that Das et al. scheme cannot achieve proper mutual authentication and vulnerable to parallel session attack. They also proposed an enhancement to overcome these limitations. In

P. K. Roy · K. Parai (✉)
Siliguri Institute of Technology, Siliguri 734009, India
e-mail: krittibas.sit@gmail.com

P. K. Roy
e-mail: prasanta201284@gmail.com

A. Hasnat
Government College of Engineering & Textile Technology, Berhampore 742101, India
e-mail: abulhasnat@gmail.com

2011, Yeh et al. [5] found that Chen et al. scheme cannot withstand insider attack and suffers from insecure password upgrade phase. They also claimed that Das et al. scheme failed to resist insider attack and forgery attack, and proposed an ECC-based enhancement. However, Yeh et al. protocol failed to achieve perfect forward secrecy, proper mutual authentication, and proper session key agreement. In 2013, Shi and Gong [6] suggested their protocol to overcome the limitations of Yeh et al. In 2014, Choi et al. [7] found that Shi et al. protocol cannot resist stolen smart card attack and session key-compromised attack. They also proposed an ECC-based enhancement against these limitations. Recently, Nam et al. [8] proposed an ECC-based authentication protocol for WSN to achieve authenticated key agreement and user anonymity. Besides all these, a number of authentication protocols still exist in recent research.

Nevertheless, in this manuscript, several disadvantages of Nam et al. [8] protocol are highlighted. Also, we propose an efficient two-factor ASKI-WSN scheme based on ECC to overcome these limitations. The proposed protocol is constructed using lightweight operations to make it more suitable for low-power practical applications of WSN with considerable computation cost.

The rest of this manuscript is organized as follows. Section 2 provides a brief overview of Nam et al. protocol. The cryptanalysis of Nam et al. protocol is demonstrated in Sect. 3. The proposed ASKI-WSN scheme is described in Sect. 4 considering three different phases, namely, registration phase, login and mutual authentication phase, and password update phase. The resistance against several well-known security threats and overall computation cost evaluation of our scheme are described in Sect. 5. Section 6 demonstrates the formal security analysis of our protocol using AVISPA tool. Finally, we conclude our work in Sect. 7.

2 Review of Nam et al. Protocol

Nam et al. [8] protocol is based on ECC and consists of three different phases: registration phase, authentication and key exchange phase, and password update phase. During initialization, GTW announces its public key $X = x \cdot P$ and secretly shares $k_{GS} = L(ID_{SR}||y)$ with SR , where $x, y \in Z_q^*$ are the master secret keys of GTW . The notations used throughout this manuscript are described in Table 1.

2.1 Registration Phase

Initially, a new USR submits its ID_{USR} to GTW via a secure channel (in offline mode). Then, GTW computes $SID_{USR} = En_{L(x)}(ID_{USR}||ID_{GTW})$ and assigns a new SC containing the parameters $\{SID_{USR}, X, ID_{GTW}, G, P, \Gamma, \Delta, L(\cdot), H(\cdot), F(\cdot)\}$ to USR . USR calculates $TID_{USR} = SID_{USR} \oplus F(ID_{USR}||pw_{USR})$ and inserts TID_{USR} into SC replacing SID_{USR} .

Table 1 Notations used

Notations	Description
$ID_{USR}, ID_{SR}, ID_{GTW}$	Identity of USR, SR, GTW , respectively
pw_{USR}	USR 's password
sk	Session key between USR and SR
\mathcal{A}	The adversary
$L(\cdot), H(\cdot), F(\cdot)$	Collision-resistant one-way hash function
$En_k(\cdot)/De_k(\cdot)$	Keyed encryption/decryption function with key k
Mc	Message authentication code
$Mc_k(\cdot)/Vr_k(\cdot)$	Mc generation/verification with key k
$Vr(\cdot)$	Keyless verification function
T_1, T_2	Timestamps
\oplus	Bitwise exclusive-OR operation
\parallel	Concatenation operation
Γ	Pair of $\{En_k(\cdot), De_k(\cdot)\}$ functions
Δ	Pair of $\{Mc_k(\cdot), Vr_k(\cdot)\}$ functions
$?$	Equality check operator
$=$	
G	Elliptic curve group of prime order q
P	Point on elliptic curve
$a, b, c, x, y \in Z_q^*$	Freshly generated random integers

2.2 Login and Authentication with Key Exchange Phase

To access the useful sensor data, USR needs to generate a valid login request. To do this, USR inserts his SC into a suitable card reader and inputs ID_{USR} and pw_{USR} . Then, SC generates T_1 , and computes $A = a \cdot P$, $K_{UG} = a \cdot X$, $k_{UG} = L(T_1||A||K_{UG})$, $SID_{USR} = TID_{USR} \oplus F(ID_{USR}||pw_{USR})$ and $C_{USR} = En_{k_{UG}}(SID_{USR}||ID_{USR}||ID_{SR})$. Then, SC sends the message $M_1 = \{ID_{GTW}, T_1, A, C_{USR}\}$ to SR via public channel. Upon receiving M_1 , SR verifies the freshness of received T_1 . If verified successfully, SR generates its timestamp T_2 and computes $\delta_{SR} = Mc_{k_{GS}}(ID_{SR}||T_2||B||M_1)$, where $B = b \cdot P$. Then, SR sends M_1 and $M_2 = \{ID_{SR}, T_2, B, \delta_{SR}\}$ to GTW via insecure channel. After receiving the message from SR , GTW checks the freshness of T_1 and T_2 . Also, GTW computes $Vr_{k_{GS}}(ID_{SR}||T_2||B||M_1, \delta_{SR}) \stackrel{?}{=} 1$ to verify whether both parameters of $Vr_{k_{GS}}(\cdot)$ function are producing same result (if yes, the outcome produced by $Vr_{k_{GS}}$ will be 1). Upon successful verification, GTW calculates $K_{UG} = x \cdot A$, $k_{UG} = L(T_1||A||K_{UG})$, $De_{k_{UG}}(C_{USR})$, and $De_{L(x)}(SID_{USR})$. GTW again checks the validity of decrypted ID_{USR} and ID_{SR} . Once verified successfully, GTW sends the messages $M_3 = \{ID_{GTW}, ID_{SR}, B, \delta_{GTW}^{USR}\}$

and $M_4 = \{ID_{GTW}, ID_{SR}, A, \delta_{GTW}^{USR}\}$ to SR via insecure channel, where $\delta_{GTW}^{USR} = Mc_{k_{UG}}(ID_{GTW}||ID_{SR}||A||B)$ and $\delta_{GTW}^{SR} = Mc_{k_{GS}}(ID_{GTW}||ID_{SR}||B||A)$. Upon receiving the message $\{M_3, M_4\}$, SR verifies whether $Vr_{k_{GS}}(ID_{GTW}||ID_{SR}||B||A, \delta_{GTW}^{SR}) \stackrel{?}{=} 1$ holds. If verified, SR computes $K_{SU} = b \cdot A$ and $sk = H(A||B||K_{SU})$. Then, SR forwards the message M_3 to USR via public channel. On receipt of M_3 , USR validates $Vr_{k_{UG}}(ID_{GTW}||ID_{SR}||A||B, \delta_{GTW}^{USR}) \stackrel{?}{=} 1$ and computes $K_{SU} = a \cdot B$ and the shared session key $sk = H(A||B||K_{SU})$.

2.3 Password Upgrade Phase

For security reasons, USR needs to change his password after certain interval of time. To do this, USR inserts his SC into the card reader and inputs ID_{USR}, pw_{USR} and newly chosen password pw'_{USR} . Subsequently, SC computes $TID'_{USR} = TID_{USR} \oplus F(ID_{USR}||pw_{USR}) \oplus F(ID_{USR}||pw'_{USR})$. Finally, SC stores TID'_{USR} replacing TID_{USR} .

3 Cryptanalysis of Nam et al

The weaknesses of Nam et al. protocol [8] considering the characteristics of Dolev–Yao threat model [9] are explained as follows.

3.1 Denial of Service

On receipt of login request M_1 , SR checks the freshness of T_1 as $T'_1 - T_1 < t$, where T'_1 is the current timestamp and t is a valid time interval [10]. After successful verification, SR generates M_2 for GTW . Assume \mathcal{A} blocks the incoming login request M_1 for $\geq t$ amount of time, and then forwards it to SR without tampering the message. It is obvious that the freshness checking of T_1 will not hold and USR 's login request will be rejected immediately.

3.2 No Login Phase Verification

To generate a valid login request, USR inputs ID_{USR} and pw_{USR} . Then, SC generates the message M_1 and sends it to SR without verifying the legitimacy of USR . If USR mistakenly provides wrong inputs, still SC will be able to generate a login request

and send it to SR . This may result in extra computation cost and communication overhead.

3.3 Clock Synchronization Problem

The entire protocol of [8] is based on the verification of timestamps of USR and SR . A successful session will take place if and only if all the clocks generating timestamps (maintained by USR , SR , and GTW) are properly synchronized. This may lead to excessive computation due to synchronization problem.

3.4 Improper Mutual Authentication

Proper mutual authentication means both sender and receiver should verify the authenticity of each other [11]. In Nam et al. scheme, after verifying GTW , SR computes sk and forwards the message M_3 to USR . On receipt of M_3 , USR verifies GTW and computes sk . Unfortunately, there is no way defined for USR to check whether SR has generated the same session key sk as he does, or vice versa.

3.5 Key-Compromised Impersonation and Man-in-the-Middle Attack

In this kind of attack, if the long-term secret key of any one of the communicating parties is compromised with \mathcal{A} , he may be able to launch an impersonation attack [12]. Unfortunately, in Nam et al. protocol, if the long-term secret key k_{GS} of SR is compromised with \mathcal{A} , he can easily impersonate USR , SR , and GTW as follows.

- Step 1: \mathcal{A} blocks the message $M_1 = \{ID_{GW}, T_1, A, C_{USR}\}$, where $A = a \cdot P$, $C_{USR} = En_{k_{UG}}(SID_{USR} || ID_{USR} || ID_{SR})$ and $SID_{USR} = En_{L(x)}(ID_{USR} || ID_{GTW})$. Then, \mathcal{A} sends his own message $M_1^* = \{ID_{GTW}, T_1, C, C_{USR}\}$ to SR with $C = c \cdot P$. \mathcal{A} saves M_1 for future reference.
- Step 2: On receipt of M_1^* , SR verifies the freshness of T_1 . If verified, he computes $\delta_{SR} = Mc_{k_{GS}}(ID_{SR} || T_2 || B || M_1^*)$, where $B = b \cdot P$ with $b \in Z_q^*$. Then, SR sends $\{M_1^*, M_2 = \{ID_{SR}, T_2, B, \delta_{SR}\}\}$ to GTW .
- Step 3: \mathcal{A} again intercepts the incoming message from SR and saves M_2 . Then, \mathcal{A} generates its own message $\{M_1, M_2^* = \{ID_{SR}, T_2, C, \delta_{SR}^*\}\}$ and sends it to GTW , where $\delta_{SR}^* = Mc_{k_{GS}}(ID_{SR} || T_2 || C || M_1)$.
- Step 4: On receipt of $\{M_1, M_2^*\}$, GTW validates T_1 and T_2 , and verifies whether $Vr_{k_{GS}}(ID_{SR} || T_2 || C || M_1, \delta_{SR}^*) \stackrel{?}{=} 1$ holds. It is obvious that all the verifi-

cations done by GTW will hold successfully. Then, GTW calculates $K_{UG} = x \cdot A$, $k_{UG} = (T_1||A||K_{UG})$, $De_{k_{UG}}(C_{UR})$ and $De_{L(x)}(SID_{USR})$ successively. GTW again verifies the decrypted ID_{SR} and ID_{USR} . If verified, GTW computes $\delta_{GTW}^{USR} = Mc_{k_{UG}}(ID_{GTW}||ID_{SR}||A||C)$ and $\delta_{GTW}^{SR} = Mc_{k_{GS}}(ID_{GTW}||ID_{SR}||C||A)$. Then, GTW generates the message $\{M_3 = \{ID_{GTW}, ID_{SR}, C, \delta_{GTW}^{USR}\}, M_4 = \{ID_{GW}, ID_{SR}, C, \delta_{GTW}^{SR}\}\}$ and sends it to SR .

- Step 5: \mathcal{A} further intercepts $\{M_3, M_4\}$, and computes $M_3^* = \{ID_{GTW}, ID_{SR}, B, \delta_{GTW}^{USR}\}$ and $M_4^* = \{ID_{GTW}, ID_{SR}, C, \delta_{GTW}^{SR*}\}$, where $\delta_{GTW}^{SR*} = Mc_{k_{GS}}(ID_{GTW}||ID_{SR}||B||C)$. Then, \mathcal{A} sends the message $\{M_3^*, M_4^*\}$ to SR .
- Step 6: SR validates $Vr_{k_{GS}}(ID_{GTW}||ID_{SR}||B||C, \delta_{GTW}^{SR*}) \stackrel{?}{=} 1$, and computes $K_{SU}^1 = b \cdot C = b \cdot c \cdot P$ and the session key $sk_1 = H(C||B||K_{SU}^1)$. Then, SR sends the message $\{M_3^*\}$ to GTW .
- Step 7: \mathcal{A} again blocks M_3^* and sends M_3 to USR .
- Step 8: After receiving M_3 , USR validates $Vr_{k_{UG}}(ID_{GTW}||ID_{SR}||A||C, \delta_{GTW}^{USR}) \stackrel{?}{=} 1$, and computes $K_{SU}^2 = a \cdot C = a \cdot c \cdot P$ and the session key $sk_2 = H(C||B||K_{SU}^2)$.
- Step 9: \mathcal{A} computes $K_{SU}^1 = c \cdot B = b \cdot c \cdot P$ and $K_{SU}^2 = c \cdot A = a \cdot c \cdot P$, and finally, the session keys $sk_1 = H(C||B||K_{SU}^1)$ and $sk_2 = H(C||B||K_{SU}^2)$.

Hence, \mathcal{A} can successfully impersonate USR , SR and GTW . He will also be able to create two different sessions using sk_1 and sk_2 with USR and SR , respectively. Thus, Nam et al. protocol [8] is vulnerable to key-compromised impersonation and man-in-the-middle attack.

3.6 Unverified Password Update Phase

For security measures, it is important to update USR 's password frequently. In Nam et al. protocol, USR inserts his SC into the card reader and inputs ID_{USR} , pw_{USR} and the newly chosen password pw'_{USR} . Eventually, SC accepts pw'_{USR} , computes $TID'_{USR} = TID_{USR} \oplus F(ID_{USR}||pw_{USR}) \oplus F(ID_{USR}||pw'_{USR})$, and stores TID'_{USR} replacing TID_{USR} . Consider that USR mistakenly submits a wrong identity or password. Still, SC will be able to compute TID'_{USR} and replace the actual TID_{USR} . This may lead to unwanted alteration of the actual login information stored in SC and also, may damage the SC due to unverified acceptance of pw'_{USR} . This may also lead to lack of user-friendliness.

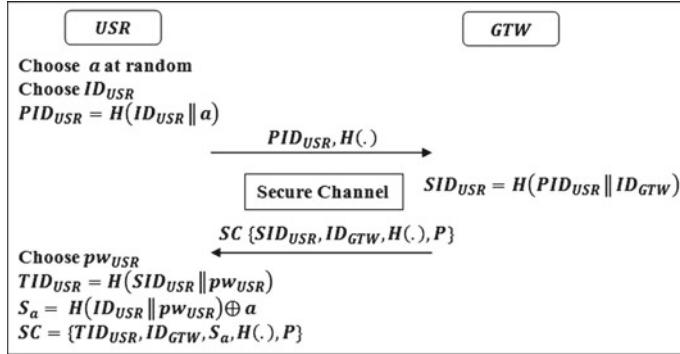


Fig. 1 Registration phase of the proposed ASKI-WSN scheme

4 Proposed ASKI-WSN Protocol

This section demonstrates our proposed ECC-based ASKI-WSN scheme. We assume that, during initiation, *GTW* announces his public key $K_g = g \cdot P$, where $g \in Z_q^*$ is a secret of *GTW*.

4.1 Registration Phase

To access ubiquitous services of WSN, a new *USR* has to register his credentials to *GTW* via a secure channel as demonstrated in Fig. 1. First, *USR* chooses ID_{USR} , pw_{USR} and a random number $a \in Z_q^*$. Then, *USR* submits his registration request $\{PID_{USR} = H(ID_{USR} \parallel a), H(.)\}$ to *GTW*. On receipt of the registration request, *GTW* computes $SID_{USR} = H(PID_{USR} \parallel ID_{GTW})$ and assigns a new *SC* containing the parameters $\{SID_{USR}, ID_{GTW}, H(.), P\}$ to *USR*. Subsequently, *USR* computes $TID_{USR} = H(SID_{USR} \parallel pw_{USR})$ and $S_a = H(ID_{USR} \parallel pw_{USR}) \oplus a$. Then, *USR* stores TID_{USR} and S_a into the *SC* replacing SID_{USR} . Finally, *SC* consists of the parameters $\{TID_{USR}, ID_{GTW}, S_a, H(.), P\}$.

4.2 Login and Mutual Authentication Phase

To access the useful data from sensor node, *USR* has to successfully login to its desired *SR*. Then, mutual authentication with session key agreement will take place between *USR* and *SR* via insecure channel as depicted in Fig. 2.

Step 1: Initially, *USR* inserts his *SC* into the card reader and inputs ID_{USR} and pw_{USR} . *SC* computes $a = S_a \oplus H(ID_{USR} \parallel pw_{USR})$ and checks the validity of ID_{USR} and

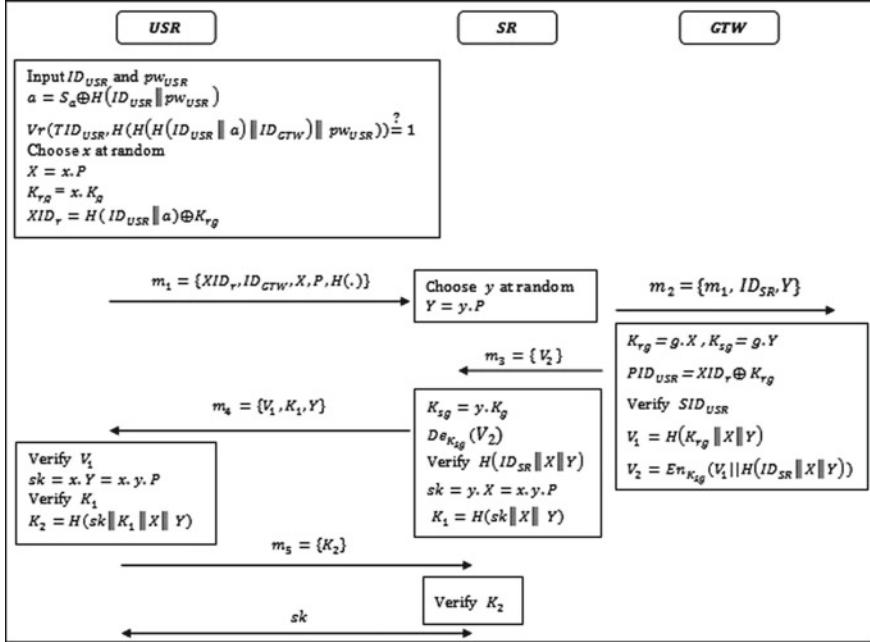


Fig. 2 Login and mutual authentication phase of the proposed ASKI-WSN scheme

- pw_{USR}, as $Vr(TID_{USR}, H(H(H(ID_{USR}||a)||ID_{GTW})||pw_{USR})) \stackrel{?}{=} 1$. If verified successfully, SC generates $x \in Z_q^*$ and calculates $X = x \cdot P$, $K_{rg} = x \cdot K_g$ and $XID_r = H(ID_{USR}||a) \oplus K_{rg}$. Then, SC sends the login request $m_1 = \{XID_r, ID_{GTW}, X, P, H(.)\}$ to SR.
- Step 2: On receipt, SR calculates $Y = y \cdot P$, where $y \in Z_q^*$. Then, SR sends the message $m_2 = \{m_1, ID_{SR}, Y\}$ to GTW.
- Step 3: Upon receiving the message m_2 , GTW calculates $K_{rg} = g \cdot X$, $K_{sg} = g \cdot Y$, $PID_{USR} = XID_r \oplus K_{rg}$, and $Vr(SID_{USR}, H(PID_{USR}||ID_{GTW})) \stackrel{?}{=} 1$. If verified, GTW calculates $V_1 = H(K_{rg} || X || Y)$ and $V_2 = En_{K_{sg}}(V_1 || H(ID_{SR} || X || Y))$. Then, GTW sends the message $m_3 = \{V_2\}$ to SR as a response.
- Step 4: Upon receiving m_3 , SR computes $K_{sg} = y \cdot K_g$ and decrypts V_2 using K_{sg} to check the validity of decrypted message $H(ID_{SR} || X || Y)$. If validated, SR calculates the session key $sk = y \cdot X = x \cdot y \cdot P$ and $K_1 = H(sk || X || Y)$. Then, SR sends the message $m_4 = \{V_1, K_1, Y\}$ to USR.
- Step 5: After receiving m_4 , USR checks whether $Vr(V_1, H(K_{rg} || X || Y)) \stackrel{?}{=} 1$ holds. On successful verification, USR computes the session key as $sk = x \cdot Y = x \cdot y \cdot P$ and sends $K_2 = H(sk || K_1 || X || Y)$ to SR.

Step 6: Finally, SR checks the validity of K_2 and if validated, the session key sk will be established between USR and SR .

4.3 Password Upgrade Phase

For security reasons, USR needs to update his password after a certain period of time. In our proposed ASKI-WSN scheme, USR can solely update his password pw_{USR} using his SC . First, USR inserts his SC into the card reader and inputs ID_{USR} , his old password pw_{USR} , and the newly chosen password pw'_{USR} . Subsequently, SC computes $a = S_a \oplus H(ID_{USR} || pw_{USR})$ and $Vr(TID_{USR}, H(H(H(ID_{USR} || a) || ID_{GTW}) || pw_{USR})) \stackrel{?}{=} 1$ to verify inputted ID_{USR} and pw_{USR} . If verified, SC accepts pw'_{USR} . SC further computes $S'_a = H(ID_{USR} || pw'_{USR}) \oplus a$ and $TID'_{USR} = H(H(H(ID_{USR} || a) || ID_{GTW}) || pw'_{USR})$. Finally, SC stores TID'_{USR} and S'_a replacing TID_{USR} and S_a , respectively.

5 Performance Analysis

This section analyzes the performance of our proposed ASKI-WSN scheme in terms of various security and design properties.

5.1 Security Analysis

The robustness of the proposed ASKI-WSN scheme under several security measures is explained as follows. Table 2 summarizes various security and design properties achieved by our protocol as compared to Nam et al. [8] protocol .

User anonymity: ID_{USR} is not disclosed in any messages over insecure channel. Also, during the registration phase USR submits $\{PID_{USR} = H(ID_{USR} || a), H(.)\}$ to GTW . Thus, \mathcal{A} cannot derive ID_{USR} from PID_{USR} due to the use of collision-resistant one-way hash function $H(.)$ [13].

Perfect forward secrecy: The shared session key is computed as $sk = x \cdot y \cdot P$, where the involved parameters x and y are unique and freshly generated in each new session created. Hence, \mathcal{A} cannot compute the current session key knowing the past session keys.

Insider attack: Any privileged insider who has access to HA 's database cannot derive ID_{USR} from PID_{USR} and/or SID_{USR} due to $H(.)$.

Replay attack: The shared session key is computed as $sk = y \cdot X = x \cdot Y = x \cdot y \cdot P$, where $X = x \cdot P$ and $Y = y \cdot P$ with fresh and unique values of x and y for each

Table 2 Performance analysis: security and design features

Properties	Nam et al. [8]	ASKI-WSN
Achieve anonymity	Yes	Yes
Perfect forward secrecy	Yes	Yes
Resist insider attack	Yes	Yes
Resist replay attack	Yes	Yes
Resist session key-compromised attack	Yes	Yes
Resist man-in-the-middle attack	No	Yes
Resist stolen smart card attack	Yes	Yes
Resist impersonation attack	No	Yes
Resist password guessing attack	Yes	Yes
Resist denial of service	No	Yes
Login phase verification	No	Yes
Proper mutual authentication	No	Yes
Secure password update	No	Yes
Clock synchronization problem	Yes	No
Lack of user-friendliness	Yes	No

new session. If \mathcal{A} obtains X and Y from previously established sessions and tries to replay them in the current session, still he will not be able to compute the current session key.

Session key-compromised attack: It is computationally infeasible to obtain x and y from X and Y , respectively, due to the use of Elliptic Curve Discrete Logarithmic Problem (ECDLP) [14]. Hence, \mathcal{A} cannot compute sk knowing X and Y .

Man-in-the-middle attack: \mathcal{A} will not be able to compute valid K_{rg} , K_{sg} , V_1 , and V_2 from the messages transmitted over public channel.

Stolen smart card attack: \mathcal{A} may obtain TID_{USR} and/or S_a by applying power analysis theory [15] to a stolen SC. Still, it is computationally infeasible to obtain ID_{USR} , pw_{USR} and a from TID_{USR} and/or S_a due to the use of $H(.)$.

Impersonation attack: \mathcal{A} cannot impersonate as USR or SR or GTW without knowing ID_{USR} , ID_M , pw_{USR} , a , K_{rg} , and K_{sg} .

Password guessing attack: \mathcal{A} cannot guess pw_{USR} neither from TID_{USR} nor from S_a without knowing valid ID_{USR} and a .

Denial of service: In our proposed ASKI-WSN scheme, authenticity of the sender is verified immediately after receiving each message. No timestamp is used to ensure authenticity of the sender. Hence, there is no way to reject the request coming from a legitimate sender.

Login phase verification: Before generating a login request SC validates ID_{USR} and pw_{USR} by computing Vr ($TID_{USR}, H(H(H(ID_{USR}||a)||ID_{GTW})||pw_{USR})$) $\stackrel{?}{=}$ 1. Only after proper verification, SC generates a login request for SR . Thus, there is no way to generate a login request using fake identity and/or password.

Table 3 Performance analysis: computation cost

Protocol	Computation cost	Time (s)
Yeh et al. [5]	$10T_{Mul} + 11T_{Hash}$	0.63625
Shi et al. [6]	$6T_{Mul} + 15T_{Hash}$	0.38595
Nam et al. [8]	$6T_{Mul} + 6T_{Hash} + 3T_{En/De} + 6T_{Mc/Vr}$	0.41055
ASKI-WSN	$8T_{Mul} + 9T_{Hash} + 2T_{En/De}$	0.5265

Proper mutual authentication: Both USR and SR verify the authenticity of each other using K_1 and K_2 , respectively, immediately after computing sk .

Secure password update: SC accepts the new password pw'_{USR} only after proper verification of ID_{USR} and pw_{USR} .

5.2 Computation Cost Analysis

In Table 3, we have compared the computation cost of our proposed ASKI-WSN protocol with Yeh et al. [5], Shi and Gong [6], and Nam et al. [8] considering the login and mutual authentication phase. The overall computation cost is evaluated based on the number of elliptic curve scalar multiplication operations (T_{Mul}), hash functions (T_{Hash}), and encryption/decryption operations ($T_{En/De}$) used to construct the protocol. We ignore the computation cost of bitwise XOR and concatenation operations as they are negligible. We assume that the time required to perform T_{Mul} , T_{Hash} , and $T_{En/De}$ are 0.063075 s, 0.0005 s, and 0.0087 s, respectively [10]. The computation costs for Mc_k/Vr_k functions ($T_{Mc/Vr}$) are assumed to be same as T_{Hash} [8]. It is obvious that the computation cost of our ASKI-WSN scheme is better than Yeh et al. [5]. However, it is slightly higher than Shi and Gong [6] and Nam et al. [8] but our proposed ASKI-WSN scheme achieves more security and design features than that of Yeh et al. [5], Shi et al. [6], and Nam et al. [8].

6 Formal Verification Using AVISPA

We have utilized widely accepted AVISPA tool [16] to formally verify the security strengths of our proposed ASKI-WSN scheme using 32-bit Ubuntu 16.04 LTS operating system running on a 2 GHz Intel(R) Core(TM) i3-6006U processor with 4 GB RAM. The implementation phase is done using High-Level Protocol Specification Language (HLPSL) [17] under On-the-Fly Model Checker (OFMC) back-end [18] considering three different agents: USR , GTW , and SR . We assume that USR and GTW share a symmetric key during the registration phase to make a secure channel between them. The communication channel for the entire login and mutual authentication phase is considered as insecure and followed by Dolev–Yao threat model

(a)	(b)
% OFMC	% OFMC
% Version of 2006/02/13	% Version of 2006/02/13
SUMMARY	SUMMARY
SAFE	UNSAFE
DETAILS	DETAILS
BOUNDED_NUMBER_OF_SESSIONS	ATTACK_FOUND
PROTOCOL	PROTOCOL
/home/prasanta/avispa-1.1/testsuite	/home/prasanta/avispa-1.1/testsuite
/results/ASKI-WSN.if	/results/NAM_Et-Al.if
GOAL	GOAL
as_specified	authentication_on_ur_sr_a
BACKEND	BACKEND
OFMC	OFMC
COMMENTS	COMMENTS
STATISTICS	STATISTICS
parseTime: 0.00s	parseTime: 0.00s
searchTime: 0.06s	searchTime: 0.17s
visitedNodes: 8 nodes	visitedNodes: 11 nodes
depth: 3 plies	depth: 2 plies

Fig. 3 HLPSL implementation result: **a** ASKI-WSN, **b** Nam et al. [8]

[9]. We have considered five secrecy goals and three authentication goals, namely, secrecy of ID_{USR} , a , g , x and y , and authentication on USR , SR , and GTW , respectively. Figure 3 demonstrates the HLPSL implementation result of our proposed ASKI-WSN scheme along with Nam et al. protocol. It can be easily noticed that our protocol remains “SAFE” under Dolev–Yao model check and successfully achieves all of the secrecy and authentication goals. However, in contrast, Nam et al. protocol [8] failed to achieve authentication between USR and SR , and hence, declared as “UNSAFE”.

7 Conclusion

In this manuscript, we have highlighted several limitations of Nam et al. protocol such as denial of service, no login phase verification, clock synchronization problem, improper mutual authentication, key-compromised impersonation and man-in-the-middle attack, and unverified password update phase. Also, we have proposed an ECC-based efficient ASKI-WSN scheme to overcome these limitations. The performance analysis phase of the proposed ASKI-WSN scheme proves that it has the ability to achieve several well-known security and design goals under Dolev–Yao channel. However, our ASKI-WSN scheme achieves slightly higher computation cost than Nam et al. protocol but it exhibits more security and design features. Hence, our proposed authentication scheme is more suitable than Nam et al. protocol in terms of security. In future, we will look forward to extend this work toward three-factor authentication.

References

1. I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, Survey on sensor network. *IEEE Commun. Mag.* **40**, 102–114 (2002)
2. K.H.M. Wong, Y. Zheng, J. Cao, S. Wang, A dynamic user authentication scheme for wireless sensor networks, in *Proceedings of IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing* (Taichung, Taiwan, 2006), pp. 244–251
3. M.L. Das, Two-factor user authentication in wireless sensor networks. *IEEE Trans. Wirel. Commun.* **8**(3), 1086–1090 (2009)
4. T.H. Chen, W.K. Shih, A robust mutual authentication protocol for wireless sensor networks. *ETRI J.* **32**(5), 704–712 (2010)
5. H.L. Yeh, T.H. Chen, P.C. Liu, T.H. Kim, H.W. Wei, A secured authentication protocol for wireless sensor network using elliptic curves cryptography. *Sensors* **11**(5), 4767–4779 (2011)
6. W. Shi, P. Gong, A new user authentication protocol for wireless sensor networks using elliptic curves cryptography. *Int. J. Distrib. Sens. Netw.*, Article ID 730831 (2013)
7. Y. Choi, D. Lee, J. Kin, J. Jung, J. Nam, D. Won, Security enhanced user authentication protocol for wireless sensor networks using elliptic curves cryptography. *Sensors* **14**(6), 10081–10106 (2014)
8. J. Nam, M. Kim, J. Paik, Y. Lee, D. Won, A provably-secure ECC-based authentication scheme for wireless sensor networks. *Sensors* **14**(11), 21023–21044 (2014)
9. D. Dolev, A.C. Yao, On the security of public key protocols. *IEEE Trans. Inf. Theor.* **29**(2), 198–208 (1983)
10. Q. Xie, D. Hong, M. Bao, N. Dong, D.S Wong, Privacy-preserving mobile roaming authentication with security proof in global mobility networks. *Int. J. Distrib. Sens. Netw.* **10**(5), Article ID 325734 (2014)
11. C. Boyd, A. Mathuria, *Protocols for Authentication and Key Establishment*. Springer Science and Business Media (2013)
12. Y. Lu, L. Li, Y. Yang, Robust and efficient authentication scheme for session initiation protocol. *Math. Prob. Eng.*, Article ID 894549 (2015)
13. P. Rogaway, T. Shrimpton, Cryptographic hash function basic: definitions, implications, and separations for preimage resistance, second-preimage resistance, and collision resistance. in *Proceedings of International Workshop on Fast Software Encryption* (2004), pp. 371–388
14. N. Koblitz, Elliptic curve cryptosystem. *Math. Comput.* **48**(177), 203–209 (1987)
15. T.S. Messerges, E.A. Dabbish, R.H. Sloan, Examining smart-card security under the threat of power analysis attacks. *IEEE Trans. Comput.* **51**(5), 541–552 (2002)
16. A. Armando, The AVISPA tool for the automated validation of internet security protocol and applications. in *Computer Aided Verification* (Springer, Berlin, Germany, 2005), pp. 281–285
17. D.V. Oheimb, The high-level protocol specification language HLPSL developed in the EU project AVISPA, in *Proceedings of APPSEM Workshop* (2005), pp. 1–17
18. D. Basin, S. Modersheim, L. Vigano, OFMC: a symbolic model checker for security protocol. *Int. J. Inf. Secur.* **4**(3), 181–208 (2005)

Secured Mobile Collaborative Application in Cloud Environments



Enakshmi Nandi, Ranjan Kumar Mondal, Payel Ray and Debabrata Sarddar

1 Introduction

The current trend of mobile devices presents a high competition between manufacturers. Several devices are manufactured with large no backup and computing, backup capabilities. This growth made people more habituated with different mobile applications. To satisfy users' demand, mobile devices have altered the interaction process with our social environment which made the devices of choice to collaborate with customers and friends. Even several customers coordinate and manipulate their shared data in a real-time system. Nowadays, mobile applications have faced several issues in case of energy consumption, security field, and maintaining consistency of shared data in mobile applications. To grow interest in cloud services, different collaboration systems based on cloud deployment model have been developed such as relegation of mobile clones in the cloud. In fact, there are several algorithms that have been evolved based on this collaborative work to main synchronization and consistency of the data. Through our study, we know that the main problems in this collaborative application are insufficient storage, energy consumption issues due to offloading of mobile computing, and interpreting in mobile applications, and intruders' challenge.

Cloud computing is autonomic computing based on virtualization concept, which covers resources of mobile devices by minimizing load from mobile to cloud where different virtual machines of mobile applications are running. In mobile collaborative application, cloud environment gives flexible utilities of hardware and software devices throughout the network. Maintenance of synchronization between virtual machines with mobile devices and virtual devices to virtual devices by controlling concurrency and ensuring maximum security is an important factor. Some editing

E. Nandi (✉) · R. K. Mondal · P. Ray · D. Sarddar

Department of Computer Science and Engineering, University of Kalyani, Kalyani,
Nadia, India

e-mail: enakshminanditechno14@gmail.com

works have done on mobile collaborative applications in a cloud environment. Some existing collaborative editing protocols are CloneDoc, OptiCloud, etc. In the collaboration, work joins set of mobile devices with a set of virtual machines in this way such that distinct mobile customers acquired their individual virtual machines in the cloud.

In our approach, we used Diffie–Hellman cryptosystem for remembering security issues and dynamic early detection technique for avoiding congestion when several virtual networks want to synchronize with mobile collaborative applications. Created congestion makes slow the synchronization process, even at this time, the third party may attack in this communication process. Our process also maintains the consistency of shared data involved in the collaboration process which is executed on the virtual machine.

2 Related Works

Recent technology massively attracted by computing treatment of cloud platform. Different mobile applications gain from this technology in case of storage and offloading issues. Several approaches have been proposed in the field of the collaborative system in a cloud environment. SPORC [1] is an example of such a collaborative system, which depends on operational transformation method and individual server to serve a global place to simultaneous customer's updates. But it has less battery life, so it does not comfort with mobile devices. Clone Doc [2] is another name of the collaborative system, which is distributed in the peer-to-peer system for virtual machines in the cloud with smartphones. It helps to minimize the massive collaborative editing task of mobile devices to the cloud. But main issues here are server breakdown and requirement of additional operation transformation in case of interaction problem occurred between mobile and virtual server in the cloud. Another name of collaborative editor framework for cloud platform is Red-Black tree [3]. This system helps to minimize the amount of data required for encryption method by following updated content analyzed by the collaborative system, here, complex structure and the unhandled situation in case of server failure. Some examples of collaborative software are Google Docs, Zoho Office Suite [4], which are developed on the basis of the client–server model is based on a client–server model in which the server ensures the data consistency. This server-centric software is suitable for the services and permits clients to alter and modify the same data with the help of central coordination server at the same time. The replication and migration technique is used for fault tolerance occurred in case of convergence with mobile device with its clones (virtual server). The work of Hyrax [5] applies replication method for re-execution of tasks on a static novel node after the occurrence of a failure. In the case of dynamic instantiation of the communication process, proxy migration technique is used to create an instance of proxy service of a novel node after the occurrence of a failure of host node [6]. Mobile Cloud Middleware (MCM) [7] is a framework where migration

of mobile tasks with a dynamic allotment of cloud infrastructure is possible. Here comes the lex heterogeneity nature of cloud platform must be revealed [8].

In our paper, we want to an emphasis on the security challenges faced by the collaboration process of mobile devices in a cloud environment. We developed a model to focus on this issue by giving a better solution with the help of designing our model.

3 Problem Statement

In this paper, we present a research work and we want to develop a model for managing distributive access and designed a cryptography-based model to preserve the security of shared documents of several resources. The combination of mobile and cloud computing helps to improve the availability of data of shared resources. In case the collaborative system improves local reply, improved concurrency, scalability, improved security, and failure retrieval, coordination in a decentralized way is necessary. To fulfill these requirements is not possible easily due to cost, lack of proper infrastructure, lack of synchronization strategy, and theft attack in the way of communication. Even congestion has created in the process of communication. But avoiding congestion in the convergence process between mobile and with its clone is a critical problem, so we add some dynamic early detection technique to control avoid this congestion. To get better performance, designing improved collaborative system by maintaining all factors is a crucial factor. In this case, we focused on just the security issue. We used the Diffie–Hellman cryptographic technique [9] for controlling theft challenges and designing our model. In some other cases, there are distinct methods invented to provide collaborative editing service to mobile users in the cloud based on different software which is required to installed, configured, and monitored properly.

4 Proposed Work

In this section, we present our secured collaborative model for controlling shared data irrespective of different constraints with the help of mobile devices in the cloud platform. We design our model first using Diffie–Hellman cryptographic algorithm and then used detection technique for avoiding congestion faced at the stage of convergence between mobile devices and clones in the cloud platform.

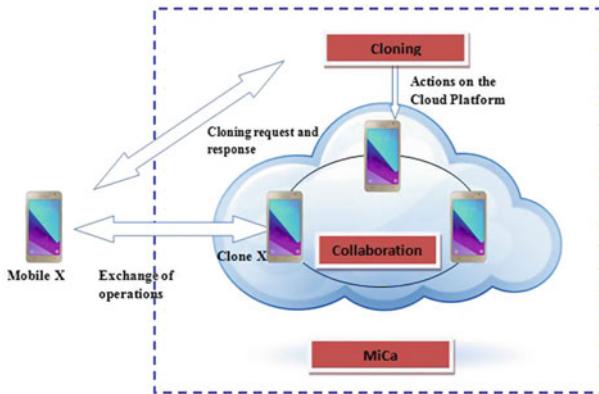


Fig. 1 Mobile collaboration model

4.1 *Collaboration Model Description*

Mobile collaboration is a new technology used in remote areas to communicate with several geographically dispersed users through mixing with electronic pulses and software. Mobile collaboration supports wireless, cellular, and broadband connections by providing different services to discreet customers on their handheld mobile devices, which is independent of location. New generation depends on mobile devices especially smartphones for different attractive features and application found in devices. Thus, mobile collaboration services become more effective services to us. Utilization of mobile collaboration technology spreads in industry, public security purpose, government sector, healthcare, etc. This technology supports several location's users in real-time system to provide advanced audio, video, and telestrator activities through combining with user's input data and complex problems in the work field by involving multiple experts in remote areas.

According to Fig. 1 which is already designed [10], it allows web-based service to provide resource consuming computation by activating several users to clone or virtualized the communication on their mobile devices with the cloud. Here, users build ad hoc groups depending on their virtual connections in the peer-to-peer virtual private network platform. Even new users can join this collaboration group at any time and edit the shared data instantly.

4.2 *Diffie–Hellman Cryptographic Algorithm*

Diffie–Hellman is asymmetric-key cryptography, where two parties generate symmetric session keys for exchanging data without saving the key value or remembering

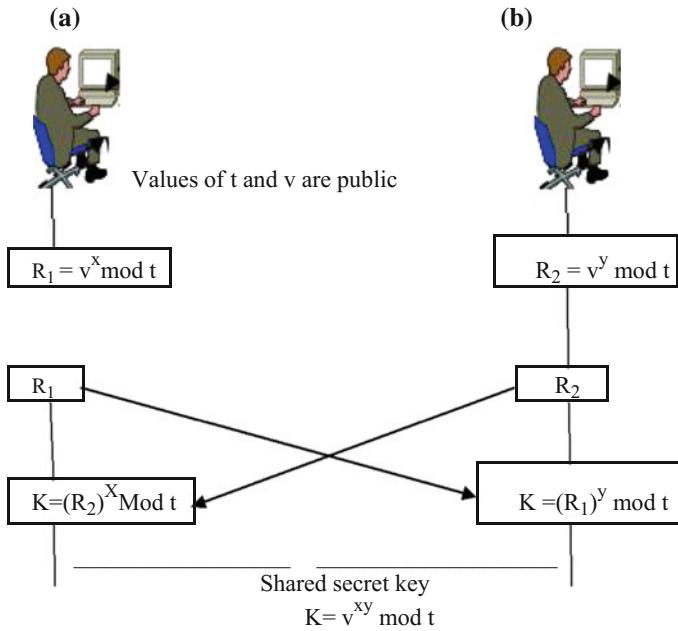


Fig. 2 Diffie–Hellman method

it for future utilization. To explain this cryptographic method, we show an example with respective method explanation and Fig. 2.

Here, A and B want to communicate with each other through exchanging key. At first, they select two numbers t and v before setting up a symmetric key. First number v is a large prime number. The range of this number is in the order of 300 decimal digits (1024 bits), and in case of another number has taken in the random manner. These two distinct numbers sent through the Internet and not required to be kept as secret.

4.3 Diffie–Hellman Algorithm

- Step 1: A selects large number as random manner x and determine $R_1 = v^x \bmod t$.
- Step 2: B selects another large number y randomly. Then determines $R_2 = v^y \bmod t$.
- Step 3: A sends R_1 to B. In this case, A does not send the value of x , she sends R_1 only.
- Step 4: Next B sends R_2 to A. Here, B also sends R_2 only, not the value of y .
- Step 5: Then, A determines $K = (R_2)^x \bmod t$.
- Step 6: B determines $K = (R_1)^y \bmod t$.

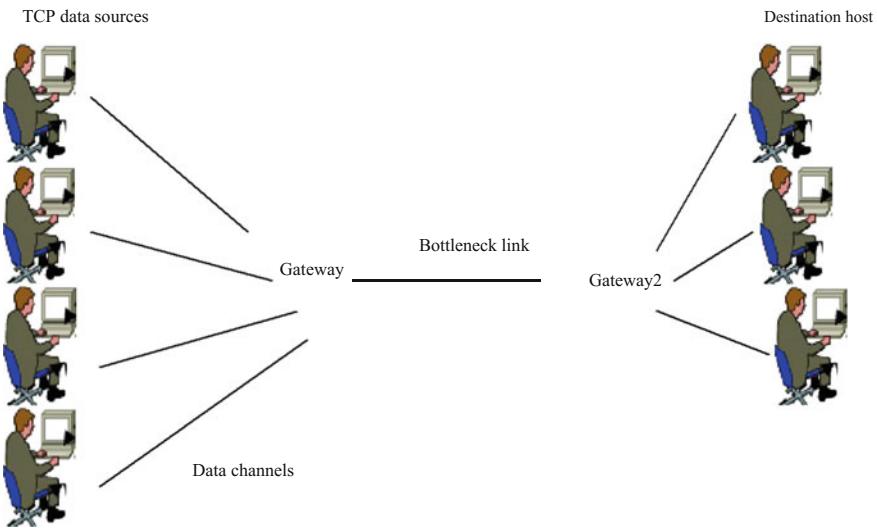


Fig. 3 A network model for developing DRED method

Here, symmetric key for the session is $K = v^{xy} \bmod t$. A cannot determine the value y utilized by B due to calculation performed in modulo t . A receives $v^y \bmod t$ from B, not v^y .

4.4 Dynamic Early Detection Method (DRED) for Congestion Control

The rapid development of Internet technology spread the congestion control mechanism at the transport layer in an effective manner. Present architecture arranged by end-to-end TCP congestion control which is completed by end hosts. This end-to-end congestion control mechanism can be progressed by advanced congestion control method. DRED network model comprises with TCP host which menace source and destination host, a gateway that refers as G1 and G2, data channels connected between TCP source host to gateway or G1 and gateway or G2 to the destination point, and bottle-neck channel established between gateways, that is, G1 and G2 (Fig. 3).

This innovative method DRED helps to identify nascent congestion by changing the original random early detection method with permitting transient congestion unconditionally. In this method, adjustment of maximum probability value depending on reconfigured waiting queue value has been done in dynamically. Maximum probability value relies on which subphase the recent average queue length contains. Here, maximum probability value enhances according to the average queue length that enhances from the lesser subphase to higher subphase. In this method, the actual

size of the queue is dynamically monitored. Here, updated threshold value is called warning line, depending on calculating real queue size according to the packet that can be received. In this process waiting for queue, value divides into two phases. If the original queue size value is less than the warning line, then waiting for queue value should be the same value with the value of the real RED model. But in case of the original size of the queue higher than warning line, then queue weight value is higher, and the ratio of surplus to buffer size will be maximum. Here, maximum drop probability (P_b) value is as follows:

$$P_b = \frac{\text{maximum probability value (average value} - \text{minimum threshold value})}{\text{maximum threshold value} - \text{minimum threshold value}}$$

5 Proposed Architecture

Our main objective is to deliver a secure mobile collaborative application service in a cloud environment by permitting various users; those are geographically dispersed for editing the shared document at any time. In our approach, we used Diffie–Hellman cryptosystem for remembering security issues and dynamic early detection technique for avoiding congestion when several virtual networks want to synchronize with mobile collaborative applications at a time. Created congestion makes slow the synchronization process, and even in this time, the third party may attack in this communication process. Our process also maintains the consistency of shared data involved in the collaboration process which is executed on the virtual machine. Here, a local copy of shared documents and update information related to documents with respective users are locally executed and transfers that information to all users those are involved in this collaboration service. Here, we mainly concern with security issue so that the collaboration process should not hamper.

Mobile collaborative service in cloud environment has done by minimizing loads from the mobile application to virtual machines, running a cloud platform. The first copy of shared documents of collaboration operation saved in the mobile device and a second copy saved on a cloud platform. Customers can modify the mobile copy and send the required information to its respective clone for updating the second copy in the clone and simultaneously transfer these information related to updating sends to other mobile devices. Thus, handling multiple collaboration works with maintaining synchronization is a challenging issue. In this case, congestion may occur, or the third party has entered for disturbing this collaboration process or creating a delay in this synchronization process.

According to our model as in Fig. 4, cloning supervisor manages the cloning protocol where total task divided into three components:

Cloning Setup Machine: These machines help customers to clone their mobile devices in the cloud platform to enrich most of the computation process related to resource consuming. It permits regular collaboration process with other partners

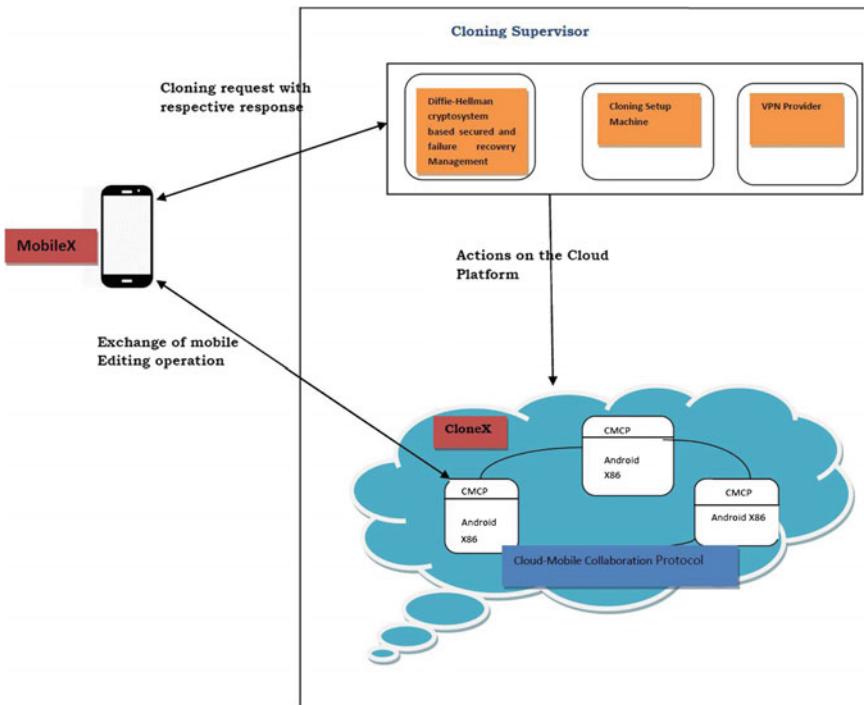


Fig. 4 Architecture of the proposed model

which also have mobile clones in the cloud by restoring energy expended on the mobile devices.

Virtual Private Network (VPN) Provider: This is mainly a peer-to-peer virtual network platform, which helps to combine with cloning setup machine to handle user divergence transparently and fresh users joining the group arrivals. In this platform, customers originate ad hoc groups and depend on their clones to hire a common goal.

Diffie–Hellman cryptosystem-based secured and Failure Recovery Management: We use dynamic early detection method based Diffie–Hellman cryptosystem for ensuring improved security and gives failure recovery service with the help of cloning supervisor. Here, the customer can also recover his shared document copy from his clone copy even during disconnection by means of the mobile devices' copy. Our fault-tolerant protocol can identify clone and failures in the network connection, which occurred at the time of clone deployment and collaboration between clones. This permits a failed clone to fix its consistent situation and reconnect its collaborative groups.

Cloud Mobile Collaboration Protocol (CMCP): It manages synchronization process for influencing shared data by variable users in the cloud. This type of protocol gives the assurance of consistency of replicated shared data in the real-time environment. This protocol [11] supports continuous access to shared documents using

optimistic replication scheme. It involves the operational transformation approach for managing consistency. Here, reconciliation of divergence copies has performed automatically in a decentralized way. Here, weak computing task has performed in the mobile device, and heavy computing job has performed on the cloud. This protocol consists of the local state which involves shared documents saved in clone, operation log which involves the history of editing operation executed by the clone, and operation queue which involves remote editing operation.

6 Conclusions

In this paper, we have designed secured mobile collaborative application in a cloud environment, which supports mobile customers to access to abundant computing power and data storage. Our model is based on two protocols: (i) a cloning deployment protocol and (ii) Cloud Mobile Collaborative Protocol (CMCP) which involves secure mechanisms for synchronizing collaborative works in a decentralized way using our methods. We hope that our model will give a better response to serve mobile collaboration work in a cloud environment in a flexible way by reducing energy consumption by giving better security.

7 Future Works

Our next focus on implementing our model in a web interface, which belongs the information cloning a mobile device and connecting a collaborative group deploying in the Axis2 web server and based on the standard Web Services Description Language (WSDL) with proper evaluating the total impact in case of network traffic, the amount of energy consumed and response time with assuring improved security when assigning treatments to clones in the cloud in near future.

References

1. A.J. Feldman, W.P. Zeller, M.J. Freedman, E.W. Felten, SPORC: Group Collaboration Using Untrusted Cloud Resources
2. S. Kosta, V.C. Perta, J. Stefa, P. Hui, A. Mei, Clonedoc: exploiting the cloud to leverage secure group collaboration mechanisms for smartphones, in *2013 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)* (IEEE, 2013), pp. 19–20
3. R. Kienzler, R. Bruggmann, A. Ranganathan, N. Tatbul, Stream as you go: the case for incremental data access and processing in the cloud, in *2012 IEEE 28th International Conference on Data Engineering Workshops (ICDEW)*, (IEEE, 2012), pp. 159–166
4. R. Wójcikowicz, Office Online Suits as a Tool for Supporting Electronic Document Management. *Prace Naukowe Uniwersytetu Ekonomicznego we Wrocławiu*, (232 Knowledge Acquisition and Management), pp. 146–155 (2011)
5. E.E. Marinelli, *Hyrax: Cloud Computing on Mobile Devices Using MapReduce* (No. CMU-CS-09-164). Carnegie-mellon univ Pittsburgh PA, School of Computer Science (2009)
6. C.A. Ellis, S.J. Gibbs, Concurrency Control in Groupware Systems **18**, 399–407 (1989)
7. R. Satta, G. Fumera, F. Roli, M. Cristani, V. Murino, A multiple component matching frameworks for person re-identification. In *International Conference on Image Analysis and Processing* (Springer, Berlin, Heidelberg, 2011), pp. 140–149
8. N. Fernando, S.W. Loke, W. Rahayu, Mobile cloud computing: a survey. *Future Gen. Comput. Syst.* **29**(1), 84–106 (2013)
9. P.C. Kocher, Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems, in *Annual International Cryptology Conference* (Springer, Berlin, Heidelberg, 1996), pp. 104–113
10. M.D. Mechaoui, N. Guetmi, A. Imine, MiCa: Lightweight and mobile collaboration across a collaborative editing service in the cloud. *Peer-To-Peer Netw. Appl.* **9**(6), 1242–1269 (2016)
11. N. Guetmi, A. Imine, Designing mobile collaborative applications for cloud environments, in *Modern Software Engineering Methodologies for Mobile and Cloud Environments* (IGI Global, 2016), pp. 34–60

Investigation of the Effect of Barrier Layer Engineering on DC and RF Performance of Gate-Recessed AlGaN/GaN HEMT



Shubham Mondal, Sritoma Paul and Angsuman Sarkar

1 Introduction

The ever increasing demand for faster and high-efficiency transistor amplifiers and switching devices along with the unprecedented performance characteristics of GaN-based High Electron Mobility Transistors (HEMTs) have propelled extensive research to establish them as the next-generation power electronic devices, owing to their peerless high electric breakdown field of 3.3 MV/cm, high electron saturation velocity of 2.5×10^7 cm/s, high mobility of around 200 cm²/V s [1], fast switching speed, lower on-state resistance, and lower semiconductor loss [2, 3]. In addition, low-voltage GaN transistors do not require expensive packaging and thereby reduce the cost of device manufacturing [3]. It is well known that Aluminum (Al) molar fraction in $\text{Al}_x\text{Ga}_{1-x}\text{N}$ for a fixed barrier thickness in HEMT structure determines the sheet carrier concentration ns and the confinement of two-dimensional electron gas (2DEG) located close to the AlGaN/GaN interface as a result of spontaneous and piezoelectric polarization [4]. Based on the nature of application, the Al content is usually varied roughly between 14 and 30%. A high Al content is desirable for devices operating at high speed as they require a large current and thus a higher carrier concentration. However, a low Al content results in a lower sheet carrier concentration and thus reduces power dissipation as well as self-heating of the devices. Hence, a high breakdown voltage can be achieved [5]. The effects of changing Al molar fraction in the barrier layer as well as barrier layer thickness modulation on DC characteristic parameters like threshold voltage, maximum transconductance, sub-threshold swing, and channel capacitance have been studied by Majumdar et al. [6]. The effect of barrier layer thickness modulation on gate leakage current has been reported by Turuvekere et al. [7]. Owing to the high breakdown voltage and high electron saturation velocity required for RF amplifiers [8], RF performance of

S. Mondal · S. Paul · A. Sarkar (✉)

ECE Department, Kalyani Government Engineering College, Kalyani, India

e-mail: angsumansarkar@hotmail.com

recessed gate AlGaN/GaN HEMTs has been extensively studied and improved over the years. In this brief, we have endeavored to investigate the effects of barrier layer engineering by introducing a second barrier layer having a different molar fraction of Al beneath the AlGaN barrier layer and its effect on the DC and RF performance of the normally off (E-mode) HEMT device has been studied.

2 Device Structure and Simulation

Schematic of the device structure used for the simulation of enhancement mode n++ GaN/AlGaN/GaN HEMT with ohmic contacts is shown in the figure below. The design and simulation presented in this paper have been done using Silvaco ATLAS device simulator (Fig. 1 and Tables 1 and 2).

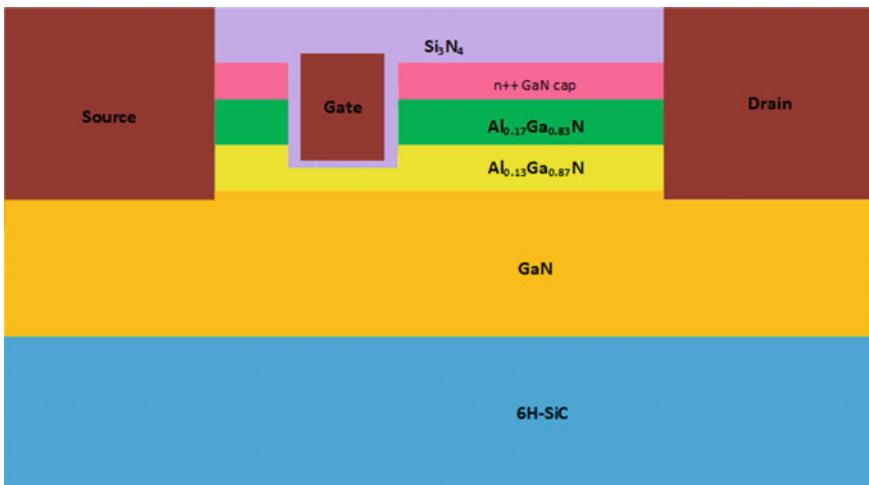


Fig. 1 Schematic of the device structure used for the simulation of E-mode n++ GaN/AlGaN/GaN HEMT with double-barrier layer

Table 1 Structure specifications

Layer	Material	Thickness	Doping
Passivation	Si ₃ N ₄	0.5 μm	NA
Cap	GaN	10 nm	2e+20 cm ⁻³
Barrier I	Al _{0.17} Ga _{0.85} N	5 nm	1e+15
Barrier II	Al _{0.13} Ga _{0.87} N	3 nm	1e+15
Buffer	GaN	2 μm	1e+16 (Fe doping)
Substrate	6H-SiC	1 μm	NA

Table 2 Gate specifications

Name	Dimension
L_g	120 nm
L_{gs}	340 nm
L_{gd}	3.43 μm
Gate-recess	6 nm
Work function	4.75 eV

It is found that decreasing both gate length (L_g) and gate-to-source length (L_{gs}) results in a significant improvement in the drain current and peak transconductance [9]. The proposed device structure has a gate work function of 4.75 eV. Metals like antimony (4.79 eV), copper (4.65 eV), and rhodium (4.72 eV) have work functions in close proximity to the taken work function. The multilayer barrier is composed of $\text{Al}_x\text{Ga}_{1-x}\text{N}$, where x denotes the molar fraction of aluminum. In this brief, the mole fractions of the two barrier layers are taken as $x=0.17$ and 0.13 , respectively, while the simulation of the single-barrier layer HEMT is performed by taking the mole fraction of Al equal to 0.17 and the width of the barrier layer is maintained at 8 nm. The gate is recessed up to the upper portion of the multi-barrier layer to ensure it operates in the enhancement mode (E-mode). The ohmic contacts in the source and drain are assumed to penetrate the GaN buffer layer such that they reach the two-dimensional electron gas formed in the channel.

Proper mobility models are specified for GaN [1] and acceptor trap levels having a density of $7\text{e}+17 \text{ cm}^{-3}$ [10] have been specified. Fermi–Dirac, “FERMI”, concentration-dependent mobility model, “CONMOB” and Shockley–Read Hall (SRH) model using fixed minority carrier lifetimes are employed for the simulation. Numerical methods, namely, GUMMEL and NEWTON have been used. The DC analysis is performed by setting the drain-to-source voltage as 8 V, while the gate voltage is varied in small steps from 0 to 2 V. The RF analysis is performed by setting gate-to-source voltage, $V_{gs}=1.5$ V and $V_{ds}=8$ V, while the ac frequency is varied from 1 to 50 GHz.

3 Results and Discussion

The Al content in $\text{Al}_x\text{Ga}_{1-x}\text{N}$ for a fixed barrier thickness in HEMT structure determines the sheet carrier concentration n_s and the confinement of 2DEG located close to the AlGaN/GaN interface as a result of spontaneous and piezoelectric polarization. Moreover, as the Al content increases, crystal quality is degraded since stored elastic energy varies as the square of the Al content, as suggested by the following equations [11]:

$$\epsilon = (a_{\text{GaN}} - a_{\text{AlGaN}})/(a_{\text{AlGaN}} \sim 0.024x_{\text{Al}}) \quad (1)$$

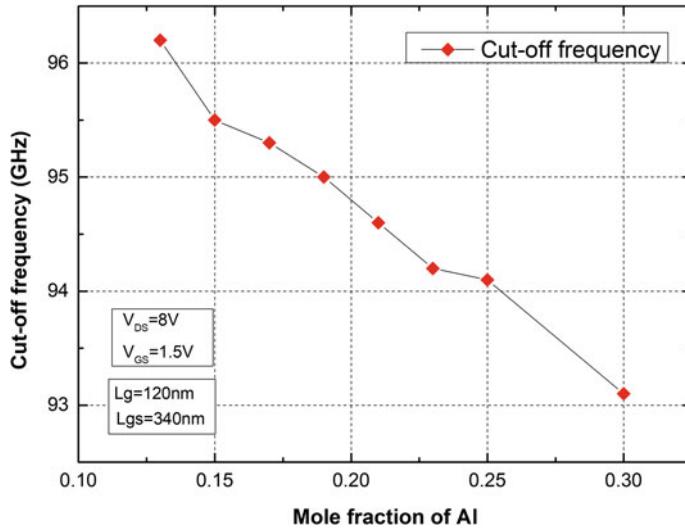


Fig. 2 Variation of cut-off frequency of SBL HEMT with molar concentration of Al

$$E = M\epsilon^2 h \sim \alpha x_{Al}^2 h \quad (2)$$

An increase in the Al content also results in a larger mismatch strain in the barrier that adds to the buffer strain [12]. Reliability is a key issue for GaN RF and power switching devices, and hence a thin barrier is generally avoided as it increases the gate leakage current due to the formation of traps that assist the tunneling of electrons from the lower part of the gate to the 2DEG [13]. Therefore, we choose a lower value of Al molar fraction for both single- and double-barrier layer HEMTs, and the thickness of the barrier layer is chosen according to the critical thickness given by Bykhovski et al. [14]. However, there are certain limitations imposed by the scarcity of 2DEG density with lower Al content in the barrier. Therefore, a trade-off is achieved and the value is chosen to be 0.13 for single-barrier layer (SBL) HEMT and 0.15 for the upper and 0.13 for the lower barrier layer in the double-barrier layer (DBL) HEMT. Figure 2 shows the variation of cut-off frequency of single-barrier layer (SBL) HEMT with molar concentration of Al in the barrier layer at a drain-to-source voltage of 8 V and gate-to-source voltage of 1.5 V. Similarly, Fig. 3 shows the variation of maximum oscillation frequency of single-barrier layer (SBL) HEMT with molar concentration of Al. This indicates that mole fraction directly influences the RF performance of the device. It is evident that a molar concentration of 0.13 yields the best RF characteristics.

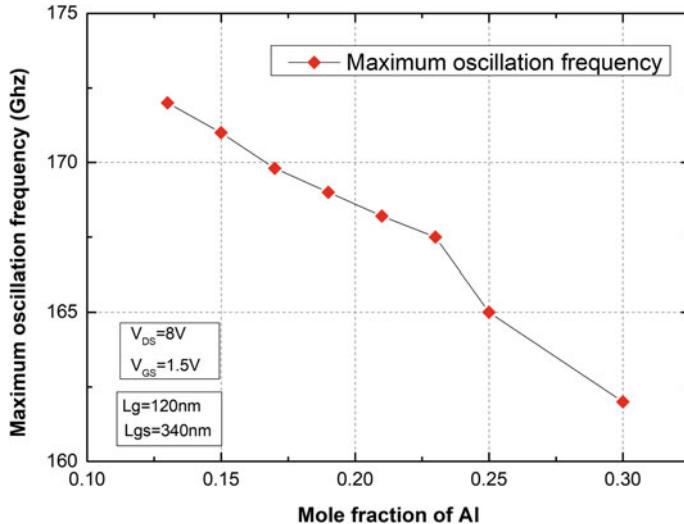


Fig. 3 Variation of maximum oscillation frequency of SBL HEMT with molar concentration of Al

Figure 4 shows the comparison of the transfer characteristics for single-barrier layer gate-recessed E-mode HEMT and double-barrier layer gate-recessed E-mode HEMT. The plot of drain current versus gate voltage at a drain-to-source voltage (V_{ds}) of 8 V reveals that for a gate voltage of 2 V the former has a drain current (I_d) of 2.22 A/mm, whereas the double-barrier layer HEMT achieves a drain current of 2.44 A/mm, thereby showing an improvement of about 10%. Linearity plays an important role in RF power amplifiers as the modulation signals have a large dynamic range [15]. The transconductance curve in Fig. 5 shows a wide linear range of transconductance close to their peak values which can be attributed to the larger absolute distance between 2DEG channel and gate by the Si_3N_4 passivation layer and implementation of n-doped GaN cap above AlGaN barrier [16]. The SBL HEMT exhibits a maximum transconductance of 1860 mS/mm, whereas the transconductance curve of the DBL HEMT attains a peak value of 2050 mS/mm, with other parameters intact. In this brief, we have attained an improvement of 10.2% in the peak transconductance value that clearly enhances the RF performance of the device. The RF performance of the SBL and DBL HEMTs is characterized from 1 to 50 GHz while maintaining $L_{gs} = 0.34 \mu\text{m}$ and $L_g = 120 \text{ nm}$. The cut-off frequency is obtained by extrapolating the parameter $|h21|^2$ with a slope of -20 dB/decade using a least square fit method in Fig. 5. The simulation result gives cut-off frequency $f_t = 95.94 \text{ GHz}$ for SBL HEMT and 102.3 GHz for DBL HEMT, thereby exhibiting an increase of 6.63%. The maximum oscillation frequency, f_{max} , values are calculated in a similar manner as shown in Fig. 6. The obtained values are $f_{max} = 172 \text{ GHz}$ for SBL HEMT and 186.2 GHz for DBL HEMT, resulting in a remarkable increase of 8.25% (Fig. 7).

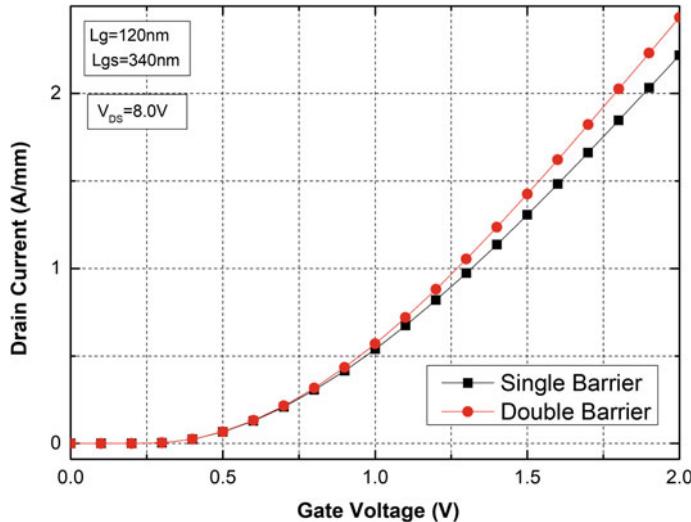


Fig. 4 Comparison of the transfer characteristics of SBL and DBL gate-recessed E-mode HEMT

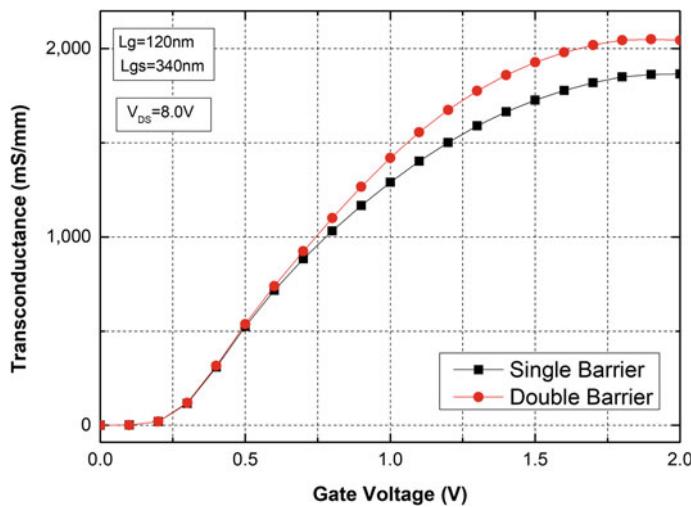


Fig. 5 Comparison of the transconductance of SBL and DBL gate-recessed E-mode HEMT

4 Conclusion

In this paper, we have investigated the effects of barrier layer engineering on gate-recessed normally off (E-mode) high electron mobility transistor. A novel HEMT structure based on double-barrier layer has been proposed and its characteristics have been studied extensively. The value of mole fraction of Al in Al_xGa_{1-x}N has

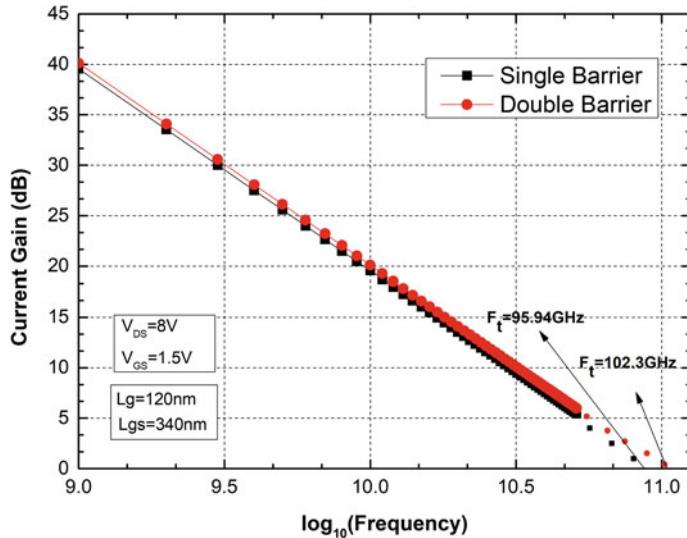


Fig. 6 Comparison of the cut-off frequency of SBL and DBL gate-recessed E-mode HEMT

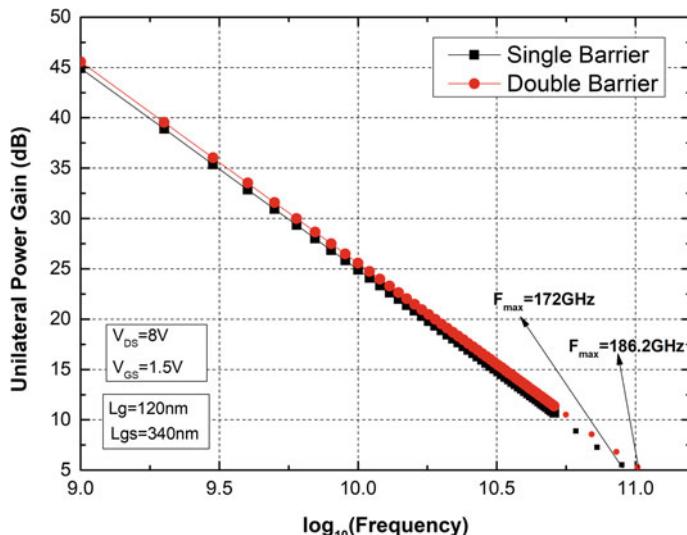


Fig. 7 Comparison of the maximum oscillation frequency of SBL and DBL gate-recessed E-mode HEMT

been optimized, and a trade-off has been achieved between feeble 2DEG density due to low values of mole fraction of Al and strain-induced effects due to higher values of mole fraction. A comparison is drawn between single-barrier layer and

double-barrier layer HEMT and significant improvement is observed in the DC and RF characteristics of DBL HEMT over SBL HEMT.

References

- U. Mishra, P. Parikh, Y.-F. Wu, AlGaN/GaN HEMTs-an overview of device operation and applications. Proc. IEEE **90**(6), 1022–1031 (2002)
- G. Tang et al., Digital integrated circuits on an E-Mode GaN power HEMT platform. IEEE Electron. Device Lett. **38**(9), 1282–1285 (2017)
- K.J. Chen, O. Häberlen, A. Lidow, C.L. Tsai, T. Ueda, Y. Uemoto, Y. Wu, GaN-on-Si power technology: devices and applications. IEEE Trans. Electron. Devices **64**(3), 779–795 (2017)
- O. Ambacher, B. Foutz, J. Smart, J.R. Shealy, N.G. Weimann, K. Chu, M. Murphy, A.J. Sierakowski, W.J. Schaff, L.F. Eastman, J. Appl. Phys. **87**, 334 (2000)
- K. Köhler, S. Müller, P. Waltereit, W. Pletschen, V. Polyakov et al., J. Appl. Phys. **109**, 053705 (2011). <https://doi.org/10.1063/1.3553866>
- S. Majumdar, S. Das, D. Biswas, Barrier layer engineering: performance evaluation of E-mode InGa/AlGaN/GaN HEMT. 1675 (2015). <https://doi.org/10.1063/1.4929179>
- S. Turuvekere, A. Dasgupta, N. DasGupta, Effect of barrier layer thickness on gate leakage current in AlGaN/GaN HEMTs. IEEE Trans. Electron. Devices **62**, 3449–3452 (2015). <https://doi.org/10.1109/ted.2015.2469151>
- U.K. Mishra, L. Shen, T.E. Kazior, Y. Wu, GaN-based RF power devices and amplifiers. Proc. IEEE **96**, 287 (2008)
- H.F. Huang et al., Investigation of a GaN-on-Si HEMT optimized for the 5th-generation wireless communication, in *2015 IEEE 11th International Conference on ASIC (ASICON)*, Chengdu (2015), pp. 1–4
- C. Tang, J. Shi, Influence of acceptor-like traps in the buffer on current collapse and leakage of E-mode AlGaN/GaN MISFETs. Semicond. Sci. Technol. **28**, 5011– (2013). <https://doi.org/10.1088/0268-1242/28/11/115011>
- Y. Yingxia, Z. Lin, C. Luan, Y. Lv, Z. Feng, M. Yang, Y. Wang, H. Chen, Influence of the channel electric field distribution on the polarization Coulomb field scattering in AlGaN/AlN/GaN heterostructure field-effect transistors. AIP Adv. **3**, 092115 (2013). <https://doi.org/10.1063/1.4821547>
- YvonCordier, Al(Ga)N/GaN high electron mobility transistors on silicon. Phys. Status Solidi A **212**(5), 1049–1058 (2015)
- D. Marti, S. Tirelli, V. Teppati, L. Lugani, J.-F. Carlin, M. Malinvern, N. Grandjean, C.R. Bolognesi, IEEE Electron. Device Lett. **36**, 17 (2015)
- A.D. Bykhovski, B.L. Gelmont, M.S. Shur, Elastic strain relaxation and piezoeffect in GaN-AlN, GaN-AlGaN and GaN-InGaNsuperlattices. J. Appl. Phys. **81**(9), 6332–6338 (1997)
- K.H. Teo, J. Shi, Simulation of GaN HEMT with wide-linear-range transconductance, TR2017-152 October 2017 (Mitsubishi Electric Research Laboratories, 2007)
- I. Khalil, E.B. Treidel, F. Schnieder, J. Würfl, Improving the linearity of GaN HEMTs by optimizing epitaxial structure. IEEE Trans. Electron. Devices **56**(3), 361–364 (2009)