

Springer Proceedings in Mathematics & Statistics

Syeda Darakhshan Jabeen
Javid Ali
Oscar Castillo *Editors*

Soft Computing and Optimization

SCOTA 2021, Ranchi, India, March 26–27

 Springer

**Springer Proceedings in Mathematics &
Statistics**

Volume 404

This book series features volumes composed of selected contributions from workshops and conferences in all areas of current research in mathematics and statistics, including data science, operations research and optimization. In addition to an overall evaluation of the interest, scientific quality, and timeliness of each proposal at the hands of the publisher, individual contributions are all refereed to the high quality standards of leading journals in the field. Thus, this series provides the research community with well-edited, authoritative reports on developments in the most exciting areas of mathematical and statistical research today.

Syeda Darakhshan Jabeen · Javid Ali ·
Oscar Castillo
Editors

Soft Computing and Optimization

SCOTA 2021, Ranchi, India, March 26–27

 Springer

Editors

Syeda Darakhshan Jabeen
Department of Mathematics
Birla Institute of Technology Mesra
Ranchi, India

Javid Ali
Department of Mathematics
Aligarh Muslim University
Aligarh, Uttar Pradesh, India

Oscar Castillo
Computer Science in the Graduate Division
Tijuana Institute of Technology
Tijuana, Mexico

ISSN 2194-1009 ISSN 2194-1017 (electronic)
Springer Proceedings in Mathematics & Statistics
ISBN 978-981-19-6405-3 ISBN 978-981-19-6406-0 (eBook)
<https://doi.org/10.1007/978-981-19-6406-0>

Mathematics Subject Classification: 41A25, 90B50, 90C25, 90C70, 49J45, 68Q12, 68Q25, 68T07, 68T10, 68W50, 68U10, 93C42

© The Editor(s) (if applicable) and The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022

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

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

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

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

Preface

Indeed, research is a continuous process and its progress outlines and illuminates a new horizon. As a result of which there is a massive enhancement in technology innovation and, at the same time, educational pursuit advances with a coherent vibration toward its perfection. There is no doubt that modern science is still striving to unravel many unexplored questions and mysteries. And in this endeavor, science is also attempting to take itself to a novel and higher level. In recent past, advances in soft computing techniques and optimization strategies have made much-needed achievements with substantial success. Indeed, soft computing has the potential to provide tractable, low cost, and comprehensible solutions even when the problem domain is partially true and ambiguous. Even more, this has emerged forth as robust, efficient and highly effective methods for solving challenging optimization problems that require faster and better decision-making efficacy. It has manifested its kaleidoscopic arrays of successful applications in solving a number of complex real-life problems with improved and progressively solvable methods, where conventional hard computing techniques have either failed to work or were partially successful. It would not be wrong to say that soft computing has not yet attained its maturity, and there are many areas and applications where soft computing techniques need to be explored, and some modifications are needed to improve their solution-finding strategies.

This book introduces the various strategies and applications of soft computing and optimization, and it will render as the basis for increasing transparency, interpretability and trustworthiness of diverse perspectives in other frameworks and for the progressive and holistic development of the miscellaneous realms. The book discusses the applications of soft computing techniques in portfolio management, image processing, machine learning, scheduling problems and other relevant application problems encountered in real life. Perhaps, soft computing is inextricably linked with many intelligent computing methods, including fuzzy logic, neural networks, bio-inspired optimization algorithms, evolutionary algorithms, etc. These are currently being used in research to develop a resilient and hybrid intelligent, affective and cognitive system. Comprehensive information about all these genres has been included in various sections of the book.

The book is broadly segregated into three parts with each part covering research papers on similar topics. The first part consists of ten chapters (Chapters “[Classification of MRI Images for Detecting Alzheimer’s Disease Using Convolutional Neural Network](#)”–“[A Soft Computing Approach for Predicting and Categorizing Learner’s Performance Using Fuzzy Model](#)”) with the main theme of soft computing methods, in which the authors introduce novel and innovative models and concepts. There is no doubt that all these chapters can fortify the basis for achieving intelligent optimization in diverse fields. These research papers deal with neural networks, particle swarm optimization methods, support vector machines, ant colony optimization and other artificial intelligence techniques. In the second part, five chapters have been assimilated (Chapters “[A Fuzzy Logic-Based Approach to Solve Interval Multi-objective Non-linear Transportation Problem: Suggested Modifications](#)”–“[Discontinuity at Fixed Point Over Partial Metric Spaces](#)). All these five chapters reflect novel theoretical and pragmatic concepts and methods pertaining to optimization principles. These include fuzzy logic, fixed point theory, multi-objective optimization, approximate optimality, etc. Subsequently, 12 chapters are placed in the third part (Chapters “[Fuzzy Logic Implementation in Home Appliances](#)”–“[Asymptotic Behavior of Resolvents on Complete Geodesic Spaces With General Perturbation Functions](#)”). These chapters are devoted to research papers covering diverse applications of various advanced topics such as fuzzy logic, machine learning and multi-objective optimization. Additionally, research papers for the pursuit of theoretical underpinnings of heuristic methods in real-world problems such as home appliances, portfolio structuring, image processing, music classification, agriculture, scheduling problems, venue allocation problems, stress level analysis, etc., are also included.

Overall, the book covers interesting selected collections of original contributions by prominent academicians and researchers who presented their latest findings in an open forum at the SCOTA 2021 international conference. We hope and believe that the readers will considerably be benefitted from the topics assimilated in different chapters. In this book, the exciting findings pertaining to theories, modeling and algorithms are illustrated in a lucid manner through simulated and experimental results for a better understanding. Editors consider that this book will in particular be bestow deep insight to practitioners and researchers already working or planning to work in this field and the students in general in their rudimentary research stage.

Ranchi, India
Aligarh, India
Tijuana, Mexico

Syeda Darakhshan Jabeen
Javid Ali
Oscar Castillo

Acknowledgements

We feel a sense of great contentment to render the book, *Soft Computing and Optimization: SCOTA 2021, Ranchi, India, March 26–27*, before the readers. Indeed, this book could take its final shape because of the substantial contributions of many learned authors. The chapters incorporated in this book are modified and subtly reviewed papers that are presented at the SCOTA 2021 held at the Department of Mathematics, Birla Institute of Technology Mesra, Ranchi, India.

First and foremost, we are deeply indebted to the beacon of inspiration from Prof. Indranil Manna, Vice-Chancellor, Birla Institute of Technology Mesra, India. We must admit that without his blessings, this conference would not have gone so well and vivacious.

We are also grateful to Prof. Seshadev Padhi, Head, Department of Mathematics, Birla Institute of Technology Mesra, for his wholehearted support in the pragmatic execution of the conference plenus.

We wish to record our deep gratitude to the dignitaries who graced the occasion. We owe our gratitude from inner core of hearts to our eminent keynote speakers: Prof. Kalyanmoy Deb, Michigan State University, USA, and Prof. Oscar Castillo, Institute of Technology Tijuana, Tijuana, Mexico. They have graciously accepted our invitation to deliver their underpinning talk at the conference. We also express our gratitude to eminent invited speakers Prof. Leopoldo Eduardo Cárdenas Barrón, Tecnológico de Monterrey, Mexico; Prof. Samir Kumar Neogy, Indian Statistical Institute Delhi, India; Prof. Ali Farajzadeh, Razi University, Kermanshah, Iran; Prof. Anindya Chatterjee, Indian Institute of Technology Kanpur, India; Prof. Armin Fugenschuh, MINT Brandenburg University of Technology, Cottbus Senftenberg, Germany; Prof. Aruna Tiwari, Indian Institute of Technology Indore, India; Prof. Debjani Chakraborty, Indian Institute of Technology Kharagpur, India; Prof. Gülder Kemalbay, Yıldız Technical University, Istanbul, Turkey; Prof. Mehdi Asadi, Islamic Azad University, Zanjan, Iran; Prof. Samarjit Kar, National Institute of Technology Durgapur, India; Prof. Pankaj Dutta, Indian Institute of Technology Bombay, India; Prof. M. Tanveer, Indian Institute of Technology Indore, India; Prof. Shiv Kumar Gupta, Indian Institute of Technology Roorkee, India; Prof. Snehashish Chakraverty, National Institute of Technology Rourkela, India; Prof. Sriparna Saha, Indian Institute of Technology

Patna, India; Prof. Tandra Pal, National Institute of Technology Durgapur, India; Prof. Yasunori Kimura, Toho University, Chiba, Japan.

We are privileged to express our gratitude to the national and international advisory committees that entail: Prof. Bimal Roy, Indian Statistical Institute Kolkata, India; Prof. Manoranjan Maiti, Vidyasagar University, India; Prof. Rathindra Nath Mukherjee, Burdwan University, India; Prof. Nita H. Shah, Gujarat University, India; Prof. M. L. Das, DA-IICT, Gandhinagar, India; Prof. M. A. Zaveri, S. V. National Institute of Technology Surat, India; Prof. T. K. Roy, Indian Institute of Engineering Science and Technology, Shibpur, India; Prof. S. Majhi, Indian Institute of Technology Patna, India; Dr. Ranjan Kumar Jana, S. V. National Institute of Technology, Surat, India; Prof. Kunal Mukhopadhyay, DoFA, Birla Institute of Technology Mesra, India; Prof. C. Jeganathan, DRIE, Birla Institute of Technology Mesra, India; Dr. Subhash Chandra Pandey, Birla Institute of Technology Mesra (Patna Campus), India; Prof. Aquil Ahmed, Aligarh Muslim University, Aligarh, India; Dr. Irfan Ali, Aligarh Muslim University, Aligarh, India; Prof. J. K. Kim, Kyungnam University, Masan, South Korea; Prof. Poom Kumam, King Mongkut's University of Technology Thonburi, Thailand; for their impeccable supports and suggestions.

Moreover, we sincerely appreciate the imperative, pertinent, and worthy comments and suggestions received from Prof. Gadadhar Sahoo, Indian Institute of Technology (ISM), Dhanbad, India; Dr. Raj Kumar Techchandani, Thapar Institute of Engineering and Technology, Patiala, India; Prof. Ved Prakash Mishra, Amity University, Dubai Campus, UAE; Dr. Leticia Amador, Tijuana Institute of Technology, Tijuana, Mexico; Prof. Mansaf Alam, Jamia Millia Islamia, New Delhi, India; Dr. Patricia Ochoa, Tijuana Institute of Technology, Tijuana, Mexico; Dr. Cinthia Peraza, Tijuana Institute of Technology, Tijuana, Mexico; Dr. Saiyed Umer, Aliah University, Kolkata, India; Dr. Rajeev Agarwal, Malaviya National Institute of Technology Jaipur, India; Dr. Manjur Kolhar, Prince Sattam Bin Abdulaziz University, Saudi Arabia; Dr. Aynur Keskin Kaymakci, Selcuk University, Konya, Turkey; Dr. Prometeo Cortes-Antonio, Tijuana Institute of Technology, Tijuana, Mexico; Dr. Mohd Nadeem, Aligarh Muslim University, Aligarh, India; Dr. Sameena Naaz, Jamia Hamdard, New Delhi, India; Dr. Pradip Kundu, Birla Global University, Bhubaneswar, India; Dr. Banshidhar Sahoo, Hiralal Bhakat College, Birbhum, India; Dr. Naeem Ahmad, Al-Jaouf University, Saudi Arabia; Dr. Rehan Ali, Jamia Millia Islamia, New Delhi, India; Dr. Ali Akbar Shaikh, The University of Burdwan, Burdwan, India; Dr. Balendu Bhooshan Upadhyay, Indian Institute of Technology Patna, India; Dr. Fayyaz Rouzkard, Farhangyan University, Tehran, Iran; Dr. Mohammad Asim, Shree Guru Gobind Singh Tricentenary University, Gurugram, India; Dr. Mohammad Tanwir Uddin Haider, National Institute of Technology Patna, India; Dr. Sanat Kumar Mahato, Sidho-Kanho-Birsha University, Purulia, India; Dr. Mubassir Alam, Patna, India; Dr. Jahangir Chauhan, Aligarh Muslim University, Aligarh, India; Dr. Aftab Alam, Jamia Millia Islamia, New Delhi, India; Dr. Manish Jain, Ahir College Rewari, Haryana, India; Dr. Mohammad Sajid, Aligarh Muslim University, Aligarh, India; Dr. Ashutosh Aggarwal, Thapar Institute of Engineering and Technology, Patiala, India; Dr. Mohammad Faisal Khan,

Saudi Electronic University, Riyadh, Saudi Arabia; Dr. Laxminarayan Sahoo, Ranganj Girl's College, Asansol, India; Dr. Yesim Sarac, Ataturk University, Erzurum, Turkey; Dr. Hazal Yükksekaya, Dicle University, Diyarbakir, Turkey; Dr. Raghav Prasad Parouha, Indira Gandhi National Tribal University, Amarkantak, India; Dr. Mohammad Parvez, Aligarh Muslim University, Aligarh, India; Prof. Saiful Islam, Aligarh Muslim University, Aligarh, India; Dr. Amit Kumar, Thapar Institute of Engineering and Technology, Patiala, India; Dr. Harish Garg, Thapar Institute of Engineering and Technology, Patiala, India; Dr. Izhar Uddin, Jamia Millia Islamia, New Delhi, India; Dr. Tanakit Thianwan, Phayao University, Thailand; Dr. Bhaskar Karn, Birla Institute of Technology Mesra, Ranchi, India; Dr. Ritesh Kumar Singh, Birla Institute of Technology Mesra, Ranchi, India; Dr. Sudip Kumar Sahana, Birla Institute of Technology Mesra, Ranchi, India; Dr. Sanjay Kumar Jha, Birla Institute of Technology Mesra, Ranchi, India; Dr. Joyjeet Ghose, Birla Institute of Technology Mesra, Ranchi, India; Dr. Shyamal Kumar Hui, University of Burdwan, Burdwan, India; Prof. Asoke Kumar Bhunia, University of Burdwan, Burdwan, India; Dr. R. Hemavathy, Queen Mary's College, Chennai, India; that enables and facilitates us to substantially augment the quality of this book. Perhaps, it would be difficult to describe the nuances of technical and canonical flavor and fragrance without these comments and suggestions accurately.

Over and above, we must extend our heartiest congratulations and appreciate the succinct contributions of the authors and participants for making this conference a great success. No doubt, all learned authors deserve a special thank for considering this platform a mode to publish their works. Further, we thank all the technical contributors, delegates, guests, chairpersons, jury members and student committees for their inextricable support and cooperation in the effective organization of the conference.

Last but not least, we would like to thank the publishing team of Springer Nature for accepting our proposal to publish the proceedings in the *Springer Proceedings in Mathematics and Statistics* series, especially Dr. Shamim Ahmad, Senior Editor, and Ms. Banu Dhayalan, Project Coordinator.

Indeed, the cherishable instances of ensemble and synergistic efforts, supports and cooperation of each other that manifested this conference a great success are still rushing back to our memory!

Syeda Darakhshan Jabeen
Javid Ali
Oscar Castillo

Contents

Classification of MRI Images for Detecting Alzheimer’s Disease Using Convolutional Neural Network	1
Rashmi Kumari, Akriti Nigam, and Shashank Pushkar	
Optimum Over Current Relays Coordination for Radial Distribution Networks Using Soft Computing Techniques	13
Shanker Godwal, S. S. Kanojia, and Akhilesh A. Nimje	
A Review of Modified Particle Swarm Optimization Method	25
Priyavada and Binay Kumar	
Automated Detection of Elephant Using AI Techniques	41
Dhrumil Patel and Sachin Sharma	
Determination of Probability of Failure of Structures Using DBSCAN and Support Vector Machine	53
Pijus Rajak and Pronab Roy	
The Ensemble of Ant Colony Optimization and Gradient Descent Technique for Efficient Feature Selection and Data Classification	65
Md. Nayer and Subhash Chandra Pandey	
s-Regularity Via Soft Ideal	79
Archana K. Prasad and S. S. Thakur	
A Comprehensive Study on Mobile Malwares: Mobile Covert Channels—Threats and Security	91
Ketaki Pattani and Sunil Gautam	
Overview of Incorporating Nonlinear Functions into Recurrent Neural Network Models	103
Farzaneh Nikbakhtsarvestani	

A Soft Computing Approach for Predicting and Categorizing Learner’s Performance Using Fuzzy Model 117
Sangita A. Jaju, Sudhir B. Jagtap, and Rohini Shinde

A Fuzzy Logic-Based Approach to Solve Interval Multi-objective Non-linear Transportation Problem: Suggested Modifications 133
Tanveen Kaur Bhatia, Amit Kumar, M. K. Sharma, and S. S. Appadoo

Interval Type-3 Fuzzy Decision-Making in Material Surface Quality Control 157
Oscar Castillo and Patricia Melin

A Generalized Nonlinear Quasi-variational-like Inclusion Problem Involving Fuzzy Mappings 171
Faizan Ahmad Khan, Javid Ali, Fatimah Nasser Albishi, and Faik Gursoy

Approximate Optimality Conditions for a Multi-objective Programming Problem 183
Bhawna Kohli

Discontinuity at Fixed Point Over Partial Metric Spaces 193
Fahimeh Mirdamadi, Hossein Monfared, Mehdi Asadi, and Hossein Soleimani

Fuzzy Logic Implementation in Home Appliances 201
Aditya Priya, Akanksha Kumari, and Mayank Singh

Portfolio Structure of Debt Mutual Funds in Indian Market 211
Soumya Banerjee, Banhi Guha, Amlan Ghosh, and Gautam Bandyopadhyay

Fixed Point Theorems for Digital Images Using Path Length Metric 221
R. Om Gayathri and R. Hemavathy

Emergency Help for Road Accidents 235
Raman Prakash Verma, Nishchaya Kumar, Santosh Kumar, and S. Sruthi

Music Classification Based on Lyrics and Audio by Using Machine Learning 245
E. Arul, A. Punidha, S. Akash Kumar, R. Ragul, and V. B. Yuvaanesh

Multiobjective Optimization for Hospital Nurses Scheduling Problem Using Binary Goal Programming 255
Salman Khalil and Umar Muhammad Modibbo

On the Performance of a Flow Energy Harvester Using Time Delay 271
Zakaria Ghoul

Identification of Some Spatial Coefficients in Some Engineering Topics 283
A. Badran

Automatic Venue Allocation for Varying Class Sizes Using Scoring and Heuristic Hill-Climbing 299
Uvir Bhagirathi and Ritesh Ajoodha

DWT and Quantization Based Digital Watermarking Scheme Using Kernel OS-ELM 313
Neeraj Kumar Sharma, Subodh Kumar, Ankit Rajpal, and Naveen Kumar

Stress Level Analysis Using Bipolar Picture Fuzzy Set 329
P. Muralikrishna, P. Hemavathi, and K. Palanivel

Asymptotic Behavior of Resolvents on Complete Geodesic Spaces With General Perturbation Functions 345
Yasunori Kimura and Keisuke Shindo

Classification of MRI Images for Detecting Alzheimer's Disease Using Convolutional Neural Network



Rashmi Kumari, Akriti Nigam, and Shashank Pushkar

Abstract According to the Alzheimer Foundation of America (AFA), 4 million patients are currently affected due to Alzheimer's Disease (AD) where it forms 6th most significant cause of death of patients in the world. In 2017, the cost of treating Alzheimer's Disease had gone up to 1.1 trillion dollars which is the nation's 10% GDP. Because AD has no cure, detecting it early is the only way to slow down its progression. However, detecting AD is a cumbersome task and requires substantial computational time. Magnetic resonance imaging (MRI) is the most popular modality used by medical practitioners for detecting AD in patients. The present work aims to use a novel convolutional neural network (CNN) as a deep learning technique for accurately classifying the brain's MRI images into two cohorts, i.e., Normal Control (NC) and Alzheimer's Disease (AD), where 85% accuracy has been achieved for 100 subjects. Consequently, the accuracy has been increased to 90% when the number of subjects increases to 200.

Keywords Alzheimer's disease · Deep learning · Convolutional neural networks · Magnetic resonance imaging

1 Introduction

The most familiar type of dementia is Alzheimer's Disease. Every new case merges for AD in any part of the world for each minute. It has been regarded as a disease of the brain that acts according to the patient's age. AD symptoms are memory loss, difficulty in understanding words, inability to understand the talking of other people, a significant problem in doing daily tasks, frequent mood changes, and personality [1]. In addition, AD hampers the economic and social activities of any country and maintains no geographical boundaries. There is no available cure for Alzheimer's Disease, but proper advice and support are readily available.

R. Kumari (✉) · A. Nigam · S. Pushkar
Department of Computer Science Engineering, BIT Mesra, Ranchi 835215, India
e-mail: Rksingh210492@gmail.com

Extraction of features manually is mainly applied in machine learning algorithms. But, in convolutional neural networks, features are extracted automatically, requiring a complicated setup. In CNN, a kernel of finite size is necessary to select the best features from the image automatically. The advancements of the newest technologies such as cloud computing, GPU computing, CNN, and other deep learning models solve intricate Alzheimer’s Disease problems. However, the data acquired from 3D MRI is very complicated and high time consumption.

The complete form of MRI is magnetic resonance imaging. The advantage of using MRI is that it finds internal body images without using incursive technology. Instead, very sophisticated magnetic waves, gradient potential, and radio waves are used to apply them to give a clear idea of body internals. In MRI, body scanning occurs in the axial direction or cuts the plane into the slices’ different shapes. Each of the pieces formed is 2D images. All the 2D pictures amalgamated to form 3D MRI images [2]. In this paper, a novel CNN has been applied for accurately classifying Normal Control (NC) and Alzheimer’s Disease (AD) patients. Here, MRI images obtained from OASIS dataset are applied in this work (Fig. 1).

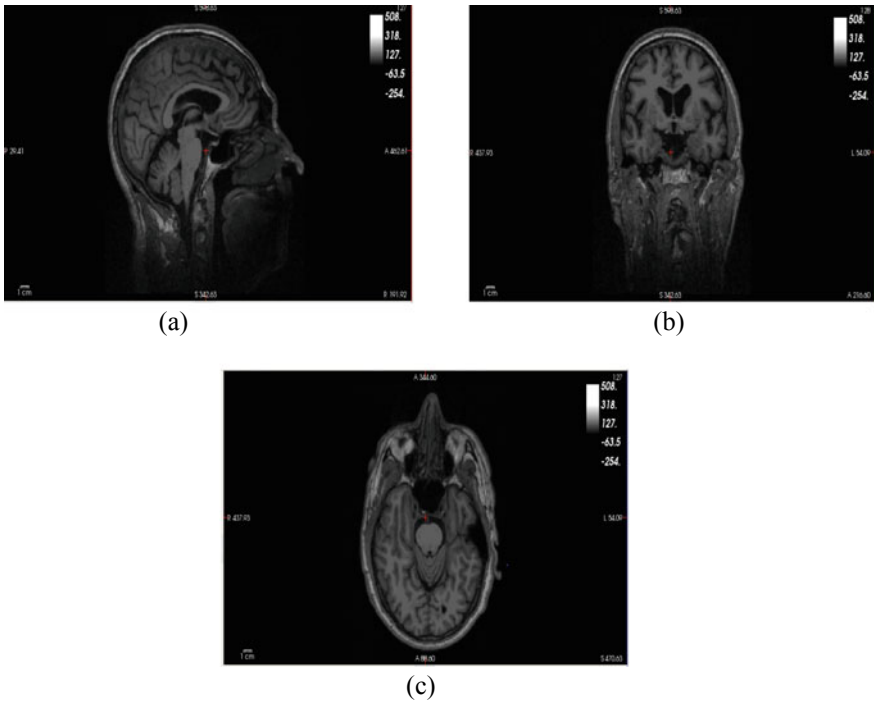


Fig. 1 a Sagittal, b coronal, c axial cross-sectional views of MRI scans

2 Related Works

Many research works are already implemented for diagnosing AD using MRI data, such as Elastic Net and Lasso logistic regressors, support vector machine (SVM), and random forest classifier. SVM is a machine learning algorithm for detecting Alzheimer's disease, applied for 417 subjects present in the OASIS dataset getting 80% accuracy [3]. The accuracy has increased to 92% when ν -SVM has been applied to OASIS dataset [4]. All the features are extracted manually by using conventional machine learning algorithms. Linear SVM is applied for detecting AD patients for MRI scans [5]. For analyzing sMRI data, variation methods and dimensional reduction methods are applied in [6]. In addition to that, SVM binary classifier and multi-class classifier are used for detecting AD. For categorizing AD from other classes, Vemuri et al. [7] applied SVM for developing three classifiers for demographic and genotype data. Gray [8] created a multimodal classification model by applying random forests classifier for the diagnosis of AD using MRI data. For classifying AD, the gray-level co-occurrence matrix (GLCM) method is applied in [9]. By utilizing both manual and automated SVM and AdaBoost algorithm, performances in the model have been compared by Morra et al. [10]. Extraction of predefined features from MRI data has been developed for MRI data. The optimal performance of the classifier is due to a mixture of classifier and features, which results in training the classifier independently [11]. Classifier ensemble algorithm has been developed by combining CNN and ensemble learning (EL) for identifying AD and MCI individuals using MRI images [12]. VGG-16 model has been trained for training datasets consisting of 2560 NC and 2519 AD cases from Kaggle website for extracting features in detecting early AD [13].

3 Materials and Methods

3.1 Dataset Description

In this work, MRI images have been obtained from Open Access Series of Imaging Studies (OASIS) [14] consisting of 416 subjects. For every subject, 3 or 4 individual scans of T1-weighted MRI are obtained in single scan sessions. The main idea of the project is to generate neuroimaging dataset, where scientists and researchers all over the world freely access the database. By collecting and disturbing data, future discoveries can be made for diagnosing different stages of AD. Here, output feature has been designed to two class problem, i.e., Alzheimer's Disease (1) or not having Alzheimer's Disease (0). The brain's gray matter depends upon Alzheimer's Disease, whereas the white value of the brain depends upon MRI scans. As a result, gray matter has been a genuine concern for Alzheimer's Disease [15].

4 Proposed Methodology

4.1 Proposed Convolutional Neural Networks (CNNs) Architecture

In the proposed CNN network, a total of six layers are used, where the first layer comprises convolutional layer [16]. To create a feature map, the filter is used for input images. For reduction in layers, pool layers are used so that they can operate with maximum efficiency. Then, one convolutional layer and pooling layer have been added for smooth operation. After that, the fully connected layer has been added at the other end with a drop out of 80%. This dropout is used only when there are dormant neurons in the preceding layer [17]. This layer consists of 248,768 neurons. Since there are two max-pooling layers with the vertical and horizontal axis with two strides, 248,768 neurons are required for preprocessing the images. Hence, these strides reduce the input size of MRI image from $50 \times 50 \times 91$ to $13 \times 13 \times 23$, which transforms the original image into the restructured image. As a result, 64 feature maps have been created before preceding the dense layer. To operate into feedforward operations, feature maps are being mapped into the dense layer, where the total number of neurons is 248,768. In the last layer, it consists of 2 neurons where it classifies Alzheimer's Disease or not. Rectified linear unit (RLU) activation function [18] is used for two convolutional layers and a fully connected layer.

Adam optimizer is being used for optimizing the whole network with a learning rate of 0.001. Hence, it can be observed that overfitting occurs within this dataset. For this, 50% dropout has been used to counterfeit this problem inserted at each convolutional layer. The probabilities used in this dropout model are 0.8 and 0.65 [19]. By applying these probabilities on this dataset, the performance of CNN has been marginally improved a lot as the data has been rightly set into the training dataset and cross-validating the dataset, which is explained in a block diagram given in Fig. 2. The pseudocode of CNN is given in Fig. 3. Calculation of layerwise and activation functions is applied to fully connected layers and convolutional layers.

$$\text{Width of Convolution} = \frac{\text{Width of (Input MRI - Filter)} + 2 \times \text{Padding}}{\text{Width of Stride}} + 1 \quad (1)$$

$$\text{Height of Convolution} = \frac{\text{Height of (Input MRI - Filter)} + 2 \times \text{Padding}}{\text{Height of Stride}} + 1 \quad (2)$$

$$\text{Total number of Neurons} = \text{Width of Convolution} \times \text{Height of Convolution} \times \text{Filters } (n) \quad (3)$$

$$\text{Size of MaxPooling} = \frac{\text{Width of Convolution}}{2} \quad (4)$$

$$\frac{\text{Parameters of Fully connected layer}}{\text{Present layer of No. of Neurons}} = \frac{\text{Previous Layer of No. of Parameters}}{\text{Present layer of No. of Neurons}} \quad (5)$$

Activation function of Rectified Linear Unit (RLU) $F(x) = 0$ for $x < 0$ and x for $x > 0$ (6)

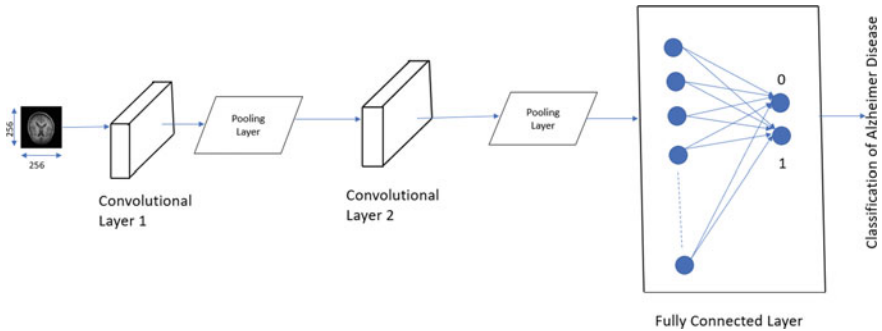


Fig. 2 Block diagram of proposed convolutional neural networks (CNNs) architecture

```

Input:
Train_X, Train_Y : Features of Training Set
Test_X, Test_Y: Features of Testing Set
Output:
Epochi, Accuracyi: Epoch and Accuracy at ith iteration
Begin:
1. for line in file_out:
2.   if 'Epoch' in line:
3.     m= line. split ()
4.     epoch.append(int(m[1]))           // Batch of training set X
5.     y.append(float(m))               // Batch of training set Y
6.   if 'Accuracy' in line:
7.     m= line. split ()
8.     accuracy.append(float*int (m [1])) // Accuracy of training set X
9.     acc.append(acc (1))             // Accuracy of training set Y
10.  EndWhile
11.  End
    
```

Fig. 3 Pseudocode of convolutional neural network (CNN)

5 Simulation Results and Discussion

This paper has executed OASIS subjects in the Python IDLE platform [20] for observing the results. Here, CNN algorithm has been implemented on this platform. It explains the loss function score concerning the number of epochs when several subjects are 50. Here, the accuracy percentage varies with increasing the number of periods as described in Figs. 4 and 5.

In Fig. 6, the loss function score is comparatively higher than the previous one. Hence, the accuracy percentages vary mainly as the number of epochs increases (Fig. 7).

In Fig. 8, it describes that the loss function score is high during lower epochs and gradually decreases as the number of epochs increases to 25 for 200 subjects. Hence, the accuracy percentage sharply varies due to increasing the number of subjects, as explained in Fig. 9.

From the above graphs, it can be concluded that 3D MRI data is sufficient for determining Alzheimer's Disease and also other types of diseases such as heart diseases and skin diseases. Hence, this paper introduces a convolutional neural network (CNN) for detecting Alzheimer's Disease. The same kind of approach is there for seeing the other diseases as data is acquired from 3D MRI scans. Furthermore, using deep learning algorithms, another type of 3D and 2D image analysis can be done.

From Fig. 10, it has also been shown that accuracy percentage is highest with 92% when the number of subjects is 200, whereas accuracy percentage is lowest of 85% when the number of subjects is 100. The parameters involved during the execution of hyperparameters of proposed CNN are given in Table 1.

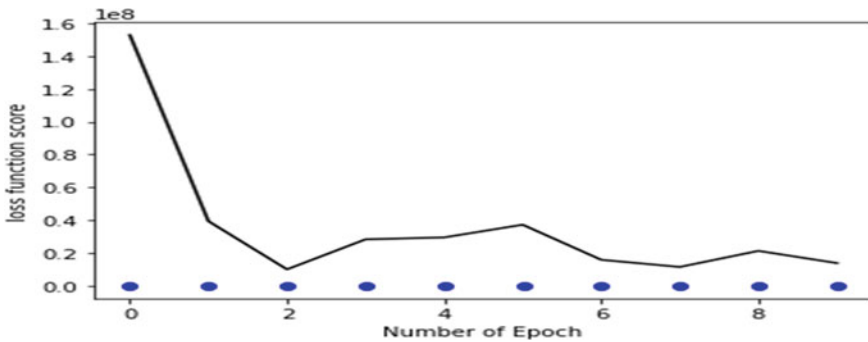


Fig. 4 Graph between the number of epochs and loss function score for 50 subjects

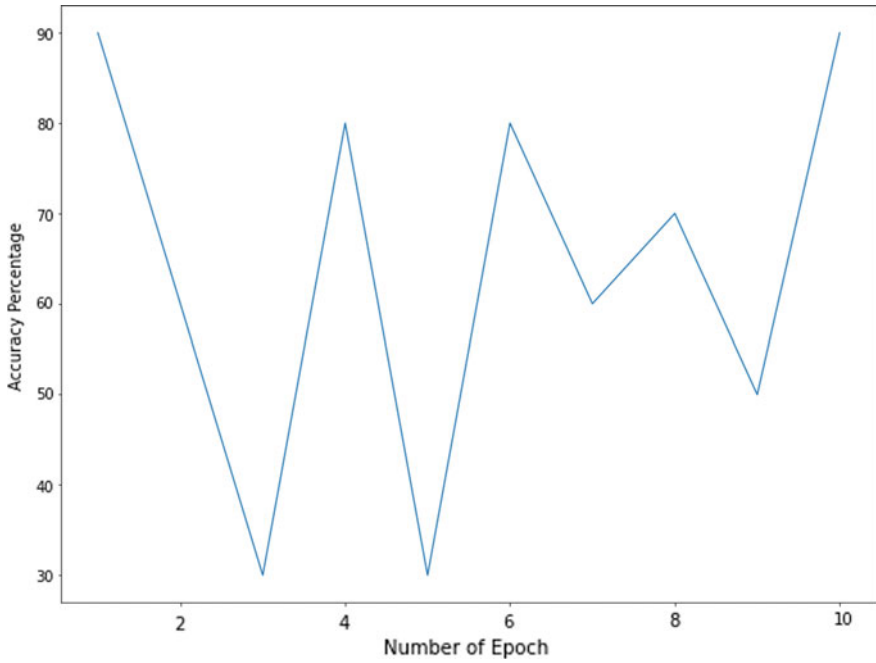


Fig. 5 Graph between epoch number and accuracy percentage for 50 subjects

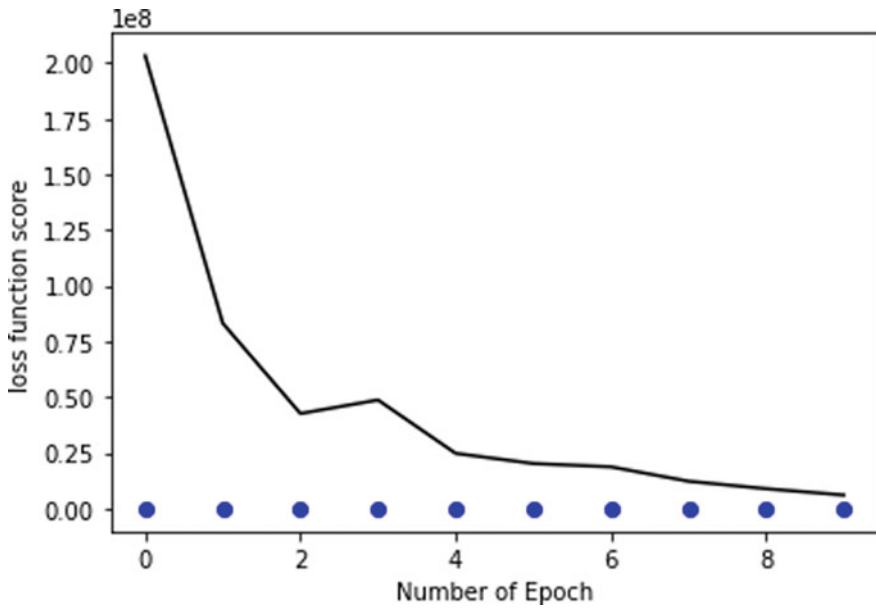


Fig. 6 Graph between the number of epochs and loss function score for 100 subjects

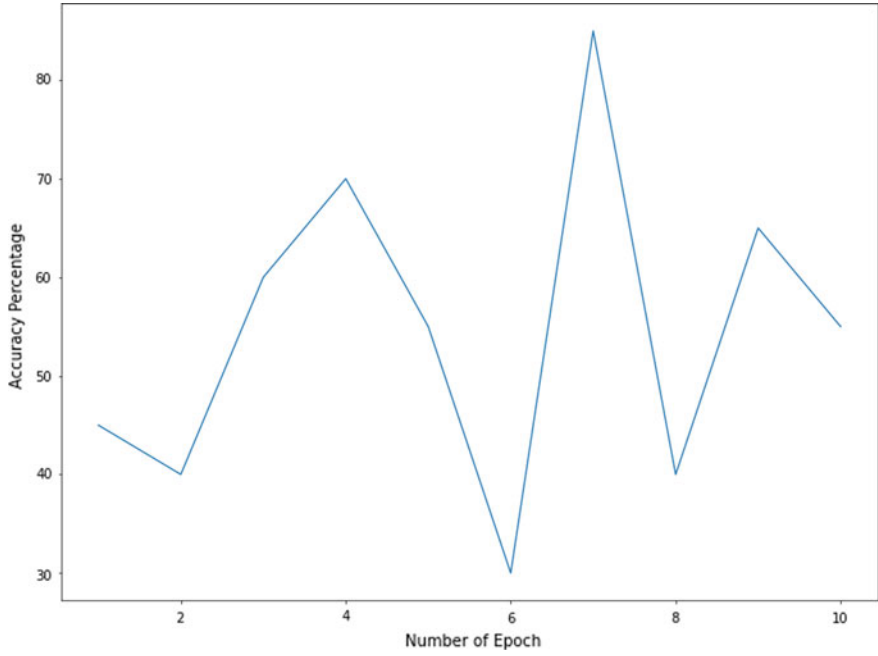


Fig. 7 Graph between the epochs number and accuracy percentage for 100 subjects

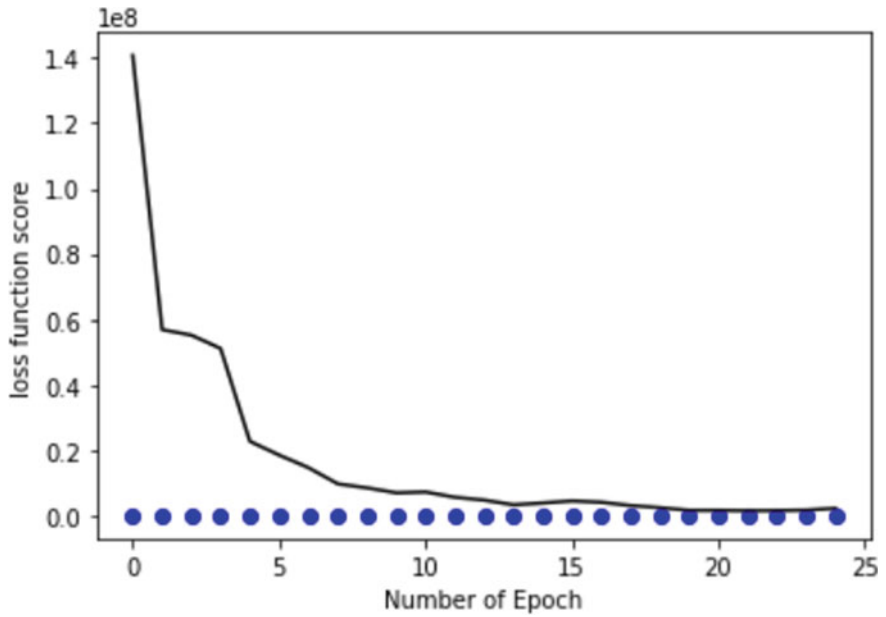


Fig. 8 Graph between epochs number and loss function score for 200 subjects

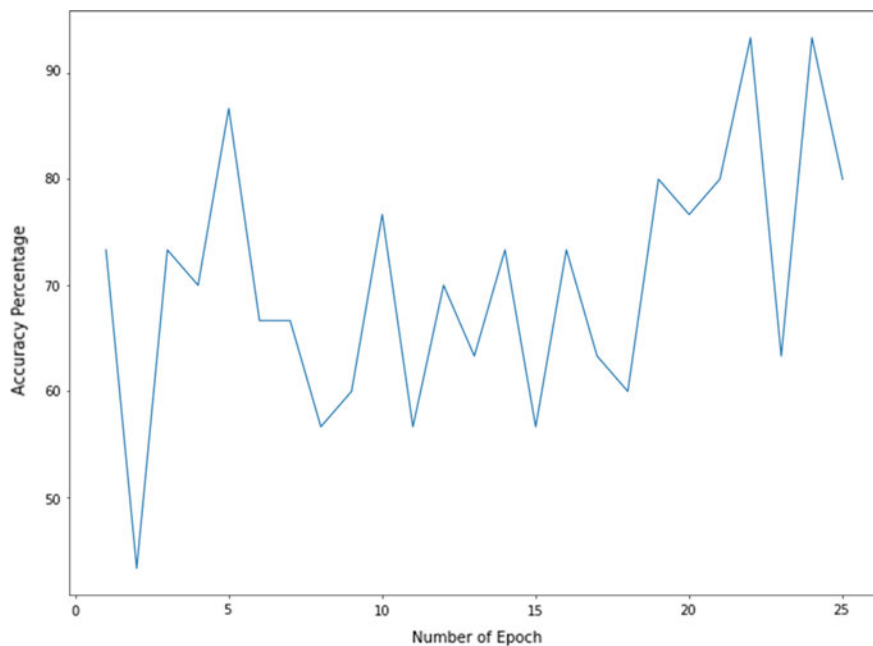


Fig. 9 Graph between the number of epochs and accuracy percentage for 200 subjects

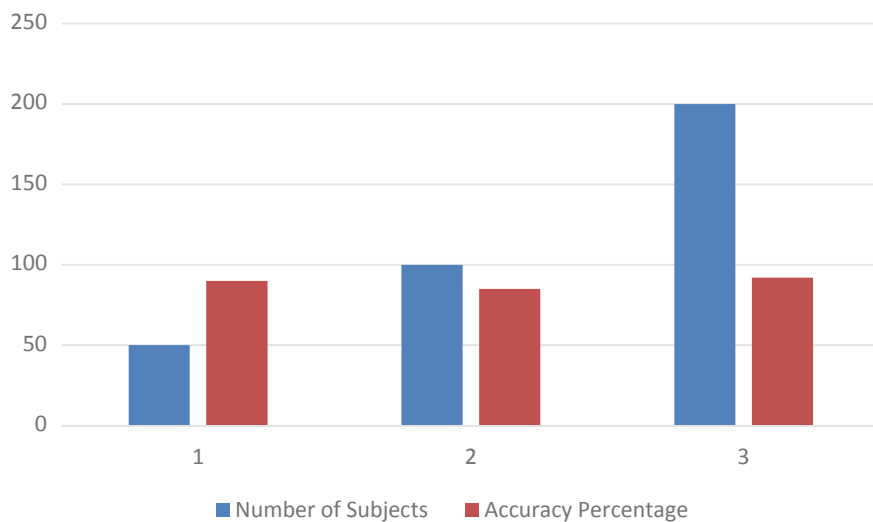


Fig. 10 Bar diagram for accuracy percentage and number of subjects

Table 1 Hyperparameters of proposed CNN

Layer (Type)	Output
conv2d_1(Conv2D)	(None, 28, 28, 32)
max_pooling_2d_1(MaxPooling2)	(None, 12, 12, 32)
conv2d_2(Conv2D)	(None, 10, 10, 64)
max_pooling_2d_2(MaxPooling2)	(None, 4, 4, 64)

6 Conclusion

This paper introduces a novel-based convolutional neural networks (CNNs) to classify AD and NC, where it achieves 85% accuracy when number of subjects taken to be 100 and gradually increases to 92% when there has been increase of 100 subjects. In the future, the study can be conducted on the 3D MRI scans where the network can be highly trained and classified. As most of the deep learning models are not perfect in classifying Alzheimer's Disease, various methods of statistics and probability can be utilized to understand the severity of Alzheimer's Disease. It could form an exciting problem when Bayesian approaches combine deep learning techniques to assess the network's performance under 3D neuroimaging data [21]. Also, models such as VGG-16, ResNet, and LeNET can be utilized for improving the classification accuracy. For these types of models, weights can be upgraded using the gradient descent method so that the classification layer can be deleted [22].

References

1. Prince, M., Graham, N., Brodaty, H., Rimmer, E., Varghese, M., Chiu, H., Acosta, D., Sczufca, M.: Alzheimer Disease international's 10/66 Dementia research group—one model for action research in developing countries. *Int. J. Geriatr. Psychiatry* **19**(2), 178–181 (2004)
2. Virtual Medica Centre, <https://www.myvmc.com/investigations/3dmagnetic-resonanceimaging-3d-mri>
3. Orr, G., Pettersson-Yeo, W., Marquand, A.F., Sartori, G., Mechelli, A.: Using support vector machine to identify imaging biomarkers of neurological and psychiatric disease: a critical review. *Neuroscience Biobehavioural Reviews* **36**(4), 1140–1152 (2012)
4. Acharya, U.R., Fernandes, S.L., WeiKoh, J.E., Ciaccio, E.J., Mohd Fabell, M.K., John Tanik, U., Rajinikanth, V., Hong Yeong, C.: Automated detection of Alzheimer's disease using brain MRI images—a study with various feature extraction techniques. *J. Med. Syst.* **43**(9), 1–14 (2019)
5. Klöppel, S., Stonnington, C.M., Chu, C., Draganski, B., Schill, R.I., Rohrer, J.D., Fox, N.C., Jack, C.R., Jr, Ashburner, J., Frackowiak, R.S.J.: Automatic classification of MR scans in Alzheimer's disease. *Brain* **131**(3): 681–689 (2008)
6. Arvesen, E.: Automatic Classification of Alzheimer's Disease from Structural MRI. Master's thesis (2015)
7. Vemuri, P., Gunter, J.L., Senjem, M.L., Whitwell, J.L., Kantarci, K., Knopman, D.S., Boeve, B.F., Petersen, R.C., Jack, C.R. Jr.: Alzheimer's disease diagnosis in individual subjects using structural MR images: validation studies. *Neuroimage* **39**(3), 1186–1197 (2008)
8. Gray, K.R.: Machine Learning for Image-Based Classification of Alzheimer's Disease. Ph.D. thesis, Imperial College London (2012)

9. Er, A., Varma, S., Paul, V.: Classification of brain MR images using texture feature extraction. *Int. J. Comput. Sci. Eng.* **5**(5), 1722–1729 (2017)
10. Morra, J.H., Tu, Z., Apostolova, L.G., Greens, A.E., Toga, A.W., Thompson, P.M.: Comparison of AdaBoost and support vector machines for detecting Alzheimer's disease through automated hippocampal segmentation. *IEEE Trans. Med. Imaging* **29**(1), 30–43 (2009)
11. Liu, M., Zhang, J., Adeli, E., Shen, D.: Landmark-based deep multi-instance learning for brain disease diagnosis. *Med. Image Anal.* **43**, 157–168 (2018)
12. Pan, D., Zeng, A., Jia, L., Huang, Y., Frizzell, T., Song, X.: Early detection of Alzheimer's disease using magnetic resonance imaging: a novel approach combining convolutional neural networks and ensemble learning. *Front. Neurosci.* **14**, 259 (2020)
13. Ebrahim, D., Ali-Eldin, A.M.T., Moustafa, H.E., Arafat, h.: Alzheimer disease early detection using convolutional neural networks. In: 2020 15th International Conference on Computer Engineering and Systems (ICCES), pp. 1–6. IEEE (2020)
14. LaMontagne, P.J., Benzinger, T.L., Morris, J.C., Keefe, S., Hornbeck, R., Xiong, C., Grant, E., Hassenstab, J., Moulder, K., Vlassenko, A. and Raichle, M.E.: OASIS-3: longitudinal neuroimaging, clinical, and cognitive dataset for normal aging and Alzheimer disease. *MedRxiv* (2019)
15. Zeppa, P., Cozzolino, I.: *Lymph Node FNC: Cytopathology of Lymph Nodes and Extranodal Lymphoproliferative Processes*. Karger Medical and Scientific Publishers (2017)
16. Albawi, S., Mohammed, T.A., Al-Zawi, S.: Understanding of a convolutional neural network. In: 2017 International Conference on Engineering and Technology (ICET), pp. 1–6. IEEE (2017)
17. Srivastava, N., Hinton, G.E., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **15**(1), 1929–1958 (2014)
18. Wang, S.-H., Phillips, P., Sui, Y., Liu, B., Yang, M., Cheng, H.: Classification of Alzheimer's disease based on eight-layer convolutional neural network with leaky rectified linear unit and max pooling. *J. Med. Syst.* **42**(5), 1–11 (2018)
19. Ketkar, N.: Introduction to keras. In: *Deep Learning with Python*, pp. 97–111. Apress, Berkeley (2017)
20. Widodo, C.E., Adi, K., Gernowo, R.: Medical image processing using python and open cv. In: *Journal of Physics: Conference Series*, vol. 1524, no. 1, p. 012003. IOP Publishing, (2020)
21. Gal, Y., Islam, R., Ghahramani, Z.: Deep Bayesian Active Learning with Image Data (2017). arXiv preprint [arXiv:1703.02910](https://arxiv.org/abs/1703.02910)
22. Donahue, J., et al.: Decaf: A deep convolutional activation feature for generic visual recognition. In: *International Conference on Machine Learning* (2014)

Optimum Over Current Relays Coordination for Radial Distribution Networks Using Soft Computing Techniques



Shanker Godwal, S. S. Kanojia, and Akhilesh A. Nimje

Abstract The protective relay coordination for a radial distribution networks are very challenging task with IDMT over current relays. In the recent past, many researchers have suggested several algorithms and mathematical formulation of relay coordination including changes in objective functions (OF) to achieve optimum relay settings for radial distribution network. In this paper, various optimization techniques are compared for achieving optimum over current relay coordination. Analysis has been done on radial distribution systems. Different optimization techniques have been used to achieve the optimum value of plug setting (PS) and time multiplier setting (TMS) for all radial distribution systems. Further, the results achieved from various soft computing techniques are compared with manual calculation of plug setting and time setting multiplier for said distribution network. In this paper, it is observed that said soft computing techniques reduced the operating time of IDMT over current relay up to an acceptable level.

Keywords Over current relay coordination · Relay characteristic · Genetic algorithm and particle swarm optimization techniques · Radial distribution networks

1 Introduction

The electrical power system, a thought that evolved to get, transmission and distribution of electrical energy to the end users area unit primarily large networks connected

S. Godwal · S. S. Kanojia (✉) · A. A. Nimje
Institute of Technology, Nirma University, Gujarat, India
e-mail: sarika.kanojia@nirmauni.ac.in

S. Godwal
e-mail: shanker.godwal@nirmauni.ac.in

A. A. Nimje
e-mail: akhilesh.nimje@nirmauni.ac.in

along that involves multiple pricey instrumentation of various ratings. It is thus necessary to guard these equipment from obtaining damaged for reliable and stable operation of electrical power grid. Protection system, which is associated, is an integral part of electrical power system protection. On the prevalence of fault, circuit breaker (CB) operates and separates the faulty portion of the system from healthy electrical part of the power system. Relays which are connected with instruments transformers sense the abnormal condition and provide trip signal to the CB. Generally, power system is split into many sections or zones, and relays are assigned to operate for a fault in an explicit zone of an electrical power system. Such a relay is considered as a primary relay. A backup is additionally provided considering the situation that a primary relay fails to work in an exceedingly explicit zone or section of a power system. The backup relay is given certain quantity of supposed time delay, in order that it operates on condition that primary relay fails to work. A relay acting as a primary may function a backup relay for neighboring zones. On failure of primary relay, after a suggested interval, backup relay should operate [1]. This is often called coordination of relaying scheme. The various MATLAB solvers using different soft computing techniques such as harmony search, Genetic algorithm, particle swarm optimization (PSO), etc. Have been taken into consideration for solving the objective function and the results obtained are compared with the conventional approach. Plug setting (PS) and time multiplier setting (TMS) obtained exploitation, and these solvers are compared to every other. The radial and parallel feeder networks are taken for the comparison purpose.

Motivation: To achieve fast and healthy protective scheme, it is required to obtain the optimum value of plug setting and time multiplier setting of the relay. Many researcher has suggested philosophy regarding change in objective function, change in relay constraints, change in relay characteristic or to adopt soft computing techniques to get reliable and healthy operation in power system. Considering the said issues, it was observe that comparative analysis of various soft computing techniques to be done and specially for the new researchers in the over current relay coordination for radial system and parallel distribution system.

1.1 Contribution

The main contribution of this paper discusses,

- Overview of various approaches for optimum relay coordination in distributions and sub-transmission system.
- The various types of relay characteristic and objective function for relay coordination problem.
- This future scope in the field of over current relay coordination which will motivate the new researcher in this field.

1.2 Paper Organization

The rest of the paper is outlined as follows. In Sect. 2, formulation of over current relay coordination problem is presented. Various constraint and objective function have been discussed. Also various types of objective functions proposed in past literature, robustness and drawback of proposed objective functions is discussed, approach based on optimization methods has been presented. Finally, in Sect. 3, the results using different solvers and conclusions of this paper are highlighted.

2 Problem Formulation for Over Current Relay Coordination

The coordination of relay problem has been consider either as a linear or nonlinear problem. Just in case of linear programming problem, the pickup current setting of relay has been kept fixed, and time multiplier setting of the relay is optimized. The values of current setting of the relay are mounted between maximum load current to minimum fault current. However, just in case of nonlinear programming issues, each TMS and plug setting of relay unit optimized at the same time with relay characteristic [2]. The complexness of the matter can increase because of distinct nature of plug setting and TMS setting [3]. That's why this can be mixed integer nonlinear programming problem. Additionally, within the uncertainties of load and topological changes, acquired setting of annotation and TMS don't seem to be optimum over all doable eventualities. Hence, the relay coordination issues should be modifying to think about all probable situation. Hence, the relay coordination problems must be modifying to consider all possible scenario.

The objective function given in Eq. 1.

$$\text{OFI} = \sum_{i=1}^m w_i t_i \quad (1)$$

Relay characteristics is as follows:

$$t_i = \frac{0.14 \times \text{TMS}_i}{(I_{c,i}/PS_i)0.02 - 1} \quad (2)$$

Various constraints are shown below:

Coordination constraints, operating time (OT),

$$\text{OT}_{i_backup} - \text{OT}_{j_primary} \geq \text{CTI}$$

Bounds on time multiplier setting (TMS)

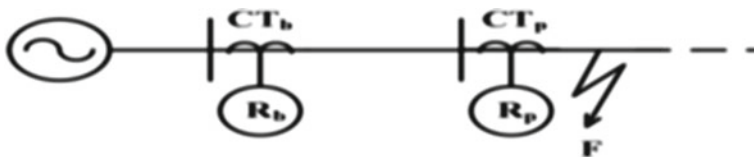


Fig. 1 Primary and backup relay pair

$$TMS_{i,\min} \leq TMS_i \leq TMS_{i,\max}$$

Bounds on plug setting (PS)

$$PS_{i,\min} \leq PS_i \leq PS_{i,\max}$$

Bounds on operating time

$$t_{i,\min} \leq t_i \leq t_{i,\max}$$

The primary and backup relay pair is shown in Fig. 1.

3 Results and Discussion

In order to evaluate the performance of various types of optimization solver, two main distribution systems have been used. These are radial feeder and parallel feeder distribution system. Radial 3 bus feeder system shown in Fig. 2 optimizes result details which are given in Tables 1, 2, 3, 4, 5, 6, 7, 8, 9, and 10, and parallel feeder network is shown in Fig. 3 and its related results are given in Tables 11, 12, 13, 14, 15, 16, 17 and 18 using different optimization techniques. Various solver [8–21] which are available in MATLAB tool box [20] are used to observe the effect on optimum value of the objective function using various approaches like conventional approach, FSolver algorithm, minmax optimization, fmincon optimization, particle swarm optimization (PSO) algorithm, pattern search method, Genetic algorithm (GA) [21–24] and using harmony search algorithm. The same is given in Tables 1, 2, 3, 4, 5, 6, 7, 8, and 9.

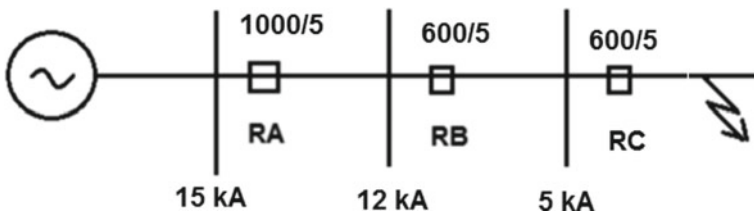


Fig. 2 3 bus radial feeder system

Table 1 Calculation of PS and TMS individual relay for radial feeder 3 bus system using conventional approach (CA)

Relay	RA	RB	RC
Plug setting (PS)	75%	100%	75%
Time multiplier setting (TMS)	0.3	0.2	0.1
Operating time (sec.)	0.043533	0.290221	0.167343
Sum of all relays operating time (sec.)	0.501098		

Table 2 Calculation of PS and TMS individual relay for radial feeder3 bus system using FSolver algorithm (FA)

Relay	RA	RB	RC
Plug setting (PS)	70.13%	99.53%	75.37%
Time multiplier setting (TMS)	0.32	0.183	0.14
Operating time (sec.)	0.045737	0.265268	0.23458
Sum of all relays operating time (sec.)	0.545585		

Table 3 Calculation of PS and TMS individual relay for radial feeder 3 bus system using minmax optimization (MO)

Relay	RA	RB	RC
Plug setting (PS)	74.13%	99.65%	72.69%
Time multiplier setting (TMS)	0.302	0.188	0.162
Operating time (sec.)	0.043707	0.272591	0.268916
Sum of all relays operating time (sec.)	0.585214		

Table 4 Calculation of PS and TMS individual relay for radial feeder 3 bus system using fmincon optimization (FO)

Relay	RA	RB	RC
Plug setting (PS)	76.36%	102.45%	74.26%
Time multiplier setting (TMS)	0.323	0.29	0.24
Operating time (sec.)	0.047063	0.423148	0.400595
Sum of all relays operating time (sec.)	0.870806		

Table 5 Calculation of PS and TMS individual relay for radial feeder 3 bus system using PSO algorithm (PSO)

Relay	RA	RB	RC
Plug setting (PS)	76.72%	98.23%	73.70%
Time multiplier setting (TMS)	0.35	0.27	0.13
Operating time (sec.)	0.051052	0.390214	0.216565
Sum of all relays operating time (sec.)	0.657831		

Table 6 Calculation of PS and TMS individual relay for radial feeder 3 bus system using pattern search method (PSM)

Relay	RA	RB	RC
Plug setting (PS)	73.56%	101.39%	74%
Time multiplier setting (TMS)	0.36	0.26	0.102
Operating time (sec.)	0.052011	0.378475	0.170099
Sum of all relays operating time (sec.)	0.600584		

Table 7 Calculation of PS and TMS individual relay for radial feeder 3 bus system using Genetic algorithm (GA)

Relay	RA	RB	RC
Plug setting (PS)	73.45%	105.2%	77.54%
Time multiplier setting (TMS)	0.40	0.35	0.23
Operating time (sec.)	0.05777	0.513804	0.388239
Sum of all relays operating time (sec.)	0.959813		

Table 8 Calculation of PS and TMS individual relay for radial feeder3 bus system using harmony search algorithm (HSA)

Relay	RA	RB	RC
Plug setting (PS)	71.23%	98.43%	73.32%
Time multiplier setting (TMS)	0.31	0.193	0.12
Operating time (sec.)	0.044463	0.279059	0.19964
Sum of all relays operating time (sec.)	0.523162		

Table 9 Calculation of PS and TMS individual relay for radial feeder 3 bus system using quadratic programming (QP)

Relay	RA	RB	RC
Plug setting (PS)	74.34%	96.23%	74.39%
Time multiplier Setting (TMS)	0.39	0.246	0.220
Operating time (sec.)	0.05648	0.35388	0.367378
Sum of all relays operating time (sec.)	0.777737		

Table 10 Calculation of PS and TMS individual relay for parallel feeder using conventional approach (CA)

Relay	RA	RB	RC	RD
Plug setting (PS)	75%	75%	50%	50%
Time multiplier setting (TMS)	0.3	0.3	0.2	0.2
Operating time (sec.)	0.851281	0.851281	0.483748	0.483748
Sum of all relays operating time (sec.)	2.670058			

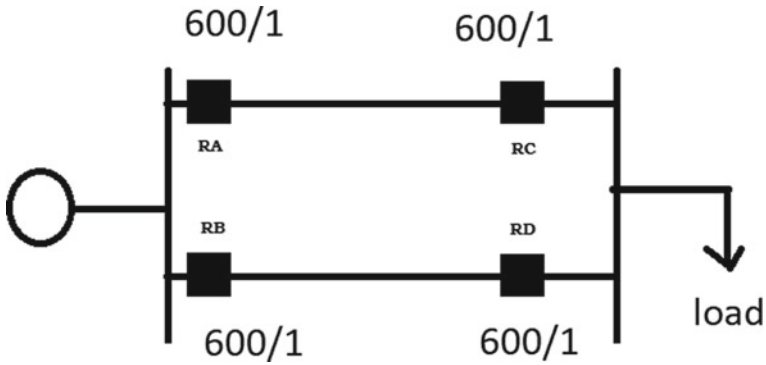


Fig. 3 Parallel feeder

Table 11 Calculation of PS and TMS individual relay for parallel feeder using FSolver algorithm (FA)

Relay	RA	RB	RC	RD
Plug setting (PS)	69.34%	69.34%	58.32%	58.32%
Time multiplier setting (TMS)	0.293	0.293	0.134	0.134
Operating time (sec.)	0.804543	0.804543	0.343403	0.343403
Sum of all relays operating time (sec.)	2.295892			

Table 12 Calculation of PS and TMS individual relay for parallel feeder using minmax optimization (MO)

Relay	RA	RB	RC	RD
Plug setting (PS)	65.01%	65.01%	52.45%	52.45%
Time multiplier setting (TMS)	0.26	0.26	0.2	0.2
Operating time (sec.)	0.69543	0.69543	0.492353	0.492353
Sum of all relays operating time (sec.)	2.375567			

Table 13 Calculation of PS and TMS individual relay for parallel feeder using fmincon optimization (FO)

Relay	RA	RB	RC	RD
Plug setting (PS)	71.21%	70.21%	52.36%	52.36%
Time multiplier setting (TMS)	0.234	0.234	0.178	0.178
Operating time (sec.)	0.649662	0.649662	0.437915	0.437915
Sum of all relays operating time (sec.)	2.175154			

Table 14 Calculation of PS and TMS individual relay for parallel feeder system using PSO algorithm (PSO)

Relay	RA	RB	RC	RD
Plug setting (PS)	75%	75%	63.23%	63.23%
Time multiplier setting (TMS)	0.312	0.312	0.198	0.198
Operating time (sec.)	0.885333	0.885333	0.523749	0.523749
Sum of all relays operating time (sec.)	2.818162			

Table 15 Calculation of PS and TMS individual relay for parallel feeder using pattern search method (PSM)

Relay	RA	RB	RC	RD
Plug setting (PS)	76.39%	76.39%	51.56%	51.56%
Time multiplier setting (TMS)	0.346	0.346	0.298	0.298
Operating time (sec.)	0.989539	0.989539	0.728968	0.728968
Sum of all relays operating time (sec.)	3.437016			

Table 16 Calculation of PS and TMS individual relay for parallel feeder using Genetic Algorithm(GA)

Relay	RA	RB	RC	RD
Plug setting (PS)	74.76%	74.76%	56.79%	56.79%
Time multiplier setting (TMS)	0.349	0.349	0.174	0.174
Operating time (sec.)	0.988975	0.988975	0.441379	0.441379
Sum of all relays operating time (sec.)	2.860709			

Table 17 Calculation of PS and TMS individual relay for parallel feeder using harmony search algorithm (HSA)

Relay	RA	RB	RC	RD
Plug setting (PS)	62.12%	62.12%	59.93%	59.93%
Time multiplier setting (TMS)	0.42	0.42	0.24	0.24
Operating time (sec.)	1.103206	1.103206	0.621583	0.621583
Sum of all relays operating time (sec.)	3.449579			

Table 18 Calculation of PS and TMS individual relay for parallel feeder using quadratic programming (QA)

Relay	RA	RB	RC	RD
Plug setting (PS)	62.43%	62.43%	61.23%	61.23%
Time multiplier setting (TMS)	0.13	0.13	0.19	0.19
Operating time (sec.)	0.342142	0.342142	0.496239	0.496239
Sum of all relays operating time (sec.)	1.676761			

It has been observed that harmony search method is much better as compared to others method for 3 bus radial feeder, and quadratic programming method is better for parallel feeder network as it gives minimum operating time with best plug setting (PS) and time multiplier setting (TMS). Figures 4 and 5 show the sum of operating time for various soft computing techniques for radial feeder and parallel feeder, respectively, and Figs. 6 and 7 show operating time of individual relay considering various soft computing techniques for radial and parallel feeder, respectively. The main limitation of soft computing techniques used in the paper can be understand that harmony search algorithm is giving optimum results for radial distribution system, whereas quadratic programming method is giving optimum result for parallel feeder which can be overcome by adopting new soft computing techniques or making some changes in these algorithm.

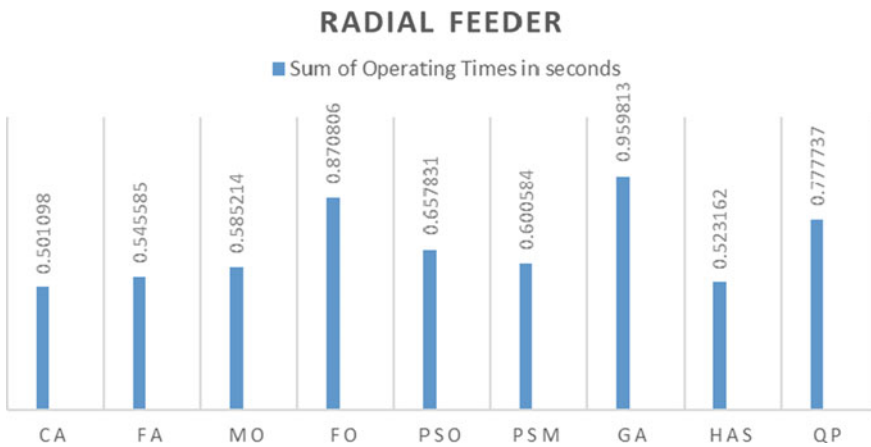


Fig. 4 Sum of operating time for various soft computing techniques for radial feeder

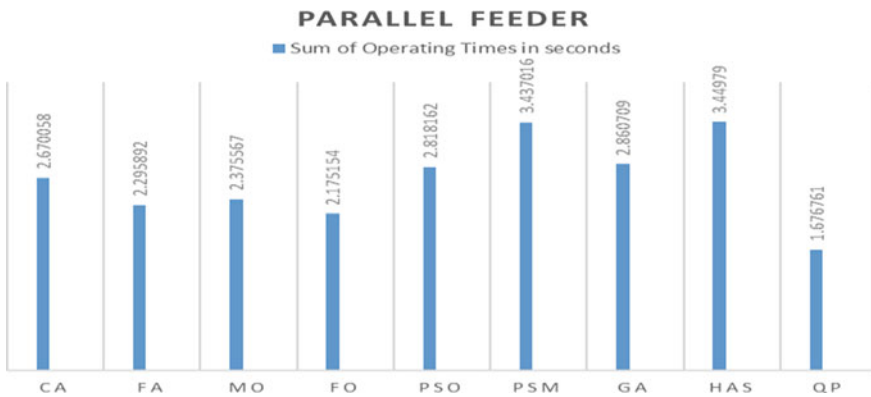


Fig. 5 Sum of operating time for various soft computing techniques for parallel feeder

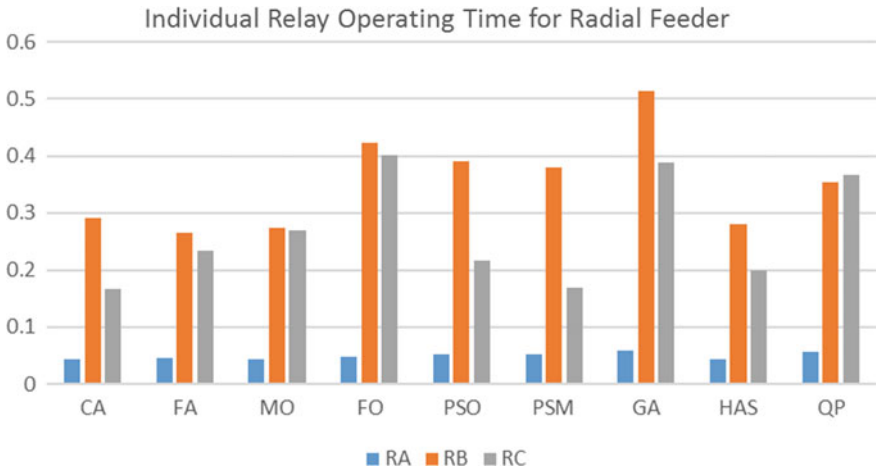


Fig. 6 Operating time of individual relay considering various soft computing techniques for radial feeder

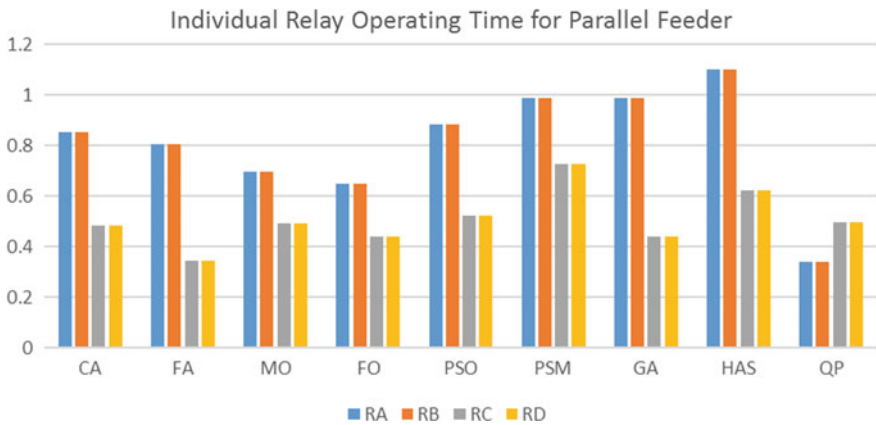


Fig. 7 Operating time of individual relay considering various soft computing techniques for parallel feeder

Conclusion and Future scope: It has been observed that based on system configuration, harmony search method is much better as compared to others method for 3 bus radial feeder and quadratic programming method which is better for parallel feeder network as it gives minimum operating time with optimum value of plug setting and time multiplier setting. Further, the work can be extended for complex power system network and for different protection scheme too.

References

1. Alam, M.N.: Adaptive protection coordination scheme using numerical directional over current relays. *IEEE Trans. Ind. Inf.* **15**(1) (2019)
2. Yu, J.: Oppositional jaya algorithm with distance adaptive coefficient in solving directional over current relays coordination problem. *IEEE Access* **7**, 150729 (2019)
3. Bedekar, P.P., Bhide, S.B., Kale, V.S.: Optimum coordination of overcurrent relays in distribution system using genetic algorithm. In: 2009 Third International Conference on Power Systems, Kharagpur, INDIA December 27–29, ICPS—247 (2009)
4. Rajput, V.N., Pandya, K.S.: On 8-bus test system for solving challenges in relay coordination. In: 2016 IEEE 6th International Conference on Power Systems (ICPS) (2016)
5. Rajput, V.N., Pandya, K.S.: coordination of directional overcurrent relays in the interconnected power systems using effective tuning of harmony search algorithm. *Sustain. Comput. Inf. Syst.* **15**, 1–15 (2017)
6. Coleman, T.F., Li, Y.: An interior, trust region approach for nonlinear minimization subject to bounds. *SIAM J. Optim.* **6**, 418–445 (1996)
7. Coleman, T.F., Li, Y.: On the convergence of reflective newton methods for large-scale nonlinear minimization subject to bounds. *Math. Program.* **67**(2), 189–224 (1994)
8. Gill, P.E., Murray, W., Wright, M.H.: *Practical Optimization*. Academic Press, London (1981)
9. Han, S.P.: A globally convergent method for nonlinear programming. *J. Optim. Theor. Appl.* **22**, 297 (1977)
10. Powell, M.J.D.: A Fast algorithm for nonlinear constrained optimization calculations. In: Watson, G.A. (ed.) *Numerical Analysis. Lecture Notes in Mathematics*, vol. 630. Springer, Berlin (1978)
11. Powell, M.J.D.: The convergence of variable metric methods for nonlinearly constrained optimization calculations. In: Mangasarian, O.L., Meyer, R.R., Robinson, S.M. (eds.) *Nonlinear Programming 3*. Academic Press (1978)
12. Dennis, J.E. Jr.: Nonlinear least-squares. In: Jacobs D (ed) *State of the Art in Numerical Analysis*. Academic Press, pp 269–312
13. Levenberg, K.: A method for the solution of certain problems in least-squares. *Quart. Appl. Math.* **2**, 164–168 (1944)
14. Marquardt, D.: An algorithm for least-squares estimation of nonlinear parameters. *SIAM J. Appl. Math.* **11**, 431–441 (1963)
15. Moré, J.J.: The levenberg-marquardt algorithm: implementation and theory. In: Watson, G.A. (ed.) *Numerical Analysis. Lecture Notes in Mathematics*, vol. 630. Springer, Berlin, pp 105–116 (1977)
16. Moré, J.J., Garbow, B.S., Hillstom, K.E.: User guide for MINPACK 1, Argonne National Laboratory, Rept. ANL-80-74 (1980)
17. Powell, M.J.D.: A fortran subroutine for solving systems of nonlinear algebraic equations. In: Rabinowitz, P. (ed.) *Numerical Methods for Nonlinear Algebraic Equations*, Chap. 7 (1970)
18. Brayton, R.K., Director, S.W., Hachtel, G.D., Vidigal, L.: A new algorithm for statistical circuit design based on Quasi-Newton methods and function splitting. *IEEE Trans. Circ. Syst.* **CAS-26**, 784–794 (1979)
19. Grace, A.C.W.: Computer-aided control system design using optimization techniques, Ph.D. Thesis, University of Wales, Bangor, Gwynedd, UK (1989)
20. [apps/matlabhelp/toolbox/optim/ref_int6.html](https://matlabhelp/toolbox/optim/ref_int6.html)
21. Amraee, T.: Coordination of directional overcurrent relays using seeker algorithm. *IEEE Trans. Power Del.* **27**(3), 1415–1422 (2012). <https://doi.org/10.1109/TPWRD.2012.2190107>
22. Venkata Rao, R.: Jaya: a simple and new optimization algorithm for solving constrained and unconstrained optimization problems. *Int. J. Ind. Eng. Comput.* **7**(1), 19–34 (2016). <https://doi.org/10.5267/j.ijiec>

23. Moravej, Z., Mohaghegh Ardebili, H.: A new objective function for adaptive distance and directional over-current relays coordination. *Int. Trans. Electr. Energy Syst.* (2018). <https://doi.org/10.1002/etep.2592>
24. Moravej, Z., Adelnia, F., Abbasi, F.: Optimal coordination of directional overcurrent relays using NSGA-II. *Electr. Power Syst. Res.* **119**, 228–236 (2014). ISSN 0378–7796. <https://doi.org/10.1016/j.epsr.2014.09.0>

A Review of Modified Particle Swarm Optimization Method



Priyavada and Binay Kumar

Abstract Particle swarm optimization (PSO) is a nature-inspired population-based evolutionary and stochastic optimization method to solve optimization problems. In PSO, particles are communicating to each other using search directions. Nowadays, PSO is advanced search algorithm and metaheuristic technique which is used in different areas of applications. Some drawbacks of PSO algorithm are local optimum solution and low convergence rate. Many researchers have modified original PSO to remove the drawback for the improvement of performance and convergence-related problems of PSO algorithm. In this paper, we have presented a review on the modified particle swarm optimization algorithms in the direction of inertia weight, discrete particle swarm optimization (DPSO), parallel particle swarm optimization (PPSO), and perspective of convergence. An attempt is made to present a systematic framework for the researchers working in the area of PSO.

Keywords Optimization problem · Particle swarm optimization · Metaheuristic · Inertia weight · Convergence

1 Introduction

PSO algorithm is the swarm-based intelligence algorithm. It is a modern optimization evolutionary technique inspired by the social behavior of flock of birds. Being its origin in 1995, time to time several modifications of PSO have been developed with many application areas. PSO is widely used in many fields because it is very easy to implement, and give the result after each iteration. After each iteration, it gives the two best values Pbest and Gbest. 1st value is related to the cognitive part, and 2nd value is related to the social part of the PSO algorithm. Theory of PSO algorithm is rapidly increasing. PSO algorithm simulates the behavior of bird's societies that

Priyavada (✉) · B. Kumar

Department of Mathematics, Lingaya's Vidyapeeth, Faridabad (Delhi NCR), India
e-mail: priyavada.parihar@gmail.com

B. Kumar

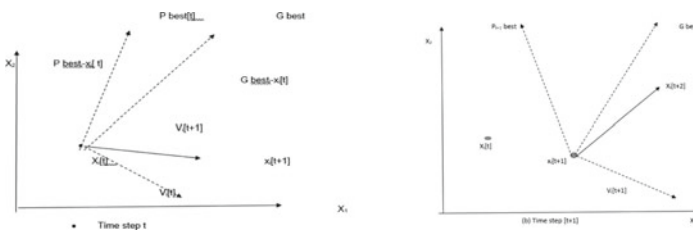
e-mail: binay@lingayasvidyapeeth.edu.in

do not have any leaders in their swarm. Each particle communicates with the other members of the swarm that have the nearest position with the food source (optimal point). This process shall be repeated several times till the best optimal point is obtained. The result of PSO totally depends on two parameters such as the inertia weight and two acceleration factors. The role of inertia weight is considered most important in PSO. Therefore, a proper control of inertia weight is considered very important to find the optimum solution. Shi and Eberhart made an improvement in the convergence of the PSO with a linearly varying inertia weights over the iterations. PSO is an evolutionary computational technique for updating velocity and position which is defined as

$$V_i[t + 1] = V_i[t] + c_1 * r_1 * (Pbest_i[t] - X_i[t]) + c_2 * r_2 * (Gbest_i - X_i[t + 1])$$

$$X_i[t + 1] = X_i[t] + V_i[t + 1]$$

velocity and position update for a particle in a two-dimensional search space



Velocity and Position update for Multi-particle in Gbest PSO.



where V_i are the velocities of the particles and X_i are positions of the particles. Pbest is the value of the personal best particle, and Gbest is value of the global best particles. Two random numbers r_1 and r_2 uniformly lie between 0 and 1. c_1 and c_2 are the two acceleration factors. PSO has a lot of applications which are discussed by different researchers. In different field like energy storage, signal processing, Networking, Design, Image and Graphics, Modeling, etc., PSO gave the best result. Naziha Ahamd Azli [1] established applications on power converter systems. Main focus of this paper is on the implementation of the harmonic elimination pulse with

modulation. Mohammed Amine Metiane [2] recently introduced about economic dispatch problem where effect of inertia weight is discussed in the problem.

Chu [3] introduced the timetable scheduling problem in PSO, where examination subjects must be settled to certain times that satisfy several of constraints. Dali [4] discussed the PSO applications based on GPU-PSO: Parallel PSO approaches on Graphical Processing Unit for constraint reasoning case of Max-CSPs. Shao [5] discussed the new application of PSO, which is PSO algorithm based on semantic relation and its engineering applications. In this paper, we have presented a review on the modified Particle swarm optimization algorithms in the direction of inertia weight, discrete particle swarm optimization (DPSO), parallel particle swarm optimization (PPSO), and perspective of convergence. Main application of Inertia weight is to control the velocity and balance between exploration process and exploitation process of swarm.

2 Review on Inertia Weight Methods

As long as the evolution of PSO algorithm, different types of PSO methods have been introduced. When the original PSO was introduced, there was no inertia weight in velocity updating equation. First modification in PSO was related to inertia weight which is applied to the velocity vector. Inertia weight plays main role in PSO technique which controls the process of local and global optimal point by maintaining a proper balance between in their capabilities. The best value of inertia weight lies between 0.4 and 0.9. Shi and Eberhart [6] proposed the inertia weight in 1998 to improve the PSO convergence speed. Inertia weight is multiplied to the velocity vector V . This helps in maintaining balance between exploration and exploitation. Larger value of w helps in exploration, and small value of w helps in exploitation in the search area. With the inertia weight, the velocity updating equation has given the better performance.

$$V_i[t + 1] = V_i[t] * w + c_1 * r_1 * (Pbest_i[t] - X_i[t]) + c_2 * r_2 * (Gbest_i - X_i[t + 1])$$

Now, we discuss different type of inertia weight strategies.

- 2.1 **Constant value of Inertia weight:** Constant I.W was proposed by Shi and Eberhart [7] in 1998 between the range [0.8, 1.2]. When the value of Inertia weight is 0.8, then PSO traps in local optimal point. By this result, they noticed that within the range (0.8, 1.2) PSO can find optimal solution in reasonable number of iterations.
- 2.2 **Random Inertia weight:** Eberhart RC, Shi [8] in 2001 introduced random I.W to permit the PSO algorithm to find the optimal point in a dynamic search space.

$$\omega = \frac{1 + \text{Rand}()}{2}$$

where $\text{rand}()$ is the random number which lies between 0 and 1 and value of inertia weigh is $W = [0.5, 1]$

- 2.3 **Adaptive Inertia Weight:** It was introduced by Clerc [9] where all the modification in the I.W is directly proportional to the comparative development of the particles. Here, this is adjusted as given below

$$w_i(t + 1) = w(0) + (w(I_{\max}) - w(0)) \times \frac{e^{m_i(t)-1}}{e^{m_i(t)+1}}$$

The relative improvement m_i is estimated as

$$m_i(t) = \frac{f(\text{gbest}(t)) - f(x_i(t))}{f(\text{gbest}(t)) + f(x_i(t))}$$

where maximum inertia weight is 0.5 and $(0) < 1$

- 2.4 **Flexible Exponent Inertia Weight:** Flexible exponent Inertia weight (FEIW) was proposed by Amoshahy [10]. Here, two inertia weight w_1 and w_2 are used. One is used in starting of the iteration, and 2nd is used at the end of a given run

$$w(0) = w_1, w_1 > 0 \quad \text{and} \quad \omega(I_{\max}) = w_2 \quad \text{where} \quad w_2 > 0.$$

FEIW strategy can adopt with any optimization problem. In suitable range and parameters, we can create many types of inertia weights like increasing and decreasing. In this way, FEIW encloses wide range of I.W. Line up of I.W strategies in FEIW one can easily change w_1 , w_2 , and φ for better optimal point. Here, w_1 , w_2 , and φ are three positive real number, and modified inertia weight is defined as

$$w(t) = \frac{-\varphi t}{\alpha_1 e^{t \max}} + \frac{\varphi t}{\alpha_2 e^{t \max}}$$

where

$$\alpha_1 = \frac{W \cdot 2e^{\varphi - \varphi e^{2\varphi}}}{1 - e^{2\varphi}}$$

$$\alpha_2 = \frac{w_1 - w_2 e^{\varphi}}{1 - e^{2\varphi}}$$

T is the current iteration time and I_{\max} = maximum iteration time.

- 2.5 **Butterworth Inertia Weight Strategy:** Butterworth Inertia Weight Strategy proposed by Zhu [11] has two main phases that are exploration and exploitation. In this strategy, there is more chance with every particle to visit more position in the large search space. This strategy is new and inspired by the Butterworth. It has a simple idea that the particles visit more positions in the search space. In the first step, value of inertia weight should be large so that all the particles have more chance for visiting in the search space and in 2nd step the value of w should be relatively small, so the accuracy of the optimal solution is guaranteed.

$$w(t) = w_{\max} * \frac{1}{1 + \left(\frac{t}{p_1}\right)^{p_2}} + w_{\min}$$

where $w(t)$ varies with iteration, $p_1 = 1/3$ of the highest number of iteration and $p_2 = 10$, highest inertia weight is set 0.5, and lowest inertia weight is set to be 0.4.

- 2.6 **Linear policies to adjust I.W:** This strategy was proposed by Eberhart and Shi [6] in 2000. With the help of linear strategies, we can improve the fine-tuning characteristic of PSO. In this method, inertia weight value is decreased from highest to lowest. In the beginning, value is large in the search space but this value is decreased to the final position. As the number of iterations is increased, the value of inertia weight is decreased. The equation is

$$w(t) = \frac{t_{\max} - t}{t_{\max}} (w_{\max} - w_{\min}) + w_{\min}$$

where t is the number of iteration and t_{\max} is highest number of iteration.

Zheng [12] had proposed an increasing inertia weight strategy. This strategy has a great effect on PSO convergence. Convergence speed is very high. Value of Inertia weight lies between (0.4 and to 0.9)

$$\omega(t) = 0.5 \times \frac{t}{t_{\max}} + 0.4$$

- 2.7 **Non-linear strategies to adjust Inertia weight:** Self-regulation particle swarm optimization (SRPSO) proposed by Tanweer [13] was introduced, which consists of a self-regulating inertia weight. It is used to find out by the best particles for better optimal point. Experimental result performs that this SRPSO has better optimal point and better result in comparison with other problems. This non-linear inertia weight strategy improved the search ability of PSO.

$$w(t) = w_i(t) - \Delta w$$

$$w(t) = w_i(t) + \eta w$$

where $\Delta\omega = \frac{\omega_{\text{start}} - \omega_{\text{end}}}{N_f}$ η : a is constant

- 2.8 **Fuzzy rule to adjust inertia weight:** Liu [14] had proposed Fuzzy rule to adjust inertia weight where two values are set for input and two values as output (FLC) was calculated. In starting work on this topic, Shi et al first used two values as input and got a result one value as output with fuzzy logic controller for best optimal point. The main approach to develop FPSO is to modify inertia weight. Adaptive FLC is applicable in dynamic search space. Two variables are used: first is input variable and other is used as a current velocity of the particles. Control the change of the velocity is determined by V_c .

$V_c = e - [10(1 + V_{ck})]$ and a new velocity updating formula is given below

$$V_{ij}(t+1) = w * V_{ij}(t) + c_1 * r_1 * (x_{ij}^*(t) - x_{ij}(t)) + c_2 * r_2 * (x_{ij}^*(t) - x_{ij}(t))$$

$$V = \begin{cases} V_{ij}, & \text{if } |V_{ij}| \geq V_c \\ \frac{u(-1,1)V_{\max}}{\rho}, & \text{if } |V_{ij}| < V_c. \end{cases}$$

- 2.9 **Random inertia weight:** Zhang [15] had proposed another random strategy to adjust inertia weight. This method may strain the assumptions of the I.W adjustable when they are randomly chosen. There is an effective stability between global and local point search potential.

$$w = \begin{cases} a_{1+\frac{r}{20}} & K > 0.05 \\ a_{2+\frac{r}{20}} & K < 0.05 \end{cases}$$

where K is given as $f(t) - f(t-10)/f(t+10)$

And $r \in [0, 1]$ $[0, 1]$, let $\alpha_1 > \alpha_2$, if $K \geq 0.05$ the expectation value.

$$E(w) = \alpha_1 + 0.25, \text{ otherwise}$$

$$E(w) = \alpha_2 + 0.25$$

- 2.10 **Adaptive Strategy for Parameters:** Zhan [16] had proposed this Inertia weight strategy which is used to equal the exploration and exploitation in the search space. In this method, I.W is applied linearly decreasing and one mapping $W(f) : R \rightarrow R$ is used for inertia weight.

$$W(f) = \frac{1}{1 + 1.5e^{-2.6f}} [0.4, 0.9], \quad \forall f \in [0, 1]$$

- 2.11 **Fixed inertia weight:** Hsieh et al. [17] proposed fixed inertia weight in 2009

$$W = \frac{1}{2 * \ln(2)}$$

2.12 **Linearly decreasing inertia weight:** Shi and Eberhart [18] introduced LDIW in 1999 and defined as

$$W = W_{\max} - \frac{W_{\max} - W_{\min}}{\text{Iter}_{\max}} \times k$$

2.13 **Simulated annealing inertia weight:** Park [19] introduced this inertia weight in 2011.

$$W_k = W_{\min} + (W_{\max} - W_{\min}) + \gamma^{(k-1)}$$

Table of modified inertia weight

S. No.	Inertia weight	Formula of inertia weight	References
1	Constant I.W	$W = c(\text{constant})$ Lies between 0.8 and 1.2	[7]
2	Random inertia weight	$w = \frac{1+\text{Rand} ()}{2}$	[8]
3	Adaptive inertia weight	$\omega_i (t + 1) = \omega (0) + (\omega (I_{\max}) - \omega (0)) \frac{e^{m_i (t)-1}}{e^{m_i (t)+1}}$ Where $m_i(t) = \frac{f(\text{gbest}(t))-f(x_i(t))}{f(\text{gbest}(t))+f(x_i(t))}$	[9]
4	Flexible exponent inertia weight	$\omega(t) = \frac{-\varphi t}{\alpha_1 e^{I_{\max}}} + \frac{\varphi t}{\alpha_2 e^{I_{\max}}}$ Here $\alpha_1 = \frac{\omega_1 - \omega_2 e^{2\varphi}}{1 - e^{2\varphi}}$ $\alpha_2 = \frac{\omega_1 - \omega_2 e^{2\varphi}}{1 - e^{2\varphi}}$	[10]
5	Butterworth inertia weight strategy	$\omega(t) = \omega_{\max} * \frac{1}{1+(\frac{t}{\rho_1})^{\rho_2}} + \omega_{\min}$	[11]
6	Linear strategies to adjust inertia weight	$\omega(t) = \frac{I_{\max}-t}{I_{\max}} (\omega_{\max} - \omega_{\min}) + \omega_{\min}$	[6]
7	Linear strategies to adjust I.W	$\omega(t) = 0.5 * \frac{t}{I_{\max}} + 0.4$	[12]
8	Non-Linear strategies to adjust inertia weight	$\omega(t) = \omega_i(t) - \Delta\omega$ $\omega(t) = \omega_i(t) + \eta\omega$ where $\Delta\omega = \frac{\omega_{\text{start}} - \omega_{\text{end}}}{N_t}$; a is constant	[13]

(continued)

(continued)

S. No.	Inertia weight	Formula of inertia weight	References
9	Fuzzy rule to adjust inertia weight	$V_c = e - [10(1 + V_{c_k})]$ and a new velocity updating formula is given below $V_{ij}(t + 1) =$ $V + C_1 r_1 (x_{ij}^*(t) - x_{ij}(t)) +$ $C_2 r_2 (x_{ij}^*(t) - x_{ij}(t))$ $V =$ $\begin{cases} V_{ij}, & \text{if } V_{ij} \geq V_c \\ u(-1, 1)V_{\max}/\rho, & \text{if } V_{ij} < V_c \end{cases}$	[14]
10	Random inertia weight	$\omega = \begin{cases} a_{1+\frac{r}{20}}, & K > 0.05 \\ a_{1+\frac{r}{20}}, & K < 0.05 \end{cases}$ $E(\omega) = \alpha_1 + 0.25, \text{ otherwise}$ $E(\omega) = \alpha_2 + 0.25$	[15]
11	Adaptive Strategy for Parameters	$\omega(f) = \frac{1}{1+1.5e^{-2.6f}} \in [0.4, 0.9],$ $\forall f \in [0, 1]$	[16]
12	Fixed inertia weight	$w = 1/2 * \ln(2)$	[17]
13	Linearly decreasing inertia weight	$W = W_{\max} - \frac{W_{\max} - W_{\min}}{\text{Iter}_{\max}} \times k$	[18]
14	Simulated annealing inertia weight	$W_k =$ $W_{\min} + (W_{\max} - W_{\min}) + \gamma^{(k-1)}$	[19]

3 Review on Discrete Particle Swarm Optimization

3.1 **Discrete PSO approach for job grid scheduling:** Discrete PSO approach for job grid scheduling was proposed by Izakian et al. [20] for the solution of Job grid scheduling optimization problem. The main objective of this problem is to minimizing make span and flow time continuously. Some experiments have performed and show that proposed PSO for job grid problem is more sufficient than PSO approach. Here, discrete PSO for J. B. S is proposed. Every particle needs to be designed in current positions. Representation step of finding the right mapping between proposed and PSO solution is the main issue. Consider a problem $m \times n$ matrix is taken where position matrix of the particles, where m denotes the available nodes and jobs are denoted by n . This matrix of every particle has two properties.

There are two values taken for all element of the matrix 0 and 1. Position of k th particles is denoted by X_k .

$$X_k(i, j) \in \{0, 1\}, \quad \forall i, \quad j \in \{1, 2, 3, \dots, m\} \quad \forall j \in \{1, 2, 3, \dots, n\}$$

In matrix, every column has assigned two value 1 and 0, 1st element value is 1 and left are 0. In position matrix of the particles, jobs represented by row and column are represented by job allocation. $m \times n$ matrix elements lie between (V_{\min}, V_{\max}) where v represents the velocity of every particle. In the same way, Pbest and nbest are $m \times n$ matrix and elements of this matrix are also same as position matrix. The k th best position is Pbest of the particle in first step, and nbest $_k$ is the best position of that particle who have come upon from the origination of the process. Star neighborhoods topology is used for nbest. In each iteration, both values are updated. First of all, fitness value of every particle is evaluated and check with Pbest $_k$, if this value is larger than change it with x_k . Two equations are proposed for update position matrix and velocity matrix. First is given

$$V_k^{(t+1)}(i, j) = V_k^t(i, j) + c_1 * r_1 * (Pbest_k^t(i, j) - X_k^t(i, j)) + c_2 * r_2 * (nbest_k^t(i, j) - X_k^t(i, j)) \quad (1)$$

$$X_k^{(t+1)}(i, j) = \begin{cases} 1 & \text{if } V_k^{(t+1)}(i, j) = \max\{V_k^{(t+1)}(i, j)\}, \forall i \in \{1, 2, 3, 4, \dots, m\} \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

In velocity updating Eq. (1), $V_k^{(t+1)}(i, j)$ is the segment in i th row and j th column, t is time step.

$X_k^{(t+1)}(i, j)$ is the position of k th particle in i th row and j th column. Make span and flow time; these are two main terms which are used to assess the outcome of scheduler. The fitness value of each solution is shown as

$$\text{Fitness} = (\lambda * \text{make span} + (1 - \lambda) * \text{mean - flow time}) \quad (3)$$

where λ regulates the effect of parameter. If the value of λ is greater, then more attention is paid and if the value of make span and flow time is small, then fitness value of particles is greater and gets a better solution.

3.2 Discrete PSO approach for timetable scheduling problem: Chu et al. [3] had proposed the discrete PSO that solve the timetable scheduling problem using PSO algorithm. By trial-and-error result, it is shown that PSO can solve discrete optimization problems. The main problems of time schedule problem are as follows

1. The status of the examination timetable.
2. The time consumed in generating the timetable.

This proposed discrete PSO gives the chance to each particle of self-changed within two slots. In this way, particles have the chance of best solution and to escape the local optimal solution. Starting steps are same as we use in standard PSO algorithm. First of all, generation is selected randomly that we take in swarm. Here, M particles are used which are shown here

$$X = \{x_1, x_2, x_3, \dots, x_m\}$$

20 particles are selected randomly. Each candidate is equivalent to a single solution. Then, progress and steps of 20 candidates are determined and set in order. The k th particle's previous last position should be kept in P_k at the i th iteration. Then select the best position from first to last and should be put in G^i . Then get a new timetable for the next generation as:

Here is one procedure by which movement of the particle can be determined with the procedure.

$$S_k^{i+1} = \text{rand-mutate}(X_k^i) \quad (4)$$

$$W_k^{i+1} = \text{rand-change}(S_k^{i+1}, P_k^i) \quad (5)$$

$$X_k^{i+1} = \text{rand-change}(W_k^{i+1}, G^i) \quad (6)$$

1st is about randomly changed slots of particles. 2nd is about copying a slot from the local best.

3rd is related to the global best. Many experimental results have discussed and find the result that PSO is best performing method for solving the DPSO of timetable scheduling.

3.3 Discrete PSO approach to distribution generation placement in Distribution method: Kumar et al. [21] had proposed a discrete PSO related to distribution generation placement in distribution method. In this problem, set of solution is generated with discrete variables. This proposed method is used in distribution generation placement problem in distribution system with aim to bring down annual energy cost and node voltage deviation. The following model indicated the k th number of iteration of the particles.

$$V_p^{(k+1)} = W * V_{pk} + c_1 * r_1 (P_{bestp} - S_p^k) + c_2 * r_2 (G_{bestk} - S_{pk})$$

$$S_p^{ks} = S_p^k + V_p^{k+1}$$

where c_1 and c_2 are acceleration coefficient and the Inertia weight is defined as

$$w = w_{\max} - \frac{w_{\max} - w_{\min}}{\text{iter}_{\max}} \times \text{iter}$$

In Distribution generation placement problem, the location is discrete variable and evaluated on 69 bus test diffusion system. A set of unconfirmed solution are created from each candidate solution of continuous decision variables. These approximate solutions do not have continuous variables. Then, all possible combinations are generated. In few locations, DG is installed. The approximation solution of candidate is not large. The fitness value of selected solution is evaluated, and candidate solution is changed by the best suitable approximation solution. Exponential decaying I.W is given by

$$W = W_{\max} * \exp\left(\frac{-kw \times \text{iter}}{\text{iter}}\right)$$

In this proposed PSO, diversity and best control of particle velocity are more, and this increases the exploitation and exploration rate of PSO. Here are some steps of DG placement problem.

1. Firstly Input data system and then obtain the power losses for every load.
2. Randomly generate the group of DPSO, then determine the best fitness of every particle and find Pbest and Gbest then updating position and velocity equations.

3.4 **Discrete PSO approach to schedule the task problem:** Sarathambekai [22] had proposed discrete PSO algorithm which is applied fluently schedule the task problem in the diversified multiprocessor structure. Intelligent should be included to create the initial swarm in order avoid the early convergence. It presents discrete PSO approach. To determine the efficiency of this DPSO, three measures are applied flow time, make span, and reliability. This is done for scheduling independent task in distributed system. We can solve heterogeneous type problems. A computing heterogeneous system presents a number of heterogeneous processor elements. These heterogeneous processor components are associated by a mesh topology. Here, n is number of task and m is number of processor.

$$T = \{T_1, T_2, \dots, T_n\}$$

$$P = \{P_1, P_2, \dots, P_m\}$$

The execution time of a job performing for each processor is different. Every job has a specific time to find on a selective processor. Each job has expected time to find and supposed to be known in advance. In this $n \times m$ matrix, first row represents the running time of ETC matrix. 1st column of ETC matrix represents estimated running time for task problem.

1. Zero value is assigned to all jobs.
2. The running time of each job is known and constant.
3. Take-over is not accepted.
4. Every processor allows to perform not more than one job.

5. First come-first served (FCFS) method is used.

DPSO performs random initialization same as other PSO problems, and this plays an important role in improvement of the convergence rate and find best solution. Main focus of DPSO is to adjust the population setup with the help of OPO method and greedy solution.

Swarm setup consists of two parts: Particles setup and allocation of processor. Particles are encoded in Permutation-based method. The opponent candidates of the original swarm are evaluated by OPO method. Best particles are sort out on the fundamental point of fitness value to form the DPSO.

The processors effectiveness is expressed by the percentage of time that processor P_j is busy during the scheduling time.

$$PU_j = \text{Aval}(P_j)/\text{Make-span}$$

$$\text{For } j = 1, 2, 3 \dots m.$$

where $\text{Aval}(P_j)$ is the utilization time of processors and RU is resource utilization and defined as

$$RU = \frac{\sum_{j=1}^m PU_j}{m}$$

In process utilization, RU is helper to find average. Two value flow time and make span run are in unparalleled range, and make span has a maximum value than flow time. The intelligent PSO has three determined criteria such as mean flow time, make span, and reliability cost. Flow time can be determined by the mean flow time. To compute the fitness value of every candidate, adaptive weighted sum is applied.

$$\begin{aligned} \text{Fitness} = & \alpha_1 * \alpha_2 * \text{Make span} + (1 - \alpha_1) * \alpha_2 * \text{Mean Flow Time} \\ & + (1 - \alpha_2) * \text{Reliability cost.} \end{aligned}$$

After calculating fitness value update both equations of PSO.

4 Review on Parallel Particle Swarm Optimization

4.1 CPU-Based Parallelization

First work related to CPU-Based Parallelization is presented by Gies and Yahya [23] in 2003. They had taken initial population with 10 particles, after on 10 nodes of a Beowulf cluster, and got result 8 times faster than their sequential execution. Mc Nabb et al. [24] had proposed Map Reduce Particle swarm optimization (MRPSO) which

execution is depending on the map reduce parallel organizing model and it presents how PSO naturally expressed in parallel programming model (PPM). As particles are mapped, they get an updated position of velocity value and individual best. In the decrease phase, it associates from other particles to revise its global optimal point. In the reduce phase, Aljarah and Ludwig [25] developed Map Reduce-based parallel particle swarm optimization clustering algorithm (MRCPSO) for facts-exhaustive applications, implemented on different sized large-scale datasets. The objective of this clustering algorithm is divided into three sections. In the 1st section, the particles swarm update centroids in Map Reduce, and in 2nd section determine the proficiency for new centroids and the 3rd section merges the new fitness values with the Pbest and Gbest centroids.

4.2 GPU-Based Parallelization

Hung and Wang [26] proposed GPU-accelerated Particle swarm optimization (GPSO), which focused to faster PSO explore function for large number of particles and higher dimension problems. Chu et al. [3] had introduced parallel PSO algorithm in which three transmission strategies are applied. First strategy displaces the best particle which is selected from every group of G_t and change G_t to restore the particles in every given group and update after each iteration. Second strategy relocates the position of best particle G_{jt} of the j th group to next group to alternative inferior particles in each receiving group for every iteration and in third strategy separating in two subgroups of all group then use transmission strategy. Two parallel methods are used pipeline in which problem is separated into tumble of jobs where every job is performed by the personal processor, and 2nd method is data parallelism which is alternate approach. Because it includes all processors which are processed for distribution of the data. The result of Parallel Particle swarm optimization is extremely depending on the level of connection between nature of transmission strategy and used parameters. Narjess Dali and Sadok [4] had introduced parallel PSO in which solution is presented by Graphical Processing Unit for constraint satisfaction problem, where number of constraints is maximum. A new approach Max-CSPs is introduced for the solution of large instance by using PSO algorithm and main aim is considered the parallel two novel approaches. The GPU is the first parallel approach used for maximum number of constraint, and second approach is GPU distributed used for Max-CSPs (GPU-DPSO), here O stands for dynamic. These two approaches show the efficiency and ability to exploit GPU architecture.

5 Conclusion

This paper presents the different modifications of PSO in respect to Inertia weight, Discrete Particle swarm optimization (DPSO), and Parallel Particle swarm optimization (PPSO). All the modifications have done in PSO with the help of parameters

and different methods. In original PSO, there was no use of Inertia weight but for getting best result many researchers introduced and implemented the Inertia weight approach. It plays an important role in whole process of PSO algorithm and controls the process of maintaining the balance of local point and global point. We find that this suggested algorithm gives excellent result on unimodal and multimodal functions with the best optimal point, rapid convergence rate, higher validity, and superior reliability. The Discrete PSO is specially introduced for the problem of discrete variables and mixed integer optimization problems. This increases the exploitation and exploration ability of Particle swarm optimization algorithm and upgrade the convergence speed. With the help of Parallel PSO, we can reduce the processing time and computational cost of PSO methods. In recent years, it has been observed that the partition clustering technique is most appropriate for clustering large dataset. In last few years, the problem of clustering has been approached from different disciplines. Many optimization algorithms have been introduced for the solution of clustering problem. The PSO and its modified inertia weight motivate us to acquire the future research work. A large work needs to be done in parameter of PSO. Appropriate parameter and topology selection is one of the research fields in PSO.

References

1. Azli, N.A., Nayan, N.M., Ayob, S.M.: In the particle swarm optimization and its applications in power converter systems. *Int. J. Artif. Intell. Soft Comput.* **3** (2013)
2. Metiane, M.A., Qudi, Y.M., Bouchiba, B., La out, A.: In impact of inertia weight strategies in PSO for solving economic dispatch problem Indonesian. *J. Electr. Eng. Comput. Sci.* **13**(1), 3772383 (2019)
3. Chu, S.-C., Chen, Y.-T., HQ, J.-H.: In time table scheduling using particle swarm optimization. In: *Proceedings of the First International Conference on Innovative Computing, Information and Control (ICICIC'06 0-7695-2616-0/06\$20.00@2006. IEEE* (2006)
4. Dali, N., Bouamama, S.: IN GPO-PSO: parallel PSO approaches on graphical processing unit for constraints reasoning: case of max-CSP. *Procedia Comput. Sci.* **60** (2018), 1010–1080. <https://doi.org/10.1016/J.Procs.2015.08.152> (2015)
5. Shao, L., Bai, Y., Qiu, Y., Du, Z.: PSO algorithm based on semantic relations and its engineering applications. *Syst. Eng. Procedia* **5**, 222–227 (2012)
6. Eberhart, R., Shi, Y.: Comparing Inertia weight and construction factors in particle swarm optimization. In: *Proceeding of the IEEE Congress on Evolutionary Computation (CEC'00)2000*, pp. 84–88 (2000)
7. Shi, Y., Eberhart, R.C.: A modified particle swarm optimizer. In: *The 1998 IEEE International Conference on Evolutionary Computation Proceedings, 1998 IEEE World Congress on Computational Intelligence, 1998. IEEE* (1998)
8. Eberhart, R.C., Shi, Y.: Tracking and optimizing dynamic systems with particle swarms. In: *2001 Proceedings of the 2001 congress on Evolutionary Computation (2001). IEEE 0-7803-6657-3/01/\$10.00 02001*
9. Engel Brecht, A.P., *Computational Intelligence an Introduction*. Wiley. ISBN: 978-0-470-03561-0 (2007)
10. Javad, M., Mousa Shamsi, A., Sedaaghi, M.H.: In a flexible inertia weight particle swarm optimization algorithm. *PLOS one*. <https://doi.org/10.1371/Journal.Pone.0160558> (2016)
11. Zhu, X., Wang, H.: In a New Inertia Weight Control strategy for Particle Swarm Optimization. Published by AIP Publishing. 978-0-7354-1654-3/\$30.0 (2018)

12. Zhang, Y., Ma, I., Zhang, L., et al.: On the convergence analysis and parameter selection in PSO. In: Proceeding of the International Conference on Machine learning and cybernetics (ICMLC'03), pp.1802–1807 (2003)
13. Tanweer, M., Suresh, S., Sundararajan, M.: Self-regulating PSO. *Inf. Sci.* **294**, 182–202 (2015)
14. Liu, H., Abraham, A.: Fuzzy adaptive turbulent particle swarm optimization. In: Proceeding of the International Conference on Hybrid Intelligent System (HIS'05), pp. 445–450 (2016)
15. Zhang, L., Yu, H., Chem, D., et al.: Analysis and improvement of PSO Algorithm. *Inf. Control* **3315**, 513–517 (2004)
16. Zhan, Z.-H., Zhang, J.: Adaptive Particle Swarm Optimization. ANTS 2008, LMCS S217, pp. 227–234 (2008)
17. Hsieh, S.-T., Sun, T.-Y., Liu, C.-C., et al.: Efficient population utilization strategy for particle swarm optimizer. *IEEE Trans. Syst. Man Cybernet. Part B Cybernet.* **39**(2), 444–456 (2009)
18. Shi, Y., Eberhart, R.C.: Parameter selection in particle swarm optimization. In: Proceedings of 7th International Conference Evolutionary Programming, 591–600 (1999)
19. Park, J.-B., Jeong, Y.-W., Shin, J.-R.: An improved particle swarm optimization for nonconvex economic dispatch problems. *IEEE Trans. Power Syst.* **25**(1), 156–166 (2010)
20. Izakian, H., Tork Ladani, B., Zamani, K., Abraham, A.: A novel Particle Swarm Optimization approach for grid job scheduling
21. Kumar, P., Gupta, N., Swarnkar, A., Niazi, K.R.: Discrete particle swarm optimization for optimal D.G placement in distribution network. 978-1-4799-5141-3/14/\$31.00©2014IEEE (2014)
22. Sarathambekai, S., Umaheshwari, K.: In intelligent discrete particle swarm optimization for multiprocessor task scheduling problem. *J. Algorithm Comput. Technol.* **11**(1) 58–67(2017)
23. Gies, D., Rahmat-Samii, Y.: Reconfigurable array design using parallel particle swarm optimization. *IEEE Int. Symp. Antennas Propag. Soc.* **1**, 177–180 (2003)
24. Andrew, W., Mc Nabb, Monson, C.K., Seppi, K.D.: In Parallel particle swarm optimization using map reduce. DOI: 10.11091 CEC, 2007.4424448. Source: IEEE explore (2015)
25. Aljarah, I., Ludwig, S.A.: Parallel particle swarm optimization clustering algorithm based on map reduce methodology. In: Fourth IEEE World Congress on Nature and Biologically Inspired Computing, pp. 104–111 (2012)
26. Hung, Y., Wang, W.: Accelerating parallel particle swarm optimization via GPU. *Optim. Methods Softw.* **27**(1), 33–51 (2012)

Automated Detection of Elephant Using AI Techniques



Dhrumil Patel and Sachin Sharma

Abstract Elephant Entry in human settlements causes a major threat to Human–Elephant Conflict (HEC). As a result, humans and elephants both face unique problems, elephants destroy crops, houses, and many other things, while for protection of crops and villages humans use electrical fencing, throw stones, and use smoke and fire. Therefore, there needs to be research carried out to address these challenges while implementing autonomous elephant detection. (Objectives): To identify suitable and highly efficient Convolutional Neural Network (CNN) models for real-time elephant detection, it is a more accurate image processing algorithm that takes help of deep learning to carry out generative as well as descriptive jobs, by implementing various models of CNN and comparing the results among one other and displaying the results. (Methods): Literature review has been performed to identify suitable elephant detection models for real-time objects and different types of methods for detecting and tracking elephants. Following this, experiments have been conducted to evaluate the performance of the selected object detection models. Two methods are used in this experiment YOLO (You Look Only Once) algorithm that detect and identify things in picture and another one is SSD_efficientnets which is single shot detector but here the backbone networks are Efficientnets trained on ImageNet and the proposed BiFPN (Bi-directional feature Pyramid Network) network serves as a feature network. This BiFPN network intakes 3 to 7 features from the backbone network and applies bottom-up and top-down bi-directional feature fusion repeatedly (Results): You Look Only Once (YOLOv3) and SSD_efficientdet_d0_512x512 have been identified from the literature review as the most suitable and efficient algorithm for detecting elephants in real time. The detection performance of these algorithms has been calculated and compared with each other. The results have been presented. (Conclusion): The accuracy of YOLOv3 has been found to be better than the SSD_efficientnets_d0_512x512 model. Therefore, it has been concluded that YOLOv3 is the best model in the real-time detection of

D. Patel

Institute of Advanced Research, Gandhinagar, Gujarat, India

S. Sharma (✉)

National Forensic Sciences University, Gandhinagar, India

e-mail: sharma.f@gmail.com

elephants. The results of YOLOv3 are better for classification performance compared to SSD_efficientdets_d0_512x512.

Keywords Elephant detection · Deep learning · Convolutional neural network · Artificial intelligence (AI)

1 Introduction

The way toward detecting objects in recordings and pictures is known as object detection. This machine learning technique allows us to detect any kind of object by training and creating models. So we are using Machine Learning for detecting elephants. Detecting and classifying objects is considered to be one of the important tasks as it helps the forest department to detect elephants near forest border areas and try to protect them from humans.

In spite of the fact that there are many Artificial Intelligence and Machine Learning technique for object detection, for example, Support Vector Machine (SVM), Convolutional Neural Network (CNN), Regional Convolutional Neural Network (R-CNN), You Only Look Once (YOLO) model, and many others, it is critical to pick the correct model for identifying elephant as it requires ongoing item identification. As machines read images in mathematical format (in matrix form) so it is very important to choose the most accurate models for elephant detection.

1.1 Aim and Objectives

The point of this undertaking is to assess the execution of the deep learning models for recognition of elephants because they are endangered. Humans are cutting down forest areas and killing elephants to live a lavish life. As these big creatures have less space to live, they started entering rural areas and villages. To stop them entering the human area, we can put an automated camera on trees or near forest borders that can detect elephants and report to the forest department if they are in danger or running out of forest.

The following objectives have been done to fulfill the main aim of this project work:

- To identify suitable and more accurate models of deep learning for elephant's detection because there are numerous models that can detect it.
- Evaluate the detection performance of the selected deep learning models.
- Analyze the location execution of the chosen model among one another and present the outcomes.

1.2 Problem Statement

Despite the fact that the learning approach for elephant detection is the same as the dog's or cat's detection, with regard to recognizing elephants, there exist exceptional difficulties for elephants as their surroundings and natural habitats are different compared to dogs and cats. Accordingly, there should be research done to assess the current best in class deep learning models also, to recognize the best deep learning model for the detection of elephants.

2 Literature Review

The conflict between humans and animals has increased drastically in the past few years due to the large-scale destruction of animal habitat. As a result, there has been an increase in the total casualties on both sides. Thereafter, many techniques have been proposed to counter this problem. Animal Detection techniques can broadly be classified into two types. Visual approaches that make use of camera or other vision sensors such as Kinect to detect animals and sensory approaches that employ some other sensor for detecting animal intrusion. So many different types of methods are used for elephant detection like Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Histogram of oriented gradients (HOG), and Local Binary Pattern (LBP) [1].

In another research paper, a method for detecting elephants using geophones is buried in the ground. In this article, the author has done analytical procedure to study elephant behavior throughout the forest area by taking migration data over the whole year and is divided into four different periods for the study. The geophone is buried in the ground near the forest border area. If an elephant crosses that geophones, the forest department is alerted through SMS to take necessary action [2].

Neil K. Sheridan has done research and made a system for elephant detection. While training models, he had tried various different methods and initially he worked with an object detector using Histogram of Oriented Gradients (HOG) and Support Vector Machines (SVM). That had promising results; giving only 26% false positives with a dataset consisting of 350 positive elephant images and 2000 negative non-elephant images and he improved results with larger datasets. And it did. He got a result of 16% false negatives with 330 positive elephant images and 3500 negative non-elephant images [3].

In "Automatic Detection and Compression for Passive Acoustic Monitoring of the African Forest Elephant," they have applied machine learning to analysis and compression of audio signals in the context of monitoring elephants in sub-Saharan Africa. However, manually monitoring tropical forests or deep oceans is intractable. For species that communicate acoustically, researchers have argued for placing audio recorders in the habitats as a cost-effective and non-invasive method, a strategy known as passive acoustic monitoring (PAM) [4].

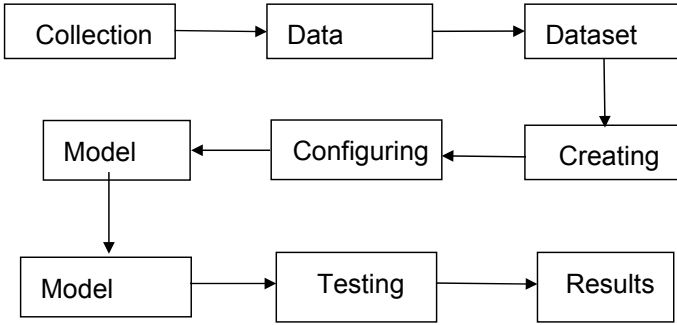


Fig. 1 Process block diagram

3 Methodology

This section gives you a brief description of the methods proposed to counter the problems of elephant’s detection.

3.1 *Detection of Elephant Using YOLOv3*

YOLO stands for “You Only Look Once” and uses convolutional neural networks (CNN) for object detection. The name of the algorithm means that a single network just once is applied to the whole image. YOLO divides image into regions and predicts bounding boxes and probabilities for every such region.

Figure 1 shows the block diagram of the proposed system. YOLO-3 is the version of YOLO that uses successive 3×3 and 1×1 convolutional layers. In total, it has 53 convolutional layers with architecture as shown on Fig. 2.

3.2 *Detection of Elephant Using SSD_EfficientDet_d0_512 × 512*

EfficientDets are a family of object detection models, which achieve state-of-the-art 55.1 mAP on the COCO test. EfficientDets are developed based on the advanced backbone, a new BiFPN, and a new scaling technique (Fig. 3):

- **Backbone:** It uses EfficientNets as a backbone network.
- **BiFPN:** we propose BiFPN, a bi-directional feature network enhanced with fast normalization, which enables easy and fast feature fusion.
- **Scaling:** we utilize a single compound scaling figure to administer the depth, width, and determination for all spine, highlight, and expectation systems.

Fig. 2 Architecture of YOLOv3

	Type	Filters	Size	Output
	Convolutional	32	3 × 3	256 × 256
	Convolutional	64	3 × 3 / 2	128 × 128
1x	Convolutional	32	1 × 1	
	Convolutional	64	3 × 3	
	Residual			128 × 128
2x	Convolutional	128	3 × 3 / 2	64 × 64
	Convolutional	64	1 × 1	
2x	Convolutional	128	3 × 3	
	Residual			64 × 64
8x	Convolutional	256	3 × 3 / 2	32 × 32
	Convolutional	128	1 × 1	
	Convolutional	256	3 × 3	
	Residual			32 × 32
8x	Convolutional	512	3 × 3 / 2	16 × 16
	Convolutional	256	1 × 1	
	Convolutional	512	3 × 3	
	Residual			16 × 16
4x	Convolutional	1024	3 × 3 / 2	8 × 8
	Convolutional	512	1 × 1	
	Convolutional	1024	3 × 3	
	Residual			8 × 8
	Avgpool		Global	
	Connected		1000	
	Softmax			

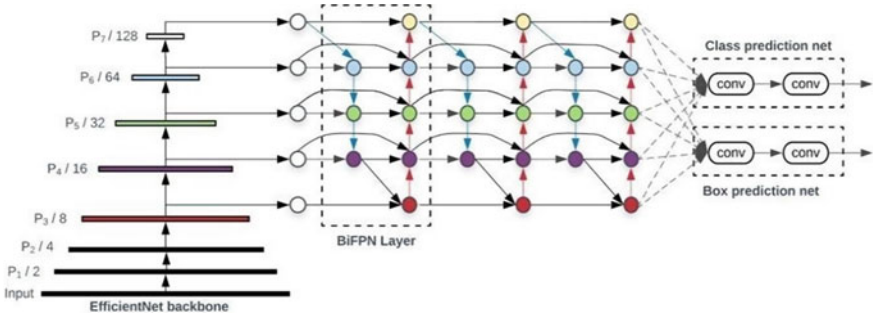


Fig. 3 Architecture of SSD_EfficientDet [5]

4 Experiment

Exploration has been chosen as the research method since when it comes to managing quantitative data, exploration is considered to be the leading strategy. In the test, we are going to utilize YOLOv3 and `ssd_efficientdet_d0` for elephant discovery.

4.1 Experimental Setup

Software: Python is widely considered as the preferred language for teaching and learning ML (Machine Learning). It is simple to learn, and data handling capacity is great and open source. To train and test models, Google Colab is used. It is a hosted Jupyter notebook service that requires no setup to use, while providing free access to computing resources including GPUs.

Few steps are required to be continued some time for training and testing model, which are as follows:

Dataset Collect: A dataset has been created by collecting the images of elephants in various angles, brightness, and contrasts. The collected images consisted of one class, a total of 1607 images.

Data Annotation: The collected images have been labeled manually using a tool known as “LabelImg.” Each image is labeled by drawing bounding boxes perfectly surrounding elephants in the image, as shown in Fig. 4. As a result, an XML file, also known as “Annotation file,” is generated for each image and saved into a specific folder. The annotation files contain details about the objects in the image such as Image name and label name, Image path, class name of the object(s), and coordinates of the bounding boxes surrounding the objects present in the image. These files are further used to train and enable the algorithm to detect the desired class objects.

Framework: TensorFlow’s object detection API is recognized to be an effective instrument, because it enables us to construct and convey picture acknowledgment

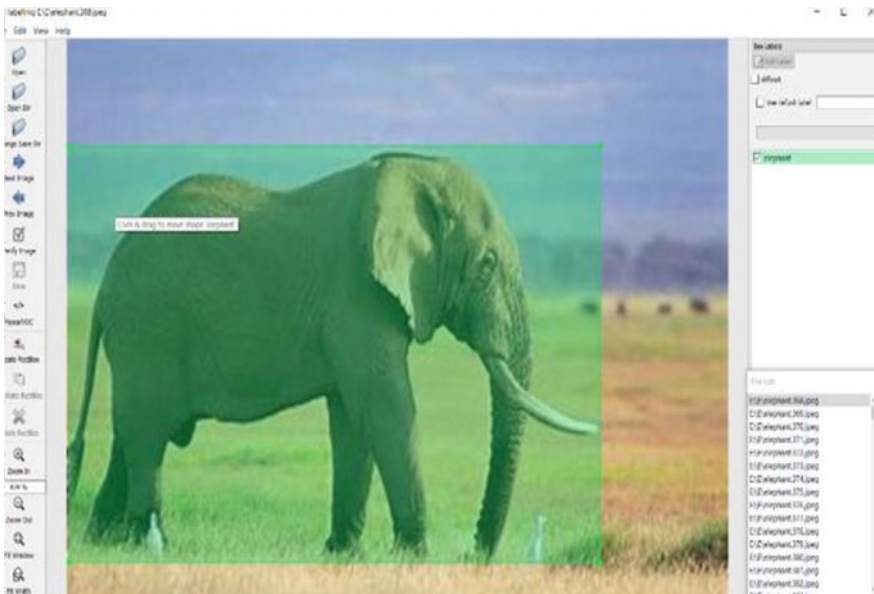


Fig. 4 Labeling of photos

```

+ Code + Text
total_bbox = 314337, rewritten_bbox = 0.342308 %
2609: 0.169853, 0.259103 avg loss, 0.001000 rate, 5.01390 seconds, 172160 images, 2.52845 hours left
Reusing, random_coef = 1.48

3x3 x 3x3
try to allocate additional workspace_size = 52.43 MB
CUDA allocate done!
loaded: 0.00020 seconds
v3 (use loss, Normalizer: (iou: 0.75, obj: 1.00, cls: 1.00) Region 82 Avg (IOU: 0.829530), count: 3, class_loss = 0.813534, iou_loss = 0.476195, total_loss = 0.858259
v3 (use loss, Normalizer: (iou: 0.75, obj: 1.00, cls: 1.00) Region 94 Avg (IOU: 0.263018), count: 3, class_loss = 0.197969, iou_loss = 0.104761, total_loss = 0.302338
v3 (use loss, Normalizer: (iou: 0.75, obj: 1.00, cls: 1.00) Region 106 Avg (IOU: 0.000000), count: 1, class_loss = 0.000000, iou_loss = 0.000000, total_loss = 0.000000
total_bbox = 314343, rewritten_bbox = 0.342381 %
v3 (use loss, Normalizer: (iou: 0.75, obj: 1.00, cls: 1.00) Region 82 Avg (IOU: 0.802042), count: 5, class_loss = 0.261684, iou_loss = 0.290286, total_loss = 0.551970
v3 (use loss, Normalizer: (iou: 0.75, obj: 1.00, cls: 1.00) Region 94 Avg (IOU: 0.677280), count: 3, class_loss = 0.308549, iou_loss = 0.301584, total_loss = 0.609733
v3 (use loss, Normalizer: (iou: 0.75, obj: 1.00, cls: 1.00) Region 106 Avg (IOU: 0.217108), count: 1, class_loss = 0.261217, iou_loss = 0.063325, total_loss = 1.229253
total_bbox = 314350, rewritten_bbox = 0.342391 %
v3 (use loss, Normalizer: (iou: 0.75, obj: 1.00, cls: 1.00) Region 82 Avg (IOU: 0.799920), count: 3, class_loss = 0.810387, iou_loss = 0.148725, total_loss = 0.164112
v3 (use loss, Normalizer: (iou: 0.75, obj: 1.00, cls: 1.00) Region 94 Avg (IOU: 0.676687), count: 5, class_loss = 0.418788, iou_loss = 0.681388, total_loss = 1.096776
v3 (use loss, Normalizer: (iou: 0.75, obj: 1.00, cls: 1.00) Region 106 Avg (IOU: 0.312200), count: 3, class_loss = 0.399442, iou_loss = 0.934428, total_loss = 1.838662
total_bbox = 314363, rewritten_bbox = 0.342399 %
v3 (use loss, Normalizer: (iou: 0.75, obj: 1.00, cls: 1.00) Region 82 Avg (IOU: 0.811175), count: 5, class_loss = 0.164181, iou_loss = 0.224870, total_loss = 0.388251
v3 (use loss, Normalizer: (iou: 0.75, obj: 1.00, cls: 1.00) Region 94 Avg (IOU: 0.848838), count: 2, class_loss = 0.026790, iou_loss = 0.049124, total_loss = 0.056904
v3 (use loss, Normalizer: (iou: 0.75, obj: 1.00, cls: 1.00) Region 106 Avg (IOU: 0.000000), count: 1, class_loss = 0.000321, iou_loss = 0.000000, total_loss = 0.000321
total_bbox = 314370, rewritten_bbox = 0.342422 %
v3 (use loss, Normalizer: (iou: 0.75, obj: 1.00, cls: 1.00) Region 82 Avg (IOU: 0.841870), count: 5, class_loss = 0.437136, iou_loss = 0.407455, total_loss = 0.578782
v3 (use loss, Normalizer: (iou: 0.75, obj: 1.00, cls: 1.00) Region 94 Avg (IOU: 0.800800), count: 1, class_loss = 0.056468, iou_loss = 0.000000, total_loss = 0.056468
v3 (use loss, Normalizer: (iou: 0.75, obj: 1.00, cls: 1.00) Region 106 Avg (IOU: 0.000000), count: 1, class_loss = 0.000000, iou_loss = 0.000000, total_loss = 0.000000
total_bbox = 314379, rewritten_bbox = 0.342426 %
v3 (use loss, Normalizer: (iou: 0.75, obj: 1.00, cls: 1.00) Region 82 Avg (IOU: 0.830416), count: 6, class_loss = 0.172791, iou_loss = 0.170658, total_loss = 0.297699
v3 (use loss, Normalizer: (iou: 0.75, obj: 1.00, cls: 1.00) Region 94 Avg (IOU: 0.940712), count: 1, class_loss = 0.004400, iou_loss = 0.002777, total_loss = 0.007277
v3 (use loss, Normalizer: (iou: 0.75, obj: 1.00, cls: 1.00) Region 106 Avg (IOU: 0.000000), count: 1, class_loss = 0.000000, iou_loss = 0.000000, total_loss = 0.000000
total_bbox = 314380, rewritten_bbox = 0.342281 %
v3 (use loss, Normalizer: (iou: 0.75, obj: 1.00, cls: 1.00) Region 82 Avg (IOU: 0.804918), count: 4, class_loss = 0.025000, iou_loss = 0.060673, total_loss = 0.060674
v3 (use loss, Normalizer: (iou: 0.75, obj: 1.00, cls: 1.00) Region 94 Avg (IOU: 0.790010), count: 7, class_loss = 0.064972, iou_loss = 0.282149, total_loss = 0.347121

```

Fig. 5 Training and classification loss for YOLOv3

computer programs rapidly. Thus, it has been chosen to prepare `ssd_efficientdet_d0` in this venture. YOLOv3 has a Darknet system which is an open-source Neural Arrange in C.

Configuration: Various changes have been made to the default configuration files of the `ssd_efficientdet_d0` provided by the TensorFlow Object Detection API, such as dataset, number of classes to be trained, batch size, and label map. For YOLOv3, changes like batch, classes, and filters are needed to be changed in the `yolo3.cfg` file provided by Darknet.

Training: After finishing all the steps mentioned above and making necessary changes in the configuration file, the training process is initialized. The step count and classification loss in each step can be seen on screen as shown in Fig. 5 for yolo and Fig. 6 for `ssd_efficientdet`.

Here, 2609 indicates the current training batch. 0.169853 indicates total loss.

0.259103 avg indicates average loss.

0.001000 rate represents the current learning rate.

5.0139 s represents the total time spent to process this batch (Fig. 6).

5 Results

Accuracy: It is defined as the number of correct predictions made by the model over the total number of predictions. This is a good measure, especially when the target variable classes are balanced in data. This can be represented as:

```
+ Code + Text
INFO:tensorflow:Step 15500 per-step time 0.4095s loss=0.162
I0119 18:13:19.130892 139942355920768 model_lib_v2.py:651] Step 15500 per-step time 0.4095 loss=0.162
2021-01-19 18:13:54.330965: W tensorflow/core/lib/png/png_io.cc:88] PNG warning: ICCP: known incorrect sRGB profile
INFO:tensorflow:Step 15600 per-step time 0.4055s loss=0.149
I0119 18:13:59.028349 139942355920768 model_lib_v2.py:651] Step 15600 per-step time 0.4055 loss=0.149
INFO:tensorflow:Step 15700 per-step time 0.3675s loss=0.178
I0119 18:14:38.697562 139942355920768 model_lib_v2.py:651] Step 15700 per-step time 0.3675 loss=0.178
2021-01-19 18:14:40.083921: W tensorflow/core/lib/png/png_io.cc:88] PNG warning: ICCP: known incorrect sRGB profile
2021-01-19 18:14:40.866544: W tensorflow/core/lib/png/png_io.cc:88] PNG warning: ICCP: known incorrect sRGB profile
INFO:tensorflow:Step 15800 per-step time 0.4025s loss=0.293
I0119 18:15:19.052085 139942355920768 model_lib_v2.py:651] Step 15800 per-step time 0.4025 loss=0.293
INFO:tensorflow:Step 15900 per-step time 0.441s loss=0.153
I0119 18:15:59.180678 139942355920768 model_lib_v2.py:651] Step 15900 per-step time 0.441s loss=0.153
INFO:tensorflow:Step 16000 per-step time 0.407s loss=0.234
I0119 18:16:39.302400 139942355920768 model_lib_v2.py:651] Step 16000 per-step time 0.407s loss=0.234
2021-01-19 18:16:45.253576: W tensorflow/core/lib/png/png_io.cc:88] PNG warning: ICCP: known incorrect sRGB profile
INFO:tensorflow:Step 16100 per-step time 0.3725s loss=0.303
I0119 18:17:19.844341 139942355920768 model_lib_v2.py:651] Step 16100 per-step time 0.3725 loss=0.303
2021-01-19 18:17:26.239121: W tensorflow/core/lib/png/png_io.cc:88] PNG warning: ICCP: known incorrect sRGB profile
INFO:tensorflow:Step 16200 per-step time 0.404s loss=0.247
I0119 18:18:00.220031 139942355920768 model_lib_v2.py:651] Step 16200 per-step time 0.404s loss=0.247
2021-01-19 18:18:16.467826: W tensorflow/core/lib/png/png_io.cc:88] PNG warning: ICCP: known incorrect sRGB profile
INFO:tensorflow:Step 16300 per-step time 0.345s loss=0.464
I0119 18:18:40.339895 139942355920768 model_lib_v2.py:651] Step 16300 per-step time 0.345s loss=0.464
INFO:tensorflow:Step 16400 per-step time 0.474s loss=0.156
I0119 18:19:20.397118 139942355920768 model_lib_v2.py:651] Step 16400 per-step time 0.474s loss=0.156
INFO:tensorflow:Step 16500 per-step time 0.393s loss=0.174
I0119 18:20:00.544723 139942355920768 model_lib_v2.py:651] Step 16500 per-step time 0.393s loss=0.174
2021-01-19 18:20:26.113724: W tensorflow/core/lib/png/png_io.cc:88] PNG warning: ICCP: known incorrect sRGB profile
INFO:tensorflow:Step 16600 per-step time 0.422s loss=0.264
I0119 18:20:40.769758 139942355920768 model_lib_v2.py:651] Step 16600 per-step time 0.422s loss=0.264
Traceback (most recent call last):
  File "model_main_tf2.py", line 113, in <module>
    tf.compat.v1.app.run()
  File "/usr/local/lib/python3.6/dist-packages/tensorflow/python/platform/app.py", line 40, in run
    _run(main=main, argv=argv, flags_parser=_parse_flags_tolerate_undef)
```

Fig. 6 Training and classification loss for ssd_efficientdet

$$\text{Accuracy} = \text{Correct predictions} / \text{Total Predictions} \tag{1}$$

(a) FOR YOLOv3:

$$\begin{aligned} \text{No. of correct predictions (CP)} &= \text{True Positives} + \text{True Negatives} \\ &= 138 + 4 \\ &= 142 \end{aligned}$$

$$\begin{aligned} \text{Total no. of Predictions (TP)} &= \text{True Positives} + \text{True Negatives} \\ &\quad + \text{False Positives} + \text{False Negatives} \\ &= 138 + 4 + 2 + 13 \\ &= 157 \end{aligned} \tag{3}$$

$$\begin{aligned} \text{Accuracy} &= \text{Correct predictions} / \text{Total predictions} \\ &= 142 / 157 \\ &= \mathbf{0.90 \text{ accuracy}} \end{aligned}$$

(b) FOR ssd_efficientdets:

$$\text{No. of correct predictions (CP)} = \text{True Positives} + \text{True Negatives}$$

Table 1 Result table

	YOLOv3	ssd_efficientdets
Total predicted	157	157
True positive	138	105
True negative	4	5
False positive	2	3
False negative	13	44
Accuracy	0.90	0.70
Specificity	0.66	0.625

$$= 105 + 5$$

$$= 110$$

$$\begin{aligned}
 \text{Total no. of Predictions (TP)} &= \text{True Positives} + \text{True Negatives} \\
 &+ \text{False Positives} + \text{False Negatives} \\
 &= 105 + 5 + 3 + 44 \\
 &= 157
 \end{aligned} \tag{3}$$

$$\begin{aligned}
 \text{Accuracy} &= \text{Correctpreconditions}/\text{TotalPredictions} \\
 &= 110/157 \\
 &= 0.70 \text{ accuracy}
 \end{aligned}$$

Specificity: It is the metric that evaluates a model's ability to predict true negatives of each available category (Table 1).

$$\text{Specificity} = \text{True Negative}/(\text{True Negative} + \text{False Positive}) \tag{4}$$

For YOLOv3

$$\begin{aligned}
 \text{Specificity} &= \text{TrueNegatives}/(\text{TrueNegatives} + \text{FalsePositives}) \\
 &= 4/(4 + 2) \\
 &= 0.66 \text{ Specificity}
 \end{aligned}$$

For ssd_efficientdets

$$\begin{aligned}
 \text{Specificity} &= \text{True Negatives}/(\text{True Negatives} + \text{False Positives}) \\
 &= 5/(5 + 3) \\
 &= 0.625 \text{ Specificity}
 \end{aligned}$$

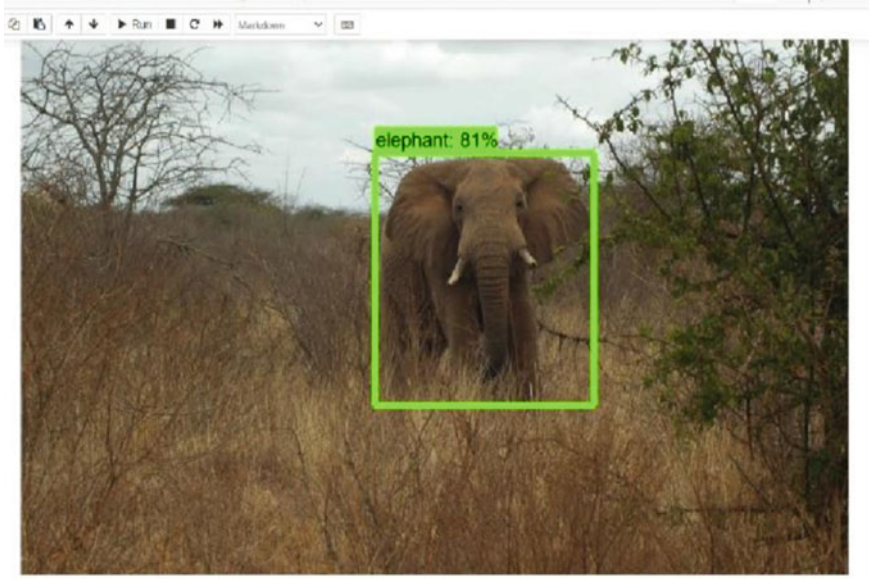


Fig. 7 Prediction made by `ssd_efficientdets`

Figures 7 and 8 represent the screenshots of the detections made by YOLOv3 and `ssd_efficientdets` along with the confidence intervals of the detections that have been made.

The test was also carried out multiple's times, by providing images of elephants as test data.

6 Conclusion and Future Work

6.1 Conclusion

This project report discusses the most suitable deep learning models for object detection and evaluates the performance of these algorithms on the detection and recognition of elephants in images. An experiment has been carried out to evaluate the detection performance of these deep learning algorithms. After the preparation of the dataset, the algorithms have been trained on the train dataset. The trained models have been tested on the test images, from which the number of true positives, true negatives, false positives, and false negatives has been identified for each image of the detections made by the two deep learning models on the test images. Using these results, the Accuracy, Sensitivity, and specificity of the models have been calculated

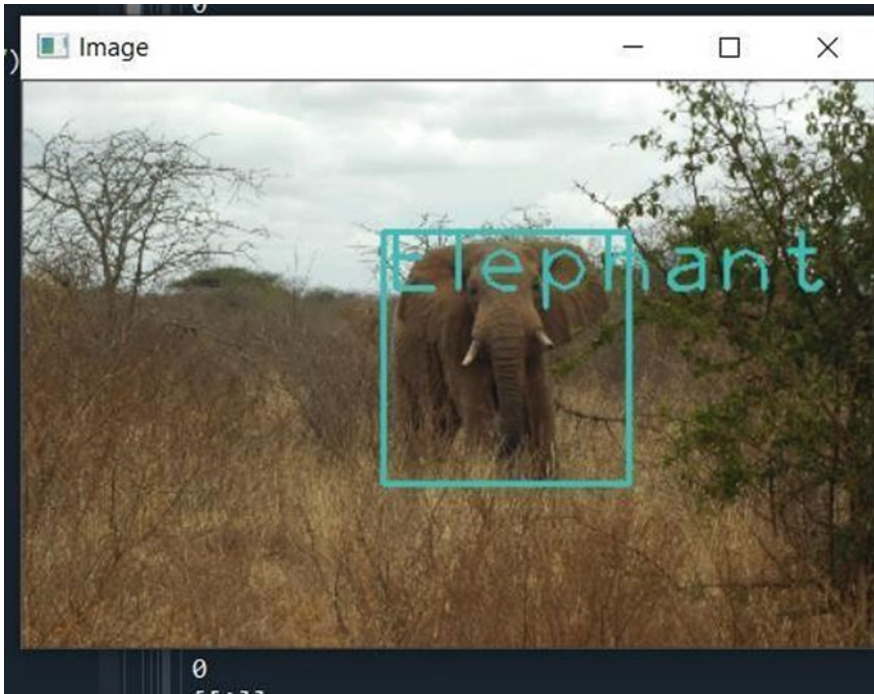


Fig. 8 Prediction made by YOLOv3

and the performance of the YOLOv3 and `ssd_efficientdet` model have been evaluated and compared. It has been concluded from the analysis that out of the two deep learning models selected for this project, YOLOv3 showed best classification performance.

Future Work Future work can be done by testing models on videos and on live camera which models show best accuracy and to implement in real world in future we will try to implement code in NVIDIA Jetson Nano or on Raspberry pi.

References

1. Shukla, P, Dua, I., Raman, B., Mittal, A.: A computer vision framework for detecting and preventing human elephant collisions (2017). https://openaccess.thecvf.com/content_ICCV_2017_workshops/papers/w41/Shukla_A_Computer_Vision_ICCV_2017_paper.pdf
2. Sugumar, S.J., Jayaparvathy, R.: An early warning system for elephant intrusion along the forest border area. *Curr. Sci.* **5**(1) (2018). ISSN No 2349-2503
3. Sheridan, N.K.: Elephant AI (2022). <https://github.com/nksheridan/elephantAI>
4. Bjorck, J., Rappazzo, B.H., Chen, D., Bernstein, R., Wrege, P., Gomes, C.: Automatic detection and compression for passive acoustic monitoring of the African forest elephant. In: Proceedings

- of the AAAI conference on artificial intelligence, vol. 33, pp. 476–484 (2019). <https://doi.org/10.1609/aaai.v33i01.3301476>
5. https://imadelhanafi.com/posts/object_detection_yolo_efficientdet_mobilenet/

Determination of Probability of Failure of Structures Using DBSCAN and Support Vector Machine



Pijus Rajak and Pronab Roy

Abstract Nowadays, the advanced machine learning method, support vector machine (SVM), is used to determine the probability of failure of a system. The aim of the paper is to find an efficient machine learning method to determine the probability of failure of structures with low computational cost. The density-based spatial clustering of applications with noise (DBSCAN) algorithm is being used to exclude noise and outliers' samples. The rest of the data is used to train SVM regression, and then, the probability of failure of the structures is found out based on Monte Carlo simulation (MCS) method. Three examples such as a three-span beam, a two-span beam, and a frame structure with known exact performance functions illustrate the application and effectiveness of the proposed method. The results show that it is suitable for less training data for low probability of failure problems. The effectiveness of the suitable algorithm is examined and compared with the direct MCS method and first-order reliability method (FORM).

Keywords Reliability analysis · Probability of failure · Density-based spatial clustering of applications with noise · Support vector machine · Monte Carlo simulation · First-order reliability analysis

1 Introduction

Structural reliability analysis is a valuable method for evaluating engineering structures' safety and allows for more realistic risk assessments. Machine learning-based reliability analysis is a promising field of research. The researchers of all over the world are working in this field. Some of the relevant studies are discussed here.

P. Rajak (✉) · P. Roy (✉)

Department of Civil Engineering, National Institute of Technology Durgapur, Durgapur 713209, India

e-mail: pr.19ce1105@phd.nitdgp.ac.in

P. Roy

e-mail: pronab.roy@ce.nitdgp.ac.in

Jamwal et al. [1] proposed a machine learning (ML)-based sustainable manufacturing (SM) framework to provide guidelines on the management of SM practices. The application of ML approaches helps to minimize cycle time and handle complex data generated by industrial processes for monitoring and prediction. Recently, a lot of papers have been devoted to applying SVM to reliability analysis. SVM is a method of statistical learning. Rocco and Morano [2] have first introduced SVM in the reliability analysis. They have developed a model estimate algorithm by training a restricted dataset model and replacing system performance assessment with a simplified calculation. Hurtado and Alvarez [3] used SVM in conjunction with a stochastic finite element to tackle reliability analysis as a classification task for pattern recognition. Dai et al. [4] proposed a better approach for performing reliability analyses by combining the SVM with the important sampling technique and demonstrating that the improved approach is effective. Hurtado [5] has developed a computationally efficient reliability algorithm based on importance sampling and SVM in the context of the traditional important sampling method. Combining these two methods based on linear and nonlinear transformations is also very powerful in the classic series problem. Bourinet et al. [6] combined subset simulation with the SVM (SS-SVM) and demonstrated that this approach could address reliability problems with low failure probabilities and large numbers of up to a few hundred random variables. Li et al. [7] showed that SVM could be used as a limit state approximator for non-training data small prediction errors can guarantee. Basudhar and Missoum [8] achieved a precise approximate decision function using the SVM with less function evaluations. The SVM-based design is capable of handling discontinuous responses. Syriopoulos et al. [9] defined a suitable support vector regression (SVR) model whose predictive capacity gives a satisfactory, robust, and promising way to predict valuable benefits and merits in the shipping industry for ship price forecasts. Zhu et al. [10] developed a least squares SVM with mixed kernels to forecast asset prices and show that the suggested method can achieve greater forecasting accuracy and trading performance. Zhang et al. [11] used SVM-based MCS to compute failure probability's lower and upper bounds. They demonstrated that the suggested structural reliability analysis approach is accurate and efficient when using random and interval variables. Roy et al. [12] used a support vector regression (SVR)-based metamodel and the direct MCS technique, and the findings suggest that the SVR-based approach is closer to MCS reliability outcomes than the conventional polynomial response surface (PRS) approach. Wang et al. [13] developed an adaptive method based on the Bayesian SVR model with a distance constraint term added to the learning function to improve the uniformity of samples. Because no embedded optimization method or iso-probabilistic transformation is required for reliability analysis, the results show that it is simple to implement. You et al. [14] proposed a random fuzzy SVM-based Particle Swarm Optimization (PSO) approach to address structural reliability problems involving implicit limit state function (LSF). Different numbers of particles or different termination conditions in optimization will produce a small variation in the results compared to the extreme response surface based on the simulated annealing (SAERS) method and MCS.

Emadi and Mazinani [15] used density-based detection techniques to assess the accuracy of DBSCAN algorithm input data. The points in low-density regions are detected as anomalies by this algorithm. Gong et al. [16] showed that the density-based algorithm’s output improves the SVMs. Amami and Smiti [17] used DBSCAN with incremental learning to detect noisy samples. The output model is then fed into a SVM classifier with a radial basis function (RBF) kernel to distinguish between normal and pathological voices. Ahmed and Razak [18] combined DBSCAN with one-class SVM for noise reduction. DBSCAN analysis reveals that accurate thresholds are a crucial requirement and that their absence leads to suboptimal results. Through SVM regression, Zhou et al. [19] established the nonlinear relationship between Wi-Fi channel state information (CSI) fingerprints and target locations. Principal component analysis (PCA) is used to extract the most useful features and reduce CSI fingerprint dimension, and DBSCAN is used to reduce CSI fingerprint noise.

SVM-based structural reliability analysis affects training samples with a low failure probability. The aim of this study is to develop a low-cost machine learning technique for predicting the probability of structural failure. If the load samples controlled uniformly at the interval $[\mu, \mu + K\sigma]$ and the strength samples at the interval $[\mu - K\sigma, \mu]$ then finding the K value is a tedious method. To overcome this drawback, use the DBSCAN algorithm, which is straightforward and reduces computational effort. DBSCAN is a base algorithm for density-based clustering. Clusters of different sizes can be discovered from a dataset containing noise and outliers. DBSCAN algorithm is used to exclude noisy samples. Then, the rest of the data is used to train SVM regression and the possible failure find out using MCS methods (Fig. 1).

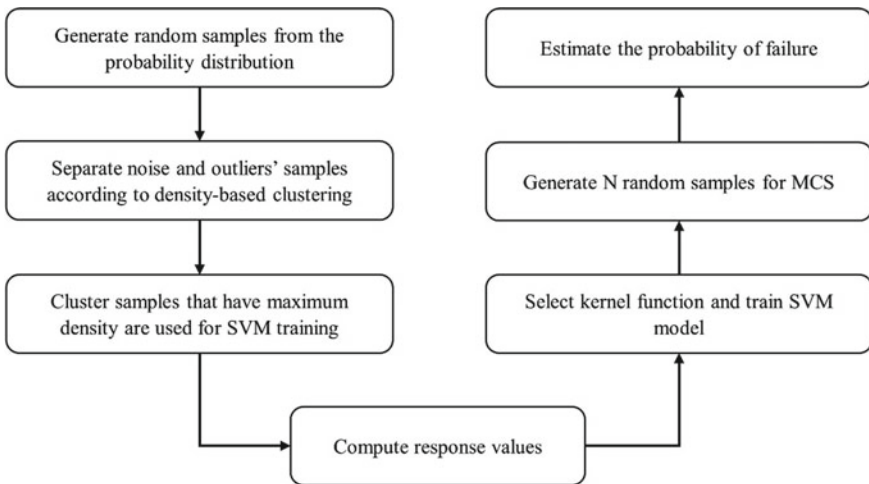


Fig. 1 Flowchart to estimate failure probability using DBSCAN and SVM

2 Sampling Procedure and Data Scaling

MinPts and eps are two parameters used by the DBSCAN algorithm. The number of points that must be clustered together for a region to be considered dense is given by minPts. eps (ϵ) is a distance measure used to find points in the neighborhood of a given point. DBSCAN draws an epsilon radius circle around each data point and categorizes them as core points, border points, noise, and outliers. If the circle around a data point contains at least minPts number of points, it is considered a core point. It is categorized as a border point if the number of points is less than minPts, and it is categorized as noise and outliers if there are no other data points within epsilon radius of any data point. MinPts are considered as $2 \times$ number of independent variables. The distance between the two points is computed using the Euclidean distance formula in this algorithm.

$$\text{dist}(p, q) = \sqrt{\sum_i^n (q_i - p_i)^2} \quad (1)$$

where p and q are two Euclidean n -space points.

Starting from the initial point, q_i , p_i are the Euclidean vectors and n is the n -space.

Instability of the SVM training because of its various physical and dimensional properties of the input variables can be alleviated or rounding error of the computer can be reduced by scaling the data using the following equation:

$$X_i = \frac{x_i - \mu_i}{\sigma_i} \quad (2)$$

where the random variable's i th sample is x_i , mean is μ_i and the standard deviation is σ_i , and X_i is the corresponding x_i i th scaled sample.

3 Nonlinear SVM

SVM algorithm that uses training data to generate an optimal classification hyperplane or decision function. Future data points can be classified or regression results can be easily predicted once the decision function is established. Linearly not separable data can be classified by linear SVM with soft margin or nonlinear SVM. However, the soft margin SVM may be trained with decision boundaries with very high margins, thus misclassifying many training data. In this case, we introduced nonlinear SVM. The nonlinear SVM's key principle is that nonlinear input data can be converted into a higher-dimensional space as a linear representation of data as shown in Fig. 2, since the number of supporting vectors or training data defines SVM efficiency, not through the dimension of the data. Then search for the linear decision function to separate the transformed higher-dimensional data.

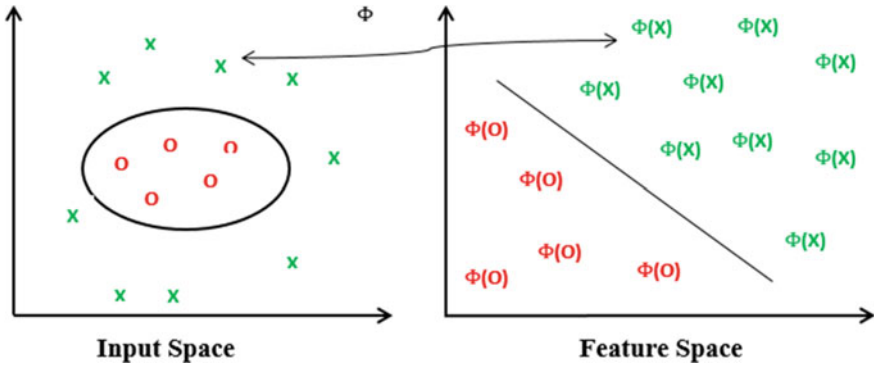


Fig. 2 A nonlinear separating region transformed into a linear one

Kernel function $K(x)$ takes a dot product of the higher-dimensional feature vectors such that,

$$K(x^i, x^j) = \Phi(x^i) \cdot \Phi(x^j) \tag{3}$$

where $K(x^i, x^j)$ is a kernel function satisfying mercer condition, input vectors x^i and x^j then the feature vector $\Phi(x^i)$ or even the mapping $\Phi(x^j)$ itself would not have to be known. In SVM, this is done by using kernel functions.

After applying the kernel, the training problems become

$$\begin{aligned} \max_{L(\alpha)} &= \sum_{i=1} \alpha^i - \frac{1}{2} \sum_{i=1} \sum_{j=1} \alpha^i \alpha^j y^i y^j K(x^i, x^j) \\ \text{s.t.} \quad &\alpha^i \geq 0; \quad i = 1, 2, \dots, m \\ &\sum_{i=1} \alpha^i y^i = 0 \end{aligned} \tag{4}$$

4 Computational Methods for Reliability Analysis

By defining the function (Z) between strength and load, the reliability of structures can be demonstrated. This function is called limit state function (LSF). The LSF is separated from its region of failure and safety. If load (L) and resistance (R), then LSF is written as:

$$Z = R - L = g(X) \tag{5}$$

where $X = (X_1, X_2, \dots, X_n)$. LSFs are negative ($g(X) \leq 0$) for system failure conditions and LSFs are positive ($g(X) > 0$) when the system is stable. Therefore, by integrating across the region $g(X) \leq 0$, then failure probability can be calculated.

$$P_f = \int_{g(X) \leq 0} \int \dots \int f_x(x_1, x_2, \dots, x_n) dx_1 \cdot dx_2 \dots dx_n \tag{6}$$

For the random variables X_1, X_2, \dots, X_n , the f_x function is a joint probability density function. Implicit LSF brings complexity to reliability analysis for most of the practical structures.

5 Numerical Examples

Three numerical examples, such as a three-span beam, a two-span beam, and a frame structure with known exact performance functions, illustrate the application and effectiveness of the proposed method. The distribution of the basic variables was used to generate the training samples.

Example 1 Reliability analysis of three-span continuous beam.

Length of each span $L = 5$ m. Consider the maximum deflection of a three-span beam is $\frac{L}{360}$, the failure probability of this beam is presented by Li et al. [7]. LSF written as,

$$g(w, E, I) = \frac{L}{360} - 0.0069 \frac{wL^4}{EI} \tag{7}$$

where w indicates for distributed loads, E for elasticity modulus, and I for moment of inertia. Table 1 shows the distribution parameters of the basic variables, which are normally distributed. Generate 30 data from their distribution and exclude 3 data according to the DBSCAN algorithm as shown in Fig. 3, assumed by epsilon = 1.6 Euclidean and minimum points = 6. Then, the rest of the 27 data were used in SVM. Responses of 27 data points shall be analyzed using the LSF. There were 21 data points utilized for training and 6 for testing. The polynomial kernel function has been used in SVM. Train SVM model and to find out best parameters used grid search algorithm. Generate $N (=10^5)$ random samples for MCS and estimate failure probabilities using SVM model. The results of direct MCS, FORM, and the proposed method are shown in Table 2.

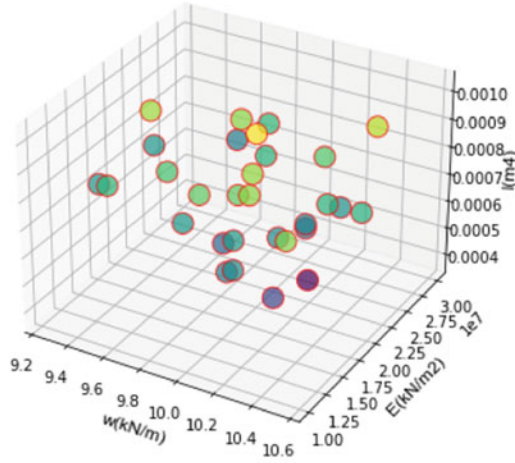
Example 2 Reliability analysis of a two-span beam as shown in Fig. 4. Length of each span $L = 6$ m. Consider the maximum deflection of a two-span beam is $\frac{L}{360}$.

LSF of two-span beam written as:

Table 1 The distribution parameters of variables

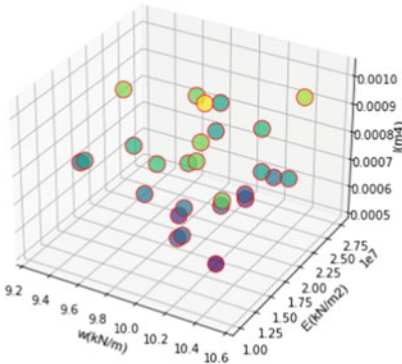
Random variables	Mean	Standard deviation
w (kN/m)	10	0.4
E (kN/m ²)	2×10^7	0.5×10^7
I (m ⁴)	8×10^{-4}	1.5×10^{-4}

Dataset



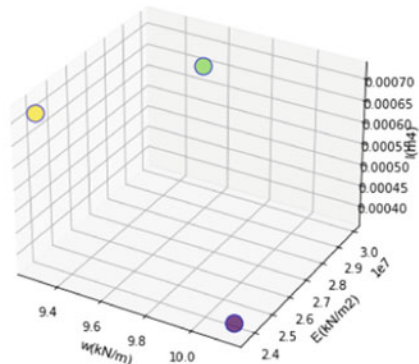
(a) Dataset (30 points)

Core and Border point



(b) Training Samples (27 points)

Noise and outliers



(c) Noise and Outliers Samples (3 points)

Fig. 3 Separate noisy data from the randomly generated dataset for Example 1

Table 2 The probability of failures using different methods of Example 1

Methods	Data	Probability of failure (P_f)
SVM-MCS	27	0.00084
MCS	100,000	0.00087
FORM	–	0.000735

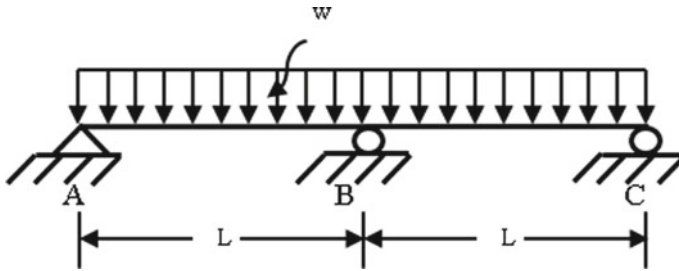


Fig. 4 Schematic of two-span beam

$$g(w, E, I) = \frac{L}{360} - \frac{wL^4}{185EI} \tag{8}$$

where w indicates for distributed loads applied on the beam, E for elasticity modulus, and I for moment of inertia. The distribution parameters of the basic variables are shown in Table 3, which are normally distributed. Generate 30 data from their distribution and exclude 3 data according to the DBSCAN algorithm as shown in Fig. 5, assumed by $\epsilon = 1.6$ Euclidean and minimum points = 6. Then, the rest of the 27 data were used in SVM. Responses of 27 data points shall be analyzed using the LSF. There were 21 data points utilized for training and 6 for testing. The polynomial kernel function has been used in SVM. The SVM parameters can be found using the grid search algorithm. Generate $N (=10^5)$ random samples for MCS and estimate failure probabilities using SVM model. The results of direct MCS, FORM, and the proposed method are shown in Table 4.

Example 3 A frame structure built into foundations A and D as shown in Fig. 6. Transverse point load (P) applied at the midpoint of BC , and a horizontal load (P) applied at point C . The maximum deflection is considered $\frac{L}{240}$.

LSF of frame structure written as:

Table 3 The mean and standard deviation of all variables

Random variables	Mean	Standard deviation
w (kN/m)	12	0.5
E (kN/m ²)	2×10^7	0.5×10^7
I (m ⁴)	8×10^{-4}	1.5×10^{-4}

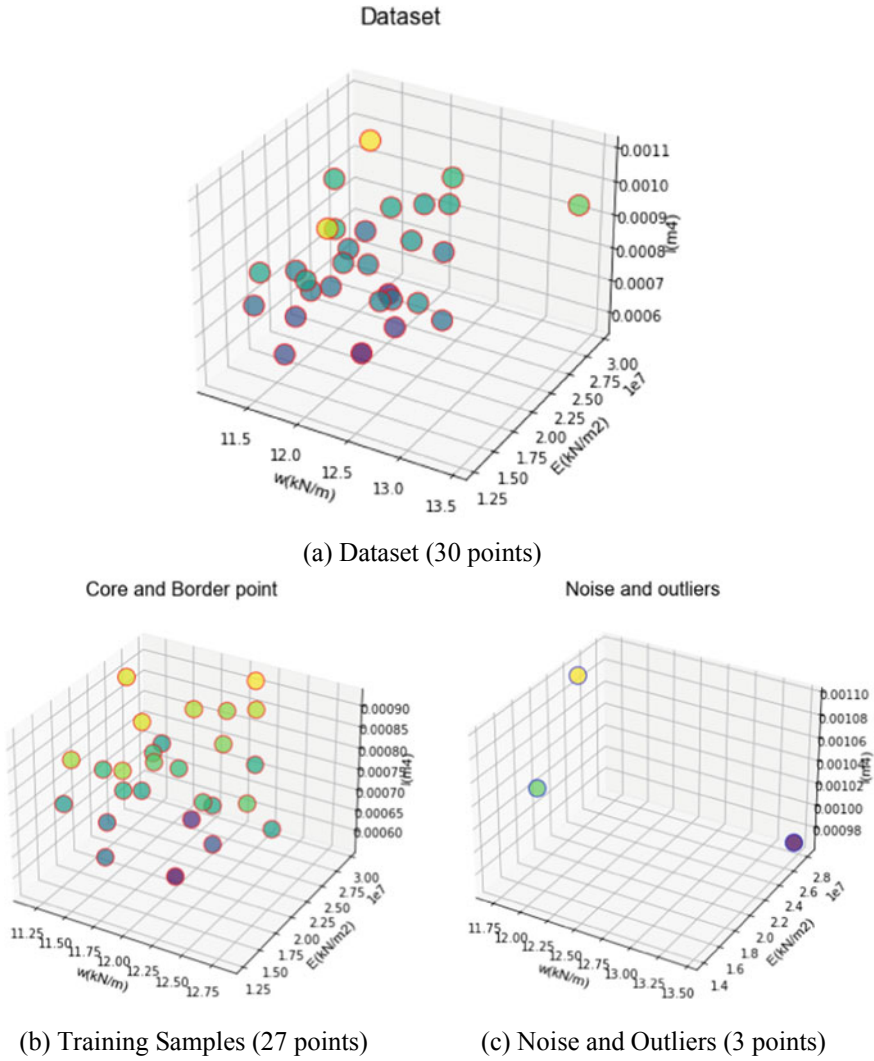
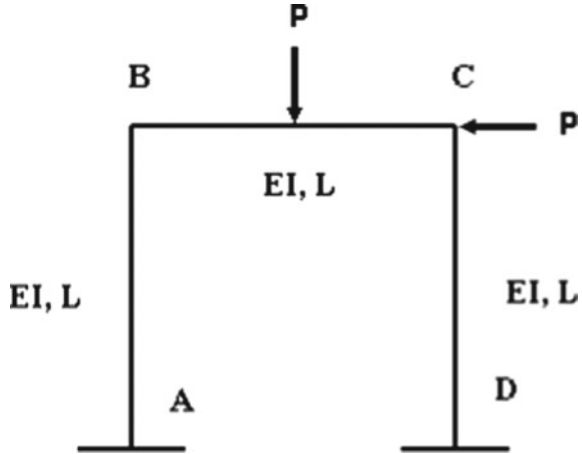


Fig. 5 Separate noisy data from the randomly generated dataset for Example 2

Table 4 The failure probabilities using different methods of Example 2

Methods	Data	Probability of failure (P_f)
SVM-MCS	27	0.00572
MCS	100,000	0.00537
FORM	–	0.00417

Fig. 6 Frame structure



$$g(L, P, E, I) = \frac{L}{240} - \frac{5PL^3}{84EI} \tag{9}$$

where L indicates the length of each member, E for the modulus of elasticity and I for the moment of inertia. Table 5 shows the distribution parameters of the basic variables, which are normally distributed. Generate 30 data from their distribution and exclude 7 data according to the DBSCAN algorithm assumed by $\epsilon = 1.7$ Euclidean and minimum points = 8. Then, the rest of the 23 data were used in SVM. Responses of 23 data points shall be analyzed using the LSF. There were 18 data points utilized for training and 5 for testing. The polynomial kernel function has been used in SVM. The SVM parameters can be found using the grid search algorithm. Generate $N (=10^5)$ random samples for MCS and estimate failure probabilities using SVM model. The results of direct MCS, FORM, and the proposed method are shown in Table 6.

Table 5 The distribution parameters of random variables

Random variables	Mean	Standard deviation
L (m)	3	0.4
P (kN)	2	0.2
E (kN/m ²)	2×10^8	2×10^7
I (m ⁴)	1.8×10^{-5}	0.5×10^{-5}

Table 6 The failure probabilities using different methods of Example 3

Method	Data	Probability of failure (P_f)
SVM-MCS	23	0.00030
MCS	100,000	0.00029
FORM	–	0.00043

6 Conclusions

The impact of SVM-based structural reliability analysis on training samples with a low failure probability is significant. The DBSCAN algorithm is being used to obtain appropriate training data to address this issue. The DBSCAN algorithm excludes noise and outlier samples after samples are generated from their distribution. The remaining data are used as input of SVM regression method, and the probability of failure of the structures can be found out based on MCS method. The proposed method is illustrated on three practical engineering problems. The proposed method's efficiency is investigated and compared with the direct MCS method and FORM. It is found that the proposed method is straightforward to implement and takes less computational cost. 27, 27, and 23 samples are used for three-span beam, two-span beam, and frame structure problems. So, this method is suitable for less number of training data to get acceptable limit of failure probability. The suggested methodology of the present study focuses on limited applications for known exact performance functions, and all random variables are considered as normally distributed. This proposed framework can be validated in implicit performance functions for large and complicated structures in future studies, providing us with a better understanding of the potential of the proposed techniques.

References

1. Jamwal, A., Agrawal, R., Sharma, M., Kumar, A. Kumar, V., Garza-Reyes, J. A. A.: Machine learning applications for sustainable manufacturing: A bibliometric-based review for future research. *J. Enterprise Inf. Manage.*, 1741–0398 (2021)
2. Rocco, C.M., Moreno, J.A.: Fast monte carlo reliability evaluation using support vector machine. *Reliab. Eng. Syst. Saf.* **76**, 237–243 (2002)
3. Hurtado, J.E., Alvarez, D.A.: Classification approach for reliability analysis with stochastic finite-element modeling. *J. Struct. Eng.* **129**, 1141–1149 (2003)
4. Dai, H., Zhang, H., Wang, W.: A support vector density-based importance sampling for reliability assessment. *Reliab. Eng. Syst. Saf.* **106**, 86–93 (2012)
5. Hurtado, J.E.: Filtered importance sampling with support vector margin: a powerful method for structural reliability analysis. *Struct. Saf.* **29**, 2–15 (2007)
6. Bourinet, J.M., Deheeger, F., Lemaire, M.: Assessing small failure probabilities by combined subset simulation and support vector machines. *Struct. Saf.* **33**, 343–353 (2011)
7. Li, H.S., Lu, Z.Z., Yue, Z.F.: Support vector machine for structural reliability analysis. *Appl. Math. Mech.* **27**(10), 1295–1303 (2006)
8. Basudhar, A., Missoum, S.: Adaptive explicit decision functions for probabilistic design and optimization using support vector machines. *Comput. Struct.* **86**, 1904–1917 (2008)
9. Syriopoulos, T., Tsatsaronis, M., Karamanos, I.: Support vector machine Algorithms: An Application to ship price forecasting. *Comput. Econ.* **57**, 55–87 (2021)
10. Zhu, B., Ye, S., Wang, P., Chevallier, J., Wei, Y.M.: Forecasting carbon price using a multi-objective least squares support vector machine with mixture kernels. *J. Forecast.* **41**, 100–117 (2022)
11. Zhang, J., Xiao, M., Gao, L., Chu, S.: Probability and interval hybrid reliability analysis based on adaptive local approximation of projection outlines using support vector machine. *Comput. Aided Civ. Infrastruct. Eng.* **34**, 991–1009 (2019)

12. Roy, A., Manna, R., Chakraborty, S.: Support vector regression based metamodeling for structural reliability analysis. *Probab. Eng. Mech.* **55**, 78–89 (2019)
13. Wang, J., Li, C., Xu, G., Li, Y., Kareem, A.: Efficient structural reliability analysis based on adaptive bayesian support vector regression. *Comput. Methods Appl. Mech. Eng.* **387**, 114172 (2021)
14. You, L.F., Zhang, J.G., Zhou, S., Wu, J.: A novel mixed uncertainty support vector machine method for structural reliability analysis. *Acta Mech.* **232**, 1497–1513 (2021)
15. Emadi, H.S., Mazinani, S.M.: A novel anomaly detection algorithm using DBSCAN and SVM in wireless sensor networks. *Wireless Pers. Commun.* **98**, 2025–2035 (2018)
16. Gong, L., Yamamoto, T., Morikawa, T.: Identification of activity stop locations in GPS trajectories by DBSCAN-TE method combined with support vector machines. *Transp. Res. Procedia* **32**, 146–154 (2018)
17. Amami, R., Smiti, A.: An incremental method combining density clustering and support vector machines for voice pathology detection. *Comput. Electric. Eng.*, 1–9 (2016)
18. Ahmed, K.N., Razak, T.A.: Scalable density based spatial clustering with integrated one-class SVM for noise reduction. *International J. Eng. Technol.* **7**, 28–32 (2018)
19. Zhou, R., Chen, J., Lu, X., Wu, J.: CSI fingerprinting with SVM regression to achieve device-free passive localization. *IEEE, Macau, China* (2017)

The Ensemble of Ant Colony Optimization and Gradient Descent Technique for Efficient Feature Selection and Data Classification



Md. Nayer and Subhash Chandra Pandey

Abstract Feature selection is of the paramount importance in the domain of data classification task especially when the number of features is huge. Further, gradient descent is an important technique for training the perceptron network and to find the hyper-plane that can classify the nonlinearly classifiable data sets with least error. However, modifying the synaptic weights of each input neuron is cumbersome and also time taking. In this work, an ensemble approach of ant colony optimization (ACO) and gradient descent technique is used. In this ensemble model, ACO will first select the important features, and subsequently, the gradient descent technique will modify the synaptic weights of these input features. This combined approach renders reduced training time of the perceptron network with enhanced classification accuracy of the test data sets. Experimental analysis and validation of proposed ensemble technique has also been incorporated for standard data sets. Over and above, the proposed technique have also been implemented for the purpose of identifying different commonly used mechanical parts.

Keywords Feature selection · Ant colony optimization · Gradient descent · Classification · Mechanical part identification

1 Introduction

It is an obvious fact that the data volume is increasing with the augmented growth of information technology and various state-of-the-art techniques within the domain of machine learning are being used unrevealing the information from data such as artificial intelligence (AI) and deep learning [1–3]. Ant colony optimization is a probabilistic, metaheuristic and iterative method often used for solving the optimization problems. It is based on well-known mechanism known as the foraging behaviour of

Md. Nayer (✉) · S. C. Pandey
Birla Institute of Technology, Patna Campus, Bihar, India
e-mail: mohdnaiyer@gmail.com

S. C. Pandey
e-mail: s.pandey@bitmesra.ac.in

the ants. In this foraging phenomenon, the ants attempt to search shortest path from original source to destination source [4]. It has been noticed that ants leave funky material termed as pheromone. They used this pheromone for tracing the path and also strive to render the quality of path that so that other ants can follow the same path in subsequent discourse of time. Researchers have observed that ants always preferred that paths where amount of pheromone is comparatively high. Owing to the fact that the ants choose the smaller path for their establishment, the smaller route entails largest amount of pheromone in comparison with long routes.

The ACO was first time deployed in travelling sales problem (TSP) [5]. Later on, it is implemented to solve several discrete optimization issues. It is pertinent to mention that initially it is deployed to solve the classical problems such that graph-colouring, scheduling issues and routing problems of vehicles. [6]. In recent past, ACO is used in different applications like intelligent scheduling, bioinformatics issues, communication network model, etc. The fact has also been revealed by literature survey that nowadays, researchers are also working on applications of ACO algorithm to manipulate the stochastic, dynamic and multi-objective issues. In the ACO algorithm, virtual pheromone is generated for updating their path by the artificial agent, i.e. artificial ants using the concept of decision graph. The artificial agent adopts quantity and density of pheromone for updating its path and thus guesses the solution efficiency compared with investigated solutions obtained by former agents [4].

Gradient descent is the succinct training strategy for single-layer perceptron (SLP) for classifying the nonlinearly classifiable data sets. It offers an easy learning strategy for the entire system of synaptic development across network system. The investigation of gradient descent learning is beneficial for two different causes, i.e. the various observation yields in the problems that are connected to learning and another one is if it is applied in artificial neural systems, then it will resolve various practical issues [7].

Feature selection is a method for selecting a subset of suitable features for the analysis of predictive model. The main concept regarding feature selection is such that data have several features that are insufficient, irrelevant or redundant has to be removed. Elimination of such types of features is necessary because it creates the model cumbersome and complex. Feature selection has to be done using some searching techniques that creates the feature subset and execution evaluator. The procedure of feature selection is being carried out by mingling searching and selection operation. However, accuracy of feature selection is one of the important and it requires due consideration. The feature selection approach may be classified as heuristic search, stochastic search and complete search. Four main methods used for feature selection are [8]:

- Filter selection method
- Wrapper selection method
- Hybrid feature selection approach
- Embedded feature selection methods.

In this work, we will use the ACO technique for feature selection that will entail filter-based feature selection. The feature thus obtained will be used for constructing the architecture of SLP. The Kolmogorov complexity of SLP constructed with reduced number of feature is considerably low and its training with gradient descent algorithm would be easy and fast.

Rest of the paper is organized as follows: Sect. 2 delineates brief literature review pertaining to ant colony optimization for the pursuit of feature selection. Section 3 illustrates the ACO for feature selection, gradient descent and ensemble algorithm of ACO and gradient descent. Further, Sect. 4 presents the data sets taken for experimentation and discussions pertaining to the results obtained. Finally, concluding remarks are given in Sect. 5.

2 Related Work

The extensive literature review revealed the fact that the ACO algorithm has prominently been used to solve the different problems. In the late 2019, ACO is used for feature selection and is applied to reduce the dimension and its performance is compared with feature subset selection obtained using PSO [9]. Further, in [10], a hybrid approach of ACO and streaming (ACO-SU) is proposed for feature selection that used the concept of symmetric uncertainty. In [4], feature selection using ACO is proposed for six different channel pre-treatment electroencephalogram (EEG) data using propagation neural network (BPNN) classification method. Furthermore, in [11], a modified graph clustering-based ant colony optimization (GCACO) algorithm is proposed to substantially enhance the performance of GCACO. This approach is called MGCACO. The MGCACO algorithm is tested on different standard benchmark data sets including sleep EEG data and its performance is highly appreciated. In [12], a wrapper-filter combination of ACO is implemented to reduce the computational complexity. This approach has been assessed on different real-life data sets, taken from UCI machine learning repository and NIPS2003 FS challenge, using K-nearest neighbours and multi-layer perceptron classifiers [12]. In addition, in [13], a persuasive feature selection algorithm that incorporates the leverage of ant colony optimization and brain storm optimization is proposed. This technique simulates the behaviour of human beings.

Further, shape recognition is an important task within the domain of computation. In this context, the viability of easy and smooth retrieval and subsequent reuse or modification of existing CAD models would certainly be of paramount importance to render the swift life cycle of new products. In literature, different techniques have been used for this pursuit. In [14–16], several shape comparison algorithms have been given. Further, in [17], an approach has been proposed for recognizing different biological shapes. Moreover, in [18], the problem of similarity measure between 3D shapes is discussed.

3 Proposed Methodology

We propose an ensemble of ant colony optimization and gradient descent technique for efficient feature selection and data classification approach. In this section, we discuss the state-of-the-art algorithm ACO and gradient descent technique for feature selection and data classification.

3.1 Ant Colony Optimization (ACO)

Ant colony optimization is a swarm intelligence-based algorithm inspired by the phenomenon of food foraging rendered by ants [19, 20]. It has been observed experimentally that without any direct communication, ants can identify the shortest route between food and their eyrie. It has also been observed that the ant releases a special type of chemical known as pheromone along the path between food source and their eyrie. Moreover, the route having greatest concentration of pheromone is chosen by other ants to pursue the shortest route. ACO algorithm uses the rightness measure to select the suitable route to solve many problems including the travelling sales person problem. The selection probability from i th to the j th node is computed using Eq. (1).

$$P_{i,j} = \frac{(\sigma_{i,j}^\alpha)(\varepsilon_{i,j}^\beta)}{\sum k S(\sigma_{i,j}^\alpha)(\varepsilon_{i,j}^\beta)} \quad (1)$$

Here $\sigma_{i,j}$ represents the pheromone's concentration onward from the route from i th to j th node and it is given in Eq. (2), $\varepsilon(i, j)$ represents heuristic function to explain the cost of selecting j th node, and α and β represents pheromone and heuristic function. Further, the equation is divided into the sum of products of heuristic and pheromone values of entire nodes that are mutually connected to the i th node. The other node is selected on the basis of the value of the node $P_{i,j}$. Once the ant completes its path, the values of pheromone are modified along with the route mentioned as:

$$\sigma'_{i,j} = \sigma_{i,j} + \sigma_{i,j} * \left[\left(1 - \frac{1}{1 + P_{i,j}} \right) \right] \quad (2)$$

Here, $\Delta\sigma_{i,j}$ shows the ant's fitness along the path and $\sigma_{i,j}$ represents the preceding values of the pheromone. The ACO algorithm does not mingle locally because it dissipates some of the pheromone on each path which is given as:

$$\sigma_{i,j} = (1 - \rho) \times \sigma'_{i,j} + \Delta\sigma_{i,j} \quad (3)$$

where, ρ is the percentage of the pheromone to be evaporated. After various iterations, when a convergence criterion is fulfilled, the algorithm stops. ACO algorithm is given

in [21] and attempt has been made to solve the plethora of problems by implementing the ACO for feature selection [22, 23]. The ensemble ACO algorithm with gradient descent is displayed in Fig. 3.

3.2 ACO Algorithm for Feature Selection

In this sub-section, we will describe the proposed methodology of using the ACO algorithm for feature selection with the help of undirected connected graph $G = (V, E)$, where the set of vertices V and E is the set of edges. Every vertex in the graph represents different feature and edges E denote interrelation between two features. The count of ants taken in ACO algorithm is also equal to the no. of features, and thus attempt has been taken to avoid getting trapped in local optimum. Let the number of features be F then $F = \text{Number of ant } (N_{\text{ANT}})$. The max count of features in a subset of feature is showed by F_{max} . Thus, $0 < F_{\text{max}} < F$. Every ant starts traversing from a vertex and keeps on traversing taxonomy of vertices. The pheromone associated with each feature is set to a steady at first. In the feature subset selection of F_{max} , features are carried out in five steps which are as follows. Different parameters used for ACO algorithm are given as F_{max} which is taken on the basis of features to be selected. $F - \text{Leader} = 25\%$ of F_{max} . Maximum number of iteration = 40. Different steps included in this approach are given below.

- Step 1: this step includes N subsets, each subset having m maximum number of features and these N subsets are evaluated by ACO.
- Step 2: Every subset gets $F - \text{Filter}$ features by choosing a degree of features randomly.
- Step 3: The f -arbitrary features are chosen that are still not incorporated into partially built subset having highest level of pheromone.
- Step 4: The remaining $\{F - \text{Leader} = (F - \text{Remain}) - (F - \text{Arbitrary})\}$ features are selected using tandem run approach.
- Step 5: All subsets are evaluated by using SLP and subset. The subset having maximum accuracy is marked as 'Local best subset'. If the accuracy of local best is superior, then the global best then local best subset is set as global best subset and considered as leader subset for future iterations.

It should be noted that in this approach, the first step executes only once, while the remaining steps executed until the stopping criteria do not satisfied. Further, after the execution of all iterations, the leader subset is treated as best subset that provides the maximum accuracy. The flow diagram for these steps is given in Fig. 1.

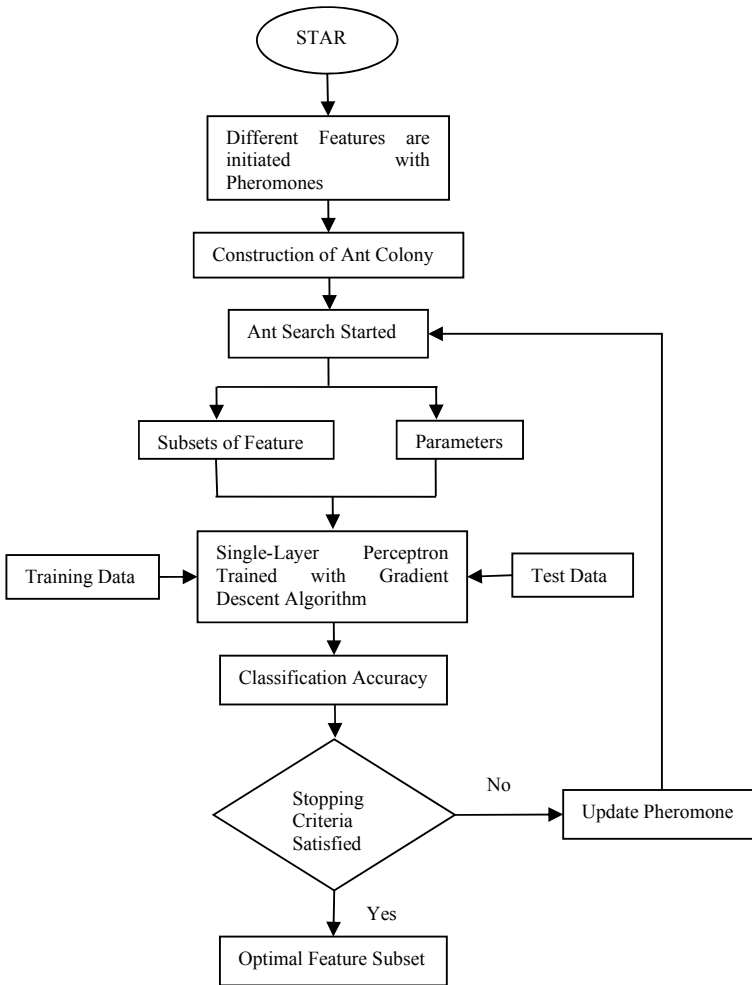


Fig.1 Flow diagram of ACO for feature selection

3.3 Gradient Descent Algorithm for Single-Layer Perceptron(SLP)

The gradient descent algorithm is used for the training purpose of single-layer perceptron (SLP), while the data sets are nonlinearly classifiable. The gradient descent algorithm provides the hyper-plane of least error as shown in Fig. 2. Different parameters taken for gradient descent technique are as follows. The slope parameter α is taken as one and the learning rate η is taken as 0.6. The activation function used is sigmoidal. The proposed ensemble algorithm of ACO and gradient descent is given in Fig. 3. The architectural details of SLP are given in Fig. 4.

Fig.2 Hyper-plane of least error

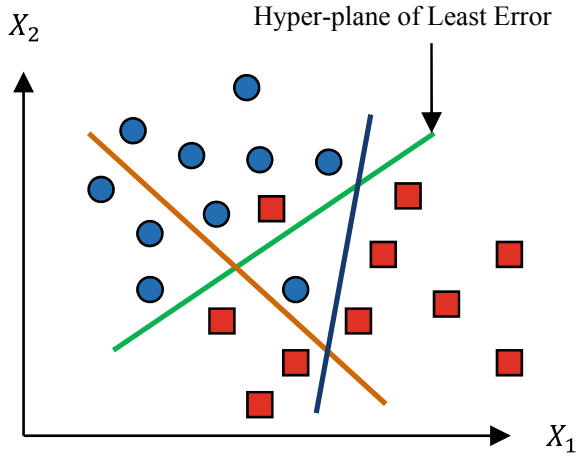


Fig.3 Ensemble of ACO and GD

```
Pheromone values initialization
repeat
  for ant A: e ( 1 , . . . , F )
    construct a solution
  end for
  for all pheromone values do
    {evaporation} the value is decreased by
    certain percentage
  end for
  for (all pheromone values corresponding to good
  solutions) do
    intensification is performed and thus the
    value is increased
  end for
until stopping criterion is met
Construct the SLP with reduced feature
Train the SLP with Gradient Descent(GD)
algorithm
```

4 Experimental Results and Discussions

In this section, we will describe different data sets considered for the purpose of experimentation and the analysis of the results obtained.

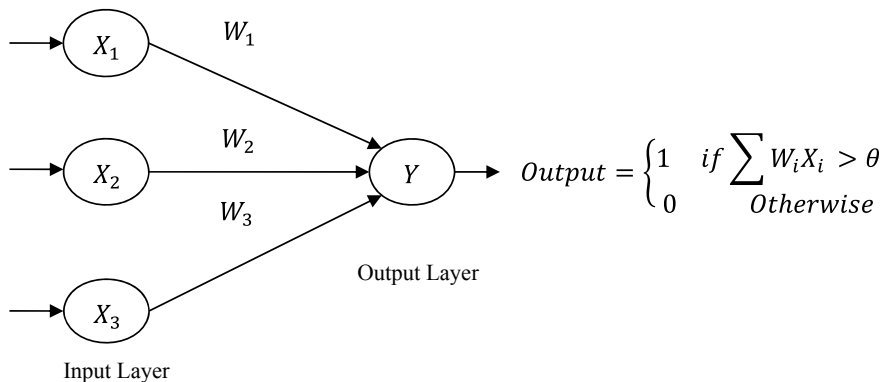


Fig.4 Architectural details of SLP

4.1 Data Set Details

The proposed ensemble algorithm is implemented for the classification task of the three data sets. These are the Wisconsin breast cancer data set, Ionosphere and Dermatology [24]. The Wisconsin breast cancer data set entails 32 features and 569 examples, and the number of classes is 2. The Ionosphere data set is having 34 features and 351 examples and 2 classes (binary classification), where the values of features can be real or integer. Further, the Dermatology data set consists of 33 features and 366 examples and 6 classes. In Dermatology data set, the features are only integer. These three data sets are freely available on UCI machine learning repository.

4.2 Classification Performance Using SLP with Gradient Descent

The SLP trained with gradient descent algorithm is of the paramount importance specifically in the domain of data classification. It provides the optimum classification accuracy for the nonlinearly classifiable data sets. Classification of binary data sets is considered as naive classification problem. However, from a different perspective, it is difficult to classify because so many hyper-planes can be drawn for binary data sets for the purpose of classification. In large data sets, in which numbers of features are substantially high, it is considerably difficult to use the SLP trained with gradient descent algorithm because of the associated complexities. The proposed ensemble approach reduces the number of features succinctly low and thus enables the SLP to be implemented and trained with gradient descent easily with high classification accuracy. Table 1 gives the classification accuracy of SLP trained with gradient descent algorithm for Wisconsin breast cancer data set for different values of number of features selected, value of F – Filter, value of F – Leader and value of F –

Arbitrary. Classification accuracy of SLP trained with gradient descent algorithm with these parameters for Ionosphere and Dermatology data set is given in Tables 2 and 3.

In this work, 70% of all the data sets are taken as training data set and remaining 30% is taken as the test data set. Further, the classification accuracy is calculated using the formula given in Eq. 4.

Table 1 Classification performance for Wisconsin breast cancer data set

No. of feature selected	Value of F-filter	Value of F-leader	Value of F-arbitrary	Classification accuracy (%) of SLP trained with gradient descent
20	11	5	6	86.34
22	13	5	6	88.21
24	14	6	6	91.11
26	16	6	6	93.33
27	17	7	6	93.84

Table 2 Classification performance for Ionosphere data set

No. of feature selected	Value of F-filter	Value of F-leader	Value of F-arbitrary	Classification accuracy (%) of SLP trained with gradient descent
20	11	5	6	88.78
22	13	5	6	91.47
24	14	6	6	94.63
26	16	6	6	95.23
27	17	7	6	96.51

Table 3 Classification performance for Dermatology data set

No. of feature selected	Value of F-filter	Value of F-leader	Value of F-arbitrary	Classification accuracy (%) of SLP trained with gradient descent
20	11	5	6	83.67
22	13	5	6	85.32
24	14	6	6	89.89
26	16	6	6	91.46
27	17	7	6	91.97

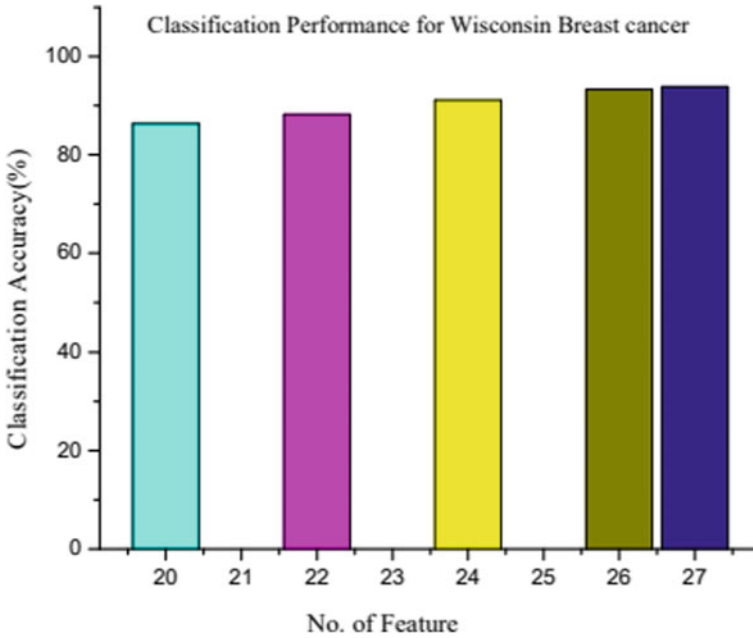


Fig. 5 Classification accuracy of Wisconsin breast cancer data set

$$\text{Classification Accuracy} = \frac{T_P + T_N}{T_P + T_N + F_P + F_N} \quad (4)$$

where T_P = True Positive, T_N = True Negative,
 F_P = False Positive, F_N = false Negative

The bar-graphs of number of feature selected versus classification accuracy are shown in Figs. 5, 6 and 7. It is explicit from experimental results that the classification accuracies of all these three data sets are increasing with the increase of number of features. The classification accuracies render the fact that the proposed model performs satisfactory.

4.3 Identification of Mechanical Parts

The proposed ensemble technique of ACO and GD is also used for 3D model classification. For this purpose, authors implemented the proposed algorithm for classifying the multi-class data set of different mechanical parts. Authors created this data set by creating their 3D model. Indeed, this data set contains the 3D images of different mechanical parts generated by computer-aided design (CAD) tool. Descriptions of mechanical part taken in this experimentation are given in Table 4. Further,

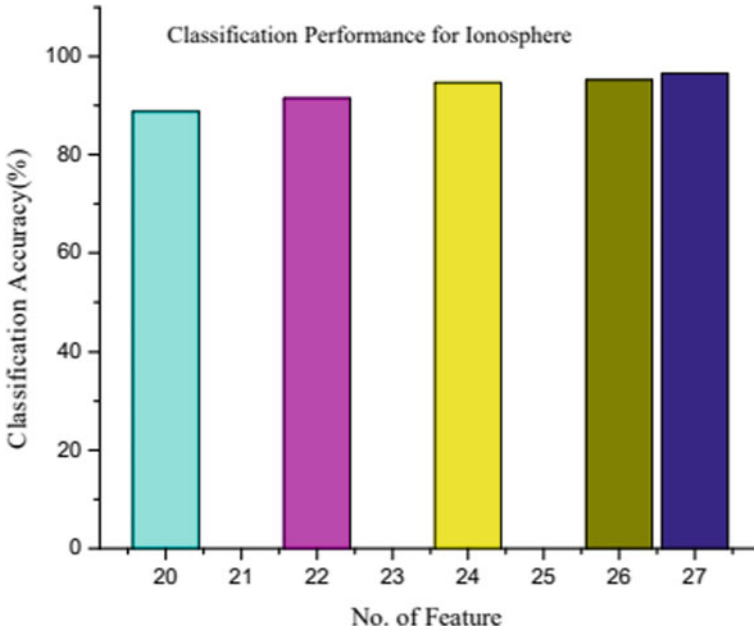


Fig. 6 Classification accuracy of Ionosphere cancer data set

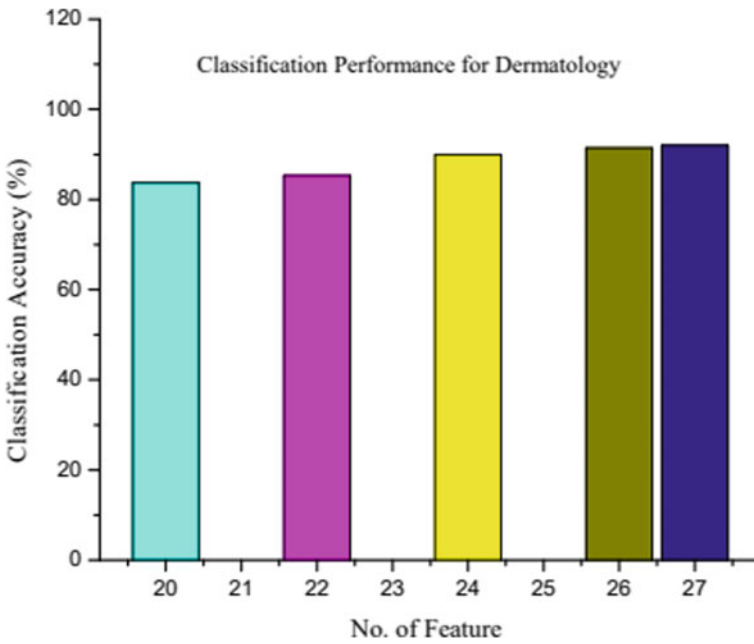


Fig. 7 Classification accuracy of Dermatology data set

Table 4 List of mechanical part considered in experimentation

S. No./	Name of the mechanical parts	Number of parts considered	S. No.	Name of the mechanical parts	Number of parts considered
1	Screws	28	11	Studs	18
2	Washers	25	12	Pins	19
3	Nuts	21	13	Bearings	15
4	Bolts	27	14	Bushes	21
5	Rivets	18	15	Springs	24
6	Plugs	20	16	Locating pins	22
7	Seals	24	17	Gears	26
8	Slides	25	18	Axis	27
9	Clamps	20	19	Brackets	23
10	Keys	20	20	Ball plungers	18

we obtained different features of each element from the 3D models generated by the CAD. Thus, authors developed a data set composed by twenty categories and containing 441 elements with their respective features of minimum box sizes (MBS) required containing the elements. Some pertinent features are:

- Volume of MBS required containing the element
- Length of MBS
- Width of MBS
- Height of MBS
- Ratio of length and height of MBS
- Ratio of length and width of MBS
- Ratio of height and width of MBS
- Diagonal length
- Surface area of the mesh.

Further, authors applied the five-fold cross validation. For this purpose, the number of each part is divided almost five equal parts. Furthermore, $1/5^{\text{th}}$ of each part is collected together and this collection is taken as test data. The remaining $4/5^{\text{th}}$ of each part is also collected together and it acts as training data. This process is repeated five times and classification accuracy of different part is obtained by the proposed algorithm in every repetition. Finally, average of all these five classification accuracies is worked out. The average classification accuracy for each part as obtained by proposed technique is given in Table 5.

Table5 Average classification accuracy obtained after five-fold cross validation

S. No.	Name of the mechanical parts	Average classification accuracy (%)	S. No.	Name of the mechanical parts	Average classification accuracy (%)
1	Screws	85.32	11	Studs	87.89
2	Washers	89.65	12	Pins	84.76
3	Nuts	88.72	13	Bearings	90.01
4	Bolts	86.12	14	Bushes	89.94
5	Rivets	88.73	15	Springs	84.32
6	Plugs	20	16	Locating pins	84.53
7	Seals	86.84	17	Gears	85.86
8	Slides	87.31	18	Axis	85.97
9	Clamps	87.61	19	Brackets	89.42
10	Keys	85.47	20	Ball plungers	86.11

5 Conclusions

We proposed an ensemble model of ACO and SLP trained with gradient descent technique. Further, we used this ensemble model for the feature selection and for the purpose of data classification. Performance of proposed model has experimentally been analysed for three data sets, namely Wisconsin breast cancer data set, Ionosphere data set and Dermatology data set. The classification accuracy of proposed model for these three data sets is satisfactory. Moreover, the performance of proposed technique also rendered satisfactory results for the identification of commonly used mechanical parts using 3D models and thus can be used for the automatic classification of different mechanical parts. Therefore, proposed method can be used for the matching and the retrieval of 3D CAD data. The implementation of proposed ensemble model for different other tasks such as clustering, anomaly detection and for the purpose of intrusion detection can be considered as the future research endeavour of the proposed work. The performance comparison of proposed ensemble model with other state-of-the-art techniques for miscellaneous tasks can also be considered as the future research direction.

References

1. Li, J., Cheng, K., Wang, S., Morstatter, F., Trevino, R.P., Tang, J., Liu, H.: Feature selection: a data perspective. *ACM Comput. Surv.* **50**(6), 94 (2018)
2. Sharma, R., Bist, A.S.: Machine learning: a survey. *Int. J. Eng. Sci. Res. Technol.* **4**(3), 708–716 (2015)
3. Kilani, A., Ben Hamida, A., Hamam, H.: Artificial intelligence review. In: Khosrow-Pour, M. (ed.) *Encyclopaedia of Information Science and Technology*, 4th ed. Hershey, pp. 106–119, PA, USA, IGI Global, (2018), <https://doi.org/10.4018/978-1-5225-2255-3.ch010>

4. Erguzel, T.T., Ozekes, S., Gultekin, S., Tarhan, N.: Ant colony optimization based feature selection method for QEEG data classification. *Psychiatry Investig.* **11**(3), 243 (2014)
5. Dorigo, M., Gambardella, L.M.: Ant colony system: a cooperative learning approach to the traveling salesman problem. *IEEE Trans. Evol. Comput.* **1**, 53–66 (1997)
6. Katiyar, S., Ibraheem, N., Ansari, A.Q.: Ant colony optimization: a tutorial review. In: National Conference on Advances in Power and Control, pp. 99–110 (2015)
7. Baldi, P.: Gradient descent learning algorithm overview: A general dynamical systems perspective. *IEEE Trans. Neural Netw.* **6**(1), 182–195 (1995)
8. Rajesh, D., Rahul, K., Ebenezer, J., Vishnu, K.: An. Ant colony optimization based feature selection for data classification. *Int. J. Recent Technol. Eng. (IJRTE)*, **7**(5S4) (2019), ISSN: 2277–3878
9. Selvarajan, D., Jabar, A.S.A., Ahmed, I.: Comparative analysis of PSO and ACO based feature selection techniques for medical data preservation. *Int. Arab J. Inf. Technol.* **16**(4), 731–736 (2019)
10. Fahad, L.G., Tahir, S.F., Shahzad, W., Hassan, M., Alquhayz, H., Hassan, R.: Ant colony optimization-based streaming feature selection: an application to the medical image diagnosis. *Sci. Program.*, (2020)
11. Ghimatgar, H., Kazemi, K., Helfroush, M.S., Aarabi, A.: An improved feature selection algorithm based on graph clustering and ant colony optimization. *Knowl.-Based Syst.* **159**, 270–285 (2018)
12. Ghosh, M., Guha, R., Sarkar, R., Abraham, A.: A wrapper-filter feature selection technique based on ant colony optimization. *Neural Comput. Appl.*, pp 1–19 (2019)
13. Liang, H., Wang, Z., Liu, Y.: A new hybrid ant colony optimization based on brain storm optimization for feature selection. *IEICE Trans. Inf. Syst.* **102**(7), 1396–1399 (2019)
14. Mak, L., Grandison, S., Morris, R.J.: An extension of spherical harmonics to region-based rotationally invariant descriptors for molecular shape description and comparison. *J. Mol. Graph. Model.* **26**(7), 1035–1045 (2008)
15. Rothganger, F., Lazebnik, S., Schmid, C., Ponce, J.: 3D object modeling and recognition using local affine-invariant image descriptors and multi-view spatial constraints. *Int. J. Comput. Vision* **66**(3), 231–259 (2006)
16. Ritchie, D.W., Kemp, G.J.: Fast computation, rotation, and comparison of low resolution spherical harmonic molecular surfaces. *J. Comput. Chem.* **20**(4), 383–395 (1999)
17. Yu-Shen, L., Fang, Y., Ramani, K.: IDSS: deformation invariant signatures for molecular shape comparison. *BMC Bioinformatics* **10**(1), 1 (2009)
18. Osada, R., Funkhouser, T., Chazelle, B., Dobkin, D.: Shape distributions. *CM Trans. Graphics (TOG)* **21**(4), 807–832 (2002)
19. Kennedy, J., Russell, E.: Particle Swarm Optimization, IEEE, Portland, OR, USA (1995)
20. Khan, Z., Fang, S., Koubaa, A., Fan, P., Abbas, F., Farman, H.: Street-centric routing scheme using ant colony optimization based clustering for bus-based vehicular ad-hoc network. *Comput. Electr. Eng.* **86**, 106736–106746 (2020)
21. Martens, D., Backer, M., Haesen, R., Member, S., Vanthienen, J., Snoeck, M., Baesens, B.: Classification with ant colony optimization. *IEEE Trans. Evol. Comput.* **11**(5), 651–665 (2007)
22. Abd-Alsabour, N., Randall, M.: Feature selection for classification using an ant colony system. In: Proceedings of 6th IEEE International Conference on E-Science Workshops, Brisbane, pp. 86–91 (2010)
23. Dorigo, M., Thomas, S.: The Ant Colony Optimization Metaheuristic: Algorithms, Applications, and Advances, pp. 250–285. *Handbook of Metaheuristics*, Boston (2003)
24. Dua, D., KarraTaniskidou, E.: UCI Machine Learning Repository [<http://archive.ics.uci.edu/ml>]. Irvine, CA, University of California, School of Information and Computer Science (2017)

s-Regularity Via Soft Ideal



Archana K. Prasad and S. S. Thakur

Abstract In this paper, the concept of soft s - \mathcal{J} -regular spaces in soft ideal topological spaces has been introduced and studied.

Keywords Soft sets · Soft topology · Soft ideal · Soft \mathcal{J} -regular · Soft s - \mathcal{J} -regular spaces

1 Introduction

The concept of soft set was introduced by Molodtsov [23] as a general mathematical tool for dealing with problems that contain uncertainty. Many authors such as Ali and his coworkers [2], Maji and his coworkers [21], Pie and Miao [28], Das and Samant [6], Sahin and Kuçuk [32] and others contributed to the development of soft set theory. Molodtsov [23] in his pioneering work provided some applications of soft sets in different fields and elaborated its merits compared with probability theory and fuzzy set theory which deals with vagueness or uncertainties. Recently, soft set theory has become very prevalent tool among researchers to overcome the problems of uncertainty in different fields such as Game Theory [23], Operations Research [23, 24], Perron Integration [24], Information Theory [28], Lattice Theory [26], Recommendation System [18], Decision Theory [20], Pattern Recognition [41], Texture Classifications [25], Data Mining [11, 18, 33], Medical Sciences [42] and Education Technology [44]. In 2011, Shabir and Naz [35] introduced the concept of soft topological spaces as a generalization of topological spaces. In the same paper, they studied concepts of soft closure, soft interior, soft neighborhood, soft subspaces and soft separation axioms in soft topological spaces. The soft separation axioms further studied by Georgiou and his coworkers [8], Prasad and Thakur [29] using

A. K. Prasad (✉)

Department of Mathematics, Swami Vivekanand Government College, Lakhnadon, Seoni, Madhya Pradesh 480886, India
e-mail: akkumariprasad@gmail.com

S. S. Thakur

Department of Applied Mathematics, Jabalpur Engineering College, Jabalpur, Madhya Pradesh 482011, India

the notion of soft points. The study of soft ideal topological spaces was initiated by Kandil et al. [14] in the year 2014. In 2015, Guler and Kale [9] introduced and studied soft \mathcal{J} -regular spaces. The present paper extended the axiom of s-regularity due to Maheshwari and Prasad [19] in soft ideal topological spaces and present their studies.

2 Preliminaries

Throughout this paper, X denotes a nonempty set, Ω denotes the set of parameters and $S(X, \Omega)$ refers to the family of all soft sets of X relative to Ω . For the definition and notations of soft set and soft topology, the readers should refer [3, 6, 10, 12, 13, 16, 17, 27, 30, 31, 34, 36–40].

A nonempty collection Γ of subsets of a nonempty set X is said to be topology on X if it contains empty set and whole set and closed with respect to arbitrary union and finite intersection. The triplet (X, Γ, Ω) is called a soft topological space. The members of Γ are known as soft open sets in X and their complements called soft closed sets in X . The family of all soft closed sets over (X, Ω) is denoted by $SC(X, \Omega)$.

Lemma 2.1 [35] *Let (X, Γ, Ω) be a soft topological space. Then the collection $\Gamma_\alpha = \{\xi(\alpha) : (\xi, \Omega) \in \Gamma\}$ for each $\alpha \in \Omega$, defines a topology on X .*

Definition 2.1 [35, 45] *Let (X, Γ, Ω) be a soft topological space and a soft set (ξ, Ω) over X .*

- (i) The soft closure of (ξ, Ω) is the soft set $Cl(\xi, \Omega) = \bigcap \{(\sigma, \Omega) : (\sigma, \Omega) \text{ is soft closed and } (\sigma, \Omega) \supseteq (\xi, \Omega)\}$.
- (ii) The soft interior of (ξ, Ω) is the soft set $Int(\xi, \Omega) = \bigcup \{(\sigma, \Omega) : (\sigma, \Omega) \text{ is soft open and } (\sigma, \Omega) \subseteq (\xi, \Omega)\}$.

Lemma 2.2 [4, 35, 45] *Let (X, Γ, Ω) be a soft topological space and let $(\xi, \Omega), (\sigma, \Omega) \in S(X, \Omega)$. Then:*

- (i) (ξ, Ω) is soft closed if and only if $(\xi, \Omega) = Cl(\xi, \Omega)$
- (ii) If $(\xi, \Omega) \subseteq (\sigma, \Omega)$, then $Cl(\xi, \Omega) \subseteq Cl(\sigma, \Omega)$.
- (iii) (ξ, Ω) is soft open if and only if $(\xi, \Omega) = Int(\xi, \Omega)$.
- (iv) If $(\xi, \Omega) \subseteq (\sigma, \Omega)$, then $Int(\xi, \Omega) \subseteq Int(\sigma, \Omega)$.
- (v) $(Cl(\xi, \Omega))^C = Int((\xi, \Omega)^C)$.
- (vi) $(Int(\xi, \Omega))^C = Cl((\xi, \Omega)^C)$.

Definition 2.2 [13] *Let (X, Γ, Ω) be a soft topological space over X and Y be a nonempty subset of X . Then $\Gamma_Y = \{(\xi_Y, \Omega) : (\xi, \Omega) \in \Gamma\}$ is said to be the soft relative topology on Y and (Y, Γ_Y, Ω) is called a soft subspace of (X, Γ, Ω) .*

Lemma 2.3 [35] *Let (Y, Γ_Y, Ω) be a soft subspace of a soft topological space (X, Γ, Ω) and (ξ, Ω) be a soft set over X , then:*

- (i) (ξ, Ω) is soft open in Y if and only if $(\xi, \Omega) = \tilde{Y} \cap (\sigma, \Omega)$ for some soft open set (σ, Ω) in X .
- (ii) (ξ, Ω) is soft closed in Y if and only if $(\xi, \Omega) = \tilde{Y} \cap (\sigma, \Omega)$ for some soft closed set (σ, Ω) in X .

Definition 2.3 [43] A soft set (ξ, Ω) in a soft topological space (X, Γ, Ω) is said to be:

- (i) Soft regular open set if $(\xi, \Omega) = \text{Int}(\text{Cl}(\xi, \Omega))$.
- (ii) Soft regular closed set if $(\xi, \Omega) = \text{Cl}(\text{Int}(\xi, \Omega))$.

Remark 2.1 [43] Every soft regular open (resp. soft regular closed) set is soft open (resp. soft closed). But the converse may not be true.

Definition 2.4 [15] Let $S(X, \Omega)$ and $S(Y, \Psi)$ be families of soft sets over X and Y . Let $u : X \rightarrow Y$ and $p : \Omega \rightarrow \Psi$ be mappings. Then a mapping $f_{pu} : S(X, \Omega) \rightarrow S(Y, \Psi)$ is defined as:

- (i) Let (ξ, Ω) be a soft set in $S(X, \Omega)$. The image of (ξ, Ω) under f_{pu} is written as $f_{pu}(\xi, \Omega) = (f_{pu}(\xi), p(\Omega))$, is a soft set in $S(Y, \Psi)$ such that

$$(f_{pu}(\xi))(k) = \begin{cases} \cup_{e \in p^{-1}(k) \cap \Omega} u(\xi(e)), & p^{-1}(k) \cap \Omega \neq \phi \\ \phi, & p^{-1}(k) \cap \Omega = \phi, \end{cases}$$

for all $k \in \Psi$.

- (ii) Let (μ, Ψ) be a soft set in $S(Y, \Psi)$. The inverse image of (μ, Ψ) under f_{pu} is given by

$$f_{pu}^{-1}(\mu)(e) = \begin{cases} u^{-1}\mu(p(e)), & p(e) \in \Psi \\ \phi, & \text{otherwise} \end{cases}$$

for all $e \in \Omega$.

Definition 2.5 [22] Let $f_{pu} : S(X, \Omega) \rightarrow S(Y, \Psi)$ be a mapping and $u : X \rightarrow Y$ and $p : \Omega \rightarrow \Psi$ be mappings. Then f_{pu} is soft injective (respectively, surjective, bijective) if $u : X \rightarrow Y$ and $p : \Omega \rightarrow \Psi$ are injective (respectively, surjective, bijective).

Definition 2.6 [9, 45, 46] Let (X, Γ, Ω) and (Y, η, Ψ) be soft topological spaces. A soft mapping: $f_{pu} : (X, \Gamma, \Omega) \rightarrow (Y, \eta, \Psi)$ is called:

- (i) Soft continuous if $f_{pu}^{-1}(\mu, \Psi)$ is soft open in X for all soft open sets (μ, Ψ) in Y .
- (ii) Soft open if $f_{pu}(\xi, \Omega)$ is soft open in Y , for all soft open sets (ξ, Ω) in X .
- (iii) Soft homeomorphism if f_{pu} is bijection, soft continuous and soft open.

Definition 2.7 [45] The soft set $(\xi, \Omega) \in S(X, \Omega)$ is called a soft point denoted by x_β , if there exists $x \in X$ and $\beta \in \Omega$ such that $\xi(\beta) = \{x\}$ and $\xi(\beta^c) = \phi$ for each $\beta^c \in \Omega - \{\beta\}$. The family of all soft points over X is denoted by $\text{SP}(X, \Omega)$.

Definition 2.8 [45] The soft point x_β is said to be in the soft set (σ, Ω) , given by $x_\beta \in (\sigma, \Omega)$ if $x_\beta \subseteq (\sigma, \Omega)$.

Lemma 2.4 [9] Let $(\xi, \Omega), (\sigma, \Omega) \in S(X, \Omega)$ and $x_\beta \in SP(X, \Omega)$. Then we have:

- (a) $x_\beta \in (\xi, \Omega) \Leftrightarrow x_\beta \notin (\xi, \Omega)^c$.
- (b) $x_\beta \in (\xi, \Omega) \cup (\sigma, \Omega) \Leftrightarrow x_\beta \in (\xi, \Omega)$ or $x_\beta \in (\sigma, \Omega)$.
- (c) $x_\beta \in (\xi, \Omega) \cap (\sigma, \Omega) \Leftrightarrow x_\beta \in (\xi, \Omega)$ and $x_\beta \in (\sigma, \Omega)$.
- (d) $(\xi, \Omega) \subseteq (\sigma, \Omega) \Leftrightarrow x_\beta \in (\xi, \Omega) \Rightarrow x_\beta \in (\sigma, \Omega)$.

Definition 2.9 [43] A soft set (ξ, Ω) of a STS (X, Γ, Ω) is called soft regular open if $(\xi, \Omega) = \text{Int}(\text{Cl}(\xi, \Omega))$.

Definition 2.10 [43] The complement of a soft regular open set is called soft regular closed set. The family of all soft regular closed sets is denoted by $\text{SRC}(X, \Omega)$.

Remark 2.2 [43] Every soft regular open (resp. soft regular closed) set is soft open (resp. soft closed) but the converse may not be true.

Definition 2.11 [5] A soft subset (ξ, Ω) of a STS (X, Γ, Ω) is called soft semi open if $(\xi, \Omega) \subseteq \text{Cl}(\text{Int}(\xi, \Omega))$. The family of all soft semi open sets over (X, Ω) will be denoted by $\text{SSO}(X, \Omega)$.

Definition 2.12 [5] The complement of a soft semi open set is called soft semi closed. The family of all soft semi closed sets over (X, Ω) will be denoted by $\text{SSC}(X, \Omega)$.

Remark 2.3 [5] Every soft open (resp. soft closed) set is soft semi open (resp. soft semi closed) set but converse may not be true.

Remark 2.4 [5] Any union (resp. intersection) of soft semi open (resp. soft semi closed) set is soft semi open (resp. soft semi closed). But intersection (resp. union) of two soft semi open (resp. soft semi closed) sets may not be soft semi open (resp. soft semi closed). However, the intersection (union) of a soft open (resp. soft closed) sets and soft semi open (resp. soft semi closed) set is soft semi open (resp. soft semi closed).

Definition 2.13 [5] The intersection of all the soft semi closed sets containing a soft set (ξ, Ω) in a STS (X, Γ, Ω) is called semi closure of (ξ, Ω) and it is denoted by $s\text{Cl}(\xi, \Omega)$.

Definition 2.14 [1] A soft set (ξ, Ω) of a STS (X, Γ, Ω) is called soft α -open if $(\xi, \Omega) \subseteq \text{Int}(\text{Cl}(\text{Int}(\xi, \Omega)))$. The family of all soft α -open sets over X will be denoted by $S\alpha\text{O}(X, \Omega)$.

Definition 2.15 [1] The complement of a $S\alpha\text{O}(X, \Omega)$ set is called soft α -closed.

Remark 2.5 [1] Every soft open (resp. soft closed) set is $S\alpha\text{O}(X, \Omega)$ (resp. soft α -closed) but the converse may not be true.

Definition 2.16 [8] A STS (X, Γ, Ω) is said to be soft regular if $\forall (\xi, \Omega) \in \text{SC}(X, \Omega)$ and each soft point x_β such that $x_\beta \notin (\xi, \Omega)$, \exists soft open sets $(\mu, \Omega), (\lambda, \Omega) \in \Gamma$ of X such that $x_\beta \in (\mu, \Omega)$, $(\xi, \Omega) \subseteq (\lambda, \Omega)$ and $(\mu, \Omega) \cap (\lambda, \Omega) = \phi$.

Definition 2.17 [14] Let \mathcal{J} be a nonempty collection of soft sets on X with set of parameters Ω . Then a family of soft sets \mathcal{J} over (X, Ω) is said to be a soft ideal on X if,

- (a) $(\xi, \Omega) \in \mathcal{J}$ and $(\sigma, \Omega) \in \mathcal{J} \Rightarrow (\xi, \Omega) \cup (\sigma, \Omega) \in \mathcal{J}$.
- (b) $(\xi, \Omega) \in \mathcal{J}$ and $(\sigma, \Omega) \subseteq (\xi, \Omega) \Rightarrow (\sigma, \Omega) \in \mathcal{J}$.

A soft topological space (X, Γ, Ω) with a soft ideal \mathcal{J} is called soft ideal topological space and is denoted by $\text{SITS}(X, \Gamma, \Omega, \mathcal{J})$.

Definition 2.18 [5] A $\text{SITS}(X, \Gamma, \Omega, \mathcal{J})$ is said to be a soft \mathcal{J} -regular if for each soft set $(\xi, \Omega) \in \text{SRC}(X, \Omega)$ and a soft point $x_\beta \notin (\xi, \Omega)$, \exists soft sets $(\mu, \Omega), (\lambda, \Omega) \in \Gamma$ such that $x_\beta \in (\mu, \Omega), (\sigma, \Omega) - (\lambda, \Omega) \in \mathcal{J}$ and $(\mu, \Omega) \cap (\lambda, \Omega) = \phi$.

3 Soft s-Regular Spaces

Definition 3.1 A STS (X, Γ, Ω) is soft s-regular if \forall soft set $(\lambda, \Omega) \in \text{SC}(X, \Omega)$ and a soft point x_β such that $x_\beta \notin (\lambda, \Omega)$, $\exists (\mu, \Omega), (\nu, \Omega) \in \text{SSO}(X, \Omega)$ such that $x_\beta \in (\mu, \Omega)$, $(\lambda, \Omega) \subseteq (\nu, \Omega)$ and $(\mu, \Omega) \cap (\nu, \Omega) = \phi$.

Theorem 3.1 Every soft regular space is soft s-regular.

Proof It follows from the Definition 3.1 and Definition 2.16 and Remark 2.3.

Remark 3.1 The converse of Theorem 3.1 is not true in general. For,

Example 3.1 Let $X = \{x_1, x_2, x_3\}$, $\Omega = \{\beta_1, \beta_2\}$ and the soft sets (λ, Ω) , (μ, Ω) and (ν, Ω) be defined as follows:

$$\begin{aligned} (\lambda, \Omega) &= \{(\beta_1, \{x_1\}), (\beta_2, \{x_2\})\}, \\ (\mu, \Omega) &= \{(\beta_1, \{x_2\}), (\beta_2, \{x_1\})\}, \\ (\nu, \Omega) &= \{(\beta_1, \{x_1, x_2\}), (\beta_2, \{x_1, x_2\})\}. \end{aligned}$$

Then $\text{STS}(X, \Gamma, \Omega)$, where $\Gamma = \{\phi, X, (\lambda, \Omega), (\mu, \Omega), (\nu, \Omega)\}$ is soft s-regular but not soft regular.

Definition 3.2 A STS (X, Γ, Ω) is soft almost regular if \forall soft set $(\lambda, \Omega) \in \text{SRC}(X, \Omega)$ and \forall soft point x_β such that $x_\beta \notin (\lambda, \Omega)$, \exists soft sets $(\mu, \Omega), (\nu, \Omega) \in \Gamma$ such that $x_\beta \in (\mu, \Omega)$, $(\lambda, \Omega) \subseteq (\nu, \Omega)$ and $(\mu, \Omega) \cap (\nu, \Omega) = \phi$.

Remark 3.2 The axioms soft s-regularity and soft almost regularity are independent to each other. The STS (X, Γ, Ω) of Example 3.1 is soft s-regular but not soft almost regular and the soft topological space of the following example is soft almost regular but not soft s-regular.

Example 3.2 Let $X = \{x_1, x_2, x_3\}$, $\Omega = \{\beta_1, \beta_2\}$ and the soft sets (λ, Ω) , (μ, Ω) and (ν, Ω) are defined as follows:

$$(\lambda, \Omega) = \{(\beta_1, \{x_1\}), (\beta_2, \{x_1\})\},$$

$$(\mu, \Omega) = \{(\beta_1, \{x_3\}), (\beta_2, \{x_3\})\},$$

$$(\nu, \Omega) = \{(\beta_1, \{x_1, x_2\}), (\beta_2, \{x_1, x_2\})\},$$

$$(\delta, \Omega) = \{(\beta_1, \{x_1, x_3\}), (\beta_2, \{x_1, x_3\})\}.$$

Then $STS(X, \Gamma, \Omega)$, where $\Gamma = \{\phi, X, (\lambda, \Omega), (\mu, \Omega), (\nu, \Omega), (\delta, \Omega)\}$ is soft almost regular but not soft s-regular.

Definition 3.2 A $STS(X, \Gamma, \Omega)$ is said to be a soft T_0 -space, if \forall soft point $x_\beta \neq y_\beta$ of X , \exists soft open sets $(\xi, \Omega), (\sigma, \Omega) \in \Gamma$ such that $x_\beta \in (\xi, \Omega)$ and $y_\beta \notin (\xi, \Omega)$ or $x_\beta \notin (\sigma, \Omega)$ and $y_\beta \in (\sigma, \Omega)$.

Definition 3.3 A $STS(X, \Gamma, \Omega)$ is said to be a soft semi- T_2 , if \forall soft point $x_\beta \neq y_\beta$ of X , \exists soft sets $(\xi, \Omega), (\sigma, \Omega) \in SSO(X, \Omega)$ such that $x_\beta \in (\xi, \Omega)$, $y_\beta \in (\sigma, \Omega)$ and $(\xi, \Omega) \cap (\sigma, \Omega) = \phi$.

Theorem 3.2 If a $STS(X, \Gamma, \Omega)$ is soft s-regular and soft- T_0 space, then it is soft semi- T_2 .

Proof Suppose (X, Γ, Ω) be a soft s-regular and let x_β, y_β be two soft points of X such that $x_\beta \neq y_\beta$. Without loss of generality, let $(\xi, \Omega) \in \Gamma$, which contains x_β but not y_β . Then $(\xi, \Omega)^c \in SC(X, \Omega)$ such that $y_\beta \in (\xi, \Omega)^c$ and $x_\beta \notin (\xi, \Omega)^c$. Since X is soft s-regular, \exists soft sets $(\mu, \Omega), (\lambda, \Omega) \in SSO(X, \Omega)$ such that $x_\beta \in (\mu, \Omega)$ and $(\xi, \Omega)^c \subseteq (\lambda, \Omega)$ and $(\mu, \Omega) \cap (\lambda, \Omega) = \phi$. Consequently, $(\mu, \Omega), (\lambda, \Omega) \in SSO(X, \Omega)$ such that $x_\beta \in (\mu, \Omega)$, $y_\beta \in (\lambda, \Omega)$ and $(\mu, \Omega) \cap (\lambda, \Omega) = \phi$. Hence, (X, Γ, Ω) is soft semi T_2 .

The following theorem gives several characterizations for soft s-regular spaces.

Theorem 3.3 The following statements are equivalent for a $STS(X, \Gamma, \Omega)$:

- (X, Γ, Ω) is soft s-regular.
- For each soft point x_β and each $(\lambda, \Omega) \in \Gamma$ containing x_β , $\exists (\mu, \Omega) \in SSO(X, \Omega)$ such that $x_\beta \in (\mu, \Omega) \subseteq sCl(\mu, \Omega) \subseteq (\lambda, \Omega)$.
- $(\xi, \Omega) = \bigcap \{(v, \Omega) : (v, \Omega) \in \Gamma \cap SRC(X, \Omega) \text{ and } (\xi, \Omega) \subseteq (v, \Omega)\}$ for every $(\xi, \Omega) \in SRC(X, \Omega)$
- For each soft set (v, Ω) and $(\psi, \Omega) \in \Gamma$ such that $(v, \Omega) \cap (\psi, \Omega) \neq \phi$, $\exists (\rho, \Omega) \in SSO(X, \Omega)$ such that $(v, \Omega) \cap (\rho, \Omega) \neq \phi$ and $sCl(\rho, \Omega) \subseteq (\psi, \Omega)$.
- For every soft set $(v, \Omega) \neq \phi$ and $(\psi, \Omega) \in SC(X, \Omega)$ such that $(v, \Omega) \cap (\psi, \Omega) = \phi$, $\exists (\chi, \Omega), (\delta, \Omega) \in SSO(X, \Omega)$ such that $(\chi, \Omega) \cap (v, \Omega) \neq \phi$ and $(\psi, \Omega) \subseteq (\delta, \Omega)$.

Proof (a) \Rightarrow (b) Let $(\lambda, \Omega) \in \Gamma$ such that $x_\beta \in (\lambda, \Omega)$. Then $(\lambda, \Omega)^c \in SC(X, \Omega)$ and $x_\beta \notin (\lambda, \Omega)^c$. Since X is soft s-regular, $\exists (\chi, \Omega), (\mu, \Omega) \in SSO(X, \Omega)$ such that $x_\beta \in (\mu, \Omega)$, $(\lambda, \Omega)^c \subseteq (\chi, \Omega)$ and $(\chi, \Omega) \cap (\mu, \Omega) = \phi$. Now, $sCl(\mu, \Omega) \subseteq sCl(\chi, \Omega)^c = (\chi, \Omega)^c$ for $(\mu, \Omega) \subseteq (\chi, \Omega)^c$ and $(\chi, \Omega)^c \in SSO(X, \Omega)$. Hence, $sCl(\mu, \Omega) \subseteq (\lambda, \Omega)$.

(b)⇒(c) Suppose $(v, \Omega) \in SC(X, \Omega)$ and $x_\beta \notin (v, \Omega)$. Then $(v, \Omega)^c \in \Gamma$ and contains x_β . By (b), there is $(\mu, \Omega) \in SSO(X, \Omega)$ such that $x_\beta \in (\mu, \Omega) \subseteq sCl(\mu, \Omega) \subseteq (v, \Omega)^c$. And so, $(\mu, \Omega)^c \supseteq sCl(\mu, \Omega)^c \supseteq (v, \Omega)$. Consequently, $(\mu, \Omega)^c$ is a soft semi closed semi nbd of (v, Ω) to which x_β does not belong. Hence (c) holds.

(c)⇒(d) Suppose $(v, \Omega) \cap (\psi, \Omega) \neq \phi$ where $(\psi, \Omega) \in \Gamma$. Let $x_\beta \in (v, \Omega) \cap (\psi, \Omega)$. Since x_β does not belong to the soft closed set $(\psi, \Omega)^c$, there exists a soft semi closed soft semi nbd of $(\psi, \Omega)^c$ say, (λ, Ω) such that $x_\beta \notin (\lambda, \Omega)$. Let $(\mu, \Omega) \in SSO(X, \Omega)$ for which $(\psi, \Omega)^c \subseteq (\mu, \Omega) \subseteq (\lambda, \Omega)$. Then $(\rho, \Omega) = (\lambda, \Omega)^c \in SSO(X, \Omega)$ which contains x_β and so $(\rho, \Omega) \cap (v, \Omega) \neq \phi$. Also, $(\mu, \Omega)^c \in SSC(X, \Omega)$, $sCl(\rho, \Omega) = sCl(\lambda, \Omega)^c \subseteq (\mu, \Omega)^c \subseteq (\psi, \Omega)$.

(d)⇒(e) If $(v, \Omega) \cap (\psi, \Omega) = \phi$, where $(v, \Omega) \neq \phi$ and $(\psi, \Omega) \in SC(X, \Omega)$. Then $(v, \Omega) \cap (\psi, \Omega)^c \neq \phi$ and $(\psi, \Omega)^c \in \Gamma$. Therefore by (d), $\exists (\chi, \Omega) \in SSO(X, \Omega)$ such that $(v, \Omega) \cap (\chi, \Omega) \neq \phi$, $(\chi, \Omega) \subseteq sCl(\chi, \Omega) \subseteq (\psi, \Omega)^c$. Put, $(\delta, \Omega) = sCl(\chi, \Omega)^c$. Then $(\psi, \Omega) \subseteq (\delta, \Omega)$ and $(\chi, \Omega), (\delta, \Omega) \in SSO(X, \Omega)$ such that $(\delta, \Omega) = sCl(\chi, \Omega)^c \subseteq (\chi, \Omega)^c$.

(e)⇒(a) Let x_β be a soft point and $(\xi, \Omega) \in SC(X, \Omega)$ such that $x_\beta \notin (\xi, \Omega)$. Now, $x_\beta \cap (\xi, \Omega) = \phi$ by hypothesis \exists disjoint $(\chi, \Omega), (\delta, \Omega) \in SSO(X, \Omega)$ such that $x_\beta \cap (\xi, \Omega) \neq \phi$ and $(\xi, \Omega) \subseteq (\delta, \Omega)$. Consequently, (X, Γ, Ω) is soft s-regular.

Theorem 3.4 *If (v, Ω) is SSO (X, Ω) and (μ, Ω) is $S\alpha O(X, \Omega)$ in a STS (X, Γ, Ω) , then $(\mu, \Omega) \cap (v, \Omega) \in SSO(X, \Omega)$.*

Theorem 3.5 *If $(v, \Omega) \in SSO(X, \Omega)$ and $(\mu, \Omega) \in \Gamma$ in a STS (X, Γ, Ω) , then $(\mu, \Omega) \cap (v, \Omega)$ is soft semi open in (X, Γ, Ω) .*

Proof It follows from Theorem 3.4. and Remark 2.5

Theorem 3.6 *If (Y, Γ_Y, Ω) is a soft subspace of a STS (X, Γ, Ω) and $(v, \Omega) \subseteq (Y, \Omega)$. If (v, Ω) is soft semi open in (X, Γ, Ω) then it is soft semi open in (Y, Γ_Y, Ω) .*

Proof Evident.

Theorem 3.7 *Let (X, Γ, Ω) be a STS and $(Y, \Omega) \in S\alpha O(X, \Omega)$. If $(v, \Omega) \in SSO(X, \Omega)$, then $(Y, \Omega) \cap (v, \Omega)$ is soft semi open in the soft subspace (Y, Γ_Y, Ω) .*

Proof It follows from Theorem 3.4 and Theorem 3.6.

Theorem 3.8 *Let (Y, Γ_Y, Ω) be a soft subspace of a STS (X, Γ, Ω) . Then a soft set $(v, \Omega) \in SSO(Y, \Omega) \Rightarrow (v, \Omega) \in SSO(X, \Omega)$ if and only if $(Y, \Omega) \in SSO(X, \Omega)$.*

Proof Necessity: Obvious.

Sufficiency: If $(v, \Omega) \in SSO(Y, \Omega)$ and $(Y, \Omega) \in SSO(X, \Omega)$. Then \exists a soft set $(\rho, \Omega)_{\Gamma_Y} \in \Gamma_Y$ in Y such that $(\rho, \Omega)_{\Gamma_Y} \subseteq (v, \Omega) \subseteq (Cl_Y(\rho, \Omega))_{\Gamma_Y}$. Let $(\rho, \Omega) \in \Gamma$ such that $(\rho, \Omega)_{\Gamma_Y} = (\rho, \Omega) \cap (Y, \Omega)$. Consequently, $(\rho, \Omega) \cap (Y, \Omega) \subseteq (v, \Omega) \subseteq (Cl_Y(\rho, \Omega))_{\Gamma_Y} = Cl((\rho, \Omega) \cap (Y, \Omega)) \cap (Y, \Omega) \subseteq Cl((\rho, \Omega) \cap Y)$. By Theorem 3.5, we observe that $(\rho, \Omega) \cap (Y, \Omega)$ is soft semi open for by hypothesis $(Y, \Omega) \in SSO(X, \Omega)$. Hence, by Theorem 3.6, we obtained that $(v, \Omega) \in SSO(X, \Omega)$.

Theorem 3.9 *If (Y, Γ_Y, Ω) is a soft α -open subspace of a STS (X, Γ, Ω) and $(\psi, \Omega) \subseteq (Y, \Omega)$, then $sCl(\psi, \Omega) \cap (Y, \Omega) = sCl_Y(\psi, \Omega)$.*

Proof Let $y_\beta \in sCl_Y(\psi, \Omega)$. Then $y_\beta \in (Y, \Omega)$. Let $(\lambda, \Omega) \in SSO(X, \Omega)$ such that $y_\beta \subseteq (\lambda, \Omega)$. Then (Y, Ω) being soft α -open in (X, Γ, Ω) , $(\lambda, \Omega) \cap (Y, \Omega)$ is soft semi open in (Y, Γ_Y, Ω) (Theorem 3.7) such that $y_\beta \in (\lambda, \Omega) \cap (Y, \Omega)$. And so, $((\lambda, \Omega) \cap (Y, \Omega)) \cap (\psi, \Omega) \neq \phi$. Consequently, $(\lambda, \Omega) \cap (\psi, \Omega) \neq \phi$. Therefore, $y_\beta \in sCl(\psi, \Omega)$ and hence, $y_\beta \in sCl(\psi, \Omega) \cap (Y, \Omega)$. Now let $y_\beta \in sCl(\psi, \Omega) \cap (Y, \Omega)$ and (λ, Ω) being soft semi open in (Y, Ω) such that $y_\beta \in (\lambda, \Omega)$. Now $(Y, \Omega) \in S\alpha O(X, \Omega)$ implies $(Y, \Omega) \in SSO(X, \Omega)$, by Theorem 3.8, $(\lambda, \Omega) \in SSO(X, \Omega)$. And so, $(\lambda, \Omega) \cap (\psi, \Omega) \neq \phi$, since $y_\beta \in sCl(\psi, \Omega)$. Consequently, $y_\beta \in sCl_Y(\psi, \Omega)$.

Theorem 3.10 *If $(Y, \Omega) \in S\alpha O(X, \Omega)$ in a STS (X, Γ, Ω) . If STS (X, Γ, Ω) is soft s-regular, then so is soft subspace (Y, Γ_Y, Ω) .*

Proof Let (X, Γ, Ω) be soft s-regular and $(Y, \Omega) \in S\alpha O(X, \Omega)$. Let $(\nu, \Omega)_Y \in \Gamma_Y$ and $x_\beta \in (\nu, \Omega)_Y$. Then, \exists a soft set $(\nu, \Omega) \in \Gamma$ such that $(\nu, \Omega)_Y = (\nu, \Omega) \cap (Y, \Omega)$. Now $x_\beta \in (\nu, \Omega)$ and (X, Γ, Ω) be soft s-regular, \exists a soft set $(\mu, \Omega) \in SSO(X, \Omega)$ such that $x_\beta \in (\mu, \Omega) \subseteq sCl(\mu, \Omega) \subseteq (\nu, \Omega)$. And so, $x_\beta \in (\mu, \Omega) \cap (Y, \Omega) \subseteq sCl(\mu, \Omega) \cap (Y, \Omega) \subseteq (\nu, \Omega) \cap (Y, \Omega)$. Put $(\mu, \Omega)_Y = (\mu, \Omega) \cap (Y, \Omega)$. Now, $(Y, \Omega) \in S\alpha O(X, \Omega)$, $(\mu, \Omega)_Y \in SSO(Y, \Omega)$ such that $x_\beta \in (\mu, \Omega)_Y \subseteq sCl_Y(\mu, \Omega)_Y \subseteq (\nu, \Omega)_Y$ by Theorem 3.7 and Theorem 3.8. Hence (Y, Γ_Y, Ω) is soft s-regular.

Corollary 3.1 Let $(Y, \Omega) \in \Gamma$ in a STS (X, Γ, Ω) . If (X, Γ, Ω) is soft s-regular then so it the soft subspace (Y, Γ_Y, Ω) .

Theorem 3.11 *Let $f_{pu} : (X, \Gamma, \Omega) \rightarrow (Y, \eta, \Psi)$ be a soft homeomorphism. If $(\xi, \Omega) \in SSO(X, \Omega)$, then $f_{pu}(\xi, \Omega) \in SSO(Y, \Psi)$.*

Proof Evident.

Theorem 3.12 *Let $f_{pu} : (X, \Gamma, \Omega) \rightarrow (Y, \eta, \Psi)$ be a soft homeomorphism. If (X, Γ, Ω) is soft s-regular, then so is (Y, η, Ψ) .*

Proof Let y_β be a soft point of Y such that $y_\beta \in (\Upsilon, \Psi)$ where $(\Upsilon, \Psi) \in \eta$. Since f_{pu} is soft open and soft continuous $f_{pu}^{-1}(y_\beta) \in f_{pu}^{-1}(\Upsilon, \Psi)$ and $f_{pu}^{-1}(\Upsilon, \Psi) \in \Gamma$. Since, (X, Γ, Ω) is soft s-regular by Theorem 5.1.2(b), \exists a soft set $(\rho, \Omega) \in SSO(X, \Omega)$ such that $f_{pu}^{-1}(y_\beta) \subseteq (\rho, \Omega) \subseteq sCl(\rho, \Omega) \subseteq f_{pu}^{-1}(\Upsilon, \Psi)$. Therefore, $y_\beta \in f_{pu}(\rho, \Omega) \subseteq f_{pu}(sCl(\rho, \Omega)) \subseteq (\Upsilon, \Psi)$. Since f_{pu} is soft open and soft continuous, $f_{pu}(\rho, \Omega) \in SSO(Y, \Psi)$ is soft semi open in Y and $f_{pu}(sCl(\rho, \Omega)) \subseteq sCl(f_{pu}(\rho, \Omega))$. Hence, (Y, η, Ψ) is soft s-regular.

Now we introduced and explores the study of soft s- \mathcal{J} -regularity and present their studies in soft ideal topological spaces.

Definition 3.2 A SITS $(X, \Gamma, \Omega, \mathcal{J})$ is soft s- \mathcal{J} -regular, if \forall soft set $(\lambda, \Omega) \in SC(X, \Omega)$ and a soft point x_β such that $x_\beta \notin (\lambda, \Omega)$, $\exists (\mu, \Omega), (\nu, \Omega) \in SSO(X, \Omega)$ such that $x_\beta \in (\mu, \Omega)$, $(\lambda, \Omega) - (\nu, \Omega) \in \mathcal{J}$ and $(\mu, \Omega) \cap (\nu, \Omega) = \phi$.

Remark 3.3 Soft \mathcal{J} -regularity implies soft s- \mathcal{J} -regularity and independent to soft almost \mathcal{J} -regularity. However, a soft s- \mathcal{J} -regular may fails to soft \mathcal{J} -regular.

Example 3.3 Let $X = \{x_1, x_2, x_3\}$ $\Omega = \{\beta_1, \beta_2\}$ and the soft sets (λ, Ω) , (μ, Ω) and (ν, Ω) are defined as follows:

$$\begin{aligned}(\lambda, \Omega) &= \{(\beta_1, \{x_1\}), (\beta_2, \{x_1\})\}, \\(\mu, \Omega) &= \{(\beta_1, \{x_1, x_3\}), (\beta_2, \{x_2, x_3\})\}, \\(\nu, \Omega) &= \{(\beta_1, \{x_1, x_2\}), (\beta_2, \{x_1, x_2\})\}.\end{aligned}$$

Let $\Gamma = \{\phi, X, (\lambda, \Omega), (\mu, \Omega), (\nu, \Omega)\}$ be a soft topology over (X, Ω) and $\mathcal{J} = \{\phi\}$ be a soft ideal over (X, Ω) . Then SITS $(X, \Gamma, \Omega, \mathcal{J})$ is soft s- \mathcal{J} -regular but not soft \mathcal{J} -regular.

The following theorem gives several characterizations of soft s- \mathcal{J} -regular spaces.

Theorem 3.13 *The following statements are equivalent for a SITS $(X, \Gamma, \Omega, \mathcal{J})$.*

- $(X, \Gamma, \Omega, \mathcal{J})$ is soft s- \mathcal{J} -regular.
- For every soft point x_β and for every $(\lambda, \Omega) \in \Gamma$ such that $x_\beta - (\lambda, \Omega) \in \mathcal{J}$, $\exists(\mu, \Omega) \in \text{SSO}(X, \Omega)$ such that $x_\beta - (\mu, \Omega) \in \mathcal{J}$ and $\text{sCl}(\mu, \Omega) - (\lambda, \Omega) \in \mathcal{J}$.
- $(\xi, \Omega) = \bigcap \{(\nu, \Omega) : (\nu, \Omega) \in \Gamma \cap \text{SRC}(X, \Omega) \text{ and } (\xi, \Omega) - (\nu, \Omega) \in \mathcal{J}\}$ for all $(\xi, \Omega) \in \text{SRC}(X, \Omega)$.
- For each soft set (ν, Ω) and $(\psi, \Omega) \in \Gamma$ such that $(\nu, \Omega) \cap (\psi, \Omega) \neq \phi$, $\exists(\rho, \Omega) \in \text{SSO}(X, \Omega)$ such that $(\nu, \Omega) \cap (\rho, \Omega) \neq \phi$ and $\text{sCl}(\rho, \Omega) \subseteq (\psi, \Omega)$.
- For every soft set $(\nu, \Omega) \neq \phi$ and $(\psi, \Omega) \in \text{SC}(X, \Omega)$ such that $(\nu, \Omega) \cap (\psi, \Omega) = \phi$, \exists disjoint $(\chi, \Omega), (\delta, \Omega) \in \text{SSO}(X, \Omega)$ such that $(\chi, \Omega) \cap (\nu, \Omega) \neq \phi$ and $(\psi, \Omega) - (\delta, \Omega) \in \mathcal{J}$.

Proof **(a)** \Rightarrow **(b)** Let $(\lambda, \Omega) \in \Gamma$ such that $x_\beta - (\lambda, \Omega) \in \mathcal{J}$. Then $(\lambda, \Omega)^c \in \text{SC}(X, \Omega)$ and $x_\beta \notin (\lambda, \Omega)^c$. Since $(X, \Gamma, \Omega, \mathcal{J})$ is soft s- \mathcal{J} -regular, $\exists(\chi, \Omega), (\mu, \Omega) \in \text{SSO}(X, \Omega)$ such that $(\lambda, \Omega)^c - (\chi, \Omega) \in \mathcal{J}$, $x_\beta - (\mu, \Omega) \in \mathcal{J}$ and $(\chi, \Omega) \cap (\mu, \Omega) = \phi$. Now, $\text{sCl}(\mu, \Omega) - \text{sCl}(\chi, \Omega)^c = (\chi, \Omega)^c \in \mathcal{J}$ for $(\mu, \Omega) - (\chi, \Omega)^c \in \mathcal{J}$ and $(\chi, \Omega)^c \in \text{SSC}(X, \Omega)$. Hence, $\text{sCl}(\mu, \Omega) - (\lambda, \Omega) \in \mathcal{J}$.

(b) \Rightarrow **(c)** Let $(\nu, \Omega) \in \text{SC}(X, \Omega)$ and $x_\beta \notin (\nu, \Omega)$. Then $(\nu, \Omega)^c \in \Gamma$ and contains x_β . By (b) $\exists(\mu, \Omega) \in \text{SSO}(X, \Omega)$ such that $x_\beta \in (\mu, \Omega)$ and $\text{sCl}(\mu, \Omega) - (\nu, \Omega)^c \in \mathcal{J}$. It follows that $\text{sCl}(\mu, \Omega)^c - (\mu, \Omega)^c \in \mathcal{J}$ and $(\nu, \Omega) - \text{sCl}(\mu, \Omega)^c \in \mathcal{J}$. Consequently, $(\mu, \Omega)^c$ is a soft semi closed soft semi neighborhood of (ν, Ω) to which x_β does not belong. Hence (c) holds.

(c) \Rightarrow **(d)** Let $(\nu, \Omega) \cap (\psi, \Omega) \neq \phi$ and $(\psi, \Omega) \in \Gamma$. Let $x_\beta \in (\nu, \Omega) \cap (\psi, \Omega)$. Since x_β does not belong to the soft closed set $(\psi, \Omega)^c$, \exists a soft semi closed semi soft nbd of $(\psi, \Omega)^c$ say, (λ, Ω) such that $x_\beta \notin (\lambda, \Omega)$. Let $(\psi, \Omega)^c - (\mu, \Omega) \in \mathcal{J}$, $(\mu, \Omega) - (\lambda, \Omega) \in \mathcal{J}$, where $(\mu, \Omega) \in \text{SSO}(X, \Omega)$. Then $(\rho, \Omega) = (\lambda, \Omega)^c \in \text{SSO}(X, \Omega)$ contains x_β and so $(\rho, \Omega) \cap (\nu, \Omega) \neq \phi$. Also, $(\mu, \Omega)^c \in \text{SSC}(X, \Omega)$, $\text{sCl}(\rho, \Omega) = \text{sCl}(\lambda, \Omega)^c - (\mu, \Omega)^c \in \mathcal{J}$ and $(\mu, \Omega)^c - (\psi, \Omega) \in \mathcal{J}$.

(d)⇒(e) If $(\nu, \Omega) \cap (\psi, \Omega) = \phi$, where $(\nu, \Omega) \neq \phi$ and $(\psi, \Omega) \in \text{SSC}(X, \Omega)$ then $(\nu, \Omega) \cap (\psi, \Omega)^c \neq \phi$ and $(\psi, \Omega)^c \in \Gamma$. Therefore by (d), $\exists (\chi, \Omega) \in \text{SSO}(X, \Omega)$ such that $(\nu, \Omega) \cap (\chi, \Omega) \neq \phi$, $(\chi, \Omega) - \text{sCl}(\chi, \Omega) \in \mathcal{J}$ and $\text{sCl}(\chi, \Omega) - (\psi, \Omega)^c \in \mathcal{J}$. Put $(\delta, \Omega) = \text{sCl}(\chi, \Omega)^c$. Then $(\psi, \Omega) - (\delta, \Omega) \in \mathcal{J}$ and $(\chi, \Omega), (\delta, \Omega) \in \text{SSO}(X, \Omega)$, such that $(\delta, \Omega) = \text{sCl}(\chi, \Omega)^c - (\chi, \Omega)^c \in \mathcal{J}$.

(e)⇒(a) Let x_β be a soft point of X and $(\xi, \Omega) \in \text{SC}(X, \Omega)$ be such that $x_\beta \notin (\xi, \Omega)$. Now, $x_\beta \cap (\xi, \Omega) = \phi$ by hypothesis \exists disjoint $(\chi, \Omega), (\delta, \Omega) \in \text{SSO}(X, \Omega), x_\beta \cap (\xi, \Omega) \neq \phi$ and $(\xi, \Omega) - (\delta, \Omega) \in \mathcal{J}$. Consequently, $(X, \Gamma, \Omega, \mathcal{J})$ is soft $s\text{-}\mathcal{J}$ -regular.

Theorem 3.14 *Let $(Y, \Omega) \in \Gamma$ be a soft open set in a SITS $(X, \Gamma, \Omega, \mathcal{J})$. If $(X, \Gamma, \Omega, \mathcal{J})$ is soft $s\text{-}\mathcal{J}$ -regular, then so is soft subspace $(Y, \Gamma_Y, \Omega, \mathcal{J})$.*

Proof Let SITS $(X, \Gamma, \Omega, \mathcal{J})$ be soft $s\text{-}\mathcal{J}$ -regular and $(Y, \Gamma_Y, \Omega, \mathcal{J}_Y)$ is a soft open subspace of a SITS $(X, \Gamma, \Omega, \mathcal{J})$. Let (ν, Ω) be soft open in (Y, Ω) and $x_\beta \in (\nu, \Omega)$, (Y, Ω) being soft open in $(X, \Gamma, \Omega, \mathcal{J})$, (ν, Ω) be soft open in $(X, \Gamma, \Omega, \mathcal{J})$. Since $(X, \Gamma, \Omega, \mathcal{J})$ be soft $s\text{-}\mathcal{J}$ -regular \exists a soft semi open set (μ, Ω) in $(X, \Gamma, \Omega, \mathcal{J})$ such that $x_\beta \in (\nu, \Omega)$ and $\text{sCl}(\mu, \Omega) - (\nu, \Omega) \in \mathcal{J}$. And so, $x_\beta \in (\mu, \Omega)$ and $(\text{sCl}(\mu, \Omega) \cap (Y, \Omega)) - (\nu, \Omega) \in \mathcal{J}$. Since $(\mu, \Omega) - (Y, \Omega) \in \mathcal{J}$ and $(\mu, \Omega) \in \text{SSO}(X, \Omega)$ implies $(\mu, \Omega) \in \text{SSO}(Y, \Omega)$. Now by the Theorem 3.13(b), it follows that $(Y, \Gamma_Y, \Omega, \mathcal{J}_Y)$ is soft $s\text{-}\mathcal{J}$ -regular,

Theorem 3.15 *Let $f_{\text{pu}}: (X, \Gamma, \Omega, \mathcal{J}) \rightarrow (Y, \eta, K, J)$ be a soft homeomorphism. If $(X, \Gamma, \Omega, \mathcal{J})$ is soft $s\text{-}\mathcal{J}$ -regular, then (Y, η, K, J) is soft $s\text{-}J$ -regular.*

Proof Let y_β be a soft point of Y such that $y_\beta \in (\Upsilon, \Psi)$ where $(\Upsilon, \Psi) \in \eta$. Since f_{pu} is soft open and soft continuous $f_{\text{pu}}^{-1}(y_\beta) \in f_{\text{pu}}^{-1}(\Upsilon, \Psi)$ and $f_{\text{pu}}^{-1}(\Upsilon, \Psi) \in \Gamma$. Since, $(X, \Gamma, \Omega, \mathcal{J})$ is soft $s\text{-}\mathcal{J}$ -regular by Theorem 3.13(b), \exists a soft set $(\rho, \Omega) \in \text{SSO}(X, \Omega)$ with $f_{\text{pu}}^{-1}(y_\beta) - (\rho, \Omega) \in \mathcal{J}$, $\text{sCl}(\rho, \Omega) - f_{\text{pu}}^{-1}(Y, \Psi) \in \mathcal{J}$. Therefore, $y_\beta \in f_{\text{pu}}(\rho, \Omega)$, $f_{\text{pu}}(\text{sCl}(\rho, \Omega)) - (Y, \Psi) \in \mathcal{J}$. Since f_{pu} is soft open and soft continuous, $f_{\text{pu}}(\rho, \Omega) \in \text{SSO}(Y, \Psi)$ is soft semi open in Y and $f_{\text{pu}}(\text{sCl}(\rho, \Omega)) - \text{sCl}(f_{\text{pu}}(\rho, \Omega)) \in J$. Hence, (Y, η, Ψ, J) is soft $s\text{-}J$ -regular.

4 Conclusion

In this paper, the axioms of soft s -regularity and soft $s\text{-}\mathcal{J}$ -regularity have been introduced. It is shown by theorems and examples that the axiom soft s -regularity (resp. soft $s\text{-}\mathcal{J}$ -regularity) is strictly weaker than the axiom of soft regularity (resp. soft \mathcal{J} -regularity) and independent to the axiom of soft almost regularity (resp. soft almost \mathcal{J} -regularity). Several theorems related to the characterizations, hereditary properties of soft s -regular and soft- \mathcal{J} -regular spaces have been established.

References

1. Akdag, M., Ozkan, A.: On soft -open and soft -continuous functions. *Abstr. Appl. Anal.* **2014**, 1–7 (2014)
2. Ali, M.I., Feng, F., Liu, X., Min, W.K., Shabir, M.: On some new operations in soft set theory. *Comput. Math. Appl.* **57**, 1547–1553 (2009)
3. Aygünöglü, A., Aygün, H.: Some notes on soft topological spaces. *Neural Comput. Appl.* **21**, 113–119 (2012)
4. Çağman, N., Karataş, S., Enginoglu, S.: Soft topology. *Comput. Math. Appl.* **62**, 351–358 (2011)
5. Chen, B.: Soft semi-open sets and related properties in soft topological spaces. *Appl. Math. Inf. Sci.* **7**(1), 287–294 (2013)
6. Das, S., Samanta, S.K.: Soft real sets, soft real numbers and their properties. *J. Fuzzy Math* **20**(3), 551–576 (2012)
7. Feng, F., Cho, J., Pedrycz, W., Fujita, H., Herawan, T.: Soft set based association rule mining. *Knowl.-Based Syst.* **111**, 268–282 (2016)
8. Georgiou, D.N., Megaritis, A.C., Petropoulos, V.I.: On soft topological spaces. *Appl. Math. Inf. Sci.* **7**(5), 1889–1901 (2013)
9. Guler, A.C., Kale, G.: Regularity and normality on soft ideal topological spaces. *Ann. Fuzzy Math. Inform* **9**(3), 373–383 (2015)
10. Hazra, H., Majumdar, P., Samanta, S.K.: Soft topology. *Fuzzy Inf. Eng.* **4**(1), 105–115 (2012)
11. Herawan, T., Deris, M.M.: A soft set approach for association rules mining. *Knowl.-Based Syst.* **24**(1), 186–195 (2011)
12. Hussain, S.: On soft regular-open (closed) sets in soft topological spaces. *J. Appl. Math. Inf.* **36**, 59–68 (2018)
13. Hussain, S., Ahmad, B.: Some properties of soft topological spaces. *Comput. Math. Appl.* **62**, 4058–4067 (2011)
14. Kandil, A., Tantawy, O.A.E., El-Sheikh, S.A., Abd El-latif, A.M.: Soft ideal theory soft local function and generated soft topological spaces. *Appl. Math. Inf. Sci.* **8**(4), 1595–1603 (2014)
15. Kharal, A., Ahmad, B.: Mappings on soft classes. *New Math. Nat. Comput.* **7**(3), 471–481 (2011)
16. Levine, N.: Semi open sets and semi continuity in topological spaces. *Amer. Math. Monthly* **70**(1), 36–41 (1963)
17. Lin, F.: Soft connected spaces and soft paracompact spaces. *Int. J. Math. Comput. Sci.* **7**(2), 277–283 (2013)
18. Lu, X., Liu, J.: The application of the paradigm of soft set logic formula in recommendation system. In *Journal of Physics: Conference Series*, Vol. 1903(1), p. 012028. IOP Publishing (2021)
19. Maheshwari, S.N., Prasad, R.: On s-regular spaces. *Glasnik Mat.* **10**(30), 37–40 (1975)
20. Maji, P.K., Roy, A.R., Biswas, R.: An application of soft sets in a decision making problem. *Comput. Math. Appl.* **44**(8–9), 1077–1083 (2002)
21. Maji, P.K., Biswas, R., Roy, A.R.: Soft set theory. *Comput. Math. Appl.* **45**, 555–562 (2003)
22. Min, W.K.: A note on soft topological spaces. *Comput. Math. Appl.* **62**, 3524–3528 (2011)
23. Molodtsov, D.: Soft set theory—first results. *Comput. Math. Appl.* **37**, 19–31 (1999)
24. Molodtsov, D., Leonov, V.Y., Kovkov, D.: Soft sets technique and its application. *Nechetkie Sistemy I Myagkkie Vychisleniya* **1**, 8–39 (2006)
25. Mushrif, M.M., Sengupta, S., Ray, A.K.: Texture classification using a novel, soft-set theory based classification algorithm. In: *Asian Conference on Computer Vision* (pp. 246–254). Springer, Berlin (2006)
26. Nagarajan, E.K.R., Meenambigai, G.: An application of soft sets to lattices. *Kragujevac J. Math.* **35**(35), 75–87 (2011)
27. Nazmul, S., Samanta, S.K.: Neighborhoods properties of soft topological spaces. *Ann. Fuzzy Math. Inform* **6**(1), 1–15 (2013)

28. Pei, D., Miao, D.: From soft sets to information systems. *Proc. Granular IEEE* **2**, 617–621 (2005)
29. Prasad, A.K., Thakur, S.S.: Soft almost regular spaces. *Malaya J. Matematik* **7**(3), 408–411 (2019)
30. Prasad, A.K., Thakur S.S.: Soft almost I-normal spaces. In: Presented in 23rd FAI International Conference on Emerging Trends and Adaptation of Spirituality in Management, Artificial Intelligence, Education, Health, Tourism, Science and Technology, 19–21 Feb. 2021
31. Rong, W.: The countabilities of soft topological spaces. *Int. J. Math. Comput. Sci.* **6**, 952–955 (2012)
32. Sahin, R., Kuçuk, A.: Soft filters and their convergence properties. *Ann. Fuzzy Math. Inform* **6**(3), 529–543 (2013)
33. Saraf, S.: Survey or review on soft set theory and development. *SIJ Trans. Comput. Sci. Eng. Appl. (CSEA)* **1**(3), 59–66 (2013)
34. Senel, G., Cagman, N.: Soft topological subspaces. *Ann. Fuzzy Math. Inform.* **10**(4), 525–535 (2015)
35. Shabir, M., Naz, M.: On soft topological spaces. *Comput. Math. Appl.* **61**, 1786–1799 (2011)
36. Singal, M.K., Arya, S.P.: On almost regular spaces. *Glasnik Mat.* **4**(24), 89–99 (1969)
37. Thakur, S.S., Rajput, A.S.: P-connectedness between soft sets. *Facta Univ. Ser. Math. Inf.* **31**(2), 335–347 (2016)
38. Thakur, S.S., Rajput, A.S.: Connectedness between soft sets. *New Math. Nat. Comput.* **14**(01), 53–71 (2018)
39. Thakur, S.S., Prasad, A.K.: Soft almost I-regular spaces. *J. Indian Acad. Math.* **41**(1), 53–60 (2019)
40. Varol, B.P., Aygun, H.: On soft Hausdorff spaces. *Ann. Fuzzy Math. Inform.* **5**(1), 15–24 (2013)
41. Yang, W.: New similarity measures for soft sets and their application. *Fuzzy Inf. Eng.* **5**(1), 19–25 (2013)
42. Yuksel, S., Dizman, T., Yildizdan, G., Sert, U.: Application of soft sets to diagnose the prostate cancer risk. *J. Inequalities Appl.* **1**, 1–11 (2013)
43. Yuksel, S.A., Tozlu, N., Ergul, Z.G.: Soft regular generalized closed sets in soft topological spaces. *Int. J. Math. Anal.* **8**(8), 355–367 (2014)
44. Zakri, A.H., Hossen, H.M., Erwi, L.Y., Al-Sharif, E.: Application of soft sets to diagnose the educational obstacles for students. *J. Innov. Technol. Educ.* **3**(1), 61–70 (2016)
45. Zorlutuna, I., Akdag, M., Min, W.K., Atmaca, S.: Remarks on soft topological spaces. *Ann. Fuzzy Math. Inform.* **3**(2), 171–185 (2012)
46. Zorlutuna, I., Cakir, H.: On continuity of soft mappings. *Appl. Math. Inf. Sci.* **9**(1), 403–409 (2015)

A Comprehensive Study on Mobile Malwares: Mobile Covert Channels—Threats and Security



Ketaki Pattani  and Sunil Gautam 

Abstract In current scenario, Android operating system powers billions of users, and everyday numerous Android devices are activated. Android OS bridging day-to-day communications covers 74.92% share in the mobile market with 2.6 million apps at the same time experiencing insecurity, threats and penetration. Attackers aim such systems to gain unauthorized access and accomplish malicious intent. Malwares act as an impetus to this and are evolving every day, as current status of malware development has noticed a huge rise of more than 46% detections and 400% rise since 2010. Researchers are constantly working over the requirements of detecting and developing antimalware techniques to curb these security leaks. Covert channel is one such way out for malwares causing to transmit sensitive data from source to sink unnoticed by users as well as state-of-the-art tools. However, these channels are often overlooked or bypassed from mobile security perspective when it comes to detection mechanisms or state-of-the-art tools. This paper aims at providing a detailed study in this direction, analyzing threats and hence directing toward a novel area of security in this regard. Also, this paper focuses on existing mitigation techniques and future requirements to discern and avert such attacks.

Keywords Mobile malwares · Information security · Covert channels · Malware analysis · Android security

1 Introduction

The proliferation in the usage of Android devices [1, 2] shows a growing threat for user privacy and security [3, 4]. These mobile devices store huge amount of data

K. Pattani (✉)

Department of Computer Sciences and Engineering, Institute of Advanced Research,
Gandhinagar, Gujarat, India
e-mail: ketakipattani2902@gmail.com

S. Gautam

Department of Computer Science and Engineering, Institute of Technology, Nirma University,
Ahmedabad, Gujarat, India
e-mail: gautamsunil.cmri@gmail.com

having sensitive information. IMEI number, SMS, contacts, financial information, location information, correspondence information, photographs and videos and more can be termed as sensitive information that if lost can be a threat to user privacy and security. These information can be significantly interesting for the attackers, spies, intruders, opponent organization or any fraudulent person or company [5]. User's data is at a high risk from attacks due to Trojans, spywares or any such malwares that exploit side channels in order to transmit the data outside and pose a threat [6, 7].

SoundComber [8] is one such Trojan that exploits covert channels in the Android devices and is a context-aware malware that leaks sensitive information of the user. Walkinwat [6] is another such application that sends short messages to each contact of the user's address book informing contacts regarding pirated version of Walk and Text in sender's system. Android hosts millions of applications everyday all of which may not be equally secure. This creates a threat of unnoticed and undetected malwares being transmitted thereby leaking privacy-based information. The malware development analysis suggests that the rate of malware development is increasing at a high pace [5]. In order to keep systems secure from such malwares, it should be verified against certain combating techniques. Such combatants are called anti-malwares. Thereby, to ensure security against such Android malwares, critical information about various ways by which security gets compromised and solutions to analyze such ways are undertaken.

Analysis of various existing malwares and attack methodologies have been carried out to have proactive protection and reactive detection of malwares and their infected resources. Various analysis techniques like statically analyzing the code, dynamically analyzing the application behavior or modifying the applications in order to have in-depth examination can be adapted. Information flow tracking through various channels is one such technique of analyzing these attacks. However, various channels exist that are either overlooked or bypassed by the tracking mechanisms such as TaintDroid [9] and FlowDroid [10]. Therefore, they become successful in transmitting sensitive data from source to the sink. The paper deals with detailed study and resultant analysis of these channels. It entails mobile malware analysis in section II. The paper details various types of covert channels, their behavior and covert communications in section III and related work in section IV. In the last section of the paper, conclusion and future scope of the research are mentioned.

2 Android Malware Analysis

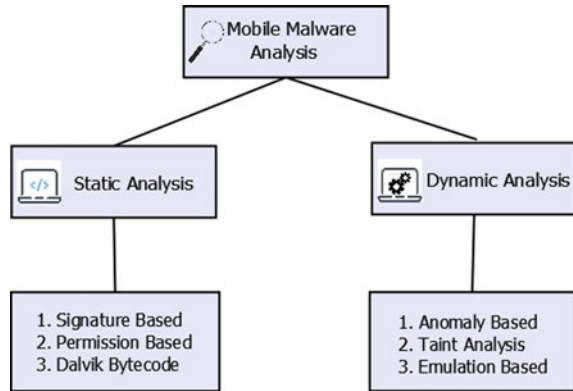
An analysis of malware categories, malware characteristics and standardized approaches is usually done to evade security and must be studied for maintaining the security too. The different types of malwares are categorized based on their operations and working which may include Trojan, Backdoor, Worm, Spyware, Botnet, Ransomware, Riskware, etc. [11] Based on their behavior, a group of malwares performing certain kind of operation are considered to be in the same family

also known as malware family. Various techniques are prevailing to analyze these malwares broadly categorized as static and dynamic analysis.

Static analysis works by checking the source code directly without executing the program or application. It checks through all the possible paths of control-flow and data-flow. However, it does not deal with dynamically injected code or values. Therefore, it also does not monitor run-time processes such as reflection. These are majorly divided into signature based, permission based and Dalvik Bytecode Detection [11]. Signature-based approach analyzes the data based on its semantics and assigns a unique signature to malwares. These signatures are thereafter used for identification of similar kind of malwares. Such methods are very easy to obfuscate as they do not identify unseen malwares. Some tools implementing signature-based approach are AndroSimilar [12] and Droid Analytics [13]. Permission-based approach checks the application permissions if they are unnecessary or certain set of permissions can be vulnerable. APK Auditor [14] and PUMA [15] are the tools implementing permission-based approach. However, applications like SoundComber easily bypass such approach [8]. Functionally, the applications developed in Android have a backend in Java and are further converted to Dalvik code which is a VM based on registry. Such an analysis of bytecode determines the functionality and feature-based behavior of the application. ScanDal [16] works over the same approach of Dalvik bytecode analysis. However, since it is done at instruction level, it has high time and space complexity and hence is unaffordable by Android systems.

Dynamic analysis analyzes the application/code in run-time environment hence detecting the dynamically loaded content too. However, at a time, it can only check one path of execution. Most of the malwares work over dynamic content loading. Dynamic malware analysis is majorly divided into anomaly-based detection, taint analysis and emulation-based detection. Anomaly-based detection, also known as behavioral malware analysis, works in training and testing phases as in tools like Crowdroid [17] and ADroid [18]. In training phase, the detector tries to learn from the behaviors of malware and that learning is applied on other applications in testing phase. One noticeable detection of anomaly-based detection is zero-day attack. However, false positives are likely to occur for benign applications having uncertain behavior. Taint analysis utilizes information flow analysis for detecting sensitive information flow toward unintended point. TaintDroid [9] and XmanDroid [19] are tools implementing taint analysis. Taint analysis can be obfuscated with evasive mechanisms. Emulation-based approach detects both monomorphic as well as polymorphic malwares, detailing its behavior as well as sequence. DroidScope [20] works to have emulation-based detection. However, covert channels have shown significant escape from these detection mechanisms and therefore needs to be concentrated upon (Fig. 1).

Fig. 1 Mobile malware analysis types



3 Covert Channel

Covert channels allow a communication that violates the policy of data transmission. The term covert channels refer to the channels that are originally intended to have transmission of control information, but are effectively used so as to transfer privacy-based information to let it pass undetected by the defense mechanisms. The defense mechanisms are continuously observing the data channels to detect the sensitive information leakage from source to sink. However, covert channels are meant to transfer control information usually, and therefore, go unnoticed by the detection mechanisms. It thus has the potential to convey information objects between processes that are not supposed to communicate as restricted by security policies.

For the communication channels to be covert channels, they must have following properties:

- I. **Detectability:** The covert channels should be detected only by specific intended users.
- II. **Non-distinguishability:** They should not be differentiated or recognized.
- III. **Size:** They should not replace whole channel legitimate data for which it is intended.

Covert channel techniques are being studied since years and are found in multiple areas of information technology and computer systems, such as in local systems [21, 22], in Internet protocol (IP) addressing and transmission control protocol (TCP) [23–25], in virtual machines [26]—just to state a few. Also, prevention, mitigation and detection of these covert channels have been studied since decades [27–29]. However, the idea of covert channels in mobile devices is not under discussion since earlier stage and still has negligible focus. Therefore, security from these channels become a primary concern for business organizations as well as society.

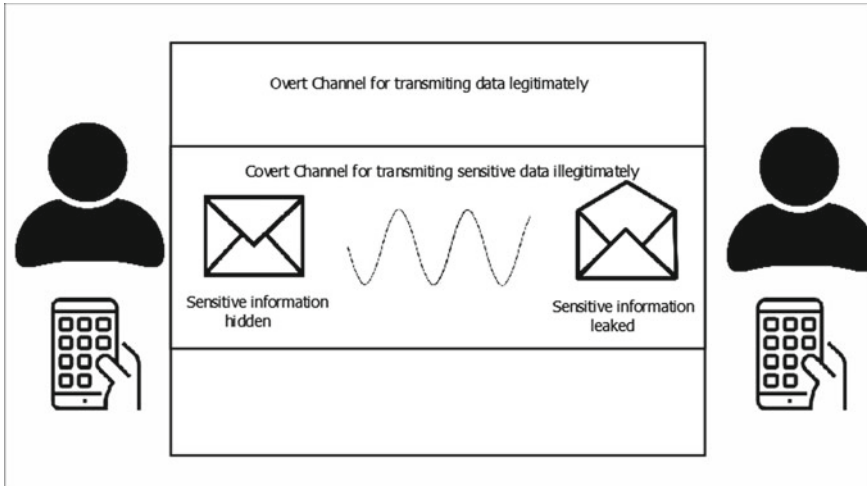


Fig. 2 Transmission of information in covert and overt channels

3.1 Types of Covert Channels

In Android, there are various overt and covert channels. Overt channels are the ones actually intended to have data transfer between systems, whereas covert channels do not need any extra permission to bypass the security mechanisms like Taintdroid [9]. This proves to be a great advantage as existing security systems also have a check over the permissions demanded and certain combinations of permissions are considered as dangerous. Covert channels thus confirm a stealthy medium of communication between the users. The usage of covert channels has advanced within last few years as it supports freedom of speech as they allow to neglect Internet restrictions or censorship and also enables botnets to implement stealthy communication based on controls.

Figure 2 depicted here shows the transmission of information that would occur in overt and covert channels. Overt channels would allow legitimate information transmission, whereas covert channels would leak illegitimate information from one end to another. This transmission can occur from victim to attacker or may occur within the system of the victim amongst applications which would later leak the information toward the attacker.

3.2 Android-Based Covert Channels

Android-based covert channels work with a similar baseline and exploit legitimate channels to transmit sensitive information. Android has many such covert channels that behave differently during communication and leak information. This can have

overt and covert channels like shared settings, system logs, file, broadcast intents, event intents, UNIX socket, thread enumeration, free space, processor statistics and processor frequency [6]. Table 1 gives a list of these channels very few of which have side effects or noticeable effects. Shared settings may include settings that are shared by multiple applications having access to those settings like volume settings, vibration settings or screen settings which are exploited to transmit the data from source to sink. Apart from the mentioned covert channels, there is an additional covert channel, i.e., sound-based covert channel. The sound above 20 kHz is called ultrasound and is inaudible to human ear. The sound between the frequency 17–20 kHz is termed as near ultrasound. This sound is produced by mobile devices, but is not audible to people above teenage. Due to this reason, this sound is used by attackers as carrier of information. On the other hand, a receiver application receives the near ultrasound and extracts information leaked [30]. Even sensors, camera, Bluetooth, network and Universal Serial Bus (USB)-based covert channels have been noticed to leak data from Android devices [31].

Control-flow graphs and data-flow graphs are tracked to determine the information flow and execution path from sensitive sources to sink. Source here means any point that acts as the initiator of sensitive information and can be user account, e-mail, contact, calendar, database, file, location logs, phone state, SMS/MMS, settings or unique deviceID such as International Mobile Equipment Identity (IMEI), International Mobile Subscriber Identity (IMSI) and Subscriber Identification Module (SIM ID). Sink may include Internet, memory card, inter-process communication message, SMS or others through which the data exits the system and can reach the intruder. Security of information being transmitted between these sources and sinks

Table 1 List of covert and overt channels [6]

Overt/covert channel type	Required permission	Side effect	Throughput
Shared settings	–	–	>100 bps
System log	READ_LOGS	–	–
File	WRITE_EXTERNAL_STORAGE	–	–
Broadcast intents	–	–	10–100 bps
Event intents	–	May raise attention	–
UNIX socket	–	–	>100 bps
Thread enumeration	–	–	>100 bps
Free space	–	–	10–100 bps
Processor statistics	–	–	<10 bps
Processor frequency	–	–	<10 bps

becomes specifically more important when covert channels get involved as a medium of transmission, creating greater chances of information getting leaked.

The capability of covert channel-based attack depends upon following: whether it has ability and permission to read from a resource, whether it has ability and permission to write to some resource, whether it can turn resources on or off and whether it can control the accessibility of the resources, i.e., if it can lock and unlock the resources. Here, Table 1 represents a list of covert and overt channels along with their required permission and side effects visible to the user. The throughput here determines the rate at which information can be transmitted. Also, the attack procedure needs to take care of certain shared resources and sequencing. Sending and receiving processes may be on different devices or on the same device. In case of different devices such as ultrasound, one end leaks the information with ultrasound and the other one listens to the sound and fetches the information. Here, sending and receiving processes must have an access to the shared attribute object. The sending object must be able to edit or make changes into the object with which the information is to be leaked. On the receiving end, the exact reference to the object must be possible, only then data can be received. It should also be able to control and manipulate incoming data and its speed. For making the process synchronous, a real-time clock pulse or indication for sequencing is required. This will make the transmissions in sequence for the sender and receiver [32].

The process begins when sender application or process initiates a channel for communication. This channel must abide by the above mentioned requirements for successful communication. The second application or process on the other hand connects to the established channel by listening to the sender. The sender now encodes the information bit by bit so as to transmit one bit at a time. The sequencing and synchronous transmission is to be handled by clock common to both the ends. The receiver listens to the bit by bit encoded data and decodes the signal or received data to get required information. Here, the most important task is to have synchronous communication as the privacy-based information is broken down into bit by bit data transmissions. Certain factors that affect this transmission include parallel application interfacing that might be run by user, uncertain user interactions and bandwidth limitation. It is possible that while the covert communication is established in backend, the user is interacting with same or different application. Also, noise caused due to multiplicity of tasks and delay has to be handled. The solution of these issues is synchronization of data and transmissions. Synchronous execution leads to proper handling of covert channel as well as application visible to the user. Any covert communication utilizing any covert channel follows the process mentioned. Parameters of communication may vary depending upon the channel selected.

4 Related Work

Various static as well as dynamic state-of-the-art tools have been designed to secure mobile device data and communications. However, many covert channels have

proved to be undetectable in turn by having lower footprints, cannot be easily detected by security mechanisms due to lowered file systems and permissions and also go undetected by the users. Tools such as TaintDroid use taint tracking techniques most of which do not successfully detect the flow of these sensitive information. Mentioned are some of the state-of-the-art tools:

I. TaintDroid

It provides system wide information or data-flow tracking within the system. It can track multiple sources of sensitive concurrently, such as camera, GPS and microphone and identifies the data released in third party developer apps or outside the system environment. It does this by labeling the sensitive information as tainted, keeping a track of these data and monitoring the sink. However, it is unable to detect the sensitive data leakage through the covert channels as these channels are not meant to have data transmission and hence go unnoticed [9].

II. DroidScope

Yan in DroidScope specifies that it is based on virtual machine (VM) introspection. DroidScope observes the whole operating system of Android by staying out of the execution environment and thus have more privileges than the malware programs. It also monitors the Dalvik semantics; thus, the privilege escalation attacks on kernel can also be detected. It is built upon QEMU. DroidDream and DroidKungFu were detected with this technique [20].

III. AntiMalDroid

The tool operates on dynamic analysis of an application to track its execution using support vector machine (SVM) algorithm. Firstly, the analysis is done for the determination of benign or the malicious application, and then, they are put into learning module after which signature is prepared. The signature is the basis and is used every time to check any application for vulnerability. This detection is dependent on the known attack signatures for future detections [33].

IV. MockDroid

The dynamic detection application allows a user to block certain application that performs malicious activities. The user shall be notified of the malicious tasks and he/she can either block it or provide fake data to that application, i.e., if an application using location information is found suspicious, user can either block it or provide false information to the application hence maintaining security. However, if any malicious application is trusted by MockDroid, it allows transmissions [34].

V. LeakMiner

The tool detects the information transmitted or leaked through static taint analysis. A system called 'Dexteroid' identifies sensitive data leakage through applications like short message service (SMS). This detection is done statically, but can be used

beforehand while applications are distributed to users. Also, it does not cause any run-time overhead to normal execution of the application [35].

VI. Inter-Component Privacy Leaks in Android Apps (IccTA)

The application is a static taint analyzer which goes beyond state-of-the-art tools by detecting the inter-component detection within the systems. There are applications such as ‘Shake It Well’ that ask for certain permissions like record audio to remove background noise. But, it works to record audio and leak it online without notifying the user. IccTA, therefore, detects inter-component communication for analysis of such data leaks. It propagates the context information along with the components [36].

VII. Dynamic Taint Analysis (DYTAN)

The tool refers to dynamic taint analysis and addresses the limitations of ad hoc taint mechanisms replacing it with dynamic taint techniques. It does data flow analysis as well as control-flow analysis of the application. Not depending upon the customized run-time environment, it has its own logic. DYTAN works at application level and is capable of dealing with external as well as system libraries. However, efficient improvement and dynamic testing for software development context are future aspects [37].

Current research suggests that existing state-of-the-art tools do not comprise potential detection or analysis mechanisms when it comes to covert channel-based privacy violations or data leakage. Covert channels in mobile devices can delegate policy-breaking transmissions without notifying the security mechanisms [6]. This poses a great threat to privacy and security and may lead to leakage of confidential information. The related work here suggests that existing static or dynamic state-of-the-art tools are not efficient to perform required covert channel-based analysis and detection. Considering the Android-specific covert channels, it becomes utmost necessary to perform detailed analysis prior to preparing a comprehensive detection. Various covert channels such as screen, processor statistics and processor frequency depict certain characteristics such as heating or memory usage that can be used to easily detect such channels. However, false positives also need to be taken care of while examining these characteristics.

Here, advanced mechanisms such as aspect-oriented programming can be utilized. Aspect-oriented programming addresses issues that can neither be solved using object-orientation nor procedural approaches. Here, the code is broken down into various modules comprising of cross-cutting concerns such as transaction management that cut across multiple objects and types. Various join points are created based on pointcuts that would define a point of execution during a method or exception handling by defining advices that represents the action to be taken at that join point. AspectJ, JBoss and Spring is some of the dominant frameworks in Java supporting modularization [38]. Also, Java reflection is a concept to modify the behavior of a class at run-time. This behavior can be used to analyze the variables, methods and classes at run-time, thereby determining the possible data-flow and control-flow [39].

5 Conclusion

The research aims at drawing attention to an unnoticed threat over the Android systems. Current security mechanisms as well as analysis tools do not necessarily focus over the threats posed by covert channels and are either overlooked or bypassed during detection. The paper deals with threats caused by different types of covert channels, their impacts, their operational differences and undetectability from existing state-of-the-art tools is depicted. Here, research comes to a conclusion that existing state-of-the-art tools are not sufficient for effective detection and analysis. Thereby, it is suggested to utilize advanced concepts of aspect-oriented programming or reflection where analysis of such channels can be done from outside of the execution environment.

The work opens up a novel scope of research in this direction. The research shows that such attacks are possible and can harm user privacy and security at a very dangerous level. Thus, it opens up wide ground for defense mechanisms to be incorporated within existing state-of-the-art tools. The research sights its future scope in terms of raising the power of existing detection algorithms or state-of-the-art tools. Also, advanced techniques using machine learning can be incorporated to better serve wider range of detections.

References

1. Lu, X. et al.: PRADA: prioritizing android devices for apps by mining large-scale usage data. In: 2016 IEEE/ACM 38th International Conference on Software Engineering (ICSE), Austin, TX, USA, pp. 3–13 (2016)
2. Samra, A.A.A., et al.: Analysis of clustering technique in android malware detection. In: Seventh International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, Taichung, Taiwan, pp. 729–733 (2013)
3. Mahindru et al.: FSDroid:—a feature selection technique to detect malware from android using machine learning techniques. *Multimed. Tools Appl.*, pp. 1–53 (2021)
4. Enck, W., et al.: A study of android application security. In: Proceedings of the 20th USENIX Security Symposium (2011)
5. Zhou, Y., et al.: Dissecting android malware: characterization and evolution. In: IEEE Symposium on Security and Privacy, San Francisco, CA, USA, pp. 95–109(2012)
6. Lalande, J., et al.: Hiding privacy leaks in android applications using low-attention raising covert channels. In: International Conference on Availability, Reliability and Security, Regensburg, Germany, pp. 701–710 (2013)
7. Quang Do, S. et al.: Exfiltrating data from android devices. *Comput. Secur.*, **48**, 74–91 (2015)
8. Roman et al.: Soundcomber: a stealthy and contextaware sound trojan for smartphones. In: Proceedings of the 18th Annual Network and Distributed System Security Symposium (NDSS), San Diego, CA, pp. 17–33 (2011)
9. Enck, W., et al.: TaintDroid: an information-flow tracking system for realtime privacy monitoring on smartphones. *ACM Trans. Comput. Syst.* **32**(2), 5.1–5.29 (2014)
10. Arzt, S. et al.: FlowDroid: precise context, flow, field, object-sensitive and lifecycle-aware taint analysis for Android apps. In: Proceedings of the 35th ACM SIGPLAN Conference on Programming Language Design and Implementation, vol. 49, no. 6, pp. 259–269 (2014)

11. Faruki, P., et al.: Android security: a survey of issues, malware penetration, and defenses. *IEEE Commun. Surv. Tutorials* **17**(2), 998–1022 (2015)
12. Faruki, P., Ganmoor, V., Laxmi, V., Gaur, M.S., Bharmal, A.: AndroSimilar: robust statistical feature signature for Android Malware detection. In: Eli, A., Gaur, M. S., Orgun, M. A., Makarevich, O. B. (eds.) *SIN*, ACM, pp. 152–159 (2013)
13. Zheng, M., Sun, M., Lui, J. C. S.: DroidAnalytics: a signature based analytic system to collect, extract, analyze and associate Android Malware, *CoRR abs/1302.7212*
14. Talha, K.A., Alper, D.I., Aydin, C.: APK auditor: permission-based Android Malware detection system. *Digit. Invest.* **13**, 1–14 (2015)
15. Sanz, B., Santos, I., Laorden, C., Ugarte-Pedrero, X., Bringas, P.G., Alvarez, G.: PUMA: permission usage to detect malware in android. In: *International Joint Conference CISIS12-ICEUTE' 12-SOCO' 12 Special Sessions*, Springer, pp. 289–298 (2013)
16. Kim, J., Yoon, Y., Yi, K., Shin, J., Center, S.: ScanDal: static analyzer for detecting privacy leaks in Android applications. In: *Proceedings of the Workshop on Mobile Security Technologies (MoST)*, in conjunction with the IEEE Symposium on Security and Privacy, (2012)
17. Burguera, I., Zurutuza, U., Nadjim-Tehrani, S.: Crowdroid: behavior based Malware detection system for Android. In: *Proceedings of the 1st ACM workshop on Security and privacy in smartphones and mobile devices*, ACM, pp. 15–26, (2011)
18. Ruiz-Heras, A., García-Teodoro, P., Sánchez-Casado, L.: ADroid: anomaly-based detection of malicious events in Android platforms. *Int. J. Inf. Secur.* **16**, 371–384 (2017)
19. Bugiel, S., Davi, L., Dmitrienko, A., Fischer, T., Sadeghi, A.-R.: Xmandroid: a new android evolution to mitigate privilege escalation attacks (2011)
20. Yan, L.K., et al.: DroidScope: seamlessly reconstructing the OS and Dalvik semantic views for dynamic Android malware analysis. In: *Proceedings of 21st USENIX Security Symposium*, p. 29 (2012)
21. Girling, et al.: Covert Channels in LANs. *IEEE Trans. Softw. Eng.*, **SE-13**(2) (1987)
22. Wolf, M.: Covert channels in LAN protocols. In: *Proceedings on Workshop of Local Area Network Security (LANSEC)*, pp. 91–101 (1989)
23. Zander, S., et al.: A survey of covert channels and countermeasures in computer network protocols. *IEEE Commun. Surv. Tutorials* **9**(3), 44–57 (2007)
24. Murdoch, S.J. et al.: Embedding covert channels into TCP/IP. In: *Proceedings of 7th Information Hiding Workshop* (2005)
25. Lucena, N.B., et al.: Covert channels in IPv6. In: *Proceedings on Privacy Enhancing Technologies (PET)*, pp. 147–66 (20005)
26. Okamura, K. et al.: Load-based covert channels between Xen virtual machines. In: *Proceedings on 2010 ACM Symposium on Applied Computing*, pp. 173–180 (2010)
27. Cabuk, S., et al.: IP covert timing channels: design and detection. In: *Proceedings on 11th ACM Conference of Computer and Communications Security (CCS)*, pp. 178–187 (2004)
28. Berk, V. et al.: Detection of covert channel encoding in network packet delays. *Computer Science Technical Report TR2005–536* (2005)
29. Goher, S.Z., et al.: Covert channel detection: a survey based analysis. *High Capacity Optical Networks and Emerging/Enabling Technologies*, Istanbul, Turkey, pp. 057–065 (2012)
30. Deshotels, L.: Inaudible sound as covert channel in mobile devices. In: *WOOT' 14 Proceedings of the 8th USENIX Conference on Offensive Technologies*, pp. 16.1–16.9 (2014)
31. Farshteindiker, B., Hasidim, N., Grosz, A., Oren, Y.: How to phone home with someone else's phone: information exfiltration using intentional sound noise on gyroscopic sensors. In: *Proceedings on USENIX Workshop Offensive Technol. (WOOT)*, pp. 1–10 (2016)
32. Chandra et al.: Towards a systematic study of the covert channel attacks in smartphones. In: *International Conference on Security and Privacy in Communication Networks*, pp. 427–435 (2014)
33. Zhao, M., Ge, F., Zhang, T., Yuan, Z.: AntiMalDroid: an efficient SVM-based malware detection framework for Android (2011)
34. Beresford, A. et al.: MockDroid: trading privacy for application functionality on smart-phones. In: *HotMobile '11 Proceedings of 12th Workshop on Mobile Computing Systems and Applications*, Phoenix, Arizona, pp. 49–54 (2011)

35. Yang, Z., et al.: LeakMiner: detect information leakage on Android with static taint analysis. In: 2012 Third World Congress on Software Engineering, Wuhan, China, pp. 101–104 (2012)
36. Li, L. et al.: IccTA: detecting inter-component privacy leaks in Android apps. In: 2015 IEEE/ACM 37th IEEE International Conference on Software Engineering, Florence, vol. 1, pp. 280–291 (2015)
37. Clause, J., et al.: Dytan: a generic dynamic taint analysis framework. In: Proceedings of the International Symposium on Software Testing and Analysis, pp. 196–206 (2007)
38. Kiczales, G. et al.: Aspect-oriented programming. In: European Conference On Object-Oriented Programming. Springer, Berlin, Heidelberg, pp. 220–242 (1997)
39. Livshits, et al.: Reflection analysis for Java. In: Asian Symposium on Programming Languages and Systems, Springer, Berlin, Heidelberg, pp. 139–160 (2005)

Overview of Incorporating Nonlinear Functions into Recurrent Neural Network Models



Farzaneh Nikbakhtsarvestani

Abstract This paper describes training recurrent neural networks (RNNs) which are able to learn features and long-range dependencies from sequential data. Although training RNNs is mostly plagued by the vanishing and exploding gradient problems, but there is a RNN architecture so called long short-term memory (LSTM) to address these issues. To demonstrate the impact of nonlinearity of activation functions on training recurrent neural networks, the limitations of the LSTM algorithm on speech recognition technology are presented in this study.

1 Introduction

Recurrent neural networks (RNNs) which have attracted great attention have been widely studied from 1986 and were based on David Rumelhart's work for modeling time series. These networks are used for different machine learning tasks which can modelize sequential data.

Recurrent neural networks have many applications, especially when the input and output have variable lengths such as handwriting recognition, speech recognition, and image to text.

The layers of connected units called artificial neurons make artificial neural networks (ANNs). The shallow network of ANN includes an input layer, an output layer, and at most a hidden layer without a recurrent connection. Recurrent connections in ANNs make them recurrent neural networks (RNNs), and the increase of complexity of network depends on the number of layers. More number of layers or recurrent connections generally increases the depth of the network and empowers it to provide various levels of data representation and feature extraction, referred to as deep learning [1]. The difference between these networks and higher layer ones is related to their units. The structure of hidden states causes RNNs to store, remember, and process past complex signals for long time periods and works as the memory of the network and state of the hidden layer at a time which is conditioned on its previous

F. Nikbakhtsarvestani (✉)

Department of Science, University of Ontario Institute of Technology, Oshawa, ON, Canada
e-mail: farzaneh.nikbakht@ontariotechu.ca

state [2]. Mapping the input sequence at the current timestep to the output sequence which is prediction the sequence in the next timestep is the another capability of RNNs. Artificial neurons and the feedback loops which are recurrent cycles over time or sequence make RNNs as a class of supervised machine learning models [3]. In this model, for training the RNNs, the dataset of input-target pairs needs to be trained, and the goal is minimizing the difference between output and target pairs via optimizing the weights of the network. This work is based on training recurrent neural network [1] and error bound for approximations [4].

Definition 1.1 (*Neural Network*) Let $d, L \in \mathbb{N}$. A neural network U with input dimension d and L layers is a sequence of matrix-vector tuples

$$U = ((A_1, b_1), (A_2, b_2), \dots, (A_L, b_L))$$

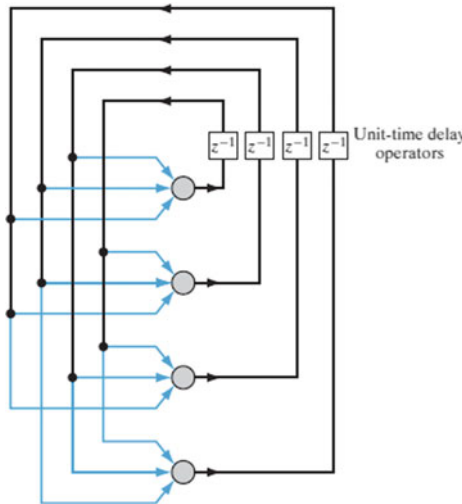
where $N_0 = d$ and $N_1, \dots, N_L \in \mathbb{N}$ and where

$$A_L := N_L \times \sum_{k=0}^{L-1} N_k,$$

and $b_L \in \mathbb{R}^{N_L}$. (Note that N_L is the definition of the output layer of U)

2 A Simple Recurrent Neural Network

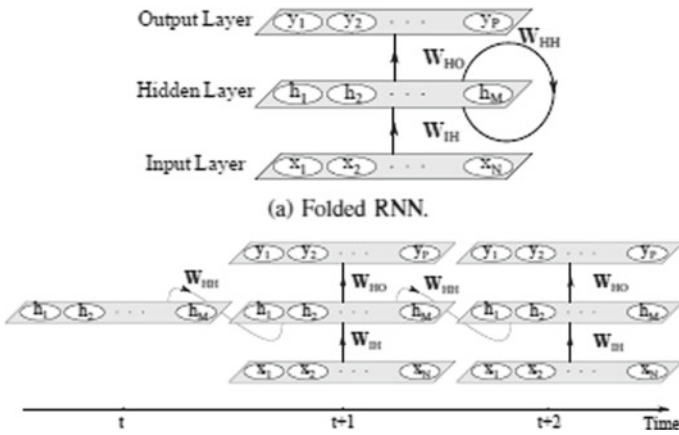
Recurrent cycles over time are called feedback loops. In the neural network literature, artificial neurons with one or more global feedback are referred to as recurrent networks [1]. RNNs are a class of supervised machine learning models. Learning capability of the RNNs and its performance depends on the amount of feedback loops.



Also where the output of a neuron is fed back into its own input, the network has self-feedback. Moreover in the situation that neural network contains nonlinear units, feedback loops include particular branches unit time delay operators (denoted by Z^{-1} in the figure).

2.1 Model Architecture

Recurrent neural network architectures can have many different forms. The layer of input, recurrent hidden, and layer of output are three layers in simple RNNs.



A sequence of vectors through time $\{\dots, x_{t-1}, x_t, x_{t+1}, \dots\}$ such that $x_t = \{x_1, x_2, \dots, x_N\}$ makes the input of input layer.

In a fully connected RNN, the input units are connected to the hidden units. In this layer, a weight matrix W_{IH} defines the connections. The hidden units $h_t = \{h_1, h_2, \dots, h_N\}$ cause the hidden layer to connect with recurrent connections through time. The stability and performance of the network depend on the initialization of the hidden units with small elements. The state space (memory) of the system is defined by hidden layer as

$$h_t = f_H(O_t) \tag{2.1}$$

such that $f_H(\cdot)$ is the activation function and

$$O_t = W_{IH}x_t + W_{HH}h_{t-1} + b_h$$

In the hidden units, b_h is the bias vector. In the third layer which is output layer, the units are computed as

$$y_t = f_O(W_{HO}h_t + b_0)$$

where $f_O(\cdot)$ is activation function and b_0 is the bias in this layer. Weighted connections W_{HO} connect the hidden layer to the output layer. A set of values which summarizes all the unique necessary information of the previous states of the network through

the time is hidden state of a RNN. These hidden states make accurate predictions at the output layer according to input vector. If a simple RNN is trained well, the network will be capable for modeling rich dynamics; however in every units, simple activation function is used [1].

2.2 Activation Function

In the output layer for training a classification model, an activation function is applied. The activation function must be continuous in order to meet differentiability requirements. Sigmoidal nonlinear functions are examples of a continuously differentiable nonlinear activation functions which are used in multi-layer perceptrons.

Some popular activation functions are rectified linear unit (ReLU) , “tanh”, and “sigmoid”. These functions (“tanh” and “sigmoid”) are two forms of sigmoidal nonlinear functions.

Both the nature of the machine learning problem and the training dataset are two important factors for choosing the proper activation function. The activation function typically takes the form of a hyperbolic tangent function which is defined as

$$\tanh(x) = \frac{e^{2x} - 1}{e^{2x} + 1}$$

or logistic(sigmoid) function which is known as common choice activation function

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

The domain of this function is real numbers, and its range is $[0, 1]$. These two activation functions which fastly saturate the neuron and cause the gradient to be vanished are related as

$$\sigma(x) = \frac{\tanh\left(\frac{x}{2}\right) + 1}{2}$$

Obviously, the scaled version of sigmoid activation function is tanh. The ReLU activation function which works on positive input values is defined as

$$\rho(x) = \max(x, 0)$$

Although comparison between ReLU activation function and another two activation functions indicates that in ReLU activation function, the acceleration of the convergence of stochastic gradient descent (SGD) is greater than tanh and sigmoid, but due to the lack of resistance of ReLU function against growing the weight matrix and the large gradient, neuron may be inactive by using of this type of activation function during training.

3 Training Recurrent Neural Network

Training the RNN in which the training loss being minimized is a main issue in such networks. Optimizing the algorithm in order to tune the weights and instantiating them are the main approaches used for minimizing the training loss. The main focus in optimizing the machine learning algorithm is on the convergence and reducing the complexity of training section of the algorithm which needs a large number of iterations. There are many approaches for training RNNs. In this paper, we study activation functions in gradient-based machine learning algorithms and their modified forms.

3.1 Gradient-Based Learning Methods

One of the most common approaches to optimize neural network is gradient descent (GD). Although this method causes the total loss to minimized, but for large datasets, this method is computationally expensive and is not appropriate for training the models as inputs arrive (i.e., online training). Basically in this way by computing the error function derivative with respect to each member of the weight matrices, the weights of the model are set. Assuming that the activation function is nonlinear and differentiable, in order to minimize the total loss, the gradient descent alters at each weight. In GD, each iteration of optimization for doing an update follows of this formula:

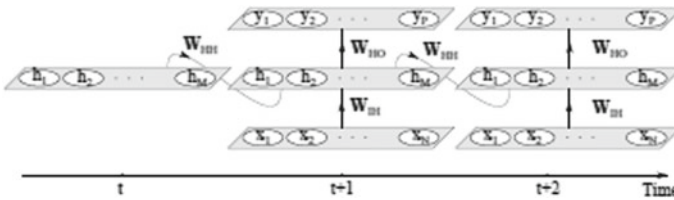
$$\theta_{t+1} = \theta_t - \frac{\lambda}{U} \sum_{k=1}^U \frac{\partial L_k}{\partial \theta}$$

where λ is the rate of training and U is the extent of training set and θ is the set of parameters. The gradient for whole dataset is computed by GD so GD is considered as batch GD. In other words, by GD method, we follow the direction of the slope of the surface created by the objective function downhill until we reach a valley [5]. Since the time is not considered in GD and RNNs include recurrent cycles over time, so this method does not work properly for training the network. For solving this problem, an extended version of GD through time is needed. It is called backpropagation through time (BPTT). Basically, backpropagation is a specific technique for implementing gradient descent in the weight space for a multi-layer perception [6]. In RNN, the connections between parameters and the dynamics are unstable, this makes computing error-derivatives through time complicated, and GD method is thus inadequate. Another shortcoming of GD is due to the difficulty in recognizing the dependencies as they increase in magnitude. The only parameter which is considered by loss function derivative with respect to weights is the distance between the updated output and its consistent target as the history of weights information is not applied. The vanishing gradient is another deficiency of applying GD method for training RNN. The exponential decay of backpropagated gradient

causes RNNs not to learn long-term temporal dependencies. In a reverse situation, GD method may lead to explode gradient issue which is due to exponentially blow-up of backpropagated gradient. This result is in unstable learning process. We are going to discuss these challenges and provide an architecture for solving these problems.

3.1.1 Backpropagation Through Time (BPTT)

The simplest type of neural network is the feedforward neural network. This network has no loop, and the information moves from input nodes, via hidden ones and finally output nodes (i.e., in only one direction). For feedforward networks, the method “BPTT” is used to train the network. BPTT is the generalization of backpropagation. In this method, the idea is making the network unfolded in time, and the signals of error propagate backwards through time [2].



In this network, the parameters can be considered as the set

$$\theta = \{W_{HH}, W_{IH}, W_{HO}, b_h, b_I, b_O\} \tag{3.1}$$

These parameters affect the loss function in the previous timesteps. The gradients of loss function with respect to this set are

$$\frac{\partial L}{\partial \theta} = \sum_{t=1}^T \frac{\partial L_T}{\partial \theta}$$

where for the loss function L , we have

$$\frac{\partial L_t}{\partial \theta} = \sum_{k=1}^t \left(\frac{\partial L_t}{\partial h_t} \cdot \frac{\partial h_t}{\partial h_k} \cdot \frac{\partial h_k^+}{\partial \theta} \right),$$

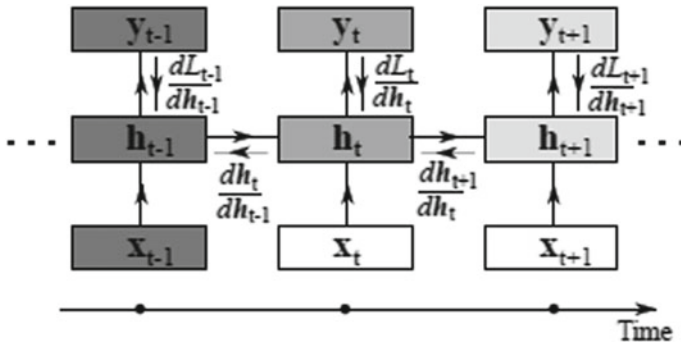
where h_t is the hidden state of network at time t and $\frac{\partial h_k^+}{\partial \theta}$ is the “immediate” partial derivative. For propagating the error signals backward in timesteps t and k which is $k < t$, we have

$$\frac{\partial h_t}{\partial h_k} = \prod_{i=k+1}^t \frac{\partial h_i}{\partial h_{i-1}},$$

and so according to Eq. (2.1) and using the Jacobian matrix for the hidden state we have:

$$\prod_{i=k+1}^t \frac{\partial h_i}{\partial h_{i-1}} = \prod_{i=k+1}^t W_{HH}^T \text{diag}|f'_H(h_{i-1})|.$$

As we see the participation of the hidden states in the network through time is obvious. In terms of the contribution of inputs and corresponding hidden states over time, two types of hidden state contribution are recognizable, such as long-term contribution (for time $k \ll t$) and short time contribution for another time.



By considering the above figure, it is evident that when the new input are admitted to the network, the sensitivity of units vanishes (decreasing the contribution of the inputs x_{t-1} through time), the activation in hidden units is overwritten by BPTT (increasing the contribution of the loss function value L_{t+1} w.r.t h_{t+1} in BPTT trough time) [1].

3.1.2 Vanishing Gradient Problem

The vanishing gradient problem causes some defects for RNNs. This is because of strong nonlinearity which is used for making complex pattern of data. When the gradient propagates back through time, its magnitude decreases exponentially. Subsequently, the long-term correlations are neglected by the network, which causes an issue in learning process of dependencies among distant events. There are two possible explanations for this:

1. The gradient of nonlinear functions which is close to zero.
2. While the gradient propagates back through time, recurrent matrix increases the gradient magnitude.

For the less than one eigenvalues of the recurrent matrix, after five to ten times of running backpropagation algorithm, the rate of convergence of gradient increases. In the RNNs learning process with extended sequences and small weights, the gradient shrinks as well. Long-term components explode in the recurrent weight matrix W_{HH}

when its spectral radius becomes more than 1 and $t \rightarrow \infty$. Because product of matrices can lead to shrinkage/explosion along several directions. In order to generalize this to nonlinear function $f'_h(\cdot)$ in Eq. (2.1), we can bound it with $\gamma \in \mathbb{R}$ s.t

$$\|\text{diag}(f'_H(h_k))\| \leq \gamma$$

And since we have

$$\prod_{i=k+1}^t \frac{\partial h_i}{\partial h_{i-1}} = \prod_{i=k+1}^t W_{\text{HH}}^T \text{diag}|f'_H(h_{i-1})|$$

so

$$\left\| \frac{\partial h_{k+1}}{\partial h_k} \right\| \leq \|W_{\text{HH}}^T\| \cdot \|\text{diag}(f'_H(h_k))\| \leq 1$$

Now for $\delta \in \mathbb{R}$, if we consider $\|\frac{\partial h_{k+1}}{\partial h_k}\| \leq \delta < 1$ for loss function component, we have

$$\left\| \frac{\partial L_t}{\partial h_t} \left(\prod_{i=1}^{t-1} \frac{\partial h_{i+1}}{\partial h_i} \right) \right\| \leq \delta^{t-k} \left\| \frac{\partial L_t}{\partial h_t} \right\|$$

in different timesteps. Since $\delta < 1$, increasing $t - k$ leads to vanishing gradient problem. Generally in recurrent matrix W_{HH} , if for the largest singular value λ_1 we have $\lambda_1 < \frac{1}{\gamma}$, the gradient vanishing problem happens.

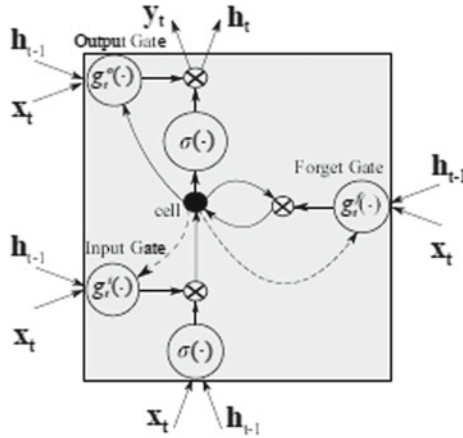
3.1.3 Exploding Gradient Problem

As its mentioned, the process of training RNNs with BPTT may be exposed by exploding problem. In training recurrent neural networks on long sequences, increasing the weights causes the norm of gradient to increase and gradients are subsequently exploded. This is a different necessary condition in comparison to the vanishing gradient problem in which for the largest singular value of recurrent matrix W_{HH} (i.e., λ_1), we have $\lambda_1 > \frac{1}{\gamma}$.

4 Long Short-term Memory

As mentioned before, the main shortcoming of BPTT method pertains to error signals flowing backwards in time. This causes gradients to vanish or explode through time which turns out to more difficulties in learning long-term dependencies. To tackle this problem, some methods have been proposed. One of the most successful techniques to strengthen the long-term dependencies is known to be the long short-term memory

(LSTM). In this method, the sigmoid or tanh hidden units are replaced with “memory cell”. This change leads to more controlled behavior of backpropagated gradients. In this approach, the input and output values of the cell memory are controlled by gates. Each cell is also matched with a forget gate that controls the decay rate of its stored values [7]. In this way, the memory cell holds its stored values during the periods that the input and output gates are off, and the forget gate is not causing decay [1]. Therefore, the gradient of the error with respect to its stored value, when backpropagated over those periods, stays constant [8]. Depending on the training application, there are varieties of LSTM structures developed by many researchers. In the following chapter, standard LSTM approach will be illustrated, and then, we will focus on bidirectional LSTM in particular to suit our applications.



4.1 Standard LSTM

As its shown in above figure, each typical memory cell has its own input, output, and forget gate and a cell activation component that provide continuous analogs of write, read, and reset operations for the cells. More precisely, the input, forget, and output gate are trainable to learn, respectively, what information to store in the memory, how long to store it, and when to read it out [9]. The activation of the cell is controlled by the designed multipliers. The input to the cells is multiplied by the activation of the input gate, the output to the net is multiplied by that of the output gate, and the previous cell values are multiplied by the forget gate. The net can only interact with the cells via the gates [10]. The input gate of LSTM is defined as

$$g_t^{\text{in}} = \sigma(W_{I_g^{\text{in}}}x_t + W_{H_g^{\text{in}}}h_{t-1} + W_{g_c^{\text{in}}}g_{t-1}^c + b_{g^{\text{in}}}) \tag{4.1}$$

where $W_{...}$ is the weight matrix as below

- $W_{I_g^{\text{in}}}$: input layer \rightarrow input gate

- $W_{\text{Hg}^{\text{in}}}$: hidden state \rightarrow input gate
- $W_{\text{g}^{\text{c}}\text{g}^{\text{in}}}$: cell activation \rightarrow input gate
- $b_{\text{g}^{\text{in}}}$: bias of the input gate
- forget gate :

$$g_t^{\text{f}} = \sigma(W_{\text{Ig}^{\text{f}}}x_t + W_{\text{Hg}^{\text{f}}}h_{t-1} + W_{\text{g}^{\text{c}}\text{g}^{\text{f}}}g_{t-1}^{\text{c}} + b_{\text{g}^{\text{f}}}) \quad (4.2)$$

where

- $W_{\text{Ig}^{\text{f}}}$: input layer \rightarrow forget gate
- $W_{\text{Hg}^{\text{f}}}$: hidden state \rightarrow forget gate
- $W_{\text{g}^{\text{c}}\text{g}^{\text{f}}}$: cell activation \rightarrow forget gate
- $b_{\text{g}^{\text{f}}}$: bias of the forget gate
- cell gate:

$$g_t^{\text{c}} = g_t^{\text{in}} \tanh(W_{\text{Ig}^{\text{c}}}x_t + W_{\text{Hg}^{\text{c}}}h_{t-1} + b_{\text{g}^{\text{c}}}) + g_t^{\text{f}} g_{t-1}^{\text{c}} \quad (4.3)$$

where

- $W_{\text{Ig}^{\text{c}}}$: input layer \rightarrow cell gate
- $W_{\text{Hg}^{\text{c}}}$: hidden state \rightarrow cell gate
- $b_{\text{g}^{\text{c}}}$: bias of the cell gate
- output gate:

$$g_t^{\text{out}} = \sigma(W_{\text{Ig}^{\text{out}}}x_t + W_{\text{Hg}^{\text{out}}}h_{t-1} + W_{\text{g}^{\text{c}}\text{g}^{\text{out}}}g_t^{\text{c}} + b_{\text{g}^{\text{out}}}) \quad (4.4)$$

where

- $W_{\text{Ig}^{\text{out}}}$: input layer \rightarrow output gate
- $W_{\text{Hg}^{\text{out}}}$: hidden state \rightarrow output gate
- $W_{\text{g}^{\text{c}}\text{g}^{\text{out}}}$: cell activation \rightarrow output gate
- $b_{\text{g}^{\text{out}}}$: bias of the output gate
- hidden state:

$$h_t = g_t^{\text{out}} \tanh(g_t^{\text{c}}) \quad (4.5)$$

The LSTM gates can prevent the rest of the network from modifying the contents of the memory cells for multiple timesteps [1].

4.2 Bidirectional LSTM

In order to train data, looking at the previous context and future context is important and has many applications such as speech recognition Bidirectional RNN (BRNN) considers all available input sequence in both the past and future for estimation of the output vector [11]. To enhance the capability of BRNNs through stacking hidden layers of LSTM cells in space, deep bidirectional LSTM (BLSTM) can be applied. BLSTM networks are more powerful than unidirectional LSTM networks [1]. This

means that the bidirectional nets and the LSTM nets did not take significantly more time to train per epoch than the unidirectional or RNN [10]. During computation, BLSTM includes all information of input sequences. Like BRNN, BLSTM model can solve the vanishing gradient problem and extend the model. But biggest difference between BRNN and BLSTM is related to their training time. The convergence of BRNN is more than eight times as long respect to BLSTM.

We consider an extended LSTM layer in multi-layer net, and the pseudocode for the forward pass is described.

4.2.1 Notation

- S input sequence
- τ time
- $x_k(\tau)$ network input to unit k at the time τ
- $y_k(\tau)$ the activation of the network input
- $E(\tau)$ output error at the time τ
- $t_k(\tau)$ training target at output unit k at time τ
- N set of all units (input units, bias units)
- W_{ij} weight from unit i to unit j
- i input gate
- ϕ forget gate
- ω output gate
- c elements of the set of cells C
- s_c state value of cell c
- f is a function of gate
- g cell input function
- h output function

Note that for each memory block, the LSTM equations are written, and these calculations can be repeated for each block. The error gradient is calculated with online BPTT, i.e., after every sequence BPTT shrink to input sequence length with the weight updates [10].

4.2.2 Forward Pass

- Re-adapt the activation to 0,
- Feed in the inputs and update the activation functions. All hidden layer and output activation functions at every timestep need to be stored,
- The activation functions are updated as:
Input Gates

$$x_t = \sum_{j \in N} \omega_{ij} y_j(\tau - 1) + \sum_{c \in C} \omega_{ic} s_c(\tau - 1)$$

$$y_t = f(x_t)$$

Forget Gates

$$x_\phi = \sum_{j \in N} \omega_{\phi j} y_j(\tau - 1) + \sum_{c \in C} \omega_{\phi c} s_c(\tau - 1)$$

5 Application of LSTM in Speech Recognition

5.1 Speech Recognition

Since RNNs are the structure through time and the signals of speech and audio change continuously over time, so RNNs can be an ideal model to learn features. Also speech recognition prediction uses the past and future sequential data, so BRNN is suitable in this field. Later applications of the connectionist temporal classification (CTC) function contributed to promote the RNNs in speech recognition. Connectionist temporal classification (CTC) is an objective function that allows an RNN to be trained for sequence transcription tasks without requiring any prior alignment between the input and target sequences [12]. CTC model has iterations like the sequence transducer and neural transducer. This property enables a second RNN to perform as a language model. This eventually leads to do the task such as online speech recognition. So by these argumentation, based on linguistic feature and prior transcriptions, the model can make the prediction.

5.2 Speech Emotion

Another application of RNNs is speech emotion. In this field, the segment of speech is organized as an emotion. Since in speech emotion recognition, the progress proceeds from the same way as that of speech recognition so the speech emotion recognition is much the same to speech recognition. Several methods have been proposed in speech application such as hidden Markov model (HMMs) and Gaussian mixture models (GMMs). With RNNs establishment, the trend of learning has improved. Because the networks were able to learn the features on their own. So RNN models have been applied for performing speech emotion recognition. LSTM-RNN has been successfully applied to speech recognition. Because in LSTM network, long-range dependencies are modeled better in order to capture the emotions. Also deep bidirectional LSTMs can capture more data through taking them in large number of frames.

5.3 *Speech Synthetic*

Another type of speech application is speech synthetic. In this field, long-term sequence learning is needed as well. HMM-based models and deep MLP neural networks can synthesize speech. However, these models have some problems. For example, in HMM-based models, statistical averaging during the training phase leads to overly smooth trajectories so the sound is not natural or MLP neural network takes each frame as an independent entity from its neighbors and fails to take into account the sequential nature of speech [13]. Introducing the RNNs in speech synthesis collaborates to leverage the sequential dependencies.

Speech synthesis also requires long-term sequence learning. HMM-based models can often produce synthesized speech, which does not sound natural. This is due to the overly smooth trajectories produced by the model, as a result of statistical averaging during the training phase [13].

Following that, LSTM performs better than RNNs. Also the ability of BLSTM model to integrate the relationship with neighboring frames in both future and past time steps [14, 15] make this model very effective in learning long-term sequential dependencies.

Acknowledgements The author would like to thank the University of Ontario Institute of Technology and the University of Manitoba for their support with this study.

References

1. Salehinejad, H., Sankar, S., Barfett, J., Colak, E., Valaee, S.: Recent Advances in Recurrent Neural Networks. [arXiv:1801.01078v3](https://arxiv.org/abs/1801.01078v3) [cs.NE, 22 Feb 2018]
2. Mikolov, T., Joulin, A., Chopra, S., Mathieu, M., Ranzato, M.: Learning longer memory in recurrent neural networks. [arXiv:1412.7753](https://arxiv.org/abs/1412.7753) (2014)
3. Haykin, S.: *Neural Networks: A Comprehensive Foundation*. Prentice Hall PTR (1994)
4. Yarotsky, D.: Error bounds for approximations with deep ReLU networks. *Neural Netw.* **94**, 103–114 (2017)
5. Ruder, S.: An overview of gradient descent optimization algorithms. [arXiv:1609.04747](https://arxiv.org/abs/1609.04747) (2016)
6. Sutskever, I., Martens, J., Dahl, G., Hinton, G.: On the importance of initialization and momentum in deep learning. In: *International Conference on Machine Learning*, pp. 1139–1147 (2013)
7. Gers, F.A., Schmidhuber, J., Cummins, F.: Learning to forget: continual prediction with LSTM (1999)
8. Le, V., Jaitly, N., Hinton, G.E.: A simple way to initialize recurrent networks of rectified linear units. [arXiv:1504.00941](https://arxiv.org/abs/1504.00941) (2015)
9. Graves, A., Schmidhuber, J.: Framewise phoneme classification with bidirectional LSTM and other neural network architectures. In: *IJCNN 2005 Conference Proceedings*. Published Under the IEEE Copyright (2014)
10. Graves, A., Schmidhuber, J.: Framewise phoneme classification with bidirectional LSTM and other neural network architectures *Neural Networks*, vol. 18, no. 5, pp. 602–610 (2005)
11. Schuster, M., Paliwal, K.K.: Bidirectional recurrent neural networks. **45**(11), 2673–2681 (1997)
12. Graves, A., Fernandez, S., Gomez, F., Schmidhuber, J.: Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. in: *Proceedings of the 23rd International Conference on Machine Learning*. ACM, pp. 369–376 (2006)

13. Fan, K., Wang, Z., Beck, J., Kwok, J., Heller, K.A.: Fast second order stochastic backpropagation for variational inference. In: *Advances in Neural Information Processing Systems*, pp. 1387–1395 (2015)
14. Fan, Y., Qian, Y., Xie, F.-L., Soong, F.K.: TTs synthesis with bidirectional LSTM based recurrent neural networks. In: *Fifteenth Annual Conference of the International Speech Communication Association* (2014)
15. Fernandez, R., Rendel, A., Ramabhadran, B., Hoory, R.: Prosody contour prediction with long short-term memory, bi-directional, deep recurrent neural networks. In: *Interspeech*, pp. 2268–2272 (2014)

A Soft Computing Approach for Predicting and Categorizing Learner's Performance Using Fuzzy Model



Sangita A. Jaju, Sudhir B. Jagtap, and Rohini Shinde

Abstract COVID-19: a pandemic situation in the world gives a turning point to education system and it becomes e-education. E-learning is emerging trend in digital era and empowerment of this trend is necessary. Traditional education system trying to adopt this new method of teaching and learning. But only teaching and learning is not sufficient in education system. We have to focus on learner and the environmental impact on them. Traditional education system unable to resolve all the issues arises due to obstacles such as understanding ability, thinking, mood and concentration. Proposed research work focusing on designing, developing and modelling of soft computing decision-making model for solving real-life problems and learners capability in education system. This research work uses fuzzy inference system (FIS) which is one of the applications in MATLAB software, for analysing learners result from obtained score and other factors related to environment. It also predicts the learner which is helpful in e-learning.

Keywords Fuzzy inference system · Bloom taxonomy · Soft computing

1 Introduction

Recently, the role of result and its analysis in education system is directly or indirectly affects in human life also it plays important role in categorizing students in different level of their learning abilities and grade. Performance evaluation is always challenges for teachers after training programme or learning process. Traditional result analysis system is worked on only scored marks by the students, averaging them and categories the students according to it. This method of averaging fails to highlight on learners 'critical thinking ability'. There are many more factors which effects on students result such as course content, difficulty level of content, time, mood,

S. A. Jaju (✉) · R. Shinde

Department of Computer Science, Dayanand Science College, Latur, Maharashtra 413512, India
e-mail: jaju.sangita@gmail.com

S. B. Jagtap

Swami Vivekanad Mahavidyalay, Udgir, Maharashtra, India

behaviour and teaching methodology. The traditional averaging method may not through light on understanding, evaluation or creation etc. Ability of student which is the standard levels for student's recognition. The grade of student must reflect their understanding of course learnt, its application in real-life problem or in similar issues. It must also reflect on how one can develop new solution by modifying existing one, etc. The final grade so obtained should give details scores which highlighting these thinking ability of student. Whenever such assessment is discussed with learner by teacher, the learner understands the natural thinking ability and strengthens the weak area for improvement by taking guidance from teacher.

Bloom's taxonomy was designed in 1965 by Dr. Benjamin Bloom [1] to improve learning standard in higher education. The taxonomy was categorizing in six levels including remembering, understanding, analysing, applying, evaluate and create. These six levels are in the increasing order of subject complexity and thinking. Now, it is globally recognized for testing skills and designing question paper to check thinking ability of learners.

In 1964, L. A. Zadeh has been invented the concept of fuzzy logic, and 1974 Mamdani developed first fuzzy logic controller which is used in predicting result when data is imprecise or vague. It is rule based and reliable. Using FIS soft computing tool, it is possible to predict the results and learning ability of the learners which is helpful for strong decision-making.

Proposed research work tries to overcome the drawbacks of conventional averaging method (CAM) of score for evaluating grade by using fuzzy averaging method (FAM). It also helps for predicting students for different learners group from their performance in different level of Bloom's taxonomy. The question paper is designed using different levels of taxonomy. The obtained score indicating the different level of questions attended by each learner from the question paper.

2 Literature Review

Fuzzy logic theory is also used in education system. Implementation of fuzzy logic for various activities of student's performance is now used many more. Student performance was evaluated using soft computing techniques in 1995. Deshmukh et al. [1, 2] were also proposed fuzzy logic-based model for evaluation of performance in network analysis course. Recently, fuzzy set for evaluation of student paper by matching answer script which becomes tedious work when records are in large number has been reported by Biswas. More generalized intelligent expert system was implemented using degree of satisfaction and extended fuzzy grade sheets [3, 4]. A cricketer performance evaluation model was presented to predict international rank of cricketer and also the effect of each parameter on performance is discussed [5]. A personalized student performance was presented in comparison with conventional statistical method and back propagation algorithm [6]. Every student is unique, and fuzzy system can evaluate student performance along with their learning progress using fuzzy system [7].

Fuzzy rules have been developed to model and evaluate the achievement of learning outcomes in information system courses [8]. It was proposed to use of fuzzy set technique to be applied in evaluation process to minimize complexity and ambiguity [9]. It has been investigated whether use of fuzzy logic is suitable for the resolution for fair assessment. It is applied for poster competition assessment for grading using fuzzy logic along with standard numerical grading. It observed that fuzzy grading method has many advantages over traditional method; it models to predict personalized student performance [10]. It also found in another study that evaluation with fuzzy logic renders more flexibility and robustness in evaluation process [11]. The fuzzy inference system has been used to obtain performance of student in different input values of student attendance, effective teaching and other facilities [12].

According to literatures survey, all researchers work motivate to develop a fuzzy expert system for predicting different learners using various learning level with taking decision for upgrade and better performance.

3 Research Experiment

3.1 Data Set

In our case, aptitude test data set is used. The test is conducted for final year student to check their logical reasoning and numerical analysis knowledge which is the base for selection in MNCs. The course is conducted prior to exam for providing related study. Learners must require the ideas of mathematical formulas, tricks, thinking power, logic, etc. The question paper is prepared for 50 marks with 50 questions, and each question carries 1 marks. The paper is designed with Bloom's taxonomy which covers all level of questions in near about equal number. The time limit for this test is one hour. Link is provided to student before 10 min of schedule time and test is conducted online. The data is collected after exam with their roll no., name, timestamp, solved questions, etc. For calculation purpose, the scored mark for each level is converted into out of 100 as given in the Table 2. In this test, total 86 learners are enrolled. We found total score of each learner in the test. Average percentage of learners in each level of questions is given in Table 1. The different levels of learning are Remembering level (RM), Understanding level (UN), Analysing level (AN), Apply level (AP), Evaluation level (EV) Creative level (CR).

The comparison of average mark with conventional method and fuzzy method is calculated (FIS output system is in the range of 0–10). The sample data of 12 learners is given in table (Table 2).

From Table 2, it can be seen that average mark calculated using fuzzy method is good agreement with average mark determined by conventional method.

Table 1 Average percentage of students in each level

Level	Percentage
RM average	72.09302
UN average	58.43023
AP average	56.10465
AN average	61.49871
EV average	54.65116
CR average	54.52196

Table 2 Sample of student data set

RM mark	UN mark	AP mark	AN mark	EN mark	CR mark	Average mark	Fuzzy method
							Average result Out of 10
13	25	0	11	25	22	16	1.47
50	25	25	11	38	11	26	2.97
13	25	50	33	13	11	24	2.88
50	75	38	44	13	11	38	3.46
63	50	50	67	25	11	44	5.00
25	38	13	33	25	22	26	2.94
25	13	38	22	13	33	24	2.94
13	63	38	11	25	44	32	4.00
88	88	63	89	88	100	86	8.00
38	13	50	44	0	11	26	2.00
100	100	88	100	100	100	98	9.36
88	88	88	100	100	100	94	9.36

4 Methodology

The proposed work designs set of rules for finding different learners category which depends on quality of their performance in aptitude test. The fuzzy expert system designed in proposed work as follows:

4.1 Crisp Value (Data)

The exam is conducted according to Bloom’s taxonomy level of thinking. The obtained score is the result of learner and which is analysed with FIS. The obtained

Table 3 Level (Criteria)-wise total questions and total marks in test

S. No.	Level	No. of questions	Total marks
1	Remember	8	80
2	Understand	8	80
3	Analyse	8	80
4	Apply	9	90
5	Evaluate	9	90
6	Create	8	80

values (of result) for each level are crisp value of input parameter for this experiment. As per taxonomy, there are six levels and which is the six input variables in this experimental work. Table 3 gives level-wise (Criteria wise) questions and marks.

4.2 Fuzzification (Fuzzy Input Value)

Membership functions (MF) are building blocks of fuzzy set theory. Such fuzziness in fuzzy set is determined by its MF. They may have different shape like triangular, trapezoidal and Gaussian, etc. Triangular or trapezoidal MF is formed using straight lines, and they have the advantage of simplicity. Due to their simple formulas and computational efficiency, both triangular MFs and trapezoidal MFs have been used extensively, especially in real-time implementations. The only condition for MF is that it must vary between 0 and 1.

4.3 Linguistic Values

In proposed work, fuzzification of six input variables (parameter) is done by linguistic values which are similar to verbal human language as Poor, Average, Good and Very good. Using trapezoidal membership function, each input parameter defined by lower limit ‘ $t1$ ’, lower support limit ‘ $t2$ ’, upper support limit ‘ $t3$ ’ and upper limit ‘ $t4$ ’, where $t1 < t2 < t3 < t4$. The following table (Table 4) gives lower and upper limit required for trapezoidal membership function. This gives the score in each category of linguistic variables. Those students score marks range 0–49 falls in poor category, marks range between 50 and 69.9 falls in average category, marks range 70–79.9 falls in good category and those who score marks above 80 falls in very good category as given in table (Table 4). For selection in MNC, minimum criteria for passing is 50%; hence, the range for poor category is up to 50 marks.

The degree of association of respective linguistic variables is represented in following Eq. (1).

Table 4 Upper and lower limits for each linguistic variable in trapezoidal function

Category	Lower limit ($t1$)	Lower support ($t2$)	Upper support ($t3$)	Upper limit ($t4$)
Poor	0	10	30	50
Average	40	50	60	70
Good	65	70	80	85
Very good	80	90	100	110

$$\text{trapezoid}(x; t1, t2, t3, t4) = \left\{ \begin{array}{ll} 0, & x \leq t1 \\ \frac{x-t1}{t2-t1}, & t1 \leq x \leq t2 \\ 1, & t2 \leq x \leq t3 \\ \frac{t4-x}{t4-t3}, & t3 \leq x \leq t4 \\ 0, & t4 \leq x \end{array} \right\} \tag{1}$$

From the given score and different limits for trapezoidal function, the process of fuzzification of six input variable is as follows.

Case Study: For example, to calculate membership value of obtained score x , if $x = 35$ marks scored in remember (RM) level which falls in poor category, then membership value can be calculated as

$$\mu_{\text{poor}}(x) = \frac{t4 - x}{t4 - t3}$$

$$\mu_{\text{poor}}(x) = \frac{50 - 35}{50 - 30}$$

$$\mu_{\text{poor}}(x) = 0.75$$

This is the membership value for poor score value **35**.

Membership function of input variable Understand (UN) is shown in Fig. 1. For remaining variables, such as Remember (RM), Analyse (AN), Apply (AP), Evaluate (EV) and Create (CR), the appearance of graph is same. From graph, the range of linguistic variable of input parameter is given in table (Table 5).

4.4 Development of Fuzzy Rules and Inference Mechanism

Fuzzy inference rules are used in inference process to relate the input and output membership functions. These rules use ‘If–Then’ statement with AND, OR and NOT operator and are flexible to construct the rules. They are formulated depending on the importance to be given to particular input parameter with some standard and expert system such as Bloom taxonomy in this proposed work. In proposed work, near about 4096 rules are formed from six input variables with each variables has

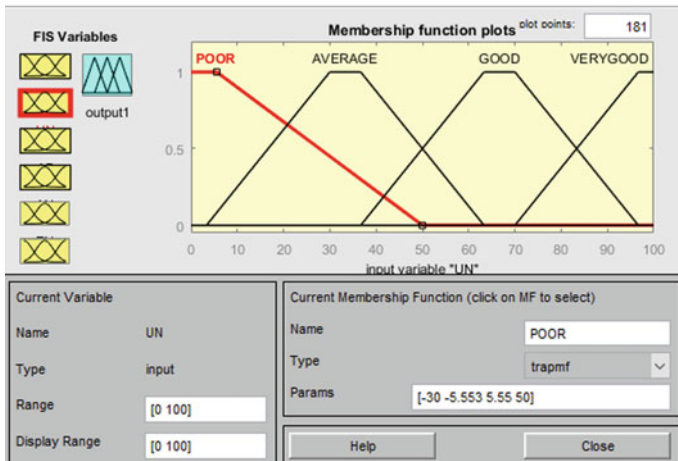


Fig. 1 Membership function plot for input variable Understand (UN)

Table 5 Score range of linguistic variables for each input parameter

Level	Poor	Average	Good	Very good
RM	< 50	10–65	40–95	>= 70
UN	< 50	10–65	40–95	>= 70
AN	< 50	10–65	40–95	>= 70
AP	< 50	10–65	40–95	>= 70
EV	< 50	10–65	40–95	>= 70
CR	< 50	10–65	40–95	>= 70

four linguistic values. Rules are nothing but various combination of values of input variables and corresponding output as $(4^6 = 4096)$. In this work, we use 54 rules for experimental purpose. Figure 2 shows interference of input to output in the form of rules.

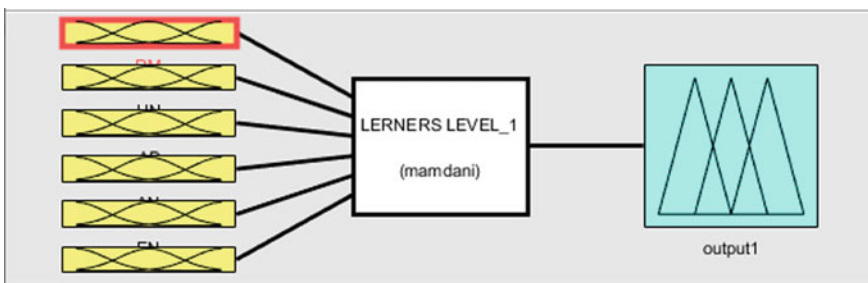


Fig. 2 Fuzzy inference system with input and output

4.5 Defuzzification of Fuzzy Output

Defuzzification is the process of representing a fuzzy set with a crisp number that is fuzzy value to precise value. The defuzzification method to find out final result as follows. There are various ways for defuzzification [13].

- a. Centre of Sums Method (COS)
- b. Centre of gravity (COG)/Centroid of Area (COA) method
- c. Centre of Area/Bisector of Area method (BOA)
- d. Weighted Average method
- e. Maxima methods.

The proposed method uses Centre of gravity (COG) method for defuzzification because this method determines the centre of area of fuzzy set and returns the corresponding crisp value. The resultant membership functions are developed by considering union of the output of each rule. It considers maximum area but overlapping area of fuzzy output is counted as one providing more result. It gives answer in more precision which tends to exactness [14]. The centroid defuzzification system is expressed as

$$Z_{\text{COG}} = \int_z^N \frac{\mu_A(x)xdx}{\mu_A(x)dx}$$

The output variable is OUTPUT1 (F) which shows different learners such as $F1$: Weak Learner (WL), $F2$: Slow Learner (SL), $F3$: Average Learner (AVGL), $F4$: Satisfactory Learner (SATL), $F5$: Fast Learner (FL) and $F6$: Extraordinary Learner (EOL). The different learners are decided by applying different rules on marks scored by student in each level of questions.

If six input variables are expressed as $f1, f2, f3, f4, f5$ and $f6$ and membership function of these six variables are $\mu(f1), \mu(f2), \mu(f3), \mu(f4), \mu(f5)$ and $\mu(f6)$, respectively, for rule $k = 1, 2, 3, \dots, r$, then the membership function of output variable F is given by Eq. (2) as [1, 2]

$$\mu(F) = \text{Max}_k[\min[\mu(f1), \mu(f2), \dots, \mu(f6)]], k = 1, 2, 3, \dots, r \quad (2)$$

This expression expresses the value of membership function for output variable overall performance for active rules for each input. The AND logical operator is used among the six input. Like linguistic variables for input, we have used linguistic variables for output also. Following table (Table 6) gives linguistic variables of output (i.e. different learners) and their mark range.

In Table 6, the ranges for linguistic values are different in each level because applying evaluating and creating levels are high-level thinking process in comparison with the previous three levels like Remember, Understand and Analyse. Bloom's taxonomy level is in increasing order of thinking and complexity hence [2]. Figure 3

Table 6 Different scores assigned for linguistic variables of output

Input Variables	Weak learner (WL)	Slow learner (SL)	Average learner (AL)	Satisfactory learner (SATL)	Fast learner (FL)	Extra ordinary learner (EOL)
Remember	< 3	3.0–3.9	3.0–6.0	5.5–7.9	7.5–8.9	>= 9.0
Understand	< 3	3.0–3.9	3.0–6.0	5.5–7.9	7.5–8.9	>= 9.0
Analyse	< 3	3.0–3.9	3.0–6.0	5.5–7.9	7.5–8.9	>= 9.0
Apply	< 3	3.0–3.9	3.0–6.0	5.5–7.9	7.5–8.4	>= 8.5
Evaluate	< 3	3.0–3.9	3.0–6.0	5.5–7.9	7.5–8.4	>= 8.5
Create	< 3	3.0–3.9	3.0–6.0	5.5–7.9	7.5–8.0	>= 8.0

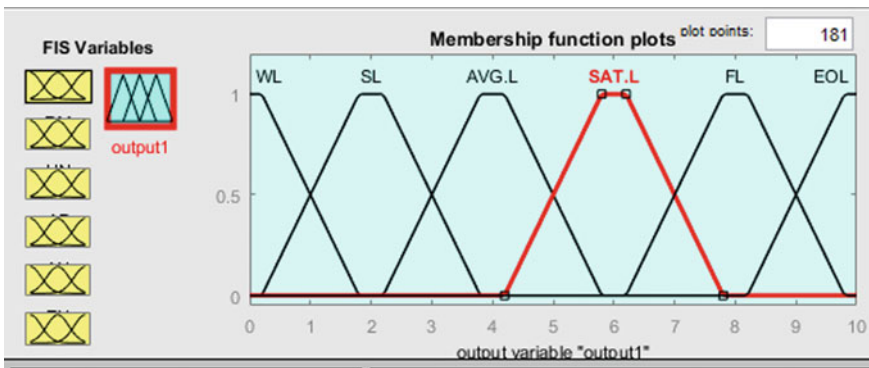


Fig. 3 Membership function plot for output as Output1

shows trapezoidal membership function plot for output which shows different learners. For example, the red coloured part in graph shows satisfactory learners which has linguistic values from 5.5 to 7.9. Its membership value is maximum, i.e. one for linguistic value 5.8 to 6.2 and changes from 0 to 1 for rest of the range.

4.6 Proposed Algorithm

Depending on different level of learning as input parameters learners learning capability is categorized in 6 categories (group) as EOL, FL, SATL, AL, SL and WL which is on the basis of the fuzzy modelling. In order to design a user friendly environment model for evaluating student learning capability, researcher proposes following algorithm.

- Step 1. Categories imprecise and vague values in ‘r’ inputs. (Say $r = 6$ two inputs)
- Step 2. Similarly, categories output as ‘n’ output. (Say $n = 1$ one output)

- Step 3. Divide each fuzzy set in k_i linguistic term. [As in proposed work first input parameter is Remembering (r_{1i}) where $i = 1 \dots 4$, i_1 (poor), i_2 (average), i_3 (Good), i_4 (Very good) score. Second input parameter is Understanding (r_{2j}) where $j = 1 \dots 4$, j_1 (poor), j_2 (average), j_3 (Good), j_4 (Very good) score and so on for remaining input parameter.
- Step 4. Form the Cartesian product L_{ij} of $r_i \times r_j$ for forming different linguistic string.
- Step 5. Apply trapezoidal membership function for each linguistic term in each fuzzy set using the available data.
- Step 6. Develop $m = k_1, k_2 \dots k_r$ linguistic strings denoted as JK using AND operation on each linguistic term L_{ij} , which is taken from each fuzzy set X_i for $i = 1 \dots r$ and $j = 1 \dots k$.

(For ex. J1 = poorRM GoodUN poorAP poorAN verygoodEV VerygoodCR,)
 J2 = avgRM avgUN avgAP avgAN goodEN goodCR and so on.)

- Step 7. Represent the state of system by fuzzy set A , in which value of each linguistic string is the minimum of all linguistic term and corresponding linguistic string.
- Step 8. With help of fuzzy expert system develop all possible if-then rules. (For ex. In proposed method, 4096 rules formed.)
- Step 10. Construct utility set for lower bound of each range as UL.
- Step 11. Construct utility set for upper bound of each range as UU.
- Step 12. Develop an optimal fuzzy utility set UOI where $I = 1, 2 \dots n$ (for n linguistic term of output) which is obtained by fuzzy intersection (\cap) on fuzzy utility set of upper bound and lower bound range.

$$\mu_{UOI}(x) = \mu_{UL}(x) \cap \mu_{UU}(x) = \min(\mu_{UL}(x), \mu_{UU}(x))$$

For each utility value x .

- Step 13. Take maximum membership value from each optimal utility set and develop a BO such as $B_o = \max(\mu_{UOI}(x), B_i) \forall \in U_{OI}$ for $I = 1, 2 \dots n$.
- Step 14. The degree of match method is applied to know the satisfactory factor for given input and computed outputs.
- Step 15. Obtain value of degree of match DM_i of input $I = 1, 2 \dots r$ between given precise value of input x_i and fuzzy input set X_i given as $DM = 2\mu_{x_i}((x_i) - 1)$.
- Step 16. Take minimum from all degree of match DM_i for $I = 1, 2 \dots r$ which is denoted by total degree of match as DM_T of input

$$DM_T = \min(DM_1, DM_2 \dots DM_r)$$

- Step 17. Also find the degree of match (from best degree of match) DM_o of alternative as $DM_o = 2(\mu_{x_i}(x_i) - 1)$.

- Step 18. Calculate difference between total degree of match of input and optimal alternative as $D = |DM_T - DM_O|$, if difference between total degree of match of input and optimal degree of match of input is near to zero, then level of satisfaction increases towards for finding student's learning capability and style.

From Table 7, it is indicated that proposed algorithm gives the Total Degree of Match (DMT) obtained from step 16, Best optimal (Do) from step 13, final result ($DM = ABS(DMT - DO)$). Step 18. For confirmation of result obtained by algorithm, it was verified with the results obtained by the rules defined in fuzzy inference system (Table 6) which is indicated with predicated output in Table 7. According to the Table 7, FIS and Algorithm results are matched. Table 7 gives the increased level of satisfaction proves that the predicted learner category and calculated learner's category are perfectly matched.

Table 7 shows that proposed algorithm is verified with the values of different case studies and results obtained by the fuzzy inference rules.

5 Results and Discussion

In proposed method, we studied rule-based classifiers like fuzzy which gives 100% accuracy for classification. The crisp values of fuzzification result are compared with classical or conventional averaging method. The Mamdani fuzzy system is used with input variables RM, UN, AP, AN, EV, as shown in Fig. 2. The output variables indicate the performance of learner's in linguistic variables as Poor < 50, Average (50–69.9), Good (70–79.9) and Very Good (above 80) and which are given in table (Table 8).

The crisp values for output for different learners category are given in Table 9. As output has six linguistic values and they are nothing but six types of learner's category and their range of crisp value is also different. If the crisp value is below 3, then the learners are Weak Learner (WL) category, if the crisp value is in the range 3.0–3.9, then the learners are Slow Learners (SL), Average Learners (AVGL) had the range between 3.0 and 6.0, for Satisfactory Learner (SATL), the range of crisp value is 5.5–7.9, for Fast Learner (FL), it is in between 7.5 and 8.9 and for Extra ordinary Learner (EOL), it is above 9.

Rule viewer of the proposed fuzzy expert system for the evaluation of overall student's performance is shown in Fig. 4. It also shows crisp value of output for respective crisp value of input parameter.

Table 1 in data set section gives average percentage of student who solved questions and scored marks in each level. From the table, it observed that the scored percentage of Remember level questions are more and are solved correctly by 72.09% which is the highest in all other level. The average percentage of understanding level is too low which is 58.43% in comparison with remembering level, and hence, their analysing and applying level are also average. Average percentage of questions solved in evaluation and creation level is also weak as the level above it such as understanding, applying and analysing levels are average.

Table 7 Comparison of calculated and predicted result

S. No.	Total degree of match	Predicted output	Best optimal	Final result	Satisfaction level
	$DM_T = \text{MIN}(D1, D2, D3, D4, D5, D6)$		D_o	$DM = \text{ABS}(DM_T - D_o)$	
1	0	WL	0.5	0.5	Increased
2	0.6	WL	0.5	0.1	Increased
3	0.85	WL	0.5	0.35	Increased
4	0.3	AVGL	0.6	0.3	Increased
5	0	WL	0.5	0.5	Increased
6	0.6	WL	0.5	0.1	Increased
7	0.85	WL	0.5	0.35	Increased
8	0.3	SL	0.6	0.3	Increased
9	0.3	SL	0.6	0.3	Increased
10	0.6	WL	0.5	0.1	Increased
11	0.6	WL	0.5	0.1	Increased
12	0.3	WL	0.5	0.2	Increased
13	0.7	EOL	0.927	0.227	Increased
14	0.3	WL	0.5	0.2	Increased
15	0.8	EOL	0.927	0.127	Increased
16	0.8	EOL	0.927	0.127	Increased
17	0.7	FL	0.822	0.122	Increased
18	0.8	FL	0.822	0.022	Increased
19	0	WL	0.5	0.5	Increased
20	0.7	AVGL	0.6	0.1	Increased
21	0.6	SL	0.6	0	Increased
22	0.7	STAL	0.8	0.1	Increased
23	0.8	EOL	0.927	0.127	Increased
24	0.7	STAL	0.8	0.1	Increased
25	0.30	FL	0.822	0.522	Decreased
26	0.70	AVGL	0.6	0.1	Increased
27	0.80	EOL	0.927	0.127	Increased
28	1.00	AVGL	0.6	0.4	Increased
29	0.90	EOL	0.927	0.027	Increased
30	0.80	EOL	0.927	0.127	Increased

Table 8 Crisp value of final score

Final score of (Linguistic variables)	Poor	Average	Good	Very good
Crisp value	< 50	50–69.9	70–79.9	>= 80

Table 9 Crisp value of different learners

Output	Weak learner	Slow learner	Average learner	Satisfactory learner	Fast learner	Extra ordinary learner
Crisp value	< 3	3.0–3.9	3.0–6.0	5.5–7.9	7.5–8.9	>= 9.0

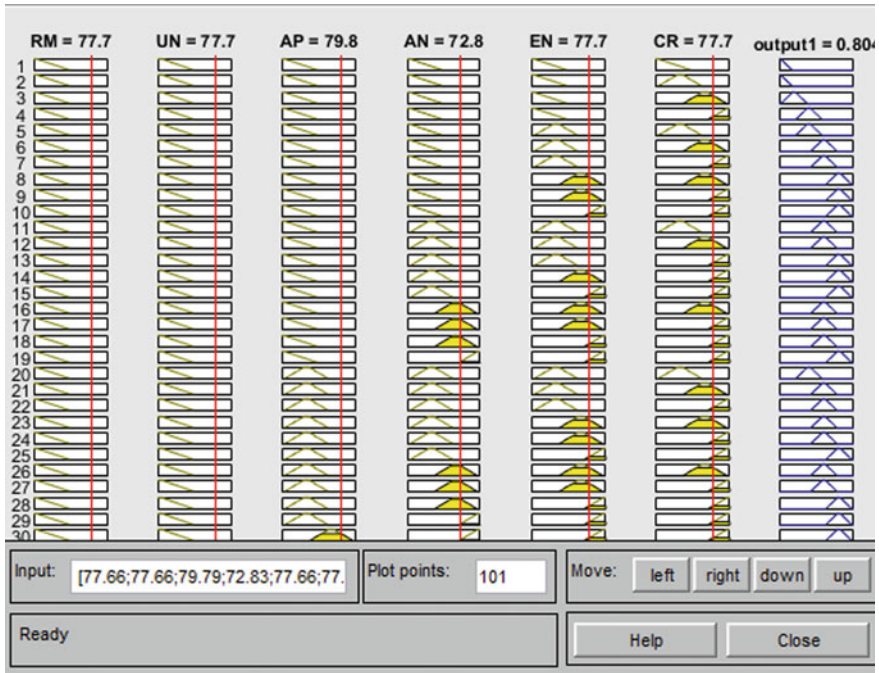


Fig. 4 Rule viewer of fuzzy inference system

Table 2 in data sets gives only sample data from population, i.e. total data of 151 cases. In the table, we find the average marks of each student by conventional averaging method and fuzzy averaging method. Both the classifiers can be used to evaluate the student performance depending upon the classification and time taken to classify. The fuzzy method gives 100% classification accuracy, and the results are compared with conventional method which are near about same, while in some cases, they differ. Though the both method gives similar answer but if data is large, the rules are more and training period is more which leads to more complexity, in such situation, fuzzy method is more comfortable and reliable for performance evaluation. It is also possible to use other soft computing tools such as neuro fuzzy system etc.

Table 10 *t*-Test: Two-sample assuming equal variances (For averaging score)

	Conventional method	Fuzzy method
Mean	4.45409	4.531667
Variance	8.934879	7.84747
Observations	12	12
Pooled variance	8.391174	
Hypothesized mean difference	0	
Df	22	
<i>t</i> Stat	-0.0656	
$P(T \leq t)$ two-tail	0.948289	
<i>t</i> Critical two-tail	2.073873	

5.1 Null Hypothesis Verification

For the validity of data analysis and calculated result, null hypothesis is applied. The result is as given in Table 10.

From Table 10, it observe that when *t*-test is conducted for checking of acceptance of null hypothesis, we get *p* value of *t*-test statistics is 0.948 (< 0.975) and *t*-statistics is -0.0656 which does not fall in rejection region.. Hence, null hypothesis is accepted. That means conventional result of averaging is equal the mean fuzzy system result. Therefore, one can apply computer-based fuzzy system approach for averaging instead of conventional method of mean which is time consuming. As the cases are increasing and more rules are formed, the complexity increases and in such situation also the fuzzy methods are more suitable.

6 Conclusion

As the fuzzy method results are compared with conventional method of averaging by applying null hypothesis and *t*-test method and it found that fuzzy averaging method gives near about 100% classification accuracy. It also found that fuzzy methods are faster than traditional method when criterion is more complex. For large amount of data, the rules and training period are more which leads to more complexity. In such cases, fuzzy method is more comfortable for performance evaluation. According to the Bloom's taxonomy, the proposed research work learner's capability is classified in RM, UN, AN, AP, EV and CR. It is also clear that remembering power of students is good but there is necessity for improving understanding, analysing and applying power of student for better evaluation and creation level. Upgrading this level is one of the tasks for student as well as teachers. Using FIS, it is possible to find in between phases and helps to upgrade lower phase to upper phase and minimize the gap between different learners. In teaching learning system, evaluation of learners

is an important phase which actual gives the proper direction to teacher to motivate, promote and upgrade the learners. Conventional method evaluates the learners depending on the marks scored, whereas proposed FIS method evaluates the learners depending on the learning capability and performance in different learning level. From the experimental result, it is found that the proposed method gives 97.36% accuracy which is highly considerable. Furthermore, we accept the null hypotheses that means conventional result is equal to the mean fuzzy system result with 95% confidence level. Hence, the computer-based fuzzy expert system is better method than conventional method in plane of time consuming conventional method and it is the requirement of present educational system.

References

1. Deshmukh, V.B., Mangalwede, S.R., Rao, D.H.: Student performance evaluation model based on scoring rubric tool for network analysis subject using fuzzy logic. In: International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICECCOT), pp. 1–5. IEEE (2017)
2. Deshmukh, V.B., Mangalwede, S.R., Rao, D.H.: Student performance evaluation model based on Bloom's Taxonomy using fuzzy logic. In: In International Conference on Electrical, Electronics, Communication, Computer, Mechanical and Computing (EECCMC). IEEE (2018)
3. Biswas: An application of fuzzy sets in students' evaluation. *Fuzzy Sets Syst.* **1995**, 187–194 (1995)
4. Chen, S.-M., Lee, C.-H.: New methods for students' evaluation using fuzzy sets. *Fuzzy Sets Syst.* **1999**, 209–218 (1999)
5. Singh, G., Bhatia, N., Singh, S.: Fuzzy logic based cricket player performance evaluator. In: IJCA Special Issue on "Artificial Intelligence Techniques—Novel Approaches & Practical Applications", pp. 11–16 (2011)
6. Arora, N., Saini, J.K.R.: A fuzzy probabilistic neural network for student's academic performance prediction. *Int. J. Innov. Res. Sci. Eng. Technol.* **2**, 4425–4432 (2013)
7. Jeremic, Z., Jovonovic, J., Gasevic, D.: Student modelling and assessment in intelligent tutoring software patterns. *Expert Syst. Appl.* **2012**, 210–212 (2012)
8. Bouslama, F., Lansari, A., Al-Rawi, A.: Fuzzy Rules in Assessing Student Learning Outcomes. American Society for Engineering Education (2006)
9. Mossin, E.A., Pantoni, R.P., Brandão, D.: Students' Evaluation based on Fuzzy Sets Theory, Fuzzy Systems (Azar, A.T. (Ed.)). InTech, ISBN: 978-953-7619-92-3. <https://doi.org/10.5772/7225>. Available from: <http://www.intechopen.com/books/fuzzy-systems/students-evaluation-based-on-fuzzy-sets-theory>
10. McLoone, S.C.: ISSC 2012. On Using Fuzzy Logic for Grading Highly Subjective Assessment Material—A Case Study. NUI May–June 28–29, 2012
11. Jamsandekar, S.S., Mudholkar, R.R.: Performance evaluation by fuzzy inference technique. *Int. J. Soft Comput. Eng. (IJSCE)* 158–164 (2013)
12. Upadhyya, M.S.: Fuzzy logic based evaluation of performance of students in colleges. *J. Comput. Appl. (JCA)* 6–9 (2012)
13. Naaz, S., Alam, A., Biswas, R.: Effect of different defuzzification methods in a fuzzy based load balancing application. *IJCSI Int. J. Comput. Sci. Issues* **8**(5), 261–267 (2011)
14. Wang, K.: Computational Intelligence in Agile Manufacturing Engineering. Department of Production and Quality Engineering, Norwegian University of Science and Technology, N-7491 Trondheim, Norway. <https://doi.org/10.1016/B978-008043567-1/50016-4>

A Fuzzy Logic-Based Approach to Solve Interval Multi-objective Non-linear Transportation Problem: Suggested Modifications



Tanveen Kaur Bhatia, Amit Kumar, M. K. Sharma, and S. S. Appadoo

Abstract To the best of author's knowledge, there does not exist any other approach to solve an interval multi-objective non-linear transportation problem having unknown interval demand (a multi-objective non-linear transportation problem in which all the parameters are represented by intervals and the interval demand of the product at each destination is unknown) except Dalman and Sivri's approach (Proc. Natl. Acad. Sci. India, Sect. A Phys. Sci. (2019) 89(2): 279–289). In the future, other researchers may use Dalman and Sivri's approach in real-life problems. However, it is observed that in one of the steps of Dalman and Sivri's approach, a mathematical incorrect assumption has been considered. Hence, it is not appropriate to use Dalman and Sivri's approach in its present form. The aim of this paper is to make the researchers aware about the mathematical incorrect assumption as well as to suggest the required modifications in Dalman and Sivri's approach.

Keywords Interval numbers · Transportation problem · Non-linear programming problem · Multi-objective programming problem

1 Introduction

Transportation problem is a well-known topic of Operations Research. A transportation problem may be classified into the following two major categories:

- (i) Single-objective transportation problem
- (ii) Multi-objective transportation problem.

The aim of a single-objective transportation problem is to determine the optimal quantity of the product that should be supplied from various sources to various

T. K. Bhatia · A. Kumar · M. K. Sharma
School of Mathematics, Thapar Institute of Engineering and Technology (Deemed to Be University), Patiala, Punjab 147004, India

S. S. Appadoo (✉)
Department of Supply Chain Management, Asper School of Business, University of Manitoba, Winnipeg, Canada
e-mail: SS.Appadoo@umanitoba.ca

destinations to minimize/maximize a single-objective function like to minimize the total transportation cost or to maximize the total profit. Similarly, the aim of a multi-objective transportation problem is to determine the optimal quantity of the product that should be supplied from various sources to various destinations to minimize/maximize more than one objective functions like to minimize the transportation cost and to minimize the transportation risk simultaneously.

In general, to find the optimal solution of a single/multi-objective transportation problem, it is assumed that the cost for transporting one unit quantity of the product from each source to each destination, the availability of the product at each source and the demand of the product at each destination are known and these parameters can be represented by non-negative real numbers. However, this assumption is not realistic. For example,

- (i) The transportation cost between two fixed places may change due to unavoidable circumstances like traffic jam, bad weather conditions, etc., and hence, it is not always possible to represent the cost for transporting one unit quantity of the product from each source to each destination by a real number.
- (ii) The availability of the product may change due to unavoidable circumstances like occurrence of natural calamities such as flood and drought, and hence, it is not always possible to represent the availability of the product by a real number.
- (iii) The demand of the product may change due to unavoidable circumstances like change in economy, change in price of the product, discount rates, change in weather conditions, etc., and hence, it is not always possible to represent the demand of the product by a real number.

Due to the same reason, in the literature [1–6], intervals have been used to represent some or all of the parameters of single/multi-objective transportation problems (named as interval single/multi-objective transportation problems). It is pertinent to mention that the existing approaches [1–5] have been proposed by considering the following assumptions:

- (i) The interval demand of the product at each destination is known.
- (ii) The quantity of the product supplied from a source will be equal to the quantity of the product received at the destinations.
- (iii) The objective function/functions is/are only depending upon the quantity of the product supplied from various sources to various destinations.

While the existing approach [6] has been proposed by considering the following assumptions:

- (i) The interval demand of the product at each destination is unknown.
- (ii) The quantity of the product supplied from a source will not necessarily be equal to the quantity of the product received at the destinations.
- (iii) The objective function/functions is/are depending upon the quantity of the product supplied from various sources to various destinations as well as the unknown interval demand at each destination.

Several researchers have proposed methods [7–21] to solve different types of transportation problems under fuzzy environment and its various extensions.

But, only the existing approach [6] can be used to solve an interval multi-objective non-linear transportation problem having unknown interval demand. Therefore, in the future, one may use the existing approach [6] to solve a real-life interval multi-objective non-linear transportation problem having unknown interval demand.

However, after a deep study of the existing approach [6], it is observed that it is inappropriate to use the existing approach [6] in its present form. The aim of this paper is to suggest the required modifications in the existing approach [6].

This paper is organized as follows: In Sect. 2, the existing approach [6] has been presented in a detailed manner. Also, the existing approach [6] has been illustrated with the help of an existing interval bi-objective non-linear transportation problem having unknown interval demand. In Sect. 3, the mathematical incorrect assumption, considered in the existing approach [6], has been pointed out. In Sect. 4, the required modifications have been suggested. In Sect. 5, the exact results of an existing interval bi-objective non-linear transportation problem having unknown interval demand [6] have been obtained. Section 6 concludes the paper.

2 Dalman and Sivri's Approach With an Illustrative Example

The aim of this paper is to make the researchers aware about a mathematical incorrect assumption, considered in one of the steps of Dalman and Sivri's approach [6] as well as to suggest the required modifications. Since, to achieve this aim, there is a need to discuss Dalman and Sivri's approach [6] as well as to illustrate Dalman and Sivri's approach [6] with the help of a numerical example. Therefore, in this section, Dalman and Sivri's approach [6] has been presented in a detailed manner. Also, Dalman and Sivri's approach [6] has been illustrated with the help of an existing interval bi-objective non-linear transportation problem having unknown interval demand.

2.1 Dalman and Sivri's Approach

Dalman and Sivri [6] have proposed the following approach to solve the interval multi-objective non-linear programming problem (P_1) corresponding to an interval multi-objective transportation problem having unknown interval demand.

Problem (P1)

$$\min \left\{ \begin{aligned} & \sum_{i=1}^m \sum_{j=1}^n [c_{ij}^e, \bar{c}_{ij}^e] [x_{ij}, \bar{x}_{ij}] + \sum_{j=1}^n [A_j^e, \bar{A}_j^e] [x_j^2, \bar{x}_j^2] \\ & + \sum_{j=1}^n [-B_j^e, -\bar{B}_j^e] [x_j, \bar{x}_j] + \sum_{j=1}^n [D_j^e, \bar{D}_j^e]; e = 1, 2, \dots, k \end{aligned} \right\}$$

Subject to

$$\sum_{i=1}^m [r_{ij}, \bar{r}_{ij}] [x_{ij}, \bar{x}_{ij}] = [x_j, \bar{x}_j], \quad j = 1, 2, \dots, n;$$

$$\sum_{j=1}^n [x_{ij}, \bar{x}_{ij}] \leq [a_i, \bar{a}_i], \quad i = 1, 2, \dots, m;$$

$$x_{ij}, x_j \geq 0, x_{ij} \leq \bar{x}_{ij}, x_j \leq \bar{x}_j, \quad i = 1, 2, \dots, m; \quad j = 1, 2, \dots, n.$$

where

- (i) m : number of sources.
- (ii) n : number of destinations.
- (iii) $[x_j, \bar{x}_j]$: the unknown interval demand of the product at the j th destination.
- (iv) $[x_{ij}, \bar{x}_{ij}]$: the interval quantity of the product that should be supplied from the i th source to the j th destination.
- (v) $[r_{ij}, \bar{r}_{ij}] [x_{ij}, \bar{x}_{ij}]$: the interval quantity of the product received at the j th destination.
- (vi) $[c_{ij}, \bar{c}_{ij}]$: the interval cost for transporting one unit quantity of the product from the i th source to the j th destination.
- (vii) $[a_i, \bar{a}_i]$: the interval quantity of the product available at the i th source.
- (viii) $\sum_{j=1}^n [A_j^l, \bar{A}_j^l] [x_j^2, \bar{x}_j^2] + \sum_{j=1}^n [-B_j^l, -\bar{B}_j^l] [x_j, \bar{x}_j] + \sum_{j=1}^n [D_j^l, \bar{D}_j^l]$: convex interval quadratic function of the unknown interval demand $[x_j, \bar{x}_j]$.

Step 1: Split the interval multi-objective non-linear programming problem (P_1) into its equivalent k -interval single-objective non-linear programming problems (P_{1e}); $e = 1, 2, \dots, k$.

Problem (P_{1e})

$$\min \left\{ \begin{aligned} & \sum_{i=1}^m \sum_{j=1}^n [c_{ij}^e, \bar{c}_{ij}^e] [x_{ij}, \bar{x}_{ij}] + \sum_{j=1}^n [A_j^e, \bar{A}_j^e] [x_j^2, \bar{x}_j^2] + \sum_{j=1}^n [-B_j^e, -\bar{B}_j^e] [x_j, \bar{x}_j] \\ & + \sum_{j=1}^n [D_j^e, \bar{D}_j^e] \end{aligned} \right\}$$

Subject to.

Constraints of the problem (P_1).

Step 2: Using the relation $[a, b][c, d] = \begin{cases} [ac, bd], & \text{if } a, b, c, d \geq 0 \\ [ad, bc], & \text{if } a, b \leq 0, c, d \geq 0 \end{cases}$, the e^{th} interval single-objective non-linear programming problem (P_{1e}); $e = 1, 2, \dots, k$ can be transformed into its equivalent e^{th} interval single-objective non-linear programming problem (P_{2e}); $e = 1, 2, \dots, k$.

Problem (P_{2e})

$$\min \left\{ \begin{aligned} & \sum_{i=1}^m \sum_{j=1}^n [c_{ij}^e x_{ij}, \bar{c}_{ij}^e \bar{x}_{ij}] + \sum_{j=1}^n [A_j^e x_j^2, \bar{A}_j^e \bar{x}_j^2] + \sum_{j=1}^n [-B_j^e \bar{x}_j, -\bar{B}_j^e x_j] \\ & + \sum_{j=1}^n [D_j^e, \bar{D}_j^e] \end{aligned} \right\}$$

Subject to.

$$\left[\sum_{i=1}^m r_{ij} \underline{x}_{ij}, \sum_{i=1}^m \bar{r}_{ij} \bar{x}_{ij} \right] = [\underline{x}_j, \bar{x}_j], \quad j = 1, 2, \dots, n;$$

$$\left[\sum_{j=1}^n \underline{x}_{ij}, \sum_{j=1}^n \bar{x}_{ij} \right] \leq [a_i, \bar{a}_i], \quad i = 1, 2, \dots, m;$$

$$\underline{x}_{ij}, \underline{x}_j \geq 0, \underline{x}_{ij} \leq \bar{x}_{ij}, \underline{x}_j \leq \bar{x}_j, \quad i = 1, 2, \dots, m; j = 1, 2, \dots, n.$$

Step 3: Using the relation $\sum_{i=1}^m [a_i, \bar{a}_i] = [\sum_{i=1}^m a_i, \sum_{i=1}^m \bar{a}_i]$, the e^{th} interval single-objective non-linear programming problem (P_{2e}); $e = 1, 2, \dots, k$ can be transformed into its equivalent e^{th} interval single-objective non-linear programming problem (P_{3e}); $e = 1, 2, \dots, k$.

Problem (P_{3e})

$$\min \left(\left[\begin{aligned} & \sum_{i=1}^m \sum_{j=1}^n c_{ij}^e x_{ij} + \sum_{j=1}^n A_j^e x_j^2 + \sum_{j=1}^n (-B_j^e \bar{x}_j) + \sum_{j=1}^n D_j^e, \sum_{i=1}^m \sum_{j=1}^n \bar{c}_{ij}^e \bar{x}_{ij} \\ & + \sum_{j=1}^n \bar{A}_j^e \bar{x}_j^2 + \sum_{j=1}^n (-\bar{B}_j^e x_j) + \sum_{j=1}^n \bar{D}_j^e \end{aligned} \right] \right)$$

Subject to

Constraints of the problem (P_{2e}).

Step 4: Using the relation $[a, b] \leq [c, d] \Rightarrow b \leq d$ and $\frac{a+b}{2} \leq \frac{c+d}{2}$, the e th interval single-objective non-linear programming problem (P_{3e}); $e = 1, 2, \dots, k$ can be transformed into its equivalent e th crisp bi-objective non-linear programming problem (P_{4e}); $e = 1, 2, \dots, k$.

Problem (P_{4e})

$$\min \left(\sum_{i=1}^m \sum_{j=1}^n \bar{c}_{ij}^e(\bar{x}_{ij}) + \sum_{j=1}^n \bar{A}_j^e(\bar{x}_j^2) + \sum_{j=1}^n (-\bar{B}_j^e(\underline{x}_j)) + \sum_{j=1}^n \bar{D}_j^e \right)$$

$$\min \left(\begin{aligned} & \sum_{i=1}^m \sum_{j=1}^n \left(\frac{c_{ij}^e(x_{ij})}{2} + \frac{\bar{c}_{ij}^e(\bar{x}_{ij})}{2} \right) + \sum_{j=1}^n \left(\frac{A_j^e(x_j^2)}{2} + \frac{\bar{A}_j^e(\bar{x}_j^2)}{2} \right) \\ & + \sum_{j=1}^n \left(\frac{-\bar{B}_j^e(x_j)}{2} + \frac{(-B_j^e(\bar{x}_j))}{2} \right) + \sum_{j=1}^n \left(\frac{D_j^e}{2} + \frac{\bar{D}_j^e}{2} \right) \end{aligned} \right)$$

Subject to

$$\sum_{i=1}^m r_{ij} x_{ij} = x_j, \quad j = 1, 2, \dots, n;$$

$$\sum_{i=1}^m \bar{r}_{ij} \bar{x}_{ij} = \bar{x}_j, \quad j = 1, 2, \dots, n;$$

$$\sum_{j=1}^n \bar{x}_{ij} \leq \bar{a}_i, \quad i = 1, 2, \dots, m;$$

$$\sum_{j=1}^n \frac{x_{ij} + \bar{x}_{ij}}{2} \leq \frac{a_i + \bar{a}_i}{2}, \quad i = 1, 2, \dots, m;$$

$$\underline{x}_{ij}, \underline{x}_j \geq 0, \underline{x}_{ij} \leq \bar{x}_{ij}, \underline{x}_j \leq \bar{x}_j, \quad i = 1, 2, \dots, m, j = 1, 2, \dots, n.$$

Step 5: Split the e th crisp bi-objective non-linear programming problem (P_{4e}); $e = 1, 2, \dots, k$ into two crisp single-objective non-linear programming problems (P_{4e1}); $e = 1, 2, \dots, k$ and (P_{4e2}); $e = 1, 2, \dots, k$, respectively.

Problem (P_{4e1})

$$\min \left(\sum_{i=1}^m \sum_{j=1}^n \bar{c}_{ij}^e(\bar{x}_{ij}) + \sum_{j=1}^n \bar{A}_j^e(\bar{x}_j^2) + \sum_{j=1}^n (-\bar{B}_j^e(\underline{x}_j)) + \sum_{j=1}^n \bar{D}_j^e \right)$$

Subject to

Constraints of the problem (P_{4e}).

Problem (P_{4e2})

$$\min \left(\begin{aligned} & \sum_{i=1}^m \sum_{j=1}^n \left(\frac{c_{ij}^e(x_{ij})}{2} + \frac{\bar{c}_{ij}^e(\bar{x}_{ij})}{2} \right) + \sum_{j=1}^n \left(\frac{A_j^e(x_j^2)}{2} + \frac{\bar{A}_j^e(\bar{x}_j^2)}{2} \right) \\ & + \sum_{j=1}^n \left(\frac{-\bar{B}_j^e(x_j)}{2} + \frac{(-B_j^e(\bar{x}_j))}{2} \right) + \sum_{j=1}^n \left(\frac{D_j^e}{2} + \frac{\bar{D}_j^e}{2} \right) \end{aligned} \right)$$

Subject to

Constraints of the problem (P_{4e}).

Step 6: Find an optimal solution and the optimal value \bar{f}_e ; $e = 1, 2, \dots, k$ of the e th crisp single-objective non-linear programming problem (P_{4e1}); $e = 1, 2, \dots, k$.

Step 7: Find an optimal solution of the e^{th} crisp single-objective non-linear programming problem (P_{4e2}); $e = 1, 2, \dots, k$.

Step 8: Find the value of the e th function $f_e = \sum_{i=1}^m \sum_{j=1}^n c_{ij}^e(x_{ij}) + \sum_{j=1}^n A_j^e(x_j^2) + \sum_{j=1}^n (-B_j^e(\bar{x}_j)) + \sum_{j=1}^n D_j^e$; $e = 1, 2, \dots, k$ corresponding to the optimal solution of e th single-objective non-linear programming problem (P_{4e2}); $e = 1, 2, \dots, k$.

Also, using the obtained value of f_e ; $e = 1, 2, \dots, k$ and the obtained value of \bar{f}_e ; $e = 1, 2, \dots, k$, find the interval optimal value $[f_e, \bar{f}_e]$; $e = 1, 2, \dots, k$ of the e th interval single-objective non-linear programming problem (P_{1e}); $e = 1, 2, \dots, k$.

Step 9: Using the obtained e th value of f_e ; $e = 1, 2, \dots, k$ and the obtained e th value of \bar{f}_e ; $e = 1, 2, \dots, k$, construct the e th membership function $\mu_e(x)$; $e = 1, 2, \dots, k$ and the e th membership function $\mu_{eC}(x)$; $e = 1, 2, \dots, k$.

$$\mu_e(x) = \begin{cases} 1, & f_e(x) \leq f_e \\ \frac{\bar{f}_e - f_e(x)}{\bar{f}_e - f_e}, & f_e < f_e(x) < \bar{f}_e \\ 0, & f_e(x) \geq \bar{f}_e \end{cases}$$

where

$$f_e(x) = \left(\sum_{i=1}^m \sum_{j=1}^n c_{ij}^e(x_{ij}) + \sum_{j=1}^n A_j^e(x_j^2) + \sum_{j=1}^n (-B_j^e(\bar{x}_j)) + \sum_{j=1}^n D_j^e \right).$$

$$\mu_{eC}(x) = \begin{cases} 1, & f_{eC}(x) \leq \underline{f}_e \\ \frac{\bar{f}_e - f_{eC}(x)}{\bar{f}_e - \underline{f}_e}, & \underline{f}_e < f_{eC}(x) < \bar{f}_e \\ 0, & f_{1C}(x) \geq \bar{f} \end{cases}$$

where

$$f_{eC}(x) = \sum_{i=1}^m \sum_{j=1}^n \left(\frac{c_{ij}^e(x_{ij})}{2} + \frac{\bar{c}_{ij}^e(\bar{x}_{ij})}{2} \right) + \sum_{j=1}^n \left(\frac{A_j^e(x_j^2)}{2} + \frac{\bar{A}_j^e(\bar{x}_j^2)}{2} \right) + \sum_{j=1}^n \left(\frac{-\bar{B}_j^e(x_j)}{2} + \frac{(-B_j^e(\bar{x}_j))}{2} \right) + \sum_{j=1}^n \left(\frac{D_j^e}{2} + \frac{\bar{D}_j^e}{2} \right).$$

Step 10: Find an optimal solution $\{x_{ij}, \bar{x}_{ij}, x_j, \bar{x}_j : i = 1, 2, \dots, m; j = 1, 2, \dots, n\}$ of the crisp single-objective non-linear programming problem (P_2) .

Problem (P_2)

$$\max \left(\sum_{e=1}^k \left(\frac{\bar{f}_e - (\sum_{i=1}^m \sum_{j=1}^n c_{ij}^e(x_{ij}) + \sum_{j=1}^n A_j^e(x_j^2) + \sum_{j=1}^n (-B_j^e(\bar{x}_j)) + \sum_{j=1}^n D_j^e)}{\bar{f}_e - \underline{f}_e} + \frac{\bar{f}_e - (\sum_{i=1}^m \sum_{j=1}^n (\frac{c_{ij}^e(x_{ij})}{2} + \frac{\bar{c}_{ij}^e(\bar{x}_{ij})}{2}) + \sum_{j=1}^n (\frac{A_j^e(x_j^2)}{2} + \frac{\bar{A}_j^e(\bar{x}_j^2)}{2}) + \sum_{j=1}^n (\frac{-\bar{B}_j^e(x_j)}{2} + \frac{(-B_j^e(\bar{x}_j))}{2}) + \sum_{j=1}^n (\frac{D_j^e}{2} + \frac{\bar{D}_j^e}{2}))}{\bar{f}_e - \underline{f}_e} \right) \right)$$

Subject to

$$\sum_{i=1}^m \sum_{j=1}^n c_{ij}^e(x_{ij}) + \sum_{j=1}^n A_j^e(x_j^2) + \sum_{j=1}^n (-B_j^e(\bar{x}_j)) + \sum_{j=1}^n D_j^e \geq \underline{f}_e; e = 1, 2, \dots, k,$$

$$\sum_{i=1}^m \sum_{j=1}^n \left(\frac{c_{ij}^e(x_{ij})}{2} + \frac{\bar{c}_{ij}^e(\bar{x}_{ij})}{2} \right) + \sum_{j=1}^n \left(\frac{A_j^e(x_j^2)}{2} + \frac{\bar{A}_j^e(\bar{x}_j^2)}{2} \right) + \sum_{j=1}^n \left(\frac{-\bar{B}_j^e(x_j)}{2} + \frac{(-B_j^e(\bar{x}_j))}{2} \right) + \sum_{j=1}^n \left(\frac{D_j^e}{2} + \frac{\bar{D}_j^e}{2} \right) \geq \underline{f}_e; e = 1, 2, \dots, k$$

and

Constraints of the problem (P_{4e}) .

Step 11: Using the optimal solution $\{x_{ij}, \bar{x}_{ij}, x_j, \bar{x}_j; i = 1, 2, \dots, m; j = 1, 2, \dots, n\}$, obtained in Step 10, find the interval optimal solution $\left\{ [x_{ij}, \bar{x}_{ij}], [x_j, \bar{x}_j]; i = 1, 2, \dots, m; j = 1, 2, \dots, n \right\}$ of the interval multi-objective non-linear programming problem (P_1) .

2.2 Illustrative Example

Dalman and Sivri [6] solved the interval bi-objective non-linear programming problem (P_3), corresponding to an interval bi-objective non-linear transportation problem having unknown interval demand, to illustrate their proposed approach.

In this section, the same problem is solved by Dalman and Sivri's approach [6].

Problem (P_3)

$$\begin{aligned} \min & ([0.8, 0.9] \otimes [\underline{x}_{11}^2, \bar{x}_{11}^2] + [0.75, 0.95] \otimes [\underline{x}_{12}^2, \bar{x}_{12}^2] + [0.35, 0.65] \otimes [\underline{x}_{21}^2, \bar{x}_{21}^2] \\ & + [0.4, 0.7] \otimes [\underline{x}_{22}^2, \bar{x}_{22}^2] + [0.2, 0.5] \otimes [\underline{x}_1^2, \bar{x}_1^2] + [-5, -3] \otimes [\underline{x}_1, \bar{x}_1] + [5, 10] \\ & + [0.15, 0.80] \otimes [\underline{x}_2^2, \bar{x}_2^2] + [-7, -4] \otimes [\underline{x}_2, \bar{x}_2] + [4, 10]) \end{aligned}$$

$$\begin{aligned} \min & ([0.7, 0.9] \otimes [\underline{x}_{11}^2, \bar{x}_{11}^2] + [0.15, 0.3] \otimes [\underline{x}_{12}^2, \bar{x}_{12}^2] + [0.5, 0.7] \otimes [\underline{x}_{21}^2, \bar{x}_{21}^2] \\ & + [0.4, 0.6] \otimes [\underline{x}_{22}^2, \bar{x}_{22}^2] + [0.3, 0.5] \otimes [\underline{x}_1^2, \bar{x}_1^2] + [-5, -4] \otimes [\underline{x}_1, \bar{x}_1] + [8, 10] \\ & + [0.7, 0.9] \otimes [\underline{x}_2^2, \bar{x}_2^2] + [-8, -6] \otimes [\underline{x}_2, \bar{x}_2] + [4, 9]) \end{aligned}$$

Subject to.

$$[0.1, 0.9] \otimes [\underline{x}_{11}, \bar{x}_{11}] + [0.8, 0.9] \otimes [\underline{x}_{21}, \bar{x}_{21}] = [\underline{x}_1, \bar{x}_1],$$

$$[0.6, 0.8] \otimes [\underline{x}_{12}, \bar{x}_{12}] + [0.7, 0.9] \otimes [\underline{x}_{22}, \bar{x}_{22}] = [\underline{x}_2, \bar{x}_2],$$

$$[\underline{x}_{11}, \bar{x}_{11}] + [\underline{x}_{12}, \bar{x}_{12}] \leq [10, 15],$$

$$[\underline{x}_{21}, \bar{x}_{21}] + [\underline{x}_{22}, \bar{x}_{22}] \leq [12, 17],$$

$$\underline{x}_{ij}, \underline{x}_j \geq 0, \underline{x}_{ij} \leq \bar{x}_{ij}, \underline{x}_j \leq \bar{x}_j, \quad i = 1, 2; j = 1, 2.$$

Using Dalman and Sivri's approach [6], the interval bi-objective non-linear programming problem (P_3) can be solved as follows.

Step 1: Using Step 1 of Dalman and Sivri's approach [6], the interval bi-objective non-linear programming problem (P_3) can be split into its equivalent interval single-objective non-linear programming problems (P_{31}) and (P_{32}).

Problem (P_{31})

$$\begin{aligned} \min & ([0.8, 0.9] \otimes [\underline{x}_{11}^2, \bar{x}_{11}^2] + [0.75, 0.95] \otimes [\underline{x}_{12}^2, \bar{x}_{12}^2] + [0.35, 0.65] \otimes [\underline{x}_{21}^2, \bar{x}_{21}^2] \\ & + [0.4, 0.7] \otimes [\underline{x}_{22}^2, \bar{x}_{22}^2] + [0.2, 0.5] \otimes [\underline{x}_1^2, \bar{x}_1^2] + [-5, -3] \otimes [\underline{x}_1, \bar{x}_1] + [5, 10] \\ & + [0.15, 0.80] \otimes [\underline{x}_2^2, \bar{x}_2^2] + [-7, -4] \otimes [\underline{x}_2, \bar{x}_2] + [4, 10]) \end{aligned}$$

Subject to.

Constraints of the problem (P_3).

Problem (P_{32})

$$\begin{aligned} \min & ([0.7, 0.9] \otimes [\underline{x}_{11}^2, \bar{x}_{11}^2] + [0.15, 0.3] \otimes [\underline{x}_{12}^2, \bar{x}_{12}^2] + [0.5, 0.7] \otimes [\underline{x}_{21}^2, \bar{x}_{21}^2] \\ & + [0.4, 0.6] \otimes [\underline{x}_{22}^2, \bar{x}_{22}^2] + [0.3, 0.5] \otimes [\underline{x}_1^2, \bar{x}_1^2] + [-5, -4] \otimes [\underline{x}_1, \bar{x}_1] + [8, 10] \\ & + [0.7, 0.9] \otimes [\underline{x}_2^2, \bar{x}_2^2] + [-8, -6] \otimes [\underline{x}_2, \bar{x}_2] + [4, 9]) \end{aligned}$$

Subject to.

Constraints of the problem (P_3).

Step 2: Using Step 2 of Dalman and Sivri's approach [6], the interval single-objective non-linear programming problems (P_{31}) and (P_{32}) can be transformed into their equivalent interval single-objective non-linear programming problems (P_{41}) and (P_{42}), respectively.

Problem (P_{41})

$$\begin{aligned} \min & ([0.8\underline{x}_{11}^2, 0.9\bar{x}_{11}^2] + [0.75\underline{x}_{12}^2, 0.95\bar{x}_{12}^2] + [0.35\underline{x}_{21}^2, 0.65\bar{x}_{21}^2] + [0.4\underline{x}_{22}^2, 0.7\bar{x}_{22}^2] \\ & + [0.2\underline{x}_1^2, 0.5\bar{x}_1^2] + [-5\underline{x}_1, -3\bar{x}_1] + [5, 10] + [0.15\underline{x}_2^2, 0.80\bar{x}_2^2] \\ & + [-7\bar{x}_2, -4\underline{x}_2] + [4, 10]) \end{aligned}$$

Subject to.

$$[0.1\underline{x}_{11}, 0.9\bar{x}_{11}] + [0.8\underline{x}_{21}, 0.9\bar{x}_{21}] = [\underline{x}_1, \bar{x}_1],$$

$$[0.6\underline{x}_{12}, 0.8\bar{x}_{12}] + [0.7\underline{x}_{22}, 0.9\bar{x}_{22}] = [\underline{x}_2, \bar{x}_2],$$

$$[\underline{x}_{11}, \bar{x}_{11}] + [\underline{x}_{12}, \bar{x}_{12}] \leq [10, 15],$$

$$[\underline{x}_{21}, \bar{x}_{21}] + [\underline{x}_{22}, \bar{x}_{22}] \leq [12, 17],$$

$$\underline{x}_{ij}, \underline{x}_j \geq 0, \underline{x}_{ij} \leq \bar{x}_{ij}, \underline{x}_j \leq \bar{x}_j, i = 1, 2; j = 1, 2.$$

Problem (P_{42})

$$\min \left(\begin{array}{l} \left[0.7\underline{x}_{11}^2, 0.9\bar{x}_{11}^2 \right] + \left[0.15\underline{x}_{12}^2, 0.3\bar{x}_{12}^2 \right] + \left[0.5\underline{x}_{21}^2, 0.7\bar{x}_{21}^2 \right] + \left[0.4\underline{x}_{22}^2, 0.6\bar{x}_{22}^2 \right] \\ + \left[0.3\underline{x}_1^2, 0.5\bar{x}_1^2 \right] + \left[-5\underline{x}_1, -4\bar{x}_1 \right] + [8, 10] \\ + \left[0.7\underline{x}_2^2, 0.9\bar{x}_2^2 \right] + \left[-8\underline{x}_2, -6\bar{x}_2 \right] + [4, 9] \end{array} \right)$$

Subject to.

Constraints of the problem (P_{41}).

Step 3: Using Step 3 of Dalman and Sivri’s approach [6], the interval single-objective non-linear programming problems (P_{41}) and (P_{42}) can be transformed into their equivalent interval single-objective non-linear programming problems (P_{51}) and (P_{52}), respectively.

Problem (P_{51})

$$\min \left(\left[\begin{array}{l} 0.8\underline{x}_{11}^2 + 0.75\underline{x}_{12}^2 + 0.35\underline{x}_{21}^2 + 0.4\underline{x}_{22}^2 + 0.2\underline{x}_1^2 - 5\underline{x}_1 + 0.15\underline{x}_2^2 - 7\underline{x}_2 + 9, 0.9\bar{x}_{11}^2 \\ + 0.95\bar{x}_{12}^2 + 0.65\bar{x}_{21}^2 + 0.7\bar{x}_{22}^2 + 0.5\bar{x}_1^2 - 3\underline{x}_1 + 0.80\bar{x}_2^2 - 4\underline{x}_2 + 20 \end{array} \right] \right)$$

Subject to.

$$\left[0.1\underline{x}_{11} + 0.8\underline{x}_{21}, 0.9\bar{x}_{11} + 0.9\bar{x}_{21} \right] = \left[\underline{x}_1, \bar{x}_1 \right],$$

$$\left[0.6\underline{x}_{12} + 0.7\underline{x}_{22}, 0.8\bar{x}_{12} + 0.9\bar{x}_{22} \right] = \left[\underline{x}_2, \bar{x}_2 \right],$$

$$\left[\underline{x}_{11} + \underline{x}_{12}, \bar{x}_{11} + \bar{x}_{12} \right] \leq [10, 15],$$

$$\left[\underline{x}_{21} + \underline{x}_{22}, \bar{x}_{21} + \bar{x}_{22} \right] \leq [12, 17],$$

$$\underline{x}_{ij}, \underline{x}_j \geq 0, \underline{x}_{ij} \leq \bar{x}_{ij}, \underline{x}_j \leq \bar{x}_j, \quad i = 1, 2; j = 1, 2.$$

Problem (P_{52})

$$\min \left(\left[\begin{array}{l} 0.7\underline{x}_{11}^2 + 0.15\underline{x}_{12}^2 + 0.5\underline{x}_{21}^2 + 0.4\underline{x}_{22}^2 + 0.3\underline{x}_1^2 - 5\underline{x}_1 + 0.7\underline{x}_2^2 - 8\underline{x}_2 + 12, 0.9\bar{x}_{11}^2 \\ + 0.3\bar{x}_{12}^2 + 0.7\bar{x}_{21}^2 + 0.6\bar{x}_{22}^2 + 0.5\bar{x}_1^2 - 4\underline{x}_1 + 0.9\bar{x}_2^2 - 6\underline{x}_2 + 19 \end{array} \right] \right)$$

Subject to

Constraints of the problem (P_{51}).

Step 4: Using Step 4 of Dalman and Sivri’s approach [6], the interval single-objective non-linear programming problems (P_{51}) and (P_{52}) can be transformed into their equivalent crisp bi-objective non-linear programming problems (P_{61}) and (P_{62}), respectively.

Problem (P_{61})

$$\min(0.9\bar{x}_{11}^2 + 0.95\bar{x}_{12}^2 + 0.65\bar{x}_{21}^2 + 0.7\bar{x}_{22}^2 + 0.5\bar{x}_1^2 - 3\underline{x}_1 + 0.80\bar{x}_2^2 - 4\underline{x}_2 + 20)$$

$$\min\left(\begin{array}{l} 0.4\underline{x}_{11}^2 + \frac{0.9}{2}\bar{x}_{11}^2 + \frac{0.75}{2}\underline{x}_{12}^2 + \frac{0.95}{2}\bar{x}_{12}^2 + \frac{0.35}{2}\underline{x}_{21}^2 + \frac{0.65}{2}\bar{x}_{21}^2 + 0.2\underline{x}_{22}^2 + \frac{0.7}{2}\bar{x}_{22}^2 \\ + 0.1\underline{x}_1^2 + \frac{0.5}{2}\bar{x}_1^2 - \frac{3}{2}\underline{x}_1 - \frac{5}{2}\bar{x}_1 + \frac{0.15}{2}\underline{x}_2^2 + 0.40\bar{x}_2^2 - 2\underline{x}_2 - \frac{7}{2}\bar{x}_2 + \frac{5}{2} + 12 \end{array}\right)$$

Subject to.

$$0.1\underline{x}_{11} + 0.8\underline{x}_{21} = \underline{x}_1,$$

$$0.9\bar{x}_{11} + 0.9\bar{x}_{21} = \bar{x}_1,$$

$$0.6\underline{x}_{12} + 0.7\underline{x}_{22} = \underline{x}_2,$$

$$0.8\bar{x}_{12} + 0.9\bar{x}_{22} = \bar{x}_2,$$

$$\bar{x}_{11} + \bar{x}_{12} \leq 15,$$

$$\frac{\underline{x}_{11}}{2} + \frac{\bar{x}_{11}}{2} + \frac{\underline{x}_{12}}{2} + \frac{\bar{x}_{12}}{2} \leq 5 + \frac{15}{2},$$

$$\bar{x}_{21} + \bar{x}_{22} \leq 17,$$

$$\frac{\underline{x}_{21}}{2} + \frac{\bar{x}_{21}}{2} + \frac{\underline{x}_{22}}{2} + \frac{\bar{x}_{22}}{2} \leq 6 + \frac{17}{2},$$

$$\underline{x}_{ij}, \underline{x}_j \geq 0, \underline{x}_{ij} \leq \bar{x}_{ij}, \underline{x}_j \leq \bar{x}_j, \quad i = 1, 2; j = 1, 2.$$

Problem (P_{62})

$$\min(0.9\bar{x}_{11}^2 + 0.3\bar{x}_{12}^2 + 0.7\bar{x}_{21}^2 + 0.6\bar{x}_{22}^2 + 0.5\bar{x}_1^2 - 4\underline{x}_1 + 0.9\bar{x}_2^2 - 6\underline{x}_2 + 19)$$

$$\min\left(\begin{array}{l} \frac{0.7}{2}\underline{x}_{11}^2 + \frac{0.9}{2}\bar{x}_{11}^2 + \frac{0.15}{2}\underline{x}_{12}^2 + \frac{0.3}{2}\bar{x}_{12}^2 + \frac{0.5}{2}\underline{x}_{21}^2 + \frac{0.7}{2}\bar{x}_{21}^2 + 0.2\underline{x}_{22}^2 + 0.3\bar{x}_{22}^2 \\ + \frac{0.3}{2}\underline{x}_1^2 + \frac{0.5}{2}\bar{x}_1^2 - 2\underline{x}_1 - \frac{5}{2}\bar{x}_1 + \frac{0.7}{2}\underline{x}_2^2 + \frac{0.9}{2}\bar{x}_2^2 - 3\underline{x}_2 - 4\bar{x}_2 + \frac{9}{2} + 11 \end{array}\right)$$

Subject to.

Constraints of the problem (P_{61}).

Step 5: Using Step 5 of Dalman and Sivri's approach [6], the crisp bi-objective non-linear programming problems (P_{61}) can be transformed into its equivalent two

crisp single-objective non-linear programming problems (P_{611}) and (P_{612}). Also, the crisp bi-objective non-linear programming problems (P_{62}) can be transformed into its equivalent two crisp single-objective non-linear programming problems (P_{621}) and (P_{622}).

Problem (P_{611})

$$\min(0.9\bar{x}_{11}^2 + 0.95\bar{x}_{12}^2 + 0.65\bar{x}_{21}^2 + 0.7\bar{x}_{22}^2 + 0.5\bar{x}_1^2 - 3\bar{x}_1 + 0.80\bar{x}_2^2 - 4\bar{x}_2 + 20)$$

Subject to.

Constraints of the problem (P_{61}).

Problem (P_{612})

$$\min \left(\begin{aligned} &0.4\bar{x}_{11}^2 + \frac{0.9}{2}\bar{x}_{11}^2 + \frac{0.75}{2}\bar{x}_{12}^2 + \frac{0.95}{2}\bar{x}_{12}^2 + \frac{0.35}{2}\bar{x}_{21}^2 + \frac{0.65}{2}\bar{x}_{21}^2 + 0.2\bar{x}_{22}^2 + \frac{0.7}{2}\bar{x}_{22}^2 \\ &+ 0.1\bar{x}_1^2 + \frac{0.5}{2}\bar{x}_1^2 - \frac{3}{2}\bar{x}_1 - \frac{5}{2}\bar{x}_1 + \frac{0.15}{2}\bar{x}_2^2 + 0.40\bar{x}_2^2 - 2\bar{x}_2 - \frac{7}{2}\bar{x}_2 + \frac{5}{2} + 12 \end{aligned} \right)$$

Subject to.

Constraints of the problem (P_{61}).

Problem (P_{621})

$$\min(0.9\bar{x}_{11}^2 + 0.3\bar{x}_{12}^2 + 0.7\bar{x}_{21}^2 + 0.6\bar{x}_{22}^2 + 0.5\bar{x}_1^2 - 4\bar{x}_1 + 0.9\bar{x}_2^2 - 6\bar{x}_2 + 19)$$

Subject to.

Constraints of the problem (P_{61}).

Problem (P_{622})

$$\min \left(\begin{aligned} &\frac{0.7}{2}\bar{x}_{11}^2 + \frac{0.9}{2}\bar{x}_{11}^2 + \frac{0.15}{2}\bar{x}_{12}^2 + \frac{0.3}{2}\bar{x}_{12}^2 + \frac{0.5}{2}\bar{x}_{21}^2 + \frac{0.7}{2}\bar{x}_{21}^2 + 0.2\bar{x}_{22}^2 + 0.3\bar{x}_{22}^2 \\ &+ \frac{0.3}{2}\bar{x}_1^2 + \frac{0.5}{2}\bar{x}_1^2 - 2\bar{x}_1 - \frac{5}{2}\bar{x}_1 + \frac{0.7}{2}\bar{x}_2^2 + \frac{0.9}{2}\bar{x}_2^2 - 3\bar{x}_2 - 4\bar{x}_2 + \frac{9}{2} + 11 \end{aligned} \right)$$

Subject to.

Constraints of the problem (P_{61}).

Step 6: According to the Step 6 of Dalman and Sivri's approach [6], there is a need to find an optimal solution and the optimal value of the crisp single-objective non-linear programming problems (P_{611}) and (P_{621}).

- (i) On solving the crisp single-objective non-linear programming problem (P_{611}), the obtained optimal solution is $\bar{x}_{11} = 0$, $\bar{x}_{11} = 0$, $\bar{x}_{12} = 0.494$, $\bar{x}_{12} = 0.494$, $\bar{x}_{21} = 1.137$, $\bar{x}_{21} = 1.137$, $\bar{x}_{22} = 0.827$, $\bar{x}_{22} = 0.827$, $\bar{x}_1 = 0.909$, $\bar{x}_1 = 1.023$, $\bar{x}_2 = 0.875$, $\bar{x}_2 = 1.140$ and the optimal value is 16.883 i.e., $\bar{f}_1 = 16.883$.

- (ii) On solving the crisp single-objective non-linear programming problem (P_{621}), the obtained optimal solution is $\underline{x}_{11} = 0$, $\bar{x}_{11} = 0$, $\underline{x}_{12} = 1.385$, $\bar{x}_{12} = 1.385$, $\underline{x}_{21} = 1.447$, $\bar{x}_{21} = 1.447$, $\underline{x}_{22} = 0.904$, $\bar{x}_{22} = 0.904$, $\underline{x}_1 = 1.158$, $\bar{x}_1 = 1.303$, $\underline{x}_2 = 1.464$, $\bar{x}_2 = 1.922$ and the optimal value is 12.290 i.e., $\underline{f}_2 = 12.290$.

Step 7: According to the Step 7 of Dalman and Sivri's approach [6], there is a need to find an optimal solution of the crisp single-objective non-linear programming problems (P_{612}) and (P_{622}).

- (i) On solving the crisp single-objective non-linear programming problem (P_{612}), the obtained optimal solution is $\underline{x}_{11} = 0.148$, $\bar{x}_{11} = 1.117$, $\underline{x}_{12} = 1.177$, $\bar{x}_{12} = 1.177$, $\underline{x}_{21} = 1.953$, $\bar{x}_{21} = 1.953$, $\underline{x}_{22} = 2.084$, $\bar{x}_{22} = 2.084$, $\underline{x}_1 = 1.577$, $\bar{x}_1 = 2.764$, $\underline{x}_2 = 2.165$, $\bar{x}_2 = 2.818$.
- (ii) On solving the crisp single-objective non-linear programming problem (P_{622}), the obtained optimal solution is $\underline{x}_{11} = 0.220$, $\bar{x}_{11} = 1.139$, $\underline{x}_{12} = 2.834$, $\bar{x}_{12} = 2.834$, $\underline{x}_{21} = 1.882$, $\bar{x}_{21} = 1.882$, $\underline{x}_{22} = 1.462$, $\bar{x}_{22} = 1.462$, $\underline{x}_1 = 1.528$, $\bar{x}_1 = 2.720$, $\underline{x}_2 = 2.724$, $\bar{x}_2 = 3.583$.

Step 8: According to the Step 8 of Dalman and Sivri's approach [6], there is a need to find the values of the functions $\underline{f}_1(x) = 0.8\underline{x}_{11}^2 + 0.75\underline{x}_{12}^2 + 0.35\underline{x}_{21}^2 + 0.4\underline{x}_{22}^2 + 0.2\underline{x}_1^2 - 5\underline{x}_1 + 0.15\underline{x}_2^2 - 7\underline{x}_2 + 9$ and $\underline{f}_2(x) = 0.7\underline{x}_{11}^2 + 0.15\underline{x}_{12}^2 + 0.5\underline{x}_{21}^2 + 0.4\underline{x}_{22}^2 + 0.3\underline{x}_1^2 - 5\underline{x}_1 + 0.7\underline{x}_2^2 - 8\underline{x}_2 + 12$ corresponding to the optimal solutions of the crisp single-objective non-linear programming problems (P_{612}) and problem (P_{622}), respectively.

- (i) Putting the optimal solution of the crisp single-objective non-linear programming problem (P_{612}) in the function $\underline{f}_1(x) = 0.8\underline{x}_{11}^2 + 0.75\underline{x}_{12}^2 + 0.35\underline{x}_{21}^2 + 0.4\underline{x}_{22}^2 + 0.2\underline{x}_1^2 - 5\underline{x}_1 + 0.15\underline{x}_2^2 - 7\underline{x}_2 + 9$, the obtained value is -19.216 i.e., $\underline{f}_1 = -19.216$.
- (ii) Putting the optimal solution of the crisp single-objective non-linear programming problem (P_{622}) in the function $\underline{f}_2(x) = 0.7\underline{x}_{11}^2 + 0.15\underline{x}_{12}^2 + 0.5\underline{x}_{21}^2 + 0.4\underline{x}_{22}^2 + 0.3\underline{x}_1^2 - 5\underline{x}_1 + 0.7\underline{x}_2^2 - 8\underline{x}_2 + 12$, the obtained value is -20.508 i.e., $\underline{f}_2 = -20.508$.

Furthermore, using the obtained values of \underline{f}_1 , \bar{f}_1 , \underline{f}_2 and \bar{f}_2 , the interval optimal values of the interval single-objective non-linear programming problems (P_{31}) and (P_{32}) are $[\underline{f}_1, \bar{f}_1] = [-19.216, 16.883]$ and $[\underline{f}_2, \bar{f}_2] = [-20.508, 12.290]$, respectively.

Step 9: Using $\underline{f}_1 = -19.216$, $\bar{f}_1 = 16.883$, $\underline{f}_2 = -20.508$ and $\bar{f}_2 = 12.290$ in Step 9 of Dalman and Sivri's approach [6],

$$(i) \quad \underline{\mu}_1(x) = \begin{cases} 1, & \underline{f}_1(x) \leq -19.216 \\ \frac{16.883 - \underline{f}_1(x)}{16.883 - (-19.216)}, & -19.216 < \underline{f}_1(x) < 16.883 \\ 0, & \underline{f}_1(x) \geq 16.883 \end{cases}$$

where

$$\underline{f}_1(x) = 0.8\underline{x}_{11}^2 + 0.75\underline{x}_{12}^2 + 0.35\underline{x}_{21}^2 + 0.4\underline{x}_{22}^2 + 0.2\underline{x}_1^2 - 5\underline{x}_1 + 0.15\underline{x}_2^2 - 7\underline{x}_2 + 9.$$

$$(ii) \quad \mu_{1C}(x) = \begin{cases} 1, & f_{1C}(x) \leq -19.216 \\ \frac{16.883 - f_{1C}(x)}{16.883 - (-19.216)}, & -19.216 < f_{1C}(x) < 16.883 \\ 0, & f_{1C}(x) \geq 16.883 \end{cases}$$

where

$$f_{1C}(x) = 0.4\underline{x}_{11}^2 + \frac{0.9}{2}\underline{x}_{11}^2 + \frac{0.75}{2}\underline{x}_{12}^2 + \frac{0.95}{2}\underline{x}_{12}^2 + \frac{0.35}{2}\underline{x}_{21}^2 + \frac{0.65}{2}\underline{x}_{21}^2 + 0.2\underline{x}_{22}^2 \\ + \frac{0.7}{2}\underline{x}_{22}^2 + 0.1\underline{x}_1^2 + \frac{0.5}{2}\underline{x}_1^2 - \frac{3}{2}\underline{x}_1 - \frac{5}{2}\underline{x}_1 + \frac{0.15}{2}\underline{x}_2^2 + 0.40\underline{x}_2^2 - 2\underline{x}_2 - \frac{7}{2}\underline{x}_2 + \frac{5}{2} + 12.$$

$$(iii) \quad \underline{\mu}_2(x) = \begin{cases} 1, & \underline{f}_2(x) \leq -20.508 \\ \frac{12.290 - \underline{f}_2(x)}{12.290 - (-20.508)}, & -20.508 < \underline{f}_2(x) < 12.290 \\ 0, & \underline{f}_2(x) \geq 12.290 \end{cases}$$

where

$$\underline{f}_2(x) = 0.7\underline{x}_{11}^2 + 0.15\underline{x}_{12}^2 + 0.5\underline{x}_{21}^2 + 0.4\underline{x}_{22}^2 + 0.3\underline{x}_1^2 - 5\underline{x}_1 + 0.7\underline{x}_2^2 - 8\underline{x}_2 + 12.$$

$$(iv) \quad \mu_{2C}(x) = \begin{cases} 1, & f_{2C}(x) \leq -20.508 \\ \frac{12.290 - f_{2C}(x)}{12.290 - (-20.508)}, & -20.508 < f_{2C}(x) < 12.290 \\ 0, & f_{2C}(x) \geq 12.290 \end{cases}$$

where

$$f_{2C}(x) = \frac{0.7}{2}\underline{x}_{11}^2 + \frac{0.9}{2}\underline{x}_{11}^2 + \frac{0.15}{2}\underline{x}_{12}^2 + \frac{0.3}{2}\underline{x}_{12}^2 + \frac{0.5}{2}\underline{x}_{21}^2 + \frac{0.7}{2}\underline{x}_{21}^2 + 0.2\underline{x}_{22}^2 + 0.3\underline{x}_{22}^2 \\ + \frac{0.3}{2}\underline{x}_1^2 + \frac{0.5}{2}\underline{x}_1^2 - 2\underline{x}_1 - \frac{5}{2}\underline{x}_1 + \frac{0.7}{2}\underline{x}_2^2 + \frac{0.9}{2}\underline{x}_2^2 - 3\underline{x}_2 - 4\underline{x}_2 + \frac{9}{2} + 11.$$

Step 10: According to the Step 10 of Dalman and Sivri's approach [6], there is a need to solve the crisp single-objective non-linear programming problem (P_4).

Problem (P_4)

$$\max \left\{ \begin{array}{l} \frac{16.883 - f_1(x)}{16.883 - (-19.216)} + \frac{16.883 - f_{1C}(x)}{16.883 - (-19.216)} + \frac{12.290 - f_2(x)}{12.290 - (-20.508)} \\ + \frac{12.290 - f_{2C}(x)}{12.290 - (-20.508)} \end{array} \right\}$$

Subject to

$$\underline{f}_1(x) \geq -19.216,$$

$$f_{1C}(x) \geq -19.216,$$

$$\underline{f}_2(x) \geq -20.508,$$

$$f_{2C}(x) \geq -20.508,$$

and

Constraints of the problem (P_{61}).

On solving the crisp single-objective non-linear programming problem (P_4), the obtained optimal solution is $\underline{x}_{11} = 0.228$, $\bar{x}_{11} = 1.049$, $\underline{x}_{12} = 2.633$, $\bar{x}_{12} = 2.633$, $\underline{x}_{21} = 1.651$, $\bar{x}_{21} = 1.651$, $\underline{x}_{22} = 0.887$, $\bar{x}_{22} = 1.415$, $\underline{x}_1 = 1.344$, $\bar{x}_1 = 2.431$, $\underline{x}_2 = 2.201$, $\bar{x}_2 = 3.380$.

Step 11: Using the optimal solution of the crisp single-objective non-linear programming problem (P_4), obtained in Step 10, the interval optimal solution of the interval bi-objective problem non-linear programming problem (P_3) is $[\underline{x}_{11}, \bar{x}_{11}] = [0.228, 1.049]$, $[\underline{x}_{12}, \bar{x}_{12}] = [2.633, 2.633]$, $[\underline{x}_{21}, \bar{x}_{21}] = [1.651, 1.651]$, $[\underline{x}_{22}, \bar{x}_{22}] = [0.887, 0.887]$, $[\underline{x}_1, \bar{x}_1] = [1.344, 2.431]$, $[\underline{x}_2, \bar{x}_2] = [2.201, 3.380]$ and the interval optimal value of the first and second objective functions of the interval bi-objective non-linear programming problem (P_3) is $[\underline{f}_1, \bar{f}_1] = [-19.216, 30.008]$ and $[\underline{f}_2, \bar{f}_2] = [-20.508, 19.834]$, respectively.

3 Mathematical Incorrect Assumption Considered in Dalman and Sivri's Approach

It is obvious from Dalman and Sivri's approach [6] that to find an interval optimal solution of the interval single-objective non-linear programming problem (P_{1e}); $e = 1, 2, \dots, k$, it has been split into two crisp single-objective non-linear programming problems (P_{4e1}) and (P_{4e2}). Finally, both the crisp single-objective non-linear programming problems (P_{4e1}) and (P_{4e2}) have been solved independently. Since, on solving the crisp single-objective non-linear programming problems (P_{4e1}) and (P_{4e2}) independently, two different optimal solutions are obtained. Hence, two

different interval optimal values are obtained for the e^{th} interval single-objective non-linear programming problem (P_{1e}); $e = 1, 2, \dots, k$. Therefore, it is not possible to conclude that whether the interval optimal solution, obtained with the help of the optimal solution of the crisp single-objective non-linear programming problem (P_{4e1}), represents the interval optimal solution of the interval single-objective non-linear programming problem (P_{1e}); $e = 1, 2, \dots, k$ or the interval optimal solution, obtained with the help of the optimal solution of the crisp single-objective non-linear programming problem (P_{4e2}), represents the interval optimal solution of interval single-objective non-linear programming problem (P_{1e}); $e = 1, 2, \dots, k$.

The following validates this claim.

It is obvious from Sect. 2.2 that to find an interval optimal solution of the interval single-objective non-linear programming problem (P_{31}), it has been broken down into two crisp single-objective non-linear programming problems (P_{611}) and (P_{612}). Also, it is obvious that

- (i) On solving the crisp single-objective non-linear programming problem (P_{611}), the obtained optimal solution is $\underline{x}_{11} = 0, \bar{x}_{11} = 0, \underline{x}_{12} = 0.494, \bar{x}_{12} = 0.494, \underline{x}_{21} = 1.137, \bar{x}_{21} = 1.137, \underline{x}_{22} = 0.827, \bar{x}_{22} = 0.827, \underline{x}_1 = 0.909, \bar{x}_1 = 1.023, \underline{x}_2 = 0.875, \bar{x}_2 = 1.140$. Using this optimal solution, the interval optimal solution for the interval single-objective non-linear programming problem (P_{31}) is $[\underline{x}_{11}, \bar{x}_{11}] = [0, 0], [\underline{x}_{12}, \bar{x}_{12}] = [0.494, 0.494], [\underline{x}_{21}, \bar{x}_{21}] = [1.137, 1.137], [\underline{x}_{22}, \bar{x}_{22}] = [0.827, 0.827], [\underline{x}_1, \bar{x}_1] = [0.909, 1.023], [\underline{x}_2, \bar{x}_2] = [0.875, 1.140]$ and the optimal interval value of the interval single-objective non-linear programming problem (P_{31}) is $[-2.906, 16.883]$.
- (ii) On solving the crisp single-objective non-linear programming problem (P_{612}), the obtained optimal solution is $\underline{x}_{11} = 0.148, \bar{x}_{11} = 1.117, \underline{x}_{12} = 1.177, \bar{x}_{12} = 1.177, \underline{x}_{21} = 1.953, \bar{x}_{21} = 1.953, \underline{x}_{22} = 2.084, \bar{x}_{22} = 2.084, \underline{x}_1 = 1.577, \bar{x}_1 = 2.764, \underline{x}_2 = 2.165, \bar{x}_2 = 2.818$. Using this optimal solution, the interval optimal solution for the interval single-objective non-linear programming problem (P_{31}) is $[\underline{x}_{11}, \bar{x}_{11}] = [0.148, 1.117], [\underline{x}_{12}, \bar{x}_{12}] = [1.177, 1.177], [\underline{x}_{21}, \bar{x}_{21}] = [1.953, 1.953], [\underline{x}_{22}, \bar{x}_{22}] = [2.084, 2.084], [\underline{x}_1, \bar{x}_1] = [1.577, 2.764], [\underline{x}_2, \bar{x}_2] = [2.165, 2.818]$ and the optimal interval value of the interval single-objective non-linear programming problem (P_{31}) is $[-19.216, 24.740]$.

Hence, it is not possible to conclude that whether the interval optimal solution, obtained in (i), or the interval optimal solution, obtained in (ii), represents the interval optimal solution of the interval single-objective linear programming problem (P_{31}).

The same problem is also occurring for the interval optimal solution for the interval single-objective non-linear programming problem (P_{32}).

4 Suggested Modifications

It is obvious that the inappropriateness of Dalman and Sivri’s approach [6] can be resolved by considering an appropriate approach for solving the e th crisp bi-objective non-linear programming problem (P_{4e}) ; $e = 1, 2, \dots, k$. There exist various approaches for solving crisp multi-objective mathematical programming problems. Weighted sum approach [22] is one of the commonly used approach for solving crisp multi-objective mathematical programming problems.

Using the weighted sum approach [22] for solving crisp multi-objective mathematical programming problems, an efficient solution of the e th crisp bi-objective non-linear programming problem (P_{4e}) can be obtained as follows.

Step 5: Transform the e th crisp bi-objective non-linear programming problem (P_{4e}) ; $e = 1, 2, \dots, k$ into its equivalent e th crisp single-objective non-linear programming problem (P_{4e1}) ; $e = 1, 2, \dots, k$.

Problem (P'_{4e1})

$$\min \left(\begin{aligned} & \sum_{i=1}^m \sum_{j=1}^n \bar{c}_{ij}^e(\bar{x}_{ij}) + \sum_{j=1}^n \bar{A}_j^e(\bar{x}_j^2) + \sum_{j=1}^n (-\bar{B}_j^e(x_j)) + \sum_{j=1}^n \bar{D}_j^e \\ & + \left(\sum_{i=1}^m \sum_{j=1}^n \left(\frac{c_{ij}^e(x_{ij})}{2} + \frac{\bar{c}_{ij}^e(\bar{x}_{ij})}{2} \right) + \sum_{j=1}^n \left(\frac{A_j^e(x_j^2)}{2} + \frac{\bar{A}_j^e(\bar{x}_j^2)}{2} \right) \right) \\ & + \left(\sum_{j=1}^n \left(\frac{-\bar{B}_j^e(x_j)}{2} + \frac{(-B_j^e(\bar{x}_j))}{2} \right) + \sum_{j=1}^n \left(\frac{D_j^e}{2} + \frac{\bar{D}_j^e}{2} \right) \right) \end{aligned} \right)$$

Subject to.

Constraints of the problem (P_{4e}) .

Step 6: Find a crisp optimal solution $\{x_{ij}, \bar{x}_{ij}, \underline{x}_j, \bar{x}_j; i = 1, 2, \dots, m; j = 1, 2, \dots, n\}^e$ of the e th crisp single-objective non-linear programming problem (P_{4e1}) ; $e = 1, 2, \dots, k$.

Step 7: Using the crisp optimal solution $\{x_{ij}, \bar{x}_{ij}, \underline{x}_j, \bar{x}_j; i = 1, 2, \dots, m; j = 1, 2, \dots, n\}^e$, obtained in Step 6, find the interval optimal solution $\left\{ \left[x_{ij}, \bar{x}_{ij} \right], \left[\underline{x}_j, \bar{x}_j \right]; i = 1, 2, \dots, m; j = 1, 2, \dots, n \right\}^e$ of the e th interval single-objective non-linear programming problem (P_{1e}) ; $e = 1, 2, \dots, k$.

Step 8: Find the value of the e^{th} interval objective function $\left[\underline{f}_e, \bar{f}_e \right] = \sum_{i=1}^m \sum_{j=1}^n \left[c_{ij}^e x_{ij}, \bar{c}_{ij}^e \bar{x}_{ij} \right] + \sum_{j=1}^n \left[A_j^e x_j^2, \bar{A}_j^e \bar{x}_j^2 \right] + \sum_{j=1}^n \left[-B_j^e \bar{x}_j, -\bar{B}_j^e \underline{x}_j \right] +$

$\sum_{j=1}^n [D_j^e, \bar{D}_j^e]$; $e = 1, 2, \dots, k$ of the interval single-objective non-linear programming problem (P_{1e}); $e = 1, 2, \dots, k$ corresponding to the obtained interval optimal solution of the e^{th} interval single-objective non-linear programming problem (P_{1e}); $e = 1, 2, \dots, k$.

5 Exact Results of the Existing Interval Bi-objective Non-linear Transportation Problem Having Unknown Interval Demand

It is obvious from Sect. 3 that Dalman and Sivri’s approach [6] is not appropriate in its present form, and hence, the results of an interval bi-objective non-linear transportation problem with unknown interval demand, obtained by Dalman and Sivri [6] with the help of the bi-objective non-linear programming problem (P_3), are not appropriate.

In this section, the exact results of the same interval bi-objective non-linear transportation problem are obtained by solving the interval bi-objective non-linear programming problem (P_3) with the help of the suggested modifications.

Step 1 to Step 4: Using Step 1 to Step 4 of Sect. 2.2, the interval bi-objective non-linear programming problem (P_3) can be transformed into its equivalent crisp bi-objective non-linear programming problems (P_{61}) and (P_{62}).

Step 5: Using Step 5 of the modified approach, the crisp bi-objective non-linear programming problems (P_{61}) and (P_{62}) can be transformed into their equivalent crisp single-objective non-linear programming problems (P'_{611}) and (P'_{621}), respectively.

Problem (P'_{611})

$$\begin{aligned} &\min(0.9\bar{x}_{11}^2 + 0.95\bar{x}_{12}^2 + 0.65\bar{x}_{21}^2 + 0.7\bar{x}_{22}^2 + 0.5\bar{x}_1^2 - 3\bar{x}_1 + 0.80\bar{x}_2^2 - 4\bar{x}_2 + 20 \\ &\quad + 0.4\bar{x}_{11}^2 + \frac{0.9}{2}\bar{x}_{11}^2 + \frac{0.75}{2}\bar{x}_{12}^2 + \frac{0.95}{2}\bar{x}_{12}^2 + \frac{0.35}{2}\bar{x}_{21}^2 + \frac{0.65}{2}\bar{x}_{21}^2 + 0.2\bar{x}_{22}^2 + \frac{0.7}{2}\bar{x}_{22}^2 \\ &\quad + 0.1\bar{x}_1^2 + \frac{0.5}{2}\bar{x}_1^2 - \frac{3}{2}\bar{x}_1 - \frac{5}{2}\bar{x}_1 + \frac{0.15}{2}\bar{x}_2^2 + 0.40\bar{x}_2^2 - 2\bar{x}_2 - \frac{7}{2}\bar{x}_2 + \frac{5}{2} + 12) \end{aligned}$$

Subject to.

Constraints of the problem (P_{61}).

Problem (P'_{621})

$$\begin{aligned} &\min(0.9\bar{x}_{11}^2 + 0.3\bar{x}_{12}^2 + 0.7\bar{x}_{21}^2 + 0.6\bar{x}_{22}^2 + 0.5\bar{x}_1^2 - 4\bar{x}_1 + 0.9\bar{x}_2^2 - 6\bar{x}_2 + 19 \\ &\quad + \frac{0.7}{2}\bar{x}_{11}^2 + \frac{0.9}{2}\bar{x}_{11}^2 + \frac{0.15}{2}\bar{x}_{12}^2 + \frac{0.3}{2}\bar{x}_{12}^2 + \frac{0.5}{2}\bar{x}_{21}^2 + \frac{0.7}{2}\bar{x}_{21}^2 + 0.2\bar{x}_{22}^2 + 0.3\bar{x}_{22}^2 \\ &\quad + \frac{0.3}{2}\bar{x}_1^2 + \frac{0.5}{2}\bar{x}_1^2 - 2\bar{x}_1 - \frac{5}{2}\bar{x}_1 + \frac{0.7}{2}\bar{x}_2^2 + \frac{0.9}{2}\bar{x}_2^2 - 3\bar{x}_2 - 4\bar{x}_2 + \frac{9}{2} + 11) \end{aligned}$$

Subject to.

Constraints of the problem (P_{61}).

Step 6: According to Step 6 of the modified approach, there is a need to find a crisp optimal solution of the crisp single-objective non-linear programming problem (P'_{611}) and (P'_{621}).

- (i) On solving the crisp single-objective non-linear programming problem (P'_{611}), the obtained optimal solution is $\underline{x}_{11} = 0.168$, $\bar{x}_{11} = 0.168$, $\underline{x}_{12} = 0.778$, $\bar{x}_{12} = 0.778$, $\underline{x}_{21} = 1.548$, $\bar{x}_{21} = 1.548$, $\underline{x}_{22} = 1.318$, $\bar{x}_{22} = 1.318$, $\underline{x}_1 = 1.256$, $\bar{x}_1 = 1.545$, $\underline{x}_2 = 1.389$, $\bar{x}_2 = 1.809$.
- (ii) On solving the crisp single-objective non-linear programming problem (P'_{621}), the obtained optimal solution is $\underline{x}_{11} = 0.158$, $\bar{x}_{11} = 0.158$, $\underline{x}_{12} = 1.986$, $\bar{x}_{12} = 1.986$, $\underline{x}_{21} = 1.710$, $\bar{x}_{21} = 1.710$, $\underline{x}_{22} = 1.153$, $\bar{x}_{22} = 1.153$, $\underline{x}_1 = 1.384$, $\bar{x}_1 = 1.681$, $\underline{x}_2 = 1.999$, $\bar{x}_2 = 2.627$.

Step 7: Using Step 7 of the modified approach,

- (i) The interval optimal solution of the interval single-objective non-linear programming problem (P_{31}) is $[\underline{x}_{11}, \bar{x}_{11}] = [0.168, 0.168]$, $[\underline{x}_{12}, \bar{x}_{12}] = [0.778, 0.778]$, $[\underline{x}_{21}, \bar{x}_{21}] = [1.548, 1.548]$, $[\underline{x}_{22}, \bar{x}_{22}] = [1.318, 1.318]$, $[\underline{x}_1, \bar{x}_1] = [1.256, 1.545]$, $[\underline{x}_2, \bar{x}_2] = [1.389, 1.809]$.
- (ii) The interval optimal solution of the interval single-objective non-linear programming problem (P_{32}) is $[\underline{x}_{11}, \bar{x}_{11}] = [0.158, 0.158]$, $[\underline{x}_{12}, \bar{x}_{12}] = [1.986, 1.986]$, $[\underline{x}_{21}, \bar{x}_{21}] = [1.710, 1.710]$, $[\underline{x}_{22}, \bar{x}_{22}] = [1.153, 1.153]$, $[\underline{x}_1, \bar{x}_1] = [1.384, 1.681]$, $[\underline{x}_2, \bar{x}_2] = [1.999, 2.627]$.

Step 8: According to Step 8 of the modified approach,

- (i) The value of the objective function ($[0.8, 0.9] \otimes [\underline{x}_{11}^2, \bar{x}_{11}^2] + [0.75, 0.95] \otimes [\underline{x}_{12}^2, \bar{x}_{12}^2] + [0.35, 0.65] \otimes [\underline{x}_{21}^2, \bar{x}_{21}^2] + [0.4, 0.7] \otimes [\underline{x}_{22}^2, \bar{x}_{22}^2] + [0.2, 0.5] \otimes [\underline{x}_1^2, \bar{x}_1^2] + [-5, -3] \otimes [\underline{x}_1, \bar{x}_1] + [5, 10] + [0.15, 0.8] \otimes [\underline{x}_2^2, \bar{x}_2^2] + [-7, -4] \otimes [\underline{x}_2, \bar{x}_2] + [4, 10]$) of the interval single-objective non-linear programming problem (P_{31}) corresponding to the obtained optimal solution $[\underline{x}_{11}, \bar{x}_{11}] = [0.168, 0.168]$, $[\underline{x}_{12}, \bar{x}_{12}] = [0.778, 0.778]$, $[\underline{x}_{21}, \bar{x}_{21}] = [1.548, 1.548]$, $[\underline{x}_{22}, \bar{x}_{22}] = [1.318, 1.318]$, $[\underline{x}_1, \bar{x}_1] = [1.256, 1.545]$, $[\underline{x}_2, \bar{x}_2] = [1.389, 1.809]$ is $[\underline{f}_1, \bar{f}_1] = [-8.773, 17.861]$.
- (ii) The value of the objective function ($[0.7, 0.9] \otimes [\underline{x}_{11}^2, \bar{x}_{11}^2] + [0.15, 0.3] \otimes [\underline{x}_{12}^2, \bar{x}_{12}^2] + [0.5, 0.7] \otimes [\underline{x}_{21}^2, \bar{x}_{21}^2] + [0.4, 0.6] \otimes [\underline{x}_{22}^2, \bar{x}_{22}^2] + [0.3, 0.5] \otimes [\underline{x}_1^2, \bar{x}_1^2] + [-5, -4] \otimes [\underline{x}_1, \bar{x}_1] + [8, 10] + [0.7, 0.9] \otimes [\underline{x}_2^2, \bar{x}_2^2] + [-8, -6] \otimes [\underline{x}_2, \bar{x}_2] + [4, 9]$) of the interval single-objective non-linear programming problem (P_{32}) corresponding to the obtained optimal solution $[\underline{x}_{11}, \bar{x}_{11}] = [0.158, 0.158]$, $[\underline{x}_{12}, \bar{x}_{12}] = [1.986, 1.986]$, $[\underline{x}_{21}, \bar{x}_{21}] = [1.710, 1.710]$, $[\underline{x}_{22}, \bar{x}_{22}] = [1.153, 1.153]$, $[\underline{x}_1, \bar{x}_1] = [1.384, 1.681]$, $[\underline{x}_2, \bar{x}_2] = [1.999, 2.627]$ is $[\underline{f}_2, \bar{f}_2] = [-11.447, 13.144]$.

Step 9: Using $\underline{f}_1 = -8.773$, $\bar{f}_1 = 17.861$, $\underline{f}_2 = -11.447$ and $\bar{f}_2 = 13.144$ in Step 9 of Dalman and Sivri's approach [6],

$$(i) \quad \mu_1(x) = \begin{cases} 1, & \underline{f}_1(x) \leq -8.773 \\ \frac{17.861 - \underline{f}_1(x)}{17.861 - (-8.773)}, & -8.773 < \underline{f}_1(x) < 17.861 \\ 0, & \underline{f}_1(x) \geq 17.861 \end{cases}$$

where

$$\underline{f}_1(x) = 0.8x_{11}^2 + 0.75x_{12}^2 + 0.35x_{21}^2 + 0.4x_{22}^2 + 0.2x_1^2 - 5\bar{x}_1 + 0.15x_2^2 - 7\bar{x}_2 + 9.$$

$$(ii) \quad \mu_{1C}(x) = \begin{cases} 1, & f_{1C}(x) \leq -8.773 \\ \frac{17.861 - f_{1C}(x)}{17.861 - (-8.773)}, & -8.773 < f_{1C}(x) < 17.861 \\ 0, & f_{1C}(x) \geq 17.861 \end{cases}$$

where

$$f_{1C}(x) = 0.4x_{11}^2 + \frac{0.9}{2}x_{11}^2 + \frac{0.75}{2}x_{12}^2 + \frac{0.95}{2}x_{12}^2 + \frac{0.35}{2}x_{21}^2 + \frac{0.65}{2}x_{21}^2 + 0.2x_{22}^2 \\ + \frac{0.7}{2}x_{22}^2 + 0.1x_1^2 + \frac{0.5}{2}x_1^2 - \frac{3}{2}x_1 - \frac{5}{2}x_1 + \frac{0.15}{2}x_2^2 + 0.40x_2^2 - 2x_2 - \frac{7}{2}x_2 + \frac{5}{2} + 12.$$

$$(iii) \quad \mu_2(x) = \begin{cases} 1, & \underline{f}_2(x) \leq -11.447 \\ \frac{13.144 - \underline{f}_2(x)}{13.144 - (-11.447)}, & -11.447 < \underline{f}_2(x) < 13.144 \\ 0, & \underline{f}_2(x) \geq 13.144 \end{cases}$$

where

$$\underline{f}_2(x) = 0.7x_{11}^2 + 0.15x_{12}^2 + 0.5x_{21}^2 + 0.4x_{22}^2 + 0.3x_1^2 - 5\bar{x}_1 + 0.7x_2^2 - 8\bar{x}_2 + 12.$$

$$(iv) \quad \mu_{2C}(x) = \begin{cases} 1, & f_{2C}(x) \leq -11.447 \\ \frac{13.144 - f_{2C}(x)}{13.144 - (-11.447)}, & -11.447 < f_{2C}(x) < 13.144 \\ 0, & f_{2C}(x) \geq 13.144 \end{cases}$$

where

$$f_{2C}(x) = \frac{0.7}{2}x_{11}^2 + \frac{0.9}{2}x_{11}^2 + \frac{0.15}{2}x_{12}^2 + \frac{0.3}{2}x_{12}^2 + \frac{0.5}{2}x_{21}^2 + \frac{0.7}{2}x_{21}^2 + 0.2x_{22}^2 + 0.3x_{22}^2 \\ + \frac{0.3}{2}x_1^2 + \frac{0.5}{2}x_1^2 - 2x_1 - \frac{5}{2}x_1 + \frac{0.7}{2}x_2^2 + \frac{0.9}{2}x_2^2 - 3x_2 - 4x_2 + \frac{9}{2} + 11.$$

Step 10: According to Step 10 of Dalman and Sivri's approach [6], there is a need to solve the crisp single-objective non-linear programming problem (P_5).

Problem (P₅)

$$\max \left\{ \begin{aligned} & \frac{17.861 - f_1(x)}{17.861 - (-8.773)} + \frac{17.861 - f_{1C}(x)}{17.861 - (-8.773)} + \frac{13.144 - f_2(x)}{13.144 - (-11.447)} \\ & + \frac{13.144 - f_{2C}(x)}{13.144 - (-11.447)} \end{aligned} \right\}$$

Subject to

$$\underline{f}_1(x) \geq -8.773,$$

$$f_{1C}(x) \geq -8.773,$$

$$\underline{f}_2(x) \geq -11.447,$$

$$f_{2C}(x) \geq -11.447,$$

and

Constraints of the problem (P₆₁).

On solving the crisp single-objective non-linear programming problem (P₅), the obtained optimal solution is $\underline{x}_{11} = 0.275$, $\bar{x}_{11} = 0.275$, $x_{12} = 3.071$, $\bar{x}_{12} = 3.071$, $\underline{x}_{21} = 1.355$, $\bar{x}_{21} = 1.355$, $\underline{x}_{22} = 0.306$, $\bar{x}_{22} = 0.306$, $\underline{x}_1 = 1.111$, $\bar{x}_1 = 1.467$, $\underline{x}_2 = 2.057$, $\bar{x}_2 = 2.733$ and the obtained optimal value is 2.828.

Step 11: Using Step 11 of Dalman and Sivri’s approach [6], the interval optimal solution of the interval bi-objective problem non-linear programming problem (P₃) is $[\underline{x}_{11}, \bar{x}_{11}] = [0.275, 0.275]$, $[\underline{x}_{12}, \bar{x}_{12}] = [3.071, 3.071]$, $[\underline{x}_{21}, \bar{x}_{21}] = [1.355, 1.355]$, $[\underline{x}_{22}, \bar{x}_{22}] = [0.306, 0.306]$, $[\underline{x}_1, \bar{x}_1] = [1.111, 1.467]$, $[\underline{x}_2, \bar{x}_2] = [2.057, 2.733]$ and the interval optimal value of the first and second objective functions of the interval bi-objective non-linear programming problem (P₃) is $[\underline{f}_1, \bar{f}_1] = [-8.771, 25.776]$ and $[\underline{f}_2, \bar{f}_2] = [-11.444, 10.195]$, respectively.

6 Conclusions

In this paper, a mathematical incorrect assumption, considered in one of the steps of Dalman and Sivri’s approach [6], has been pointed out. Also, the required modifications have been suggested followed by an illustrative numerical example.

In the future, one may generalize the suggested modifications to solve an interval multi-objective non-linear transportation problem having unknown interval demand as well as unknown interval supply.

References

1. Das, S.K., Goswami, A., Alam, S.S.: Multiobjective transportation problem with interval cost, source and destination parameters. *Eur. J. Oper. Res.* **117**(1), 100–112 (1999)
2. Yu, V.F., Hu, K.J., Chang, A.Y.: An interactive approach for the multi-objective transportation problem with interval parameters. *Int. J. Prod. Res.* **53**(4), 1051–1064 (2015)
3. Hajiagha, S.H.R., Mahdiraji, H.A., Hashemi, S.S.: Multi-objective linear programming with interval coefficients: a fuzzy set based approach. *Kybernetes* **42**(3), 482–496 (2013)
4. Güzel, N., Emiroglu, Y., Tapci, F., Guler, C., Syvry, M.: A solution proposal to the interval fractional transportation problem. *Appl. Math. Inf. Sci.* **6**(3), 567–571 (2012)
5. Dalman, H., Köçken, H.G., Sivri, M.: A solution proposal to indefinite quadratic interval transportation problem. *New Trends Math. Sci.* **1**(2), 7–12 (2013)
6. Dalman, H., Sivri, M.: A fuzzy logic based approach to solve interval multiobjective nonlinear transportation problem. *Proc. Natl. Acad. Sci., India, Sect. A* **89**(2), 279–289 (2019)
7. Rahman, M.S., Shaikh, A.A., Ali, I., Bhunia, A.K., Fügenschuh, A.: A theoretical framework for optimality conditions of nonlinear type-2 interval-valued unconstrained and constrained optimization problems using type-2 interval order relations. *Mathematics* **9**(8), 908 (2021)
8. Khan, M.F., Haq, A., Ahmed, A., Ali, I.: Multiobjective multi-product production planning problem using intuitionistic and neutrosophic fuzzy programming. *IEEE Access* **9**, 37466–37486 (2021)
9. Gupta, S., Ali, I., Ahmed, A.: An extended multi-objective capacitated transportation problem with mixed constraints in fuzzy environment. *Int. J. Oper. Res.* **37**(3), 345–376 (2020)
10. Kamal, M., Gupta, S., Ali, I.: A decentralised multi-objective sustainable supply chain model under intuitionistic fuzzy environment. *Int. J. Math. Oper. Res.* **16**(3), 376–406 (2020)
11. Gupta, S., Ali, I., Ahmed, A.: Efficient fuzzy goal programming model for multi-objective production distribution problem. *Int. J. Appl. Comput. Math.* **4**(2), 76 (2018)
12. Gupta, S., Ali, I., Ahmed, A.: Multi-choice multi-objective capacitated transportation problem—a case study of uncertain demand and supply. *J. Stat. Manage. Syst.* **21**(3), 467–491 (2018)
13. Gupta, S., Ali, I., Ahmed, A.: Multi-objective vendor selection problem of supply chain management under fuzzy environment. *J. Oper. Res. Soc. China* **9**(1), 33–62 (2021)
14. Gupta, S., Ali, I., Ahmed, A.: Multi-objective bi-level supply chain network order allocation problem under fuzziness. *Opsearch* **55**(3), 721–748 (2018)
15. Nomani, M.A., Ali, I., Ahmed, A.: A new approach for solving multi-objective transportation problems. *Int. J. Manage. Sci. Eng. Manage.* **12**(3), 165–173 (2017)
16. Ahmadini, A.A.H., Modibbo, U.M., Shaikh, A.A., Ali, I.: Multi-objective optimization modelling of sustainable green supply chain in inventory and production management. *Alex. Eng. J.* **60**(6), 5129–5146 (2021)
17. Ali, I., Fügenschuh, A., Gupta, S., Modibbo, U.M.: The LR-Type fuzzy multi-objective vendor selection problem in supply chain management. *Mathematics* **8**(9), 1621 (2020)
18. Gupta, S., Ali, I., Chaudhary, S.: Multi-objective capacitated transportation: a problem of parameters estimation, goodness of fit and optimization. *Granul. Comput.* **5**(1), 119–134 (2020)
19. Ali, I., Gupta, S., Ahmed, A.: Multi-objective linear fractional inventory problem under intuitionistic fuzzy environment. *Int. J. Syst. Assur. Eng. Manage.* **10**(2), 173–189 (2019)
20. Charles, V., Gupta, S., Ali, I.: A fuzzy goal programming approach for solving multi-objective supply chain network problems with pareto-distributed random variables. *Int. J. Unc. Fuzz. Knowl. Based Syst.* **27**(04), 559–593 (2019)
21. Gupta, S., Ali, I., Ahmed, A.: Multi-objective capacitated transportation problem with mixed constraint: a case study of certain and uncertain environment. *Opsearch* **55**(2), 447–477 (2018)
22. Fishburn, P.C.: Additive utilities with incomplete product set: Applications to priorities and assignments. *Oper. Res.* **15**(3), 537–542 (1967)

Interval Type-3 Fuzzy Decision-Making in Material Surface Quality Control



Oscar Castillo and Patricia Melin

Abstract This article is presenting a proposal for the mathematical definition of interval type-3 fuzzy sets, as a basis for later constructing all the theory needed for building interval type-3 fuzzy systems for different areas of application. It is well known that most used form of fuzzy logic is the one called type-1, which later evolved into interval and general type-2. Recently, we have seen that type-2 fuzzy has outperformed type-1 in many real-world problems when there is more uncertainty, dynamic situations or nonlinearity. For these reasons, it is interesting to explore the new area of type-3 fuzzy theory, and in this paper, we are dealing with the application side of this theory. We illustrate the potential of interval type-3 in a decision-making application in surface quality control. The proposed method uses interval type-3 fuzzy to automate quality control in material manufacturing. Surface roughness and porosity of the materials are utilized to estimate the quality of the material with a type-3 fuzzy approach. Material manufacturing requires strict quality control, and a type-3 fuzzy approach is able to outperform type-2 and type-1 in surface quality control.

Keywords Type-3 fuzzy logic · Type-2 fuzzy systems · Surface quality control

1 Introduction

In artificial intelligence, it is well accepted that fuzzy systems can have successful applications in areas, such as control, recognition, medical diagnosis and prediction. Originally, fuzzy sets (now called type-1) were proposed by Lotfi Zadeh in 1965. Later, fuzzy logic and fuzzy systems were also proposed by Zadeh, and many applications follow, mainly in control [1]. Type-1 fuzzy systems evolved to type-2 fuzzy systems with the works by Mendel in 2001 [2]. Initially, interval type-2 fuzzy systems were studied and applied to several problems [3]. These systems were applied to many problems in areas such as robotics, intelligent control and others [4, 5]. Simulation

O. Castillo (✉) · P. Melin
Tijuana Institute of Technology, Tijuana, BC, Mexico
e-mail: ocastillo@tectijuana.mx

and experimental results show that interval type-2 outperforms type-1 fuzzy systems in situations with higher levels of noise, dynamic environments or highly nonlinear problems [6–8]. Later, general type-2 fuzzy systems were considered to manage higher levels of uncertainty, and good results have been achieved in several areas of application [9–11]. Recently, it is becoming apparent that type-3 fuzzy systems could help solve even more complex problems. For this reason, in this paper we are putting forward the basic constructs of type-3 fuzzy systems by extending the ideas of type-2 fuzzy systems [12–14]. The main motivation is to continue improving the potential of fuzzy theory by constructing more powerful tools, and we believe that type-3 fuzzy will be helpful in this regard. Type-3 fuzzy will help to represent the uncertainty in determining type-2 fuzzy sets, which is expected to improve the results in complex problems.

The key contribution is the proposal of mathematical definitions of interval type-3 fuzzy sets, which were obtained by using the extension principle on the type-2 fuzzy sets definitions. In addition, the utilization of interval type-3 fuzzy in surface quality control in intelligent manufacturing has not been presented before in the literature. We consider that these are important contributions to the frontier knowledge in the soft computing area.

The remaining of the article is structured as follows: Sect. 2 is presenting the proposed definitions of interval type-3 fuzzy sets; in Sect. 3, we are outlining the proposed type-3 fuzzy system design; in Sect. 4, an application to material quality estimation based on a fuzzy system is presented; and in Sect. 5, we are presenting the conclusions that we have reached with the research done in this work.

2 Interval Type-3 Fuzzy Sets

We start by defining an interval type-3 fuzzy sets and related concepts.

Definition 1. A type-3 fuzzy sets (T3 FS) [15–17], denoted by $A^{(3)}$, are represented by the plot of a trivariate function, called membership function (MF) of $A^{(3)}$, in the Cartesian product $X \times [0, 1] \times [0, 1]$ in $[0, 1]$, where X is the universe of the primary variable of $A^{(3)}$, x . The MF of $\mu_{A^{(3)}}$ is denoted by $\mu_{A^{(3)}}(x, u, v)$ (or $\mu_{A^{(3)}}$ to abbreviate), and it is called a type-3 membership function (T3 MF) of the T3 FS. In other words, more formally,

$$\mu_{A^{(3)}} : X \times [0, 1] \times [0, 1] \rightarrow [0, 1]$$

$$A^{(3)} = \left\{ \begin{array}{l} (x, u(x), v(x, u), \mu_{A^{(3)}}(x, u, v)) \\ | x \in X, u \in U \subseteq [0, 1], v \in V \subseteq [0, 1] \end{array} \right\} \quad (1)$$

where U is the universe for the secondary variable u and V is the universe for tertiary variable v . A T3 FS, $A^{(3)}$, can also be expressed in continuous notation as follows:

$$A^{(3)} = \int_{x \in X} \int_{u \in [0,1]} \int_{v \in [0,1]} \mu_{A^{(3)}}(x, u, v) / (x, u, v) \tag{2}$$

$$A^{(3)} = \int_{x \in X} \left[\int_{u \in [0,1]} \left[\int_{v \in [0,1]} \mu_{A^{(3)}}(x, u, v) / v \right] / u \right] / x \tag{3}$$

where $\int \int \int$ is notation for the union over all the admissible x, u, v values.

Equation (3) is represented as a mapping of the T3 FS membership functions with the following equations:

$$A^{(3)} = \int_{x \in X} \mu_{A_x^{(3)}}(u, v) / x$$

$$\mu_{A_x^{(3)}}(u, v) = \int_{u \in [0,1]} \mu_{A_{(x,u)}^{(3)}}(v) / u$$

$$\mu_{A_{(x,u)}^{(3)}}(v) = \int_{v \in [0,1]} \mu_{A^{(3)}}(x, u, v) / v$$

where $\mu_{A_x^{(3)}}(u, v)$ is the primary MF, $\mu_{A_{(x,u)}^{(3)}}(v)$ is the secondary MF, and $\mu_{A^{(3)}}(x, u, v)$ is the tertiary MF of the T3 FS.

If $\mu_{A^{(3)}}(x, u, v) = 1$, the T3 FS, $A^{(3)}$, is reduced to an interval type-3 fuzzy set (IT3 FS) with the notation \mathbb{A} , defined by Eq. (4).

$$\mathbb{A} = \int_{x \in X} \left[\int_{u \in [0,1]} \left[\int_{v \in [\underline{\mu}_{\mathbb{A}}(x,u), \overline{\mu}_{\mathbb{A}}(x,u)]} 1 / v \right] / u \right] / x \tag{4}$$

where

$$\mu_{\mathbb{A}(x,u)}(v) = \int_{v \in [\underline{\mu}_{\mathbb{A}}(x,u), \overline{\mu}_{\mathbb{A}}(x,u)]} 1 / v$$

$$\mu_{\mathbb{A}(x)}(u, v) = \int_{u \in [0,1]} \left[\int_{v \in [\underline{\mu}_{\mathbb{A}}(x,u), \overline{\mu}_{\mathbb{A}}(x,u)]} 1 / v \right] / u$$

$$\mathbb{A} = \int_{x \in X} \mu_{\mathbb{A}(x)}(u, v) / x$$

Assuming that $v \in [\underline{\mu}_{\mathbb{A}}(x, u), \overline{\mu}_{\mathbb{A}}(x, u)]$ and the lower and upper MFs $\underline{\mu}_{\mathbb{A}}(x, u)$, $\overline{\mu}_{\mathbb{A}}(x, u)$ are general type-2 membership functions (T2 MF) over the plane (x, u) , Eq. (4) can be simplified as a bivariate isosurface simplification with an interval type-3 membership function (IT3 MF), $\tilde{\mu}_{\mathbb{A}}(x, u) \in [\underline{\mu}_{\mathbb{A}}(x, u), \overline{\mu}_{\mathbb{A}}(x, u)]$, defined by Eq. (5).

$$\mathbb{A} = \int_{x \in X} \int_{u \in [0,1]} \tilde{\mu}_{\mathbb{A}}(x, u)/(x, u) \quad (5)$$

where the lower T2 MF $\underline{\mu}_{\mathbb{A}}(x, u)$, is contained in the upper T2 MF $\overline{\mu}_{\mathbb{A}}(x, u)$, this is $\underline{\mu}_{\mathbb{A}}(x, u) \subseteq \overline{\mu}_{\mathbb{A}}(x, u)$, then $\underline{\mu}_{\mathbb{A}}(x, u) \leq \overline{\mu}_{\mathbb{A}}(x, u)$, and as a consequence, an IT3 FS is represented by two T2 FSs, one inferior $\underline{\mathbb{A}}$ with T2 MF $\underline{\mu}_{\mathbb{A}}(x, u)$ and another superior $\overline{\mathbb{A}}$, with T2 MF $\overline{\mu}_{\mathbb{A}}(x, u)$ defined by Eqs. (6) and (7) (see Fig. 1)

$$\underline{\mathbb{A}} = \int_{x \in X} \int_{u \in [0,1]} \underline{\mu}_{\mathbb{A}}(x, u)/(x, u) = \int_{x \in X} \left[\int_{u \in [0,1]} \underline{f}_x(u)/u \right] / x \quad (6)$$

$$\overline{\mathbb{A}} = \int_{x \in X} \int_{u \in [0,1]} \overline{\mu}_{\mathbb{A}}(x, u)/(x, u) = \int_{x \in X} \left[\int_{u \in [0,1]} \overline{f}_x(u)/u \right] / x \quad (7)$$

where the secondary MFs of $\underline{\mathbb{A}}$ and $\overline{\mathbb{A}}$ are T1 MFs of T1FS given by the Eqs. (8) and (9)

$$\mu_{\underline{\mathbb{A}}(x)}(u) = \int_{u \in J_x} \underline{f}_x(u)/u \quad (8)$$

$$\mu_{\overline{\mathbb{A}}(x)}(u) = \int_{u \in J_x} \overline{f}_x(u)/u \quad (9)$$

In this work, we utilize interval type-3 MFs that are scaled Gaussians in the primary and secondary, respectively. This function can be represented as $\tilde{\mu}_{\mathbb{A}}(x, u) = \text{ScaleGaussScaleGauss IT3MF}$, with Gaussian footprint of uncertainty $FOU(\mathbb{A})$, characterized with parameters $[\sigma, m]$ (UpperParameters) for the upper membership function UMF and for the lower membership function LMF the parameters λ (LowerScale) and ℓ (LowerLag) to form the $DOU = [\underline{\mu}(x), \overline{\mu}(x)]$. The vertical cuts $\mathbb{A}_{(x)}(u)$ characterize the $FOU(\mathbb{A})$, and are IT2 FSs with Gaussian IT2 MFs, $\mu_{\mathbb{A}(x)}(u)$ with parameters $[\sigma_u, m(x)]$ for the UMF and LMF λ (LowerScale) and ℓ (LowerLag). The IT3 MF, $\tilde{\mu}_{\mathbb{A}}(x, u) = \text{ScaleGaussScaleGaussIT3MF}(x, \{[\sigma, m]\}, \lambda, \ell)$ is described with the following equations:

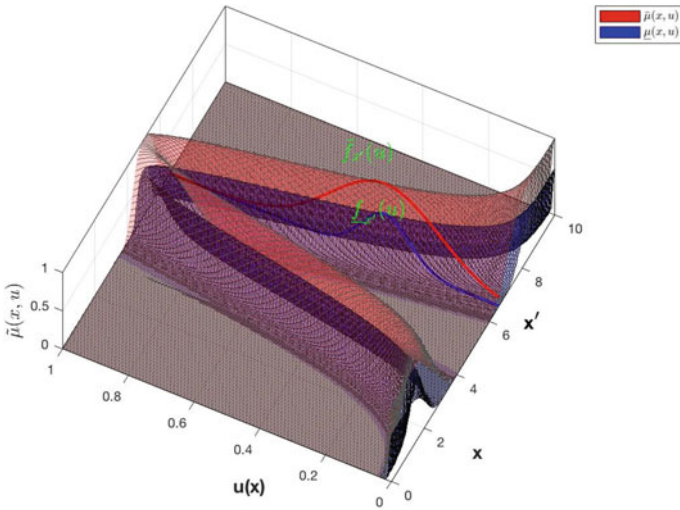


Fig. 1 IT3 FS with IT3MF $\tilde{\mu}(x, u)$, where $\underline{\mu}(x, u)$ is the LMF and $\overline{\mu}(x, u)$ is the UMF. The embedded secondary T1 MFs in x' of \underline{A} and \overline{A} are $\underline{f}_{x'}(u)$ and $\overline{f}_{x'}(u)$, respectively (the footprint of uncertainty FOU is inside)

$$\overline{u}(x) = \exp \left[-\frac{1}{2} \left(\frac{x - m}{\sigma} \right)^2 \right] \tag{10}$$

$$\underline{u}(x) = \lambda \cdot \exp \left[-\frac{1}{2} \left(\frac{x - m}{\sigma^*} \right)^2 \right] \tag{11}$$

where $\sigma^* = \sigma \sqrt{\frac{\ln(\ell)}{\ln(\varepsilon)}}$, and ε is the machine epsilon. If $\ell = 0$, then $\sigma^* = \sigma$. Then, $\overline{u}(x)$ and $\underline{u}(x)$ are the upper and lower limits of the DOU. The range $\delta(u)$ and radius σ_u of the FOU are:

$$\delta(u) = \overline{u}(x) - \underline{u}(x) \tag{12}$$

$$\sigma_u = \frac{\delta(u)}{2\sqrt{3}} + \varepsilon \tag{13}$$

The apex or core, $m(x)$, of the IT3 MF $\tilde{\mu}(x, u)$, is defined by the expression:

$$m(x) = \exp \left[-\frac{1}{2} \left(\frac{x - m}{\rho} \right)^2 \right] \tag{14}$$

where $\rho = (\sigma + \sigma^*)/2$. Then, the vertical cuts with IT2 MF, $\mu_{\mathbb{A}(x)}(u) = [\underline{\mu}_{\mathbb{A}(x)}(u), \bar{\mu}_{\mathbb{A}(x)}(u)]$, are described by the equations:

$$\bar{\mu}_{\mathbb{A}(x)}(u) = \exp\left[-\frac{1}{2}\left(\frac{u - u(x)}{\sigma_u}\right)^2\right] \tag{15}$$

$$\underline{\mu}_{\mathbb{A}(x)}(u) = \lambda \cdot \exp\left[-\frac{1}{2}\left(\frac{u - u(x)}{\sigma_u^*}\right)^2\right] \tag{16}$$

where $\sigma_u^* = \sigma_u \sqrt{\frac{\ln(\ell)}{\ln(\varepsilon)}}$. If $\ell = 0$, then $\sigma_u^* = \sigma_u$. Then, $\bar{\mu}_{\mathbb{A}(x)}(u)$ and $\underline{\mu}_{\mathbb{A}(x)}(u)$ are the UMF and LMF of the IT2 FSs of the vertical cuts of the secondary IT2MF of the IT3 FS.

3 Interval Type-3 Fuzzy Systems

The structure of an interval type-3 fuzzy system, similarly to type-2 and type-1, is composed of a fuzzifier, fuzzy rules, inference, type reduction and defuzzifier. In Fig. 2, we can find the structure of an interval type-3 system.

In more complicated situations or problems, it is possible that a hierarchical approach is needed, like in Fig. 3 where several simpler fuzzy systems are used, and then, an aggregator (higher type) fuzzy system is needed to combine the outputs of the simpler systems. We have used this architecture with type-1 individual controllers and then a type-2 fuzzy aggregator [18, 19], but it is also possible that type-2 fuzzy controllers can be aggregated by a type-3 fuzzy system. In Fig. 3, the individual fuzzy systems are shown with type $n-1$ and the aggregator with type- n .

In the next section, we describe in detail an application of interval type-3 fuzzy systems to illustrate the design method and also show the improvements on results with respect to type-1 and type-2.

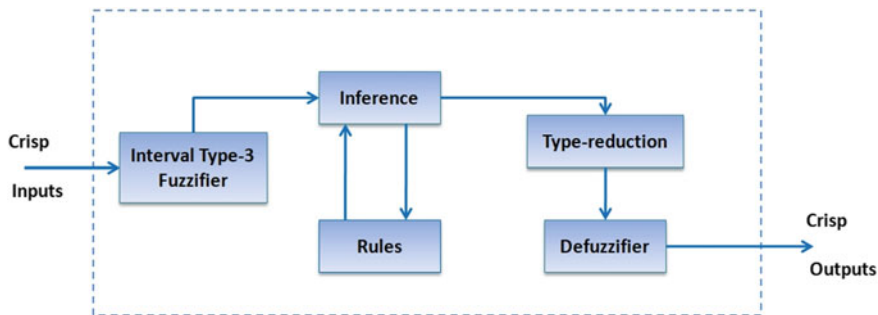


Fig. 2 Structure of an interval type-3 system

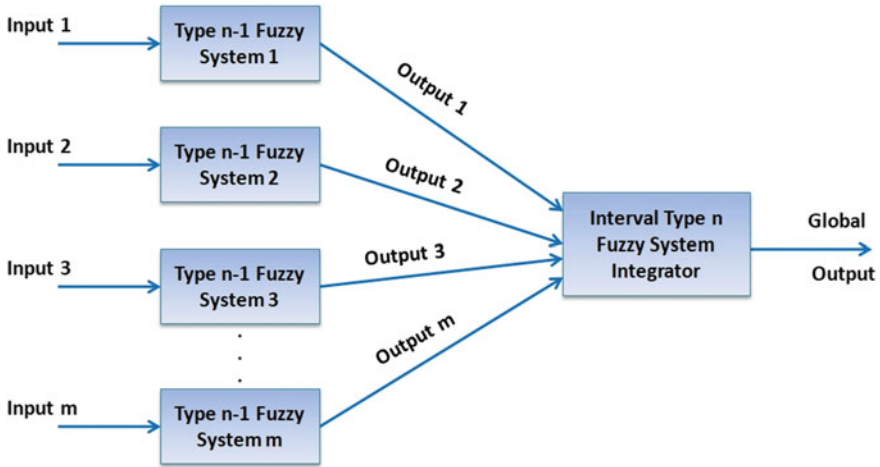


Fig. 3 Hierarchical architecture of multiple fuzzy systems

4 Decision-Making Application: Surface Quality Control

Fuzzy logic can be utilized to achieve automated quality control in many manufacturing areas. In this case, a set of fuzzy rules is established to relate relevant physical characteristics of the material (to be manufactured) with the quality control of the product. The fuzzy rules represent the knowledge of experts for the specific domain of material processing. If we assume that the product quality is a linguistic variable with values: bad, regular and excellent, we can then put forward a fuzzy rule base for surface quality control, as illustrated in Table 1. The input variables are roughness and porosity, both with linguistic values: bad, medium and good.

Table 1 Fuzzy rules for surface quality control

If	And	Then
Roughness	Porosity	Quality
Good	Good	Excellent
Good	Medium	Regular
Medium	Good	Regular
Medium	Medium	Regular
Medium	Bad	Bad
Bad	Medium	Bad
Bad	Bad	Bad
Good	Bad	Regular
Bad	Good	Regular

The interpretation of this fuzzy system is that if the surface roughness is good and porosity is good, then the product quality can be considered excellent. Other fuzzy rules can be interpreted in a similar way. We show in Fig. 4 the particular structure of the fuzzy system. We show in Fig. 5 the Gaussian interval type-3 MFs of the roughness input linguistic variable. In Fig. 6, we illustrate in a similar way the MFs for the porosity input linguistic variable. In Fig. 7, we show the MFs of quality output variable in the plane and in Fig. 8 with a 3D view. In Fig. 9, the nonlinear surface representing the complete fuzzy model is illustrated. In Fig. 10, we illustrate the 3D nature of interval type-3 membership functions with the FOU shown in green color. We have to mention that the type reduction and defuzzification methods are the same as the ones used in type-2 fuzzy theory. In this case, we use center of sets as type reduction and centroid as defuzzification.

In Table 2, we show a summary of some results obtained with the interval type-3 fuzzy approach for estimating the quality of a material based on the values of roughness and porosity.

The estimated quality in each case was compared with the human experts in quality control in the manufacturing of an oxide layer surface, and the average error was of less than 1%, which is lower than we have previously achieved with type-1 and type-2 fuzzy approaches [20].

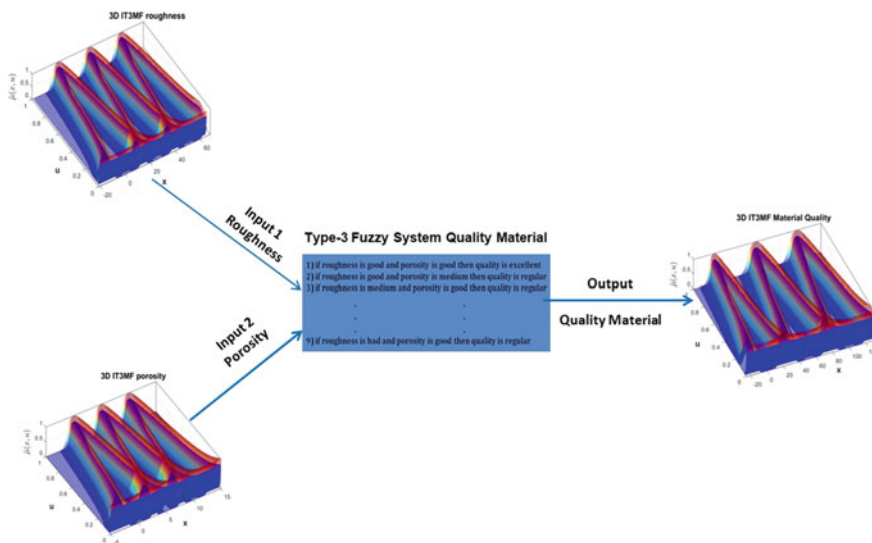


Fig. 4 Structure of type-3 system for material quality evaluation

Fig. 5 Membership functions for roughness input variable

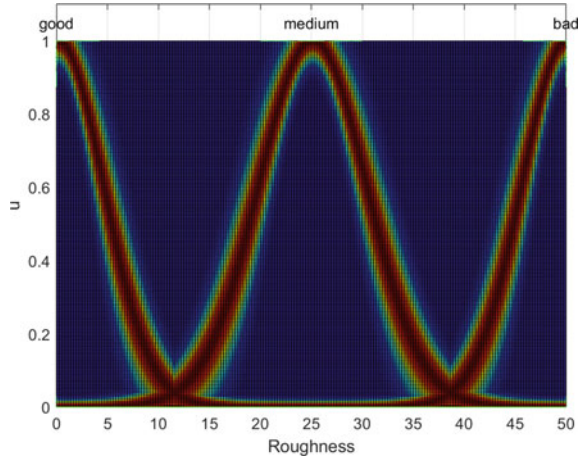
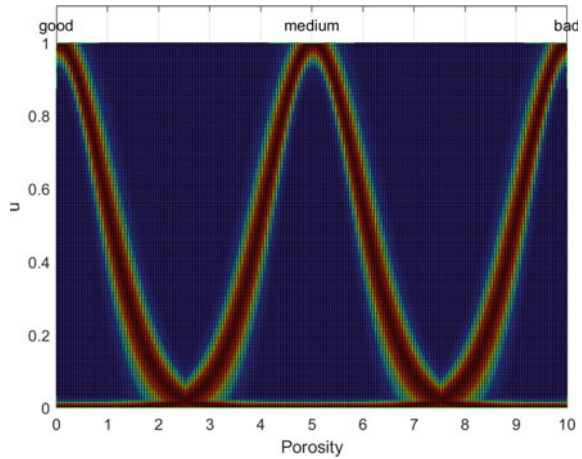


Fig. 6 Membership functions for porosity input variable



5 Conclusions

This article presented a proposal for the mathematical definition of interval type-3 fuzzy sets, as a basis for later constructing all the theory needed for building interval type-3 systems for different areas of application. It is well known that most used form of fuzzy logic is the one called type-1, which later evolved into interval and general type-2. Recently, we have noticed that type-2 fuzzy has outperformed type-1 in real-world problems when there is high level of uncertainty, dynamic situations or nonlinearity. For these reasons, it is interesting to explore the new area of type-3 fuzzy logic, and in this paper, we have proposed some mathematical definitions of type-3 fuzzy sets, as a basis for later proposing their operations. The main idea in the proposal of the interval type-3 fuzzy definitions was the use of extension principle

Fig. 7 Membership functions for quality output variable

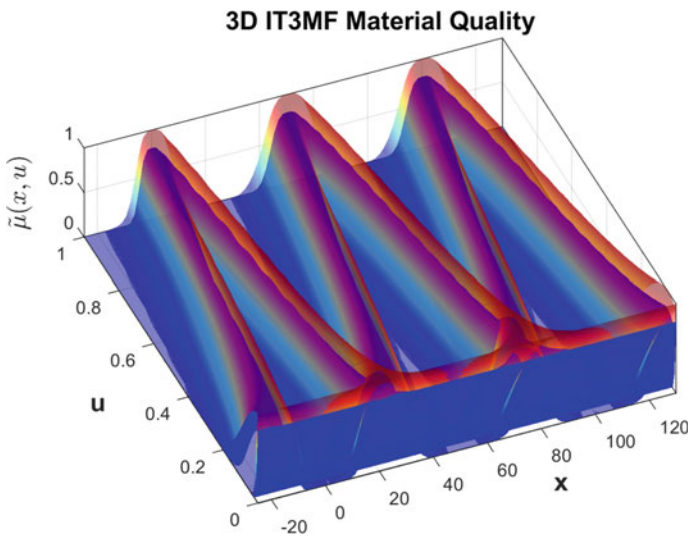
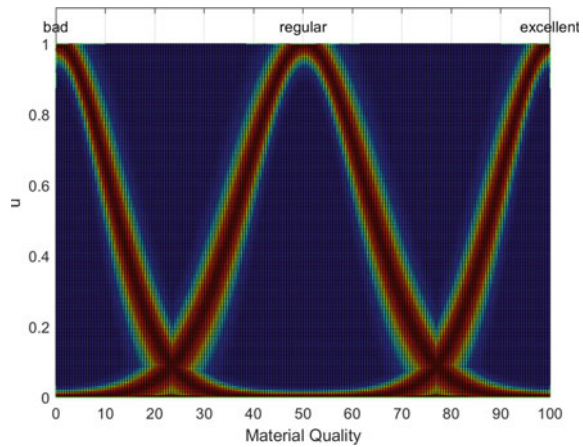


Fig. 8 Three-dimensional (3D) view of output MFs

and the previous existing definitions of general and interval type-2. In addition, we illustrate the potential of interval type-3 fuzzy in a decision-making application in surface quality control. Material manufacturing requires strict quality control, and a type-3 fuzzy system is able to outperform type-2 and type-1 in surface quality control. We have to mention that a limitation of this study is that the design of the fuzzy system was not optimized, which we plan to consider in the short term to improve results. As future work, we will continue constructing all the basis for type-3 fuzzy systems, and later their applications in different areas. Also, we have to say that we believe that interval type-3 could enhance the quality of the solutions in other

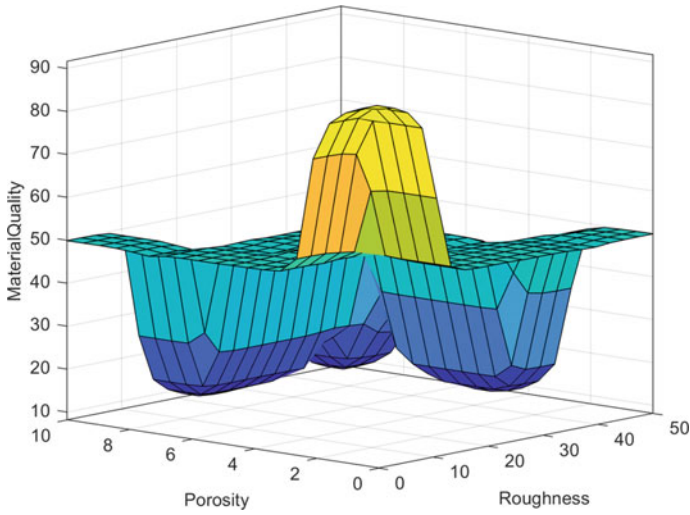


Fig. 9 Surface of the surface quality control fuzzy model

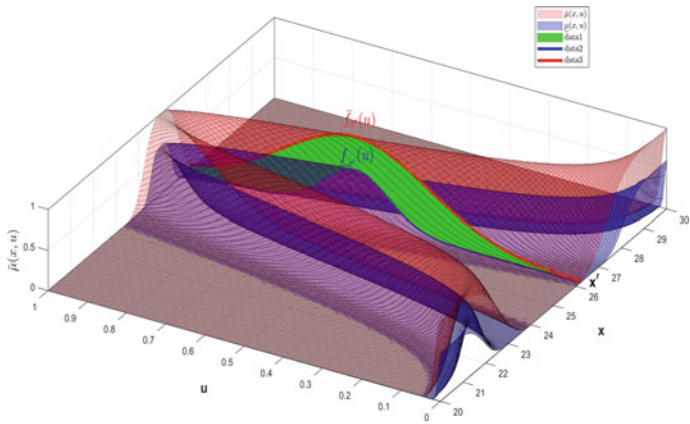


Fig. 10 Membership function as a 3D plot for illustration purposes

areas, such as in time series prediction [21], fuzzy clustering [22] and diagnosis [23], by enhancing the management of the uncertainty in the corresponding areas, and we envision working on these application areas in the near future for the benefit of society and economy. Finally, we envision in the long-term working in general on type-n fuzzy systems and their optimal design, as outlined in [24].

Table 2 Summary of results with the interval type-3 fuzzy model

Roughness	Porosity	Quality
0.2000	0.5000	91.9048
1.0000	0.1000	92.1451
3.6700	0.0000	91.0108
5.0000	0.2500	89.7966
10.2000	0.3000	72.6401
16.9700	0.3600	49.1708
18.4000	7.5000	39.8726
20.0000	6.0000	48.9439
25.0000	2.1000	48.8231
30.0000	8.1500	15.2604
40.0000	9.2000	23.3671
45.0000	6.3000	10.4162

References

1. Zadeh, L.A.: Knowledge representation in fuzzy logic. *IEEE Trans. Knowl. Data Eng.* **1**, 89 (1989)
2. Zadeh, L.A.: Fuzzy logic. *Computer* **1**(4), 83–93 (1998)
3. Mendel, J.M.: *Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions*. Prentice-Hall, Upper-Saddle River, NJ (2001)
4. Mendel, J.M.: *Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions*, 2nd edn. Springer (2017)
5. Karnik, N.N., Mendel, J.M.: Operations on type-2 fuzzy sets. *Fuzzy Sets Syst.* **122**, 327–348 (2001)
6. Moreno, J.E., et al.: Design of an interval type-2 fuzzy model with justifiable uncertainty. *Inf. Sci.* **513**, 206–221 (2020)
7. Mendel, J.M., Hagrais, H., Tan, W.-W., Melek, W.W., Ying, H.: *Introduction to Type-2 Fuzzy Logic Control*. NJ. Wiley and IEEE Press, Hoboken (2014)
8. Olivas, F., Valdez, F., Castillo, O., Melin, P.: Dynamic parameter adaptation in particle swarm optimization using interval type-2 fuzzy logic. *Soft. Comput.* **20**(3), 1057–1070 (2016)
9. Sakalli, A., Kumbasar, T., Mendel, J.M.: Towards systematic design of general type-2 fuzzy logic controllers: analysis, interpretation, and tuning. *IEEE Trans. Fuzzy Syst.* **29**(2), 226–239 (2021)
10. Ontiveros, E., Melin, P., Castillo, O.: High order α -planes integration: a new approach to computational cost reduction of general type-2 fuzzy systems. *Eng. Appl. Artif. Intell.* **74**, 186–197 (2018)
11. Castillo, O., Amador-Angulo, L.: A generalized type-2 fuzzy logic approach for dynamic parameter adaptation in bee colony optimization applied to fuzzy controller design. *Inf. Sci.* **460–461**, 476–496 (2018)
12. Cao, Y., Raise, A., Mohammadzadeh, A., et al.: Deep learned recurrent type-3 fuzzy system: application for renewable energy modeling/prediction. *Energy Rep.* **7**, 8115–8127 (2021)
13. Mohammadzadeh, A., Castillo, O., Band, S.S., et al.: A novel fractional-order multiple-model type-3 fuzzy control for nonlinear systems with unmodeled dynamics. *Int. J. Fuzzy Syst.* (2021). <https://doi.org/10.1007/s40815-021-01058-1>
14. Qasem, S.N., Ahmadian, A., Mohammadzadeh, A., Rathinasamy, S., Pahlevanzadeh, B.: A type-3 logic fuzzy system: optimized by a correntropy based Kalman filter with adaptive fuzzy kernel size *Inform. Sci.* **572**, 424–443 (2021)

15. Rickard, J.T., Aisbett, J., Gibbon, G.: Fuzzy subthood for fuzzy sets of type-2 and generalized type- n . *IEEE Trans. Fuzzy Syst.* **17**(1), 50–60 (2009)
16. Mohammadzadeh, A., Sabzalian, M.H., Zhang, W.: An interval type-3 fuzzy system and a new online fractional-order learning algorithm: theory and practice. *IEEE Trans. Fuzzy Syst.* **28**(9), 1940–1950 (2020)
17. Liu, Z., Mohammadzadeh, A., Turabieh, H., Mafarja, M., Band, S.S., Mosavi, A.: A new online learned interval type-3 fuzzy control system for solar energy management systems. *IEEE Access* **9**, 10498–10508 (2021)
18. Cervantes, L., Castillo, O.: Type-2 fuzzy logic aggregation of multiple fuzzy controllers for airplane flight control. *Inf. Sci.* **324**, 247–256 (2015)
19. Castillo, O., Cervantes, L., Soria, J., Sanchez, M., Castro, J.R.: A generalized type-2 fuzzy granular approach with applications to aerospace. *Inf. Sci.* **354**, 165–177 (2016)
20. Castillo, O., Melin, P.: *Soft Computing and Fractal Theory for Intelligent Manufacturing*. Springer-Verlag, Heidelberg, Germany (2003)
21. Castillo, O., Castro, J.R., Melin, P., Rodriguez-Diaz, A.: Application of interval type-2 fuzzy neural networks in non-linear identification and time series prediction. *Soft. Comput.* **18**(6), 1213–1224 (2014)
22. Rubio, E., Castillo, O., Valdez, F., Melin, P., Gonzalez, C.I., Martinez, G.: An extension of the fuzzy possibilistic clustering algorithm using type-2 fuzzy logic techniques. *Adv. Fuzzy Syst.* (2017). <https://doi.org/10.1155/2017/7094046>
23. Melin, P., Miramontes, I., Prado-Arechiga, G.: A hybrid model based on modular neural networks and fuzzy systems for classification of blood pressure and hypertension risk diagnosis. *Expert Syst. Appl.* **107**, 146–164 (2018)
24. Castillo, O.: Towards finding the optimal n in designing type- n fuzzy systems for particular classes of problems: a review. *Appl. Comput. Math.* **17**(1), 3–9 (2018)

A Generalized Nonlinear Quasi-variational-like Inclusion Problem Involving Fuzzy Mappings



Faizan Ahmad Khan, Javid Ali, Fatimah Nasser Albishi, and Faik Gursoy

Abstract In this paper, using η -proximal point mapping of a m - η -accretive mapping and an equivalence relation, we propose and analyze an iterative algorithm for finding the approximate solution of a generalized nonlinear quasi-variational-like inclusion problem involving fuzzy mappings in a real Banach space. Further, we prove the existence of solution of this class of inclusion problem and discuss the convergence analysis of iterative sequences generated by the proposed iterative algorithm.

Keywords Generalized nonlinear quasi-variational-like inclusion · Fuzzy mappings · m - η -accretive mapping · η -proximal point mapping · Iterative algorithm

Mathematics Subject Classification 47H10 · 49J40

1 Introduction

Variational inclusion is an important generalization of variational inequality. For application of variational inclusions, see for example [8]. In recent years, various kinds of iterative methods have been established to find the approximate solutions for variational inequalities (inclusions). Among these methods, the proximal point mapping method has been widely used by many authors for solving various classes

F. A. Khan (✉) · F. N. Albishi
Department of Mathematics, University of Tabuk, Tabuk 71491, Saudi Arabia
e-mail: fkhan@ut.edu.sa

F. N. Albishi
e-mail: 421009246@stu.ut.edu.sa

J. Ali
Department of Mathematics, A.M.U., Aligarh 202002, India
e-mail: javid.mm@amu.ac.in

F. Gursoy
Department of Mathematics, Adiyaman University, 02040 Adiyaman, Turkey
e-mail: faikgursoy02@hotmail.com

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
S. D. Jabeen et al. (eds.), *Soft Computing and Optimization*, Springer Proceedings
in Mathematics & Statistics 404, https://doi.org/10.1007/978-981-19-6406-0_13

of variational inequalities (inclusions). For details, we refer to see [1, 4, 6, 9, 11, 16, 19].

In 1965, Zadeh [20] has given the notion of fuzzy sets. In 1981, Heilpern [10] initiated the study of fuzzy mappings and established a fuzzy analog of the Nadler's fixed point theorem for multi-valued mappings. For applications of fuzzy mappings, we refer to see [2, 3, 5, 7, 10, 12, 15, 17, 18].

In 1989, Chang and Zhu [3] initiated the study of a class of variational inequalities involving fuzzy mappings. After that various classes of variational inequalities involving fuzzy mappings have been studied by many researchers, see for examples [3, 5, 12, 15, 17].

Recently, Ding and Park [7], Kazmi [13] and Park and Jeong [18] studied some generalized classes of variational inclusions involving fuzzy mappings. Note that most of results in this field have been proved in Hilbert spaces.

Motivated by the works in this field, in this paper, we shall consider and study a generalized nonlinear quasi-variational-like inclusion problem involving fuzzy mappings (for short, GNQVLIP) in a real Banach space. Further, using η -proximal point mapping of a m - η -accretive mapping and an equivalence relation, we propose and analyze an iterative algorithm for GNQVLIP. Furthermore, we prove the existence of solution of GNQVLIP and discuss the convergence analysis of the proposed iterative algorithm. The method presented in this paper may be used to generalize and advance some known results given in [3, 7, 12, 13, 15, 17, 18].

2 Preliminaries

Let E be a real Banach space equipped with the norm $\| \cdot \|$. Let $\langle \cdot, \cdot \rangle$ denote the dual pair between E and its dual space E^* and let $J : E \rightarrow 2^{E^*}$ be the normalized duality mapping, see for details [4, 13, 19]. Let $CB(E)$ be the collection of all nonempty closed and bounded subsets of E and let 2^E be the power set of E . The Hausdorff metric $\tilde{\mathcal{H}}(\cdot, \cdot)$ on $CB(E)$ is defined by

$$\tilde{\mathcal{H}}(A, B) = \max \left\{ \sup_{x \in A} \inf_{y \in B} d(x, y), \sup_{y \in B} \inf_{x \in A} d(x, y) \right\}, \quad A, B \in CB(E). \quad (2.1)$$

First, we recall and define the following concepts and known results.

Definition 2.1 ([7, 13]) Let $\mathcal{F}(E)$ be the family of all fuzzy sets over E . A mapping $F : E \rightarrow \mathcal{F}(E)$ is called a *fuzzy mapping* over E .

Remark 2.1 ([7, 13]) Let F be a fuzzy mapping over E , then F_x is a fuzzy set on E and $F_x(y)$ is the membership function of y in F_x .

Definition 2.2 ([7, 13]) A fuzzy mapping $F : E \rightarrow \mathcal{F}(E)$ is said to be *closed*, if for each $x \in E$, the function $y \rightarrow F_x(y)$ is upper semi-continuous, i.e., for any given net $\{y_\alpha\} \subset E$ satisfying $y_\alpha \rightarrow y_0 \in E$, $\limsup_\alpha F_x(y_\alpha) \leq F_x(y_0)$.

Definition 2.3 ([7]) Let $A \in \mathcal{F}(E), \alpha \in [0, 1]$. Then, the set $(A)_\alpha = \{z \in E : A(z) \geq \alpha\}$ is called a α -cut set of fuzzy set A .

Definition 2.4 ([4, 13]) A single-valued mapping $g : E \rightarrow E$ is said to be

- (i) γ -strongly accretive if, $\forall x, y \in E, \exists j(x - y) \in J(x - y)$ and $\gamma > 0$ such that

$$\langle g(x) - g(y), j(x - y) \rangle \geq \gamma \|x - y\|^2;$$

- (ii) l_g -Lipschitz continuous if, $\exists l_g > 0$ such that

$$\|g(x) - g(y)\| \leq l_g \|x - y\|, \quad \forall x, y \in E.$$

Definition 2.5 ([4, 13]) Let $\eta : E \times E \rightarrow E$ be a mapping. A multi-valued mapping $M : E \rightarrow 2^E$ is said to be

- (i) η -accretive if, $\forall x, y \in E, \exists j\eta(x, y) \in J\eta(x, y)$ such that

$$\langle u - v, j\eta(x, y) \rangle \geq 0, \quad \forall u \in M(x), v \in M(y);$$

- (ii) strictly η -accretive if, $\forall x, y \in E, \exists j\eta(x, y) \in J\eta(x, y)$ such that

$$\langle u - v, j\eta(x, y) \rangle \geq 0 \quad \forall u \in M(x), v \in M(y)$$

and equality holds if and only if $x = y$;

- (iii) ν -strongly η -accretive if, $\forall x, y \in E, \exists j\eta(x, y) \in J\eta(x, y)$ and $\nu > 0$ such that

$$\langle u - v, j\eta(x, y) \rangle \geq \nu \|x - y\|^2, \quad \forall u \in M(x), v \in M(y);$$

- (iv) m - η -accretive if, M is η -accretive and $(I + \rho M)(E) = E$ for any $\rho > 0$.

Lemma 2.1 ([19]) Let E be a real Banach space and $J : E \rightarrow 2^{E^*}$ be the normalized duality mapping. Then, for any $x, y \in E$,

$$\|x + y\|^2 \leq \|x\|^2 + 2\langle y, j(x + y) \rangle, \quad \forall j(x + y) \in J(x + y).$$

Lemma 2.2 ([4]) Let $\eta : E \times E \rightarrow E$ be a strictly accretive mapping and let $M : E \rightarrow 2^E$ be a m - η -accretive mapping, then

- (i) $\langle u - v, j\eta(x, y) \rangle \geq 0, \forall (v, y) \in \text{Graph}(M)$ implies $(u, x) \in \text{Graph}(M)$, where $\text{Graph}(M) := \{(u, x) \in E \times E : u \in M(x)\}$;
- (ii) the mapping $(I + \rho M)^{-1}$ is single-valued for all $\rho > 0$.

By Lemma 2.2, we can define η -proximal point mapping for a m - η -accretive multi-valued mapping M as follows:

$$J_\rho^M(z) = (I + \rho M)^{-1}(z), \quad \forall z \in E, \tag{2.2}$$

where $\rho > 0$ is a constant, I is identity mapping on E and $\eta : E \times E \rightarrow E$ is a strictly accretive mapping, see [4].

Lemma 2.3 ([4]) *Let $\eta : E \times E \rightarrow E$ be a δ -strongly accretive and τ -Lipschitz continuous mapping. Let $M : E \rightarrow 2^E$ be a m - η -accretive mapping, then the η -proximal point mapping J_ρ^M is $\frac{\tau}{\delta}$ -Lipschitz continuous, i.e.,*

$$\|J_\rho^M(x) - J_\rho^M(y)\| \leq \frac{\tau}{\delta} \|x - y\|, \quad \forall x, y \in E.$$

3 Formulation of Problem

Let $A, B, C, D, Q, R : E \rightarrow \mathcal{F}(E)$ be closed fuzzy mappings satisfying condition (C) then there exist functions $a, b, c, d, q, r : E \rightarrow [0, 1]$ such that for each $x \in E$, we have $(A_x)_{a(x)}, (B_x)_{b(x)}, (C_x)_{c(x)}, (D_x)_{d(x)}, (Q_x)_{q(x)}, (R_x)_{r(x)} \in CB(E)$. By using the fuzzy mappings A, B, C, D, Q, R , we can define, respectively, the multi-valued mappings $\tilde{A}, \tilde{B}, \tilde{C}, \tilde{D}, \tilde{Q}, \tilde{R} : E \rightarrow CB(E)$ by $\tilde{A}(x) = (A_x)_{a(x)}, \tilde{B}(x) = (B_x)_{b(x)}, \tilde{C}(x) = (C_x)_{c(x)}, \tilde{D}(x) = (D_x)_{d(x)}, \tilde{Q}(x) = (Q_x)_{q(x)}, \tilde{R}(x) = (R_x)_{r(x)}$, for each $x \in E$. It means that

$$\begin{aligned} \tilde{A}(x) &= (A_x)_{a(x)} = \{z \in E, A_x(z) \geq a(x)\} \in CB(E), \\ \tilde{B}(x) &= (B_x)_{b(x)} = \{z \in E, B_x(z) \geq b(x)\} \in CB(E), \\ \tilde{C}(x) &= (C_x)_{c(x)} = \{z \in E, C_x(z) \geq c(x)\} \in CB(E), \\ \tilde{D}(x) &= (D_x)_{d(x)} = \{z \in E, D_x(z) \geq d(x)\} \in CB(E), \\ \tilde{Q}(x) &= (Q_x)_{q(x)} = \{z \in E, Q_x(z) \geq q(x)\} \in CB(E), \\ \tilde{R}(x) &= (R_x)_{r(x)} = \{z \in E, R_x(z) \geq r(x)\} \in CB(E). \end{aligned}$$

Let $f, g, m : E \rightarrow E$ and $N : E \times E \times E \rightarrow E$ be single-valued mappings such that $g \neq 0$. Let $A, B, C, D, Q, R : E \rightarrow \mathcal{F}(E)$ be fuzzy mappings and let $a, b, c, d, q, r : E \rightarrow [0, 1]$ be given functions. Let $W : E \times E \rightarrow 2^E$ be a multi-valued mapping such that for each $z \in E$, $W(\cdot, z)$ is a m - η -accretive mapping with $g(x) - m(y) \in \text{dom}(W(\cdot, z)), \forall x, y \in E$.

Here, we consider the following generalized nonlinear quasi-variational-like inclusion problem involving fuzzy mappings (for short, GNQVLIP): Find $x, u, v, w, s, y, z \in E$ such that $A_x(u) \geq a(x), B_x(v) \geq b(x), C_x(w) \geq c(x), D_x(s) \geq d(x), Q_x(z) \geq q(x), R_x(y) \geq r(x)$ and

$$0 \in W(g(x) - m(y), z) + f(s) - N(u, v, w). \tag{3.1}$$

For appropriate and suitable choices of the mappings $A, B, C, D, Q, R, N, W, \eta, a, b, c, d, f, g, m, q, r$ and the space E , it is easy to see that GNQVLIP (3.1) includes a number of known generalized classes of variational inclusions (inequalities), studied by many authors, see for example [3, 7, 12, 13, 15, 17, 18].

Now, we define the following important lemma which will be used in the sequel as an equivalence relation between the solutions.

Lemma 3.1 *(x, u, v, w, s, z, y) is a solution of GNQVLIP (3.1) if and only if it satisfies the relation*

$$g(x) = m(y) + J_\rho^{W(\cdot, z)}(g(x) - m(y) - \rho[f(s) - N(u, v, w)]), \tag{3.2}$$

where $u \in \tilde{A}(x), v \in \tilde{B}(x), w \in \tilde{C}(x), s \in \tilde{D}(x), z \in \tilde{Q}(x), y \in \tilde{R}(x), J_\rho^{W(\cdot, z)} := (I + \rho W(\cdot, z))^{-1}$ and $\rho > 0$ is a constant.

4 Iterative Algorithm

In this section, using Lemma 3.1 and Nadler’s technique [14], we propose the following iterative algorithm to compute the approximate solution of GNQVLIP (3.1).

Iterative Algorithm 4.1 Let $P : E \rightarrow 2^E$ be a multi-valued mapping such that for each $x \in E, P(x) \subseteq g(E)$, where P is defined by

$$P(x) = \bigcup_{u \in \tilde{A}(x)} \bigcup_{v \in \tilde{B}(x)} \bigcup_{w \in \tilde{C}(x)} \bigcup_{s \in \tilde{D}(x)} \bigcup_{z \in \tilde{Q}(x)} \bigcup_{y \in \tilde{R}(x)} \left(m(y) + J_\rho^{W(\cdot, z)}(g(x) - m(y) - \rho[f(s) - N(u, v, w)]) \right). \tag{4.1}$$

Now, for given $x_0 \in E, u_0 \in \tilde{A}(x_0), v_0 \in \tilde{B}(x_0), w_0 \in \tilde{C}(x_0), s_0 \in \tilde{D}(x_0), z_0 \in \tilde{Q}(x_0)$ and $y_0 \in \tilde{R}(x_0)$, and let

$$p_0 = m(y_0) + J_\rho^{W(\cdot, z_0)}(g(x_0) - m(y_0) - \rho[f(s_0) - N(u_0, v_0, w_0)]) \in P(x_0) \subseteq g(E).$$

Hence, there exists $x_1 \in E$ such that $p_0 = g(x_1)$. Further, as we know that $\tilde{A}(x_0), \tilde{B}(x_0), \tilde{C}(x_0), \tilde{D}(x_0), \tilde{Q}(x_0), \tilde{R}(x_0) \in CB(E)$, and using Nadler [14], there exist $u_1 \in \tilde{A}(x_1), v_1 \in \tilde{B}(x_1), w_1 \in \tilde{C}(x_1), s_1 \in \tilde{D}(x_1), z_1 \in \tilde{Q}(x_1)$ and $y_1 \in \tilde{R}(x_1)$ such that

$$\|u_1 - u_0\| \leq (1 + (1 + 0)^{-1}) \tilde{\mathcal{H}}(\tilde{A}(x_1), \tilde{A}(x_0)),$$

$$\|v_1 - v_0\| \leq (1 + (1 + 0)^{-1}) \tilde{\mathcal{H}}(\tilde{B}(x_1), \tilde{B}(x_0)),$$

$$\begin{aligned} \|w_1 - w_0\| &\leq (1 + (1 + 0)^{-1}) \tilde{\mathcal{H}}(\tilde{\mathcal{C}}(x_1), \tilde{\mathcal{C}}(x_0)), \\ \|s_1 - s_0\| &\leq (1 + (1 + 0)^{-1}) \tilde{\mathcal{H}}(\tilde{\mathcal{D}}(x_1), \tilde{\mathcal{D}}(x_0)), \\ \|z_1 - z_0\| &\leq (1 + (1 + 0)^{-1}) \tilde{\mathcal{H}}(\tilde{\mathcal{Q}}(x_1), \tilde{\mathcal{Q}}(x_0)), \\ |y_1 - y_0| &\leq (1 + (1 + 0)^{-1}) \tilde{\mathcal{H}}(\tilde{\mathcal{R}}(x_1), \tilde{\mathcal{R}}(x_0)). \end{aligned}$$

Let $p_1 = m(y_1) + J_\rho^{W(\cdot, z_1)}(g(x_1) - m(y_1) - \rho[f(s_1) - N(u_1, v_1, w_1)]) \in P(x_1) \subseteq g(E)$. Hence there exists $x_2 \in E$ such that $p_1 = g(x_2)$. Continuing the above process inductively, we can define the following iterative sequences $\{x_n\}$, $\{u_n\}$, $\{v_n\}$, $\{w_n\}$, $\{s_n\}$, $\{z_n\}$ and $\{y_n\}$ satisfying

$$g(x_{n+1}) = m(y_n) + J_\rho^{W(\cdot, z_n)}(g(x_n) - m(y_n) - \rho[f(s_n) - N(u_n, v_n, w_n)]), \quad (4.2)$$

$$\begin{aligned} u_n \in \tilde{A}(x_n) : \|u_{n+1} - u_n\| &\leq (1 + (1 + n)^{-1}) \tilde{\mathcal{H}}(\tilde{A}(x_{n+1}), \tilde{A}(x_n)), \\ v_n \in \tilde{B}(x_n) : \|v_{n+1} - v_n\| &\leq (1 + (1 + n)^{-1}) \tilde{\mathcal{H}}(\tilde{B}(x_{n+1}), \tilde{B}(x_n)), \\ w_n \in \tilde{C}(x_n) : \|w_{n+1} - w_n\| &\leq (1 + (1 + n)^{-1}) \tilde{\mathcal{H}}(\tilde{C}(x_{n+1}), \tilde{C}(x_n)), \\ s_n \in \tilde{D}(x_n) : \|s_{n+1} - s_n\| &\leq (1 + (1 + n)^{-1}) \tilde{\mathcal{H}}(\tilde{D}(x_{n+1}), \tilde{D}(x_n)), \\ z_n \in \tilde{Q}(x_n) : \|z_{n+1} - z_n\| &\leq (1 + (1 + n)^{-1}) \tilde{\mathcal{H}}(\tilde{Q}(x_{n+1}), \tilde{Q}(x_n)), \\ y_n \in \tilde{R}(x_n) : \|y_{n+1} - y_n\| &\leq (1 + (1 + n)^{-1}) \tilde{\mathcal{H}}(\tilde{R}(x_{n+1}), \tilde{R}(x_n)), \end{aligned}$$

where $n = 0, 1, 2, 3, \dots$, and $\rho > 0$ is a constant.

5 Existence of Solution and Convergence Analysis

First, we define the following definitions.

Definition 5.1 A multi-valued mapping $A : E \rightarrow 2^E$ is said to be $\tilde{\mathcal{H}}$ -Lipschitz continuous if, there exists a constant $l_A > 0$ such that

$$\tilde{\mathcal{H}}(A(x_1), A(x_2)) \leq l_A \|x_1 - x_2\|, \quad \forall x_1, x_2 \in E.$$

Definition 5.2 Let $\eta : E \times E \rightarrow E$ be a single-valued and $A : E \rightarrow 2^E$ be a multi-valued mappings. A mapping $N : E \times E \times E \rightarrow E$ is said to be

- (i) α - η -relaxed Lipschitz with respect A in the first argument if, $\forall x_1, x_2 \in E, \exists j\eta(x_1, x_2) \in J\eta(x_1, x_2)$ and $\alpha > 0$ such that

$$\langle N(u_1, \cdot, \cdot) - N(u_2, \cdot, \cdot), j\eta(x_1, x_2) \rangle \leq -\alpha \|x_1 - x_2\|^2, \quad \forall u_1 \in A(x_1), u_2 \in A(x_2);$$

(ii) $(l_{(N,1)}, l_{(N,2)}, l_{(N,3)})$ -mixed Lipschitz continuous if, $\exists l_{(N,1)}, l_{(N,2)}, l_{(N,3)} > 0$ such that

$$\|N(x_1, y_1, z_1) - N(x_2, y_2, z_2)\| \leq l_{(N,1)} \|x_1 - x_2\| + l_{(N,2)} \|y_1 - y_2\| + l_{(N,3)} \|z_1 - z_2\|,$$

$$\forall x_1, x_2, y_1, y_2, z_1, z_2 \in E.$$

Now, we prove the existence of solution of GNQVLIP (3.1) and discuss the convergence analysis of Iterative Algorithm 4.1.

Theorem 5.1 Let $\eta : E \times E \rightarrow E$ be a δ -strongly accretive and τ -Lipschitz continuous mapping. Let $\tilde{A}, \tilde{B}, \tilde{C}, \tilde{D}, \tilde{Q}, \tilde{R} : E \rightarrow CB(E)$ be the multi-valued mappings induced by the closed fuzzy mappings A, B, C, D, Q, R , respectively. Let $\tilde{A}, \tilde{B}, \tilde{C}, \tilde{D}, \tilde{Q}$ and \tilde{R} be \tilde{H} -Lipschitz continuous with constants l_A, l_B, l_C, l_D, l_Q and l_R , respectively. Let $m, f, g : E \rightarrow E$ be Lipschitz continuous with constants l_m, l_f and l_g , respectively, and let $(g - I) : E \rightarrow E$ be μ -strongly accretive mapping, where I is identity mapping on E . Let $N : E \times E \times E \rightarrow E$ be α - η -relaxed Lipschitz with respect to \tilde{A} in the first argument and $(l_{(N,1)}, l_{(N,2)}, l_{(N,3)})$ -mixed Lipschitz continuous. Let $W : E \times E \rightarrow 2^E$ be such that for each fixed $z \in E$, $W(\cdot, z)$ is m - η -accretive mapping, and let for each $x \in E$, $P(x) \subseteq g(E)$, where P is defined by (4.1). Assume that there exist constants $\rho > 0$ and $\lambda > 0$ such that

$$\|J_\rho^{W(\cdot, x)}(z) - J_\rho^{W(\cdot, y)}(z)\| \leq \lambda \|x - y\|, \quad \forall x, y, z \in E, \tag{5.1}$$

and

$$\theta := \frac{1}{\sqrt{2\mu + 1}} \left[\left(1 + \frac{\tau}{\delta}\right) l_m l_R + \lambda l_Q + \frac{\tau}{\delta} \left\{ \rho (l_f l_D + l_{(N,2)} l_B + l_{(N,3)} l_C) + \sqrt{l_g^2 - 2\rho\alpha + 2\rho[l_{(N,1)} l_A \{l_g + \tau + \rho l_{(N,1)} l_A\}]} \right\} \right] < 1. \tag{5.2}$$

Then $(x^*, u^*, v^*, w^*, s^*, z^*, y^*)$ is a solution of GNQVLIP (3.1), where the iterative sequences $\{x_n\}, \{u_n\}, \{v_n\}, \{w_n\}, \{s_n\}, \{z_n\}$ and $\{y_n\}$ generated by Iterative Algorithm 4.1 converge strongly to $x^*, u^*, v^*, w^*, s^*, z^*$ and y^* , respectively.

Proof Given that $(g - I)$ is μ -strongly accretive mapping, we have

$$\begin{aligned} \|x_{n+2} - x_{n+1}\|^2 &= \|g(x_{n+2}) - g(x_{n+1}) + x_{n+2} - x_{n+1} - (g(x_{n+2}) - g(x_{n+1}))\|^2 \\ &\leq \|g(x_{n+2}) - g(x_{n+1})\|^2 - 2\langle (g - I)(x_{n+2}) \\ &\quad - (g - I)(x_{n+1}), j(x_{n+2} - x_{n+1}) \rangle \\ &\leq \|g(x_{n+2}) - g(x_{n+1})\|^2 - 2\mu \|x_{n+2} - x_{n+1}\|^2 \end{aligned}$$

which implies that

$$\|x_{n+2} - x_{n+1}\| \leq \frac{1}{\sqrt{2\mu + 1}} \|g(x_{n+2}) - g(x_{n+1})\|. \quad (5.3)$$

Using Lemma 2.3, Iterative Algorithm 4.1 and (5.1), we have

$$\begin{aligned} \|g(x_{n+2}) - g(x_{n+1})\| &= \|m(y_{n+1}) - m(y_n)\| + \|J_\rho^{W(\cdot, z_{n+1})}(g(x_{n+1}) - m(y_{n+1}) - \rho[f(s_{n+1}) \\ &\quad - N(u_{n+1}, v_{n+1}, w_{n+1})]) - J_\rho^{W(\cdot, z_n)}(g(x_n) - m(y_n) - \rho[f(s_n) - N(u_n, v_n, w_n)])\| \\ &\leq \left(1 + \frac{\tau}{\delta}\right) \|m(y_{n+1}) - m(y_n)\| + \lambda \|z_{n+1} - z_n\| + \frac{\tau}{\delta} \rho \|f(s_{n+1}) - f(s_n)\| \\ &\quad + \frac{\tau}{\delta} \|g(x_{n+1}) - g(x_n) + \rho[N(u_{n+1}, v_{n+1}, w_{n+1}) - N(u_n, v_n, w_n)]\|. \end{aligned} \quad (5.4)$$

Using Lipschitz continuity of m and $\tilde{\mathcal{H}}$ -Lipschitz continuity of \tilde{R} , we have

$$\begin{aligned} \|m(y_{n+1}) - m(y_n)\| &\leq l_m(1 + (1 + n)^{-1}) \tilde{\mathcal{H}}(\tilde{R}(x_{n+1}), \tilde{R}(x_n)) \\ &\leq l_m l_R(1 + (1 + n)^{-1}) \|x_{n+1} - x_n\|. \end{aligned} \quad (5.5)$$

Using Lipschitz continuity of f and $\tilde{\mathcal{H}}$ -Lipschitz continuity of \tilde{D} , we have

$$\begin{aligned} \|f(s_{n+1}) - f(s_n)\| &\leq l_f(1 + (1 + n)^{-1}) \tilde{\mathcal{H}}(\tilde{D}(x_{n+1}), \tilde{D}(x_n)) \\ &\leq l_f l_D(1 + (1 + n)^{-1}) \|x_{n+1} - x_n\|. \end{aligned} \quad (5.6)$$

Using $\tilde{\mathcal{H}}$ -Lipschitz continuity of \tilde{Q} , we have

$$\|z_{n+1} - z_n\| \leq (1 + (1 + n)^{-1}) \tilde{\mathcal{H}}(\tilde{Q}(x_{n+1}), \tilde{Q}(x_n)) \leq l_Q(1 + (1 + n)^{-1}) \|x_{n+1} - x_n\|. \quad (5.7)$$

Now, we estimate that

$$\begin{aligned} &\|g(x_{n+1}) - g(x_n) + \rho[N(u_{n+1}, v_{n+1}, w_{n+1}) - N(u_n, v_n, w_n)]\| \\ &\leq \|g(x_{n+1}) - g(x_n) + \rho[N(u_{n+1}, v_{n+1}, w_{n+1}) - N(u_n, v_{n+1}, w_{n+1})]\| \\ &\quad + \rho(\|N(u_n, v_{n+1}, w_{n+1}) - N(u_n, v_n, w_{n+1})\| + \|N(u_n, v_n, w_{n+1}) \\ &\quad - N(u_n, v_n, w_n)\|). \end{aligned} \quad (5.8)$$

Using Lemma 2.1, we have

$$\begin{aligned} &\|g(x_{n+1}) - g(x_n) + \rho[N(u_{n+1}, v_{n+1}, w_{n+1}) - N(u_n, v_{n+1}, w_{n+1})]\|^2 \\ &\leq \|g(x_{n+1}) - g(x_n)\|^2 + 2\rho \langle N(u_{n+1}, v_{n+1}, w_{n+1}) - N(u_n, v_{n+1}, w_{n+1}), j(g(x_{n+1}) - g(x_n) \\ &\quad - \rho[N(u_{n+1}, v_{n+1}, w_{n+1}) - N(u_n, v_{n+1}, w_{n+1})]) \rangle. \end{aligned}$$

Since g is Lipschitz continuous and \tilde{A} is \tilde{H} -Lipschitz continuous; N is α - η -relaxed Lipschitz with respect to \tilde{A} and mixed Lipschitz continuous, there exists $j\eta(x_{n+1}, x_n) \in J\eta(x_{n+1}, x_n)$ such that the preceding inequality reduces to

$$\begin{aligned}
 & \|g(x_{n+1}) - g(x_n) + \rho[N(u_{n+1}, v_{n+1}, w_{n+1}) - N(u_n, v_{n+1}, w_{n+1})]\|^2 \\
 & \leq l_g^2 \|x_{n+1} - x_n\|^2 + 2\rho(N(u_{n+1}, v_{n+1}, w_{n+1}) - N(u_n, v_{n+1}, w_{n+1}), j\eta(x_{n+1}, x_n)) \\
 & \quad + 2\rho(N(u_{n+1}, v_{n+1}, w_{n+1}) - N(u_n, v_{n+1}, w_{n+1}), j(g(x_{n+1}) - g(x_n))) \\
 & \quad - \rho(N(u_{n+1}, v_{n+1}, w_{n+1}) - N(u_n, v_{n+1}, w_{n+1})) - j\eta(x_{n+1}, x_n) \\
 & \leq (l_g^2 - 2\rho\alpha) \|x_{n+1} - x_n\|^2 + 2\rho\|N(u_{n+1}, v_{n+1}, w_{n+1}) - N(u_n, v_{n+1}, w_{n+1})\| \\
 & \quad \times [\|g(x_{n+1}) - g(x_n)\| + \rho\|N(u_{n+1}, v_{n+1}, w_{n+1}) - N(u_n, v_{n+1}, w_{n+1})\| + \|\eta(x_{n+1}, x_n)\|] \\
 & \leq (l_g^2 - 2\rho\alpha) \|x_{n+1} - x_n\|^2 + 2\rho l_{(N,1)} \|u_{n+1} - u_n\| [(l_g + \tau) \|x_{n+1} - x_n\| + \rho l_{(N,1)} \|u_{n+1} - u_n\|] \\
 & = [l_g^2 - 2\rho\alpha + 2\rho l_{(N,1)} l_A (1 + (1+n)^{-1}) (l_g + \tau + \rho l_{(N,1)} l_A (1 + (1+n)^{-1}))] \|x_{n+1} - x_n\|^2.
 \end{aligned} \tag{5.9}$$

Using mixed Lipschitz continuity of N and $\tilde{\mathcal{H}}$ -Lipschitz continuity of \tilde{B} and \tilde{C} , we have

$$\begin{aligned}
 & \|N(u_n, v_{n+1}, w_{n+1}) - N(u_n, v_n, w_n)\| \\
 & \leq \|N(u_n, v_{n+1}, w_{n+1}) - N(u_n, v_n, w_{n+1})\| + \|N(u_n, v_n, w_{n+1}) - N(u_n, v_n, w_n)\| \\
 & \leq (L_{(N,2)} l_B + L_{(N,3)} l_C) (1 + (1+n)^{-1}) \|x_{n+1} - x_n\|.
 \end{aligned} \tag{5.10}$$

It follows from (5.3)–(5.10) that

$$\|x_{n+2} - x_{n+1}\| \leq \theta_n \|x_{n+1} - x_n\|, \tag{5.11}$$

where $\theta_n := \frac{1}{\sqrt{2\mu+1}} \left[(1 + \frac{\tau}{\delta}) l_m l_R (1 + (1+n)^{-1}) + \lambda l_Q (1 + (1+n)^{-1}) \right.$

$$\begin{aligned}
 & \left. + \frac{\tau}{\delta} \rho (1 + (1+n)^{-1}) (l_f l_D + l_{(N,2)} l_B + l_{(N,3)} l_C) \right. \\
 & \left. + \frac{\tau}{\delta} \sqrt{l_g^2 - 2\rho\alpha + 2\rho[l_{(N,1)} l_A (1 + (1+n)^{-1}) (l_g + \tau + \rho l_{(N,1)} l_A (1 + (1+n)^{-1}))]} \right].
 \end{aligned} \tag{5.12}$$

Letting $n \rightarrow \infty$, we have $\theta_n \rightarrow \theta$, where

$$\begin{aligned}
 \theta := \frac{1}{\sqrt{2\mu+1}} & \left[(1 + \frac{\tau}{\delta}) l_m l_R + \lambda l_Q + \frac{\tau}{\delta} \left\{ \rho (l_f l_D + l_{(N,2)} l_B + l_{(N,3)} l_C) \right. \right. \\
 & \left. \left. + \sqrt{l_g^2 - 2\rho\alpha + 2\rho[l_{(N,1)} l_A (l_g + \tau + \rho l_{(N,1)} l_A)]} \right\} \right].
 \end{aligned} \tag{5.13}$$

By condition (5.2), $\theta \in (0, 1)$. Hence, $\theta_n < 1$ for n sufficiently large. Therefore, (5.11) implies that $\{x_n\}$ is a Cauchy sequence in E . Since E is complete, there exists

$x^* \in E$ such that $x_n \rightarrow x^*$ as $n \rightarrow \infty$. Further, it follows from $\tilde{\mathcal{H}}$ -Lipschitz continuity of \tilde{A} and Iterative Algorithm 4.1, we have

$$\begin{aligned} \|u_{n+1} - u_n\| &\leq (1 + (1 + n)^{-1}) \tilde{\mathcal{H}}(\tilde{A}(x_{n+1}), \tilde{A}(x_n)) \\ &\leq (1 + (1 + n)^{-1}) l_A \|x_{n+1} - x_n\|, \end{aligned}$$

which implies that $\{u_n\}$ is also a Cauchy sequence in E . Similarly, we can show that $\{v_n\}$, $\{w_n\}$, $\{s_n\}$, $\{z_n\}$ and $\{y_n\}$ are Cauchy sequences in E , hence there exist $v^*, w^*, s^*, z^*, y^* \in E$ such that $u_n \rightarrow u^*, v_n \rightarrow v^*, w_n \rightarrow w^*, s_n \rightarrow s^*, z_n \rightarrow z^*$ and $y_n \rightarrow y^*$ as $n \rightarrow \infty$.

As we know that $u_n \in \tilde{A}(x_n)$, we have

$$\begin{aligned} d(u^*, \tilde{A}(x^*)) &\leq \|u^* - u_n\| + d(u_n, \tilde{A}(x^*)) \\ &\leq \|u^* - u_n\| + \tilde{\mathcal{H}}(\tilde{A}(x_n), \tilde{A}(x^*)) \\ &\leq \|u^* - u_n\| + l_A \|x_n - x^*\| \rightarrow 0 \end{aligned}$$

which implies that $u^* \in \tilde{A}(x^*)$. Similarly, we can show that $v^* \in \tilde{B}(x^*)$, $w^* \in \tilde{C}(x^*)$, $s^* \in \tilde{D}(x^*)$, $z^* \in \tilde{Q}(x^*)$ and $y^* \in \tilde{R}(x^*)$. Hence we have $A_{x^*}(u^*) \geq a(x^*)$, $B_{x^*}(v^*) \geq b(x^*)$, $C_{x^*}(w^*) \geq c(x^*)$, $D_{x^*}(s^*) \geq d(x^*)$, $Q_{x^*}(z^*) \geq q(x^*)$ and $R_{x^*}(y^*) \geq r(x^*)$. Thus, it follows from Iterative Algorithm 4.1 that

$$g(x^*) = m(y^*) + J_\rho^{W(\cdot, z^*)}(g(x^*) - m(y^*) - \rho[f(s^*) - N(u^*, v^*, w^*)]).$$

Hence, it ensure from Lemma 3.1 that $\{x^*, u^*, v^*, w^*, s^*, z^*, y^*\}$ is a solution of GNQVLIP (3.1).

Remark 5.1 For $\lambda, \mu, \rho > 0$ and $g \neq 0$, it is clear that $\delta \leq \tau$; $2\mu + 1 > 0$; $2\rho\alpha < l_g^2 + 2\rho[l_{(N,1)}l_A\{l_g + \tau + \rho l_{(N,1)}l_A\}]$. Further, $\theta \in (0, 1)$ and condition (5.2) of Theorem 5.1 holds for some suitable values of constants. Since GNQVLIP (3.1) includes many known generalized classes of variational inclusions (inequalities) as special cases, so the technique utilized in this paper can be used to extend and advance the results given in [3, 7, 12, 13, 15, 17, 18].

References

1. Adly, S.: Perturbed algorithms and sensitivity analysis for a general class of variational inclusions. *J. Math. Anal. Appl.* **201**(3), 609–630 (1996)
2. Chang, S.-S., Huang, N.-J.: Generalized complementarity problem for fuzzy mappings. *Fuzzy Sets Syst.* **55**, 227–234 (1993)
3. Chang, S.-S., Zhu, Y.-G.: On variational inequalities for fuzzy mappings. *Fuzzy Sets Syst.* **32**, 359–367 (1989)
4. Chidume, C.E., Kazmi, K.R., Zegeye, H.: Iterative approximation of a solution of a general variational-like inclusion in Banach space. *Int. J. Math. Math. Sci.* **22**, 1159–1168 (2004)

5. Ding, X.-P.: Algorithm of solutions for mixed implicit quasi-variational inequalities with fuzzy mappings. *Comput. Math. Appl.* **38**(5–6), 231–249 (1999)
6. Ding, X.P., Luo, C.-L.: Perturbed proximal point algorithms for generalized quasi-variational-like inclusions. *J. Comput. Appl. Math.* **113**, 153–165 (2000)
7. Ding, X.P., Park, J.-Y.: A new class of generalized nonlinear quasi-variational inclusions with fuzzy mappings. *J. Comput. Appl. Math.* **138**, 243–257 (2002)
8. Hassouni, A., Moudafi, A.: A perturbed algorithm for variational inclusions. *J. Math. Anal. Appl.* **185**, 706–712 (1994)
9. He, X.: On ϕ -strongly accretive mappings and some set-valued variational problems. *J. Math. Anal. Appl.* **277**, 504–511 (2003)
10. Heilpern, S.: Fuzzy mappings and fixed point theorems. *J. Math. Anal. Appl.* **83**, 566–569 (1981)
11. Huang, N.-J.: Generalized nonlinear variational inclusions with noncompact valued mappings. *Appl. Math. Lett.* **9**(3), 25–29 (1996)
12. Huang, N.-J.: A new method for a class of nonlinear variational inequalities with fuzzy mappings. *Appl. Math. Lett.* **10**(6), 129–133 (1997)
13. Kazmi, K.R.: Iterative algorithms for generalized quasi-variational-like inclusions with fuzzy mappings in Banach spaces. *J. Comput. Appl. Math.* **188**, 1–11 (2006)
14. Nadler, S.B.: Multi-valued contractive mappings. *Pac. J. Math.* **30**, 475–488 (1969)
15. Noor, M.A.: Variational inequalities with fuzzy mappings (I). *Fuzzy Sets Syst.* **55**, 309–314 (1989)
16. Noor, M.A.: Iterative schemes for multi-valued quasi-variational inclusions. *J. Glob. Optim.* **19**, 141–150 (2001)
17. Park, J.-Y., Jeong, J.-U.: Generalized strongly quasi-variational inequalities for fuzzy mappings. *Fuzzy Sets Syst.* **99**, 115–120 (1998)
18. Park, J.-Y., Jeong, J.-U.: A perturbed algorithm of variational inclusions for fuzzy mappings. *Fuzzy Sets Syst.* **115**, 419–424 (2000)
19. Petryshyn, W.V.: A characterization of strict convexity of Banach spaces and other use of duality mappings. *J. Funct. Anal.* **6**, 282–291 (1970)
20. Zadeh, L.A.: Fuzzy sets. *Inform. Contr.* **8**, 338–353 (1965)

Approximate Optimality Conditions for a Multi-objective Programming Problem



Bhawna Kohli

Abstract In this paper, approximate Karush–Kuhn–Tucker (KKT) type necessary conditions in terms of upper convexifactors for quasi ε –weak efficient solution of a multi-objective programming problem have been established. For that, nonsmooth version of Guignard constraint qualification (GCQ) has been introduced.

Keywords Approximate optimality conditions · Upper convexifactors · Quasi ε –weak efficient solution · Constraint qualification

1 Introduction

Convexifactors, introduced by Demyanov [5] in 1994, are tools for solving nonsmooth optimization problems. They are generalized subdifferentials for scalar valued functions. They are further studied by Jeyakumar and Luc [15], Dutta and Chandra [6], Kohli [17–19], Suneja and Kohli [25], Kabgani et al. [16], and many others. Clarke subdifferential, Michel–Penot subdifferential, and other known subdifferentials in the literature are always convex and compact but these convexifactors though always closed sets may not be convex or bounded. These relaxations allow their application to large class of nonsmooth problems. Using convexifactors, we get sharp optimality conditions, duality results, and good calculus rules.

There are many situations where exact optimal solutions cannot be applied. This gives rise to the need of approximate solutions which are easy to calculate and have immense importance in practical applications. Kutateladze [20] gave the concept of ε –optimal solutions and pareto ε –optimal solutions. Motivated by Ekeland’s variational principle, Loridan [23] introduced the notion of quasi ε –solution.

Not only theoretically, multi-objective optimization problems (MOPs) are important practically also. They arise in many fields such as engineering, economics, management, medicine, and others. For the study of approximate solutions of MOPs,

B. Kohli (✉)

Department of Mathematics, P.G.D.A.V. College, University of Delhi, New Delhi, Delhi 110065, India

e-mail: bhawna_kohli@rediffmail.com

one can see [1, 2, 4, 7–10, 12, 13, 20, 24]. Ghaznavi [9] presented sufficient optimality conditions for approximate quasi (weak, proper) efficient solutions of multi-objective optimization problem. Lee and Jiao [21] established approximate optimality conditions and duality results for quasi ε -solution of the robust convex optimization problem. Later on, Jiao and Lee [14] considered convex semidefinite programs to obtain approximate optimality conditions and duality results for their quasi-approximate solutions. Most recently, Yameng and Guolin [26] gave optimality conditions and saddle point results for approximate quasi weak efficient solution of a robust multi-objective programming problem. Gutiérrez et al. [11] studied multi-objective optimization problem and established optimality conditions for their approximate proper solutions.

Recently, Capâta [3], via ε -convexifactors, developed optimality results for a point to be a quasi ε -solution and quasi ε -efficient solution, respectively, of a scalar and of a vector optimization problem with constraints and also obtained necessary conditions for a vector equilibrium problem with constraints.

Motivated by above, here, using upper convexifactors, we obtain approximate KKT conditions for a multi-objective optimization problem. Nonsmooth GCQ in terms of upper convexifactors has been introduced and employed for the same. Our results are sharper and more general than those using Clarke, Michel-Penot subdifferential, etc.

Composition of the paper is as follows. Section 2 is based on the definitions of upper convexifactors, preliminary results, and an important lemma to be used in proving our result. In Sect. 3, approximate necessary conditions are developed.

2 Preliminaries

To begin with, Dutta and Chandra [6] gave following notions of convexifactors.

Definition 1 An extended real valued function, $g : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{\pm\infty\}$, with $g(x)$ finite for $x \in \mathbb{R}^n$,

- (i) admits an upper convexifactor (UCF) $\partial^u g(x)$ at x iff $\partial^u g(x) \subseteq \mathbb{R}^n$ is a closed set and

$$(g)_d^-(x, v) \leq \sup_{x^* \in \partial^u g(x)} \langle x^*, v \rangle, \text{ for all } v \in \mathbb{R}^n.$$

- (ii) admits an upper semiregular convexifactor (USRCF) $\partial^{us} g(x)$ at x iff $\partial^{us} g(x) \subseteq \mathbb{R}^n$ is a closed set and

$$(g)_d^+(x, v) \leq \sup_{x^* \in \partial^{us} g(x)} \langle x^*, v \rangle, \text{ for all } v \in \mathbb{R}^n.$$

In particular, if equality holds in above, then, $\partial^{us}_d g(x)$ is called an upper regular convexfactor (URCF) of g at x .

Here, $(g)_d^-(x, v)$ is lower Dini derivative and $(g)_d^+(x, v)$ is upper Dini derivative of g at x in direction v defined, respectively, as follows:

$$(g)_d^-(x, v) := \liminf_{t \rightarrow 0^+} \frac{g(x + tv) - g(x)}{t} \quad \text{and}$$

$$(g)_d^+(x, v) := \limsup_{t \rightarrow 0^+} \frac{g(x + tv) - g(x)}{t}.$$

These Dini derivatives can be finite as well as infinite. When g is locally Lipschitz, both these are finite.

Throughout the paper, the closure, convex hull, closed convex hull, and cone convex hull of any set $S \subset \mathbb{R}^n$ is denoted, respectively, by $\text{cl}S$, $\text{co}S$, $\text{clco}S$, and $\text{coneco}S$.

Here, we consider the following multi-objective programming problem:

$$(P) \text{ minimize } (f_1(x), f_2(x), \dots, f_l(x))$$

$$\text{subject to } h_k(x) \leq 0, k = 1, 2, \dots, m$$

where $f_i : \mathbb{R}^n \rightarrow \mathbb{R}, i = 1, 2, \dots, l, h_k : \mathbb{R}^n \rightarrow \mathbb{R}, k = 1, 2, \dots, m$.

The feasible set for (P) is given by

$$E := \{x \in \mathbb{R}^n : h_k(x) \leq 0, k = 1, 2, \dots, m\}.$$

A point which is not an optimal solution can be a quasi ε -solution. It has been given for scalar problem in [3].

Now, we define quasi ε -weak efficient solution of (P).

Definition 2 Let $\varepsilon \geq 0, \bar{x} \in E$ is said to be a quasi ε -weak efficient solution of (P) if for any $x \in E$, following does not hold

$$f_i(x) - f_i(\bar{x}) + \varepsilon \|x - \bar{x}\| < 0 \quad i = 1, 2, \dots, l.$$

When $\varepsilon = 0$, it becomes weak efficient solution of (P).

Following lemma will be used in our main results.

Lemma 1 Let $\varepsilon \geq 0, U$ a nonempty, convex, and compact set and C be a convex cone. If

$$\sup_{u \in U} \langle u, d' \rangle + \varepsilon \|d'\| \geq 0, \quad \text{for all } d' \in C^-,$$

then, $0 \in U + \text{cl}C + \varepsilon B'$,

where $B' = \{\bar{x} \in \mathbb{R}^n : \|\bar{x}\| \leq 1\}$ is a closed ball in \mathbb{R}^n with unit radius and $C^- = \{d' \in \mathbb{R}^n | \langle c, d' \rangle \leq 0, \forall c \in C\}$ is the negative polar cone of C .

Proof We will prove by contradiction. Suppose that $0 \notin U + \text{cl}C + \varepsilon B'$. Then, $-\text{cl}C$ and $U + \varepsilon B'$ are disjoint sets. Applying separation theorem, there exists a non-zero vector $z \in \mathbb{R}^n$ and a real number α such that

$$\langle z, u \rangle < \alpha < \langle z, c' \rangle \text{ for all } u \in U + \varepsilon B', c' \in -\text{cl}C.$$

Hence, $\langle z, u \rangle < \alpha < \langle z, c' \rangle$ for all $u \in U + \varepsilon B', c' \in -C$.

Since $-C$ is a cone, we get $\alpha = 0$ and $\langle z, u \rangle < 0$ for all $u \in U + \varepsilon B'$.

That is, $\langle z, u' + \varepsilon b' \rangle < 0$ for all $u' \in U, b' \in B'$,

Hence, $\langle z, u' \rangle + \langle z, \varepsilon b' \rangle < 0$ for all $u' \in U, b' \in B'$.

By compactness, we have

$$\sup_{u' \in U} \langle z, u' \rangle + \varepsilon \|b'\| \|z\| < 0.$$

That is,

$$\sup_{u' \in U} \langle z, u' \rangle + \varepsilon \|z\| < 0. \quad (1)$$

Let $c' = -c, c \in C$.

Now, we have $\langle z, c' \rangle > 0$ for all $c' \in -C$.

Then, $\langle z, c \rangle < 0$ for all $c \in C \Rightarrow z \in C^-$.

Since $z \in C^-$, by (1), we have contradiction to the given.

Hence, result. \square .

Remark 1 Let each $f_i, i = 1, 2, \dots, l$ admits an USRCF $\partial^{us} f_i(\bar{x}), i = 1, 2, \dots, l$ at \bar{x} . If $\lambda_i > 0, i = 1, 2, \dots, l, \sum_{i=1}^l \partial^{us}(\lambda_i f_i)(\bar{x})$ is an USRCF of $\sum_{i=1}^l \lambda_i f_i$ at \bar{x} .

Proof Now

$$\left(\sum_{i=1}^l \lambda_i f_i \right)_d^+ (\bar{x}, v) \leq \sum_{i=1}^l (\lambda_i f_i)_d^+ (\bar{x}, v) = \sum_{i=1}^l \lambda_i (f_i)_d^+ (\bar{x}, v)$$

Since each $f_i, i = 1, 2, \dots, l$ admits an USRCF $\partial^{us} f_i(\bar{x}), i = 1, 2, \dots, l$ at \bar{x} , it follows from above that

$$\begin{aligned} \left(\sum_{i=1}^l \lambda_i f_i \right)_d^+ (\bar{x}, v) &\leq \lambda_1 \sup_{\xi_1 \in \partial^{us} f_1(\bar{x})} \langle \xi_1, v \rangle + \dots + \lambda_l \sup_{\xi_l \in \partial^{us} f_l(\bar{x})} \langle \xi_l, v \rangle \\ &= \sup_{\xi'_1 \in \lambda_1 \partial^{us} f_1(\bar{x})} \langle \xi'_1, v \rangle + \dots + \sup_{\xi'_l \in \lambda_l \partial^{us} f_l(\bar{x})} \langle \xi'_l, v \rangle \text{ as } \lambda_i > 0, i = 1, 2, \dots, l. \end{aligned}$$

$$= \sup_{\xi' \in \sum_{i=1}^l \lambda_i \partial^{us} f_i(\bar{x})} \langle \xi', v \rangle,$$

Using Lemma 1 (a) [22], we get

$$\left(\sum_{i=1}^l \lambda_i f_i \right)_d^+(\bar{x}, v) \leq \sup_{\xi' \in \sum_{i=1}^l \partial^{us}(\lambda_i f_i)(\bar{x})} \langle \xi', v \rangle.$$

Hence result \square .

3 Approximate Necessary Conditions

This section is devoted to approximate necessary conditions for a point to be quasi ε -weak efficient solution of (P).

For that, at first we introduce Nonsmooth GCQ as follows:

Let $K(\bar{x}) := \{k = 1, 2, \dots, m : h_k(\bar{x}) = 0\}$ and $\bar{x} \in E$. Suppose that the functions $h_k, k \in K(\bar{x})$ admit upper convexifactors $\partial^u h_k(\bar{x}), k \in K(\bar{x})$ at \bar{x} .

Nonsmooth GCQ holds at \bar{x} if

$$\left(\bigcup_{k \in K(\bar{x})} \text{co} \partial^u h_k(\bar{x}) \right)^- \subseteq \text{clco} T(E, \bar{x}),$$

where $T(E, \bar{x})$ denotes the tangent cone to E at \bar{x} .

Following are the approximate necessary conditions.

Theorem 1 *Let $\varepsilon \geq 0$. Suppose that \bar{x} is a quasi ε -weak efficient solution of (P). Let each $f_i, i = 1, 2, \dots, l$ admits a bounded USRCF $\partial^{us} f_i(\bar{x}), i = 1, 2, \dots, l$ at \bar{x} and $h_k, k = 1, 2, \dots, m$, admits UCF $\partial^u h_k(\bar{x}), k = 1, 2, \dots, m$, at \bar{x} . Suppose that the functions $f_i, i = 1, 2, \dots, l$ and $h_k, k = 1, 2, \dots, m$ are continuous and convex. Assume that Nonsmooth GCQ is satisfied at \bar{x} . Then, there exist scalars $\lambda_k^h \geq 0$ with $\sum_{k=1}^m \lambda_k^h = 1$ such that the following hold:*

$$0 \in \text{cl} \left(\sum_{i=1}^l \text{co} \partial^{us}(\lambda_i f_i)(\bar{x}) \right) + \text{clcone} \left(\sum_{k=1}^m \lambda_k^h \text{co} \partial^u h_k(\bar{x}) \right) + \varepsilon B', \quad (2)$$

$$\lambda_k^h h_k(\bar{x}) = 0, k = 1, 2, \dots, m.$$

Proof Since \bar{x} is a quasi ε -weak efficient solution of (P), therefore, there does not exist any feasible solution $x \in E$ such that the following hold:

$$f_i(x) - f_i(\bar{x}) + \varepsilon \|x - \bar{x}\| < 0, \quad i = 1, 2, \dots, l. \tag{3}$$

Using weighted sum approach, we have following scalar program from (P)

$$(P1) \text{ minimize } \sum_{i=1}^l \lambda_i f_i(x)$$

$$\text{subject to } h_k(x) \leq 0, \quad k = 1, 2, \dots, m$$

where $\lambda_i > 0, i = 1, 2, \dots, l, \sum_{i=1}^l \lambda_i = 1$.

Since $\lambda_i > 0, i = 1, 2, \dots, l$, multiplying (3) by λ_i and adding, we ensure that for all $x \in E$, following does not hold

$$\sum_{i=1}^l \lambda_i f_i(x) - \sum_{i=1}^l \lambda_i f_i(\bar{x}) + \sum_{i=1}^l \lambda_i \varepsilon \|x - \bar{x}\| < 0. \tag{4}$$

Hence, \bar{x} is a quasi ε -solution of (P1).

As Nonsmooth GCQ holds at \bar{x} , we can choose any arbitrary $v \in \text{cl}(\text{co}(T(E, \bar{x})))$.

Thus, we can find a sequence $v_j \in \text{co}(T(E, \bar{x}))$ such that $v_j \rightarrow v$ as $j \rightarrow \infty$.

Now, by Carathéodary's Theorem, we can find elements $v_j^q \in T(E, \bar{x})$ such that $v_j = \sum_{q=1}^{n+1} t_j^q v_j^q$, for all $j \in N$ with $t_j^q \geq 0$ and $\sum_{q=1}^{n+1} t_j^q = 1$, as $v_j \in \text{co}(T(E, \bar{x}))$.

Taking limit as $j \rightarrow \infty$, we get

$$v = \lim_{j \rightarrow \infty} \left(\sum_{q=1}^{n+1} t_j^q v_j^q \right). \tag{5}$$

Since $v_j^q \in T(E, \bar{x})$, we can find sequences $v_j^{q,s} \rightarrow v_j^q$ and $\lambda_j^{q,s} \downarrow 0$ such that $\bar{x} + \lambda_j^{q,s} v_j^{q,s} \in E$ for all $s \in N$.

Taking x to be $\bar{x} + \lambda_j^{q,s} v_j^{q,s}$ in (4), we get that the following does not hold

$$\frac{\sum_{i=1}^l \lambda_i f_i(\bar{x} + \lambda_j^{q,s} v_j^{q,s}) - \sum_{i=1}^l \lambda_i f_i(\bar{x})}{\lambda_j^{q,s}} + \varepsilon \|v_j^{q,s}\| < 0,$$

for sufficiently large $j, s \in N$.

Taking limit supremum in above equation, we get that the following does not hold

$$\limsup_{\lambda_j^{q,s} \rightarrow 0^+} \frac{\sum_{i=1}^l \lambda_i f_i(\bar{x} + \lambda_j^{q,s} v_j^{q,s}) - \sum_{i=1}^l \lambda_i f_i(\bar{x})}{\lambda_j^{q,s}}$$

$$+ \varepsilon \|v_j^q\| = \left(\sum_{i=1}^l \lambda_i f_i \right)_d^+ (\bar{x}, v_j^q) + \varepsilon \|v_j^q\| < 0, \quad q = 1, 2, \dots, n + 1.$$

That is, following does not hold

$$\left(\sum_{i=1}^l \lambda_i f_i \right)_d^+ (\bar{x}, v_j^q) + \varepsilon \|v_j^q\| < 0, \text{ for all } v_j^q \in T(E, \bar{x}) \text{ as } v_j^q \text{ is arbitrary, } q = 1, 2, \dots, n + 1.$$

Since $\sum_{i=1}^l \partial^{us}(\lambda_i f_i)(\bar{x})$ is an USRCF of $\sum_{i=1}^l \lambda_i f_i$ at \bar{x} , from Remark 1, we have

$$\sup_{\xi \in \sum_{i=1}^l \partial^{us}(\lambda_i f_i)(\bar{x})} \langle \xi, v_j^q \rangle + \varepsilon \|v_j^q\| < 0, \text{ for all } v_j^q \in T(E, \bar{x}), \quad q = 1, 2, \dots, n + 1 \text{ does}$$

not hold.

As $t_j^q \geq 0, q = 1, 2, \dots, n + 1$, we have

$$\sup_{\xi \in \sum_{i=1}^l \partial^{us}(\lambda_i f_i)(\bar{x})} \left\langle \xi, t_j^q v_j^q \right\rangle + \varepsilon \|t_j^q v_j^q\| < 0, \text{ for all } v_j^q \in T(E, \bar{x}), \quad q = 1, 2, \dots, n + 1$$

does not hold.

Adding above for all $q = 1, 2, \dots, n + 1$, taking limit as $j \rightarrow \infty$ and using (5), we get

$$\sup_{\xi \in \text{clco} \sum_{i=1}^l \partial^{us}(\lambda_i f_i)(\bar{x})} \langle \xi, v \rangle + \varepsilon \|v\| < 0, \text{ for all } v \in \text{cl}(\text{co}(T(E, \bar{x}))) \text{ does not hold.}$$

That is,

$$\sup_{\xi \in \text{clco} \sum_{i=1}^l \partial^{us}(\lambda_i f_i)(\bar{x})} \langle \xi, v \rangle + \varepsilon \|v\| \geq 0, \text{ for all } v \in \text{cl}(\text{co}(T(E, \bar{x}))).$$

Since Nonsmooth GCQ holds at \bar{x} , we have

$$\sup_{\xi \in \text{clco} \sum_{i=1}^l \partial^{us}(\lambda_i f_i)(\bar{x})} \langle \xi, v \rangle + \varepsilon \|v\| \geq 0, \text{ for all } v \in C^-,$$

where C^- is the negative polar cone of C and C is defined by

$$C := \text{coneco} \left(\bigcup_{k \in K(\bar{x})} \text{co} \partial^u h_k(\bar{x}) \right).$$

Using Lemma 1, and the convex hull property $\text{co}(S_1 + S_2) = \text{co}S_1 + \text{co}S_2$, of subsets S_1 and S_2 of \mathbb{R}^n , we get

$$0 \in \text{cl} \left(\sum_{i=1}^l \text{co} \partial^{us}(\lambda_i f_i)(\bar{x}) \right) + \text{clconeco} \left\{ \bigcup_{k \in K(\bar{x})} \text{co} \partial^u h_k(\bar{x}) \right\} + \varepsilon B',$$

which ensures the existence of a sequence

$$x_j \in \sum_{i=1}^l \text{co}\partial^{us}(\lambda_i f_i)(\bar{x}) + \text{coneco} \left\{ \bigcup_{k \in K(\bar{x})} \text{co}\partial^u h_k(\bar{x}) \right\} + \varepsilon B',$$

such that $x_j \rightarrow 0$ as $j \rightarrow \infty$.

As in general, convexifactors are nonconvex sets, and we can find sequence of scalars $\{\lambda_{kj}^h\}$, $\lambda_{kj}^h \geq 0$, $k \in K(\bar{x})$, $\lim_{j \rightarrow \infty} \sum_{k \in K(\bar{x})} \lambda_{kj}^h := 1$ such that

$$x_j \in \sum_{i=1}^l \text{co}\partial^{us}(\lambda_i f_i)(\bar{x}) + \text{cone} \left(\sum_{k \in K(\bar{x})} \lambda_{kj}^h \text{co}\partial^u h_k(\bar{x}) \right) + \varepsilon B'.$$

Because $\{\lambda_{kj}^h\}$, $k \in K(\bar{x})$ is bounded sequence, assuming that the sequence $\lambda_{kj}^h \rightarrow \lambda_k^h$, $k \in K(\bar{x})$ as $j \rightarrow \infty$, we have

$$0 \in \text{cl} \left(\sum_{i=1}^l \text{co}\partial^{us}(\lambda_i f_i)(\bar{x}) \right) + \text{clcone} \left(\sum_{k \in K(\bar{x})} \lambda_k^h \text{co}\partial^u h_k(\bar{x}) \right) + \varepsilon B',$$

with $\sum_{k \in K(\bar{x})} \lambda_k^h := 1$.

Now, $h_k(\bar{x}) := 0$ for $k \in K(\bar{x})$, therefore $\lambda_k^h h_k(\bar{x}) := 0$ for $k \in K(\bar{x})$.

For $k \notin K(\bar{x})$, $h_k(\bar{x}) < 0$, so let $\lambda_k^h = 0$, $k \notin K(\bar{x})$.

Then, we have $\lambda_k^h h_k(\bar{x}) := 0$, $k = 1, 2, \dots, m$ and

$$0 \in \text{cl} \left(\sum_{i=1}^l \text{co}\partial^{us}(\lambda_i f_i)(\bar{x}) \right) + \text{clcone} \left(\sum_{k=1}^m \lambda_k^h \text{co}\partial^u h_k(\bar{x}) \right) + \varepsilon B',$$

with $\sum_{k=1}^m \lambda_k^h = 1$.

Hence, theorem is proved. \square .

Following example illustrates above theorem.

Example 1 Consider the following problem

$$\text{minimize } [f_1(x, y), f_2(x, y)]$$

$$\text{subject to } h(x, y) \leq 0,$$

where $f_1, f_2, h : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$ are defined as

$$f_1(x, y) := \begin{cases} |x|, & x \geq 0, y \geq 0 \\ x^2 - |y|, & x < 0, y \in \mathbb{R} \\ x^2 + y, & x \geq 0, y < 0 \end{cases}$$

$$f_2(x, y) := \begin{cases} x^2 + y, & x \geq 0, y \in \mathbb{R} \\ y, & x < 0, y \in \mathbb{R} \end{cases}, \quad h(x, y) := -\frac{y}{2}$$

We can see that $\lambda_1 f_1$ admits an USRCF $\partial^{us}(\lambda_1 f_1)(0, 0) := \{(\frac{1}{2}, 0), (0, \frac{1}{2})\}$, $\lambda_2 f_2$ admits an USRCF $\partial^{us}(\lambda_2 f_2)(0, 0) := \{(0, \frac{1}{2})\}$ and $\sum_{i=1}^2 \text{co}\partial^{us}(\lambda_i f_i)(0, 0) := \{(\frac{1}{2}\lambda, 1 - \frac{1}{2}\lambda) | 0 \leq \lambda \leq 1\}$ with $\lambda_1 = \frac{1}{2}, \lambda_2 = \frac{1}{2}$, h admits an UCF $\partial^u h(0, 0) := \{(0, \frac{-1}{2})\}$ at $(0, 0)$.

Nonsmooth GCF holds at $(0, 0)$.

Set of feasible solutions, $E \subset \mathbb{R} \times \mathbb{R}$ is given by $E := \{(0, y) | y \geq 0\}$ and $T(E, (0, 0)) := \{(0, y) | y \geq 0\}$.

$(0, 0)$ is a quasi ε -weak efficient solution for $\varepsilon \geq 1$, but it is not an optimal solution of the problem.

Then, there exist scalar $\lambda^h = 1, \varepsilon = 1, (0, \frac{-1}{2}) \in B'$ such that $(0, 0) \in \text{cl}(\sum_{i=1}^2 \text{co}\partial^{us}(\lambda_i f_i)(0, 0)) + \text{clcone}(\lambda^h \text{co}\partial^u h(0, 0)) + \varepsilon B'$,

$$\lambda^h h(0, 0) = 0.$$

4 Conclusions

In this paper, we have developed approximate necessary optimality conditions using the concept of upper convexifactors for quasi ε -weak efficient solution of a multi-objective programming problem.

Acknowledgements The author would like to thank the two anonymous referees of this paper for their helpful comments and suggestions that has improved the paper.

References

1. Bhatia, D., Gupta, A., Arora, P.: Optimality via generalized approximate convexity and quasi efficiency. *Optim. Lett.* **7**, 127–135 (2013)
2. Beldiman, M., Panaitescu, E., Dogaru, L.: Approximate quasi efficient solutions in multiobjective optimization. *Bull. Math. Soc., Sci. Math. Roum.*, **51**(99), 109–121 (2008)
3. Capâta, A.: Optimality conditions for ε -quasi solutions of optimization problems via ε -upper convexifactors with applications. *Optim. Lett.* **13**, 857–873 (2019)
4. Choung, T.D., Kim, D.S.: Approximate solutions of multiobjective optimization problems. *Positivity* **20**, 187–207 (2016)

5. Demyanov, V.F.: Convexification and concavification of positively homogeneous function by the same family of linear functions, Report 3.208,802 Universita di pisa (1994)
6. Dutta, J., Chandra, S.: Convexifactors, generalized convexity and optimality conditions. *J. Optim. Theory Appl.* **113**, 41–65 (2002)
7. Giorgi, G., Jimenez, B., Novo, V.: Approximate Karush-Kuhn-Tucker condition in multiobjective optimization. *J. Optim. Theory Appl.* **171**, 70–89 (2016)
8. Gao, Y., Yang, X., Teo, K.L.: Optimality conditions for approximate solutions in vector optimization problems. *J. Ind. Manag. Optim.* **7**(2), 483–496 (2011)
9. Ghaznavi, M.: Optimality conditions via scalarization for approximate quasi efficiency in multiobjective optimization. *Filomat* **31**(3), 671–680 (2017)
10. Gutiérrez, C.: Optimality conditions for weak solutions of vector optimization problems through quasi interiors and improved sets. *J. Nonlin. Convex Anal.* **20**, 2507–2523 (2019)
11. Gutiérrez, C., Huegra, L., Jime'nez, B.: Optimality conditions for approximate proper solutions in multiobjective optimization with polyhedral cones. *TOP28*, 526–544 (2020)
12. Gutiérrez, C., Jime'nez, B., Novo, V.: Optimality conditions via scalarization for a new ε -efficiency concept in vector optimization problem. *Eur. J. Oper. Res.* **201**, 11–22 (2010)
13. Hejazi, M.A.: On approximate Karush-Kuhn-Tucker condition for multiobjective optimization problems. *Iranian J. Sci. Tech.* **42**(2), 873–879 (2018)
14. Jiao, L., Lee, H.L.: Approximate optimality and approximate duality for quasi approximate solutions in robust convex semidefinite programs. *J. Optim. Theory Appl.* **176**, 74–93 (2018)
15. Jeyakumar, V., Luc, D.T.: Nonsmooth calculus, maximality and monotonicity of convexificators. *J. Optim. Theory Appl.* **101**, 599–621 (1999)
16. Kabgani, A., Soleimani-damaneh, M., Zamani, M.: Optimality conditions in optimization problems with convex feasible set using convexifactors. *Math. Methods Oper. Res.* **86**, 103–121 (2017)
17. Kohli, B.: Optimality conditions for optimistic bilevel programming problem using convexifactors. *J. Optim. Theory Appl.* **152**(3), 632–651 (2012)
18. Kohli, B.: Necessary and sufficient optimality conditions using convexifactors for mathematical programs with equilibrium constraints. *Rairo Oper. Res.* **53**, 1617–1632 (2019)
19. Kohli, B.: Sufficient optimality conditions using convexifactors for optimistic bilevel programming problem. *J. Ind. Manag. Optim.* **17**(6), 3209–3221 (2021)
20. Kutateladze, S.S.: Convex ε -programming. *Sov. Math. Dokley* **20**, 391–393 (1979)
21. Lee, H. J., Jiao, L.: On quasi ε -solution for robust convex optimization problems. *Optim. Lett.* **11**, 1609–1622 (2017)
22. Li, X.F., Zhang, J.Z.: Necessary optimality conditions in terms of convexificators in lipschitz optimization. *J. Optim. Theory Appl.* **131**, 429–452 (2006)
23. Loridan, P.: Necessary conditions for ε -optimality. *Math. Program. Study* **19**, 140–152 (1982)
24. Loridan, P.: ε -solutions in vector minimization problems. *J. Optim. Theory Appl.* **43**, 265–276 (1984)
25. Suneja, S.K., Kohli, B.: Optimality and duality results for bilevel programming problem using convexifactors. *J. Optim. Theory Appl.* **150**, 1–19 (2011)
26. Yameng, Z., Guolin, Y.: Optimality conditions and saddle point theorems of approximate quasi weak efficient solutions for robust multiobjective programming. *J. Jilin Univ. Sci. Ed.* **59**(2), 250–256 (2021)

Discontinuity at Fixed Point Over Partial Metric Spaces



Fahimeh Mirdamadi, Hossein Monfared, Mehdi Asadi, and Hossein Soleimani

Abstract In this paper, we give a solution to the question of the existence of contraction that generates a fixed point for the mapping but which does not force it to be continuous at the fixed point over partial metric spaces.

Keywords Fixed point · Partial metric space

Mathematics Subject Classification: 47H10 · 54H25

1 Introduction and Preliminaries

The generalization of metric space is called partial metric spaces (PMS) that introduced by Mathews [1] as a part of the study of denotational semantics of data flow networks, and he gave a Banach fixed point result for these spaces. After that, many authors proved fixed point theorems on PMS. For more details, see [2–12].

In the sequel, let

$$\Phi := \{\phi : \mathbb{R}_+ \rightarrow \mathbb{R}_+ : \phi(t) < t, \quad \forall t > 0\} \text{ and } \mathbb{N}_0 = \mathbb{N} \cup \{0\}.$$

In 2017, Ravindra and Pant [13] proved the following fixed point theorem. In what follows we shall denote

F. Mirdamadi

Department of Mathematics, Isfahan (Khorasgan) Branch, Islamic Azad University, Isfahan, Iran

H. Monfared

Department of Mathematics, Germe Branch, Islamic Azad University, Germe, Iran

M. Asadi (✉)

Department of Mathematics, Zanjan Branch, Islamic Azad University, Zanjan, Iran

e-mail: masadi@iauz.ac.ir

H. Soleimani

Department of Mathematics, Malayer Branch, Islamic Azad University, Malayer, Iran

$$M(x, y) = \max\{p(x, y), p(x, \tau x), p(y, \tau y), [p(x, \tau y) + p(y, \tau x)]/2\}. \quad (1)$$

Theorem 1.1 *Let (X, d) be a complete metric space. If τ self-mapping on X satisfies the conditions;*

- i. $d(\tau x, \tau y) \leq \phi(M(x, y))$, where $\phi : \mathbb{R}_+ \rightarrow \mathbb{R}_+$ is such that $\phi(t) < t$ for each $t > 0$;
- ii. For a given $\epsilon > 0$ there exists a $\delta(\epsilon) > 0$ such that $\epsilon < M(x, y) < \epsilon + \delta$ implies $d(\tau x, \tau y) \leq \epsilon$,

then τ has a unique fixed point, say z . Moreover, τ is continuous at z iff

$$\lim_{x \rightarrow z} \max\{d(x, \tau x), d(z, \tau z)\} = 0.$$

Definition 1.1 ([1, 14], [Definition 1.1]) *A partial metric on a nonempty set X is a function $p : X \times X \rightarrow \mathbb{R}^+$ such that for all $x, y, z \in X$:*

- (p1) $p(x, x) = p(y, y) = p(x, y) \iff x = y$,
- (p2) $p(x, x) \leq p(x, y)$,
- (p3) $p(x, y) = p(y, x)$,
- (p4) $p(x, y) \leq p(x, z) + p(z, y) - p(z, z)$.

A partial metric space is a pair (X, p) such that X is a nonempty set and p is a partial metric on X .

By (p1) and (p2), it is easy to see that $p(x, y) = 0$ implies $x = y$. A PMS is a pair (X, p) such that X is a nonempty set and p is a partial metric on X . It is obvious that any metric is a partial metric.

Example 1.1 [1] Let $p : \mathbb{R}^+ \times \mathbb{R}^+ \rightarrow \mathbb{R}^+$ be defined by $p(x, y) = \max\{x, y\}$ for any $x, y \in X$. Then p is a partial metric on \mathbb{R}^+ .

Any partial metric p on X induces a topology on (X, p) which has a base of p -balls:

$$B_p(x, \epsilon) = \{y \in X : p(x, y) < p(x, x) + \epsilon\}$$

for all $x \in X$ and $\epsilon > 0$. We denote this topology by τ_p . The sequence $\{x_n\}$ in X is called Cauchy if $\lim_{m, n \rightarrow \infty} p(x_n, x_m)$ exists and is finite. This sequence is convergent to x if

$$p(x, x) = \lim_{n \rightarrow \infty} p(x, x_n) = \lim_{m, n \rightarrow \infty} p(x_n, x_m).$$

(X, p) is called complete if any Cauchy sequence in X converges.

A mapping $f : X \rightarrow Y$ between two PMS (X, p) and (Y, q) is called τ_p -continuous (or simply continuous) at $x_0 \in X$ if for any $\epsilon > 0$, there exists $\delta > 0$ such that

$$f(B_p(x_0, \delta)) \subseteq B_q(fx_0, \epsilon).$$

Recall that f is sequentially continuous in x_0 if $fx_n \rightarrow fx_0$ whenever $x_n \rightarrow x_0$.

Theorem 1.2 *Suppose that $f : X \rightarrow Y$ is a map between PMS (X, p) and (Y, q) . Then f is continuous if and only if it is sequentially continuous.*

Proof Suppose that f is continuous in x_0 and $\epsilon > 0$. Then there exists $\delta > 0$ such that

$$f(B_p(x_0, \delta)) \subseteq B_q(fx_0, \epsilon).$$

Then, $p(x, x_0) < p(x_0, x_0) + \delta$ implies $q(fx, fx_0) < q(fx_0, fx_0) + \epsilon$. Suppose $x_n \rightarrow x_0$. So, $p(x_n, x_0) \rightarrow p(x_0, x_0)$, then

$$\exists N \in \mathbb{N}, \forall n \geq N, |p(x_n, x_0) - p(x_0, x_0)| < \delta.$$

Therefore, $p(x_n, x_0) < p(x_0, x_0) + \delta$. So,

$$q(fx_n, fx_0) - q(fx_0, fx_0) < \epsilon.$$

On the other hand, $q(fx_0, fx_0) \leq q(fx_n, fx_0)$ and then

$$|q(fx_n, fx_0) - q(fx_0, fx_0)| < \epsilon,$$

therefore, $fx_n \rightarrow fx_0$.

Now, in contrary, suppose that f is sequentially continuous at x_0 and is not continuous in x_0 . Then

$$\exists \epsilon_0 > 0, \forall n \geq 1, \exists x_n \in X, p(x_n, x_0) < p(x_0, x_0) + \frac{1}{n}, \quad (2)$$

$$q(fx_n, fx_0) \geq q(fx_0, fx_0) + \epsilon_0.$$

So,

$$|p(x_n, x_0) - p(x_0, x_0)| = p(x_n, x_0) - p(x_0, x_0) < \frac{1}{n},$$

therefore, $p(x_n, x_0) \rightarrow p(x_0, x_0)$ and then $x_n \rightarrow x_0$ that implies $fx_n \rightarrow fx_0$. So,

$$q(fx_n, fx_0) - q(fx_0, fx_0) \rightarrow 0$$

contradicts with (2). Therefore, f is continuous at x_0 . \square

Remark 1.1 Similar to proof of the above theorem, we can prove that a function $f : X \times X \rightarrow Y$ is continuous at (x_0, y_0) if it is sequentially continuous at (x_0, y_0) .

If (X, p) is a partial metric space, then the function $p^s : X \times X \rightarrow \mathbb{R}^+$ defined by

$$p^s(x, y) = 2p(x, y) - p(x, x) - p(y, y),$$

is a metric on X . (See [1].)

Theorem 1.3 [15] *A PMS (X, p) is complete iff the metric space (X, p^s) is complete. Furthermore,*

$$\lim_{n \rightarrow \infty} p^s(a, x_n) = 0 \iff p(a, a) = \lim_{n \rightarrow \infty} p(a, x_n) = \lim_{m, n \rightarrow \infty} p(x_n, x_m).$$

Main Results

In this section, we generalized main result of [13] on PMS.

Theorem 1.4 *Let (X, p) be a complete PMS. Let τ be a self-mapping on X such that τ^2 is continuous and $\phi \in \Phi$ satisfy the conditions;*

- i. $p(\tau x, \tau y) \leq \phi(M(x, y))$;
- ii. $\forall \epsilon > 0 \exists \delta(\epsilon) > 0$ s.t. $\epsilon < M(x, y) < \epsilon + \delta \Rightarrow p(\tau x, \tau y) \leq \delta$.

Then τ has a unique fixed point x^ , and $\tau^n x \rightarrow x^*$ for each $x \in X$. Further ,*

$$\tau \text{ is discontinuous at } x^* \iff \lim_{x \rightarrow x^*} M(x, x^*) \neq 0.$$

Proof Let $x_0 \in X$ and let $x \neq \tau x$. Define $\{x_n\} \subseteq X$ by $x_{n+1} = \tau^n x_0 = \tau x_n$ and $p_n = p(x_n, x_{n+1})$ for all $n \in \mathbb{N}_0$. Then by (i)

$$p_n = p(x_n, x_{n+1}) = d(\tau x_{n-1}, \tau x_n) \leq \phi(M(x_{n-1}, x_n)) < M(x_{n-1}, x_n) = \max\{p_n, p_{n-1}\} = p_{n-1}.$$

Since

$$\begin{aligned} & M(x_{n-1}, x_n) \\ = & \max \left\{ p(x_n, x_{n-1}), p(x_n, x_{n-1}), p(x_n, x_{n+1}), \frac{p(x_{n-1}, x_{n+1}) + p(x_n, x_n)}{2} \right\} \\ \leq & \max \left\{ p(x_n, x_{n-1}), p(x_n, x_{n+1}), \frac{p(x_{n-1}, x_n) + p(x_n, x_{n+1}) - p(x_n, x_n) + p(x_n, x_n)}{2} \right\} \\ = & \max \{p_n, p_{n-1}\}. \end{aligned}$$

Thus, $\{p_n\} \subseteq \mathbb{R}^+$ is a strictly decreasing sequence and tends to a limit $p \geq 0$. We may assume that $p > 0$. Then

$$\exists k \in \mathbb{N} \forall n \quad (n \geq k \Rightarrow p < p_n < p + \delta(p)). \tag{3}$$

It follows from (ii) and $p_n < p_{n-1}$ where $p < p_n < p + \delta(p)$, for all $n \geq k$. So $p < p_{n+1} < p + \delta(p)$. By (ii) we have $p_{n+1} \leq p$ for $n \geq k$, which contradicts so $p = 0$.

Now we get

$$\lim_{n \rightarrow \infty} p(x_n, x_{n+1}) = 0.$$

On the other hand, $0 \leq p(x_n, x_n) \leq p(x_n, x_{n+1})$ for all $n \in \mathbb{N}$ and so we get

$$\lim_{n \rightarrow \infty} p(x_n, x_n) = 0.$$

To checking $\{x_n\}$ as a p -Cauchy sequence: fix an $\epsilon > 0$. Without loss of generality, we may assume that $\delta(\epsilon) < \epsilon$. Since

$$p_n \rightarrow 0 \Rightarrow \exists k \in \mathbb{N} \forall n \quad (n \geq k \Rightarrow p_n < \frac{1}{2}\delta),$$

Following Jachymski [7, 16], by induction we show that, for any $n \in \mathbb{N}$,

$$p(x_k, x_{k+n}) < \epsilon + \frac{1}{2}\delta. \tag{4}$$

The (3) is clear for $n = 1$. By the (p4), we have

$$p(x_k, x_{k+n+1}) \leq p(x_k, x_{k+1}) + p(x_{k+1}, x_{k+n+1}) - p(x_{k+1}, x_{k+1}). \tag{5}$$

It suffices to show that

$$p(x_{k+1}, x_{k+n+1}) \leq \epsilon. \tag{6}$$

To that end, we shall prove that $M(x_k, x_{k+n}) \leq \epsilon + \delta$, where

$$M(x_k, x_{k+n}) = \max \left\{ \begin{array}{l} p(x_k, x_{k+n}), p(x_k, \tau x_k), p(x_{k+n}, \tau x_{k+n}), \\ [p(x_k, \tau x_{k+n}) + p(x_{k+n}, \tau x_k)]/2 \end{array} \right\}. \tag{7}$$

By the induction hypothesis, we get

$$p(x_k, x_{k+n}) < \epsilon + \frac{1}{2}\delta, \quad p(x_k, x_{k+1}) < \frac{1}{2}\delta, \quad p(x_{k+n}, x_{k+n+1}) < \frac{1}{2}\delta.$$

Also,

$$\begin{aligned} \frac{1}{2}[p(x_k, x_{k+n+1}) + p(x_{k+1}, x_{k+n})] &\leq \frac{1}{2}[p(x_k, x_{k+n}) + p(x_{k+n+1}, x_{k+n}) - p(x_{k+n}, x_{k+n}) \\ &\quad + p(x_k, x_{k+1}) + p(x_k, x_{k+n}) - p(x_k, x_k)] \\ &< \epsilon + \delta. \end{aligned}$$

Thus, $M(x_k, x_{k+n}) < \epsilon + \delta$ hence by (ii) $p(x_{k+1}, x_{k+n+1}) \leq \epsilon$. Hence, (3) implies that $\{x_n\}$ is a p -Cauchy sequence. Since X is complete, there exists a point $x^* \in X$ such that $\lim_{n \rightarrow \infty} x_n \rightarrow x^*$. Also $\tau x_n \rightarrow x^*$ and $\tau^2 x_n \rightarrow x^*$. By Theorem 1.2 and continuity of τ^2 , we have $\tau^2 x_n \rightarrow \tau^2 x^*$. This implies $\tau^2 x^* = x^*$. We claim that $\tau x^* = x^*$. For if $x^* \neq \tau x^*$, we get

$$\begin{aligned} p(x^*, \tau x^*) &= p(\tau^2 x^*, \tau x^*) \leq \phi(M(\tau x^*, x^*)) < M(\tau x^*, x^*) \\ &= \max\{p(\tau x^*, x^*), p(\tau x^*, \tau^2 x^*), p(x^*, \tau x^*), [p(\tau x^*, \tau x^*) + p(x^*, \tau^2 x^*)]/2\} \\ &= p(x^*, \tau x^*), \end{aligned}$$

which is a contradiction. Therefore, $x^* \in \text{Fix}(\tau)$. Uniqueness deduces by (i). \square

Example 1.2 Let $p(x, y) = \max\{x, y\}$ for $x, y \in [0, 2]$. Define $\tau : [0, 2] \rightarrow [0, 2]$ by

$$\tau(x) = \begin{cases} 1, & x \leq 1; \\ 0, & x > 1. \end{cases} \quad \text{and} \quad \phi(t) := \begin{cases} 0, & t < 1; \\ 1, & 1 \leq t. \end{cases}$$

Then

$$M(x, y) = \begin{cases} 1, & x \leq y \leq 1; \\ y, & 1 < x < y, \quad x < 1 < y; \end{cases}$$

and

$$p(\tau x, \tau y) = \begin{cases} 1, & x \leq y \leq 1, \quad x < 1 < y; \\ 0, & 1 < x < y. \end{cases}$$

The conditions of Theorem 1.4 are hold. Also $\tau^2 = 1$ is continuous.

The [13, Corollaries 2.2 and 2.3] and [13, Theorem 2.1] are as our direct results.

References

1. Matthews, S.G.: Partial metric topology. Ann. NY. Acad. Sci. **728**, 183–197 (1994)
2. Ćirić L.B.: On contraction type mappings. Math. Balkanica **1**, 52–57 (1971)
3. Ćirić, L.B.: A generalization of Banach contraction principle. Proc. Am. Math. Soc. **45**(2), 267–273 (1974)
4. Altun, I., Asim, M., Imdad, M., Alfaqih, W.M.: Fixed point results for F_R -generalized contractive mappings in partial metric spaces. Math. Slovaca **69**(6), 1413–1424 (2019)
5. Boyd, D.W., Wong, J.S.: On nonlinear contractions. Proc. Am. Math. Soc. **20**, 458–464 (1969)
6. Hicks, T., Rhoades, B.E.: Fixed points and continuity for multivalued mappings. Int. J. Math. Math. Sci. **15**, 15–30 (1992)
7. Jachymski, J.: Equivalent conditions and Meir-Keeler type theorems. J. Math. Anal. Appl. **194**, 293–303 (1995)

8. Kannan, R.: Some results on fixed points-II. *Amer. Math. Monthly* **76**, 405–408 (1969)
9. Matkowski, J.: Integrable solutions of functional equations. *Dissertationes Math.* **127**, 1–68 (1975)
10. Meir, A., Keeler, E.: A theorem on contraction mappings. *J. Math. Anal. Appl.* **28**, 326–329 (1969)
11. Rhoades, B.E.: A comparison of various definitions of contractive mappings. *Trans. Am. Math. Soc.* **226**, 257–290 (1977)
12. Rhoades, B.E.: Contractive definitions and continuity. *Contemp. Math.* **72**, 233–245 (1988)
13. Ravindra, K.B., Pant, R.P.: A remark on discontinuity at fixed points. *J. Math. Anal. Appl.* **455**, 1239–1242 (2017)
14. Shatanawi, W., Postolache, M.: Coincidence and fixed point results for generalized weak contractions in the sense of Berinde on partial metric spaces. *Fixed Point Theory Appl.* **2013**, 54 (2013). <https://doi.org/10.1186/1687-1812-2013-54>
15. Oltra, S., Otero, O.: Banach's fixed point theorem for PMS. *Rend. Ist. Mat. Univ. Trieste* **36**(1–2), 17–26.1 (2004)
16. Jachymski, J.: Common fixed point theorems for some families of maps. *Indian J. Pure Appl. Math.* **25**, 925–937 (1994)

Fuzzy Logic Implementation in Home Appliances



Aditya Priya, Akanksha Kumari, and Mayank Singh

Abstract Fuzzy logic allows the vague assessments by human in computing problems. It provides better assessment of options. Currently, development work and research are going on in the fields of applications of fuzzy in software, integration of fuzzy logic with neural network, self-learning, fuzzy-control systems and adaptive genetic software systems. In this paper, fuzzy logic for three basic appliances, i.e. fan, AC and microwave have been proposed. The proposed logic in case of fan will control fan speed in accordance with the air temperature. In case of AC, the proposed logic can control compressor speed, fin direction, fan speed and mode of operation with change in temperature and humidity. In case of microwave, cooking time and cooking temperature can be controlled as per the type of food, amount of food and degree of coldness of the food. For this, rules are feed into the fuzzy logic controller. The results from fuzzy logic controller are given to the models of respective appliances. Implementing artificial intelligence this way also helps to save energy.

Keywords Fuzzy logic controller · Microwave oven · Air conditioner

1 Introduction

The concept of Fuzzy logic was introduced in 1965 by Lofti Zadeh. The fuzzy logic deals with imprecise and incomplete information linguistically by numerical methods using membership functions. The reason behind choosing Fuzzy logic is that it grants better accuracy as well as response time for the required process and also due to its control based on 'IF-THEN' rules.

The implementation of fuzzy logic-based methods in adaptive techniques provides reduced complexity in nonlinear conditions. When implemented with IoT, Fuzzy

A. Priya (✉) · A. Kumari · M. Singh
Birla Institute of Technology, Mesra, Ranchi, India
e-mail: adityapriya24458@gmail.com

M. Singh
e-mail: mayanksingh@bitmesra.ac.in

logic controller (FLC) is used to navigate and operate whole system. The outputs from IoT devices comprising of sensors and Bluetooth are feeded as inputs to the fuzzy logic controllers which decides on the basis of rules defined. Outputs from FLC are communicated through IoT devices to assist the user for that appliance.

- (i) A fuzzy logic system is flexible, so a modification can be made easily.
- (ii) Any type of information is accepted by the system even if it is imprecise, distorted and error input.
- (iii) These systems can provide solutions to any complex problems as these systems involve human reasoning and decision making.
- (iv) These systems can be constructed easily as they don't require any mathematical modelling.
- (v) These systems can be combined easily and they deliver improved efficiency with conventional control system.

Currently, appliances having artificial intelligence are preferable. These are used in washing machines, vacuum cleaners, facial pattern recognitions, etc.

To gain control over manageable network performances, the incorporation of genetic algorithm in sensor nodes and soft computing technologies like fuzzy logic has to be modified with some rules. Input of the fuzzy controllers is the output from IoT devices comprising Wi-Fi, Bluetooth, sensors, etc. Rules are developed based on requirements and conditions to generate output. These outputs are communicated to the user through IoT devices. Hence, aim of the project is to build a home automation model and fuzzy logic or rules for home appliances like air conditioner [1], microwave oven [2, 3] and fan [4, 5] so that it is safer, reliable and comfortable.

2 Fuzzy Logic

- A membership function of a fuzzy logic is a curve that represents crisp values mapped to a particular value which lies between 0 and 1. Membership functions are used in defuzzification and fuzzification steps of a Fuzzy logic system (FLS). There are different types of MFs which are triangular, trapezoidal, Gaussian, singleton and piecewise linear. Triangular membership is used in this report.
- A fuzzy rule is decided to control the output variables. Fuzzy rule is a simple 'IFTHEN' rule which contains a condition along with a conclusion.
- The possible algorithms of fuzzy logic are Mamdani model (used when there are less number of input data sets) and Sugeno model (used when there are comparatively more number of input data sets). We are using Mamdani model as it is used for small number of data sets.
- The mechanism of Mamdani Fuzzy Inference System is described below:
- Fuzzification: In this phase, input variables are fuzzified. This is the process of converting a crisp quantity into a fuzzy quantity. On the basis of available information, membership functions are defined for each input variable. For example,

if we say the temperature is 16 °C, the fuzzification process decides whether it is hot or cold for human body.

- Inference: In this phase, fuzzy rules are set. For example, if the temperature is hot, then speed of fan is high.
- Aggregation values obtained from previous phases are united to get a fuzzy set for each output variable.
- Defuzzification: In this phase, fuzzy quantity obtained is converted into crisp quantity. The technique of centroid of the area was used in this defuzzification process. In this report, we use centroid of area technique for defuzzification process.

3 Fuzzy Logic for Microwave Oven

Microwave oven [2, 3] is the one of the most extensively used home appliances. The Microwave ovens work on the principle of alternating electromagnetic field generated inside the Oven which leads to excitation, rotation and collisions of molecules and ions inside the food. The heating of foods inside it depends upon composition of food like moisture content and also on molecular arrangement of the food. So, fuzzy rules or logics are designed to control temperature and time required for cooking by keeping these in knowledge (Fig. 1).

Input variables: (i) Degree of coldness of the food: Degree of coldness of the food plays vital role in measuring degree of temperature and time required for heating the food in Microwave oven. Suppose, if a food is stored in refrigerator, it'd be colder when compared to the food stored at normal room temperature. (ii) Amount

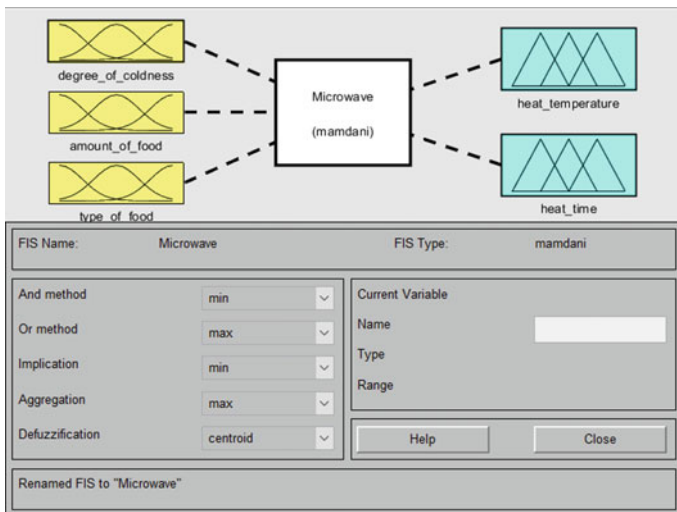


Fig. 1 The design for microwave ovens on MATLAB

of food: Amount of food being heated is another significant factor on which output parameters depends. Clearly, large amount of food requires more time for heating as compared to little amount of food. (iii) Type of food: Type of food is another factor on which the time taken to heat and heating temperature depends. For example, the liquid drinks like milk, soup takes mere little amount of time and low temperature as compared to solid foods like chicken and vegetables for heating. Solid foods require high temperature and long time for efficient heating.

Output variables: (i) Heating Temperature: The heating temperature is classified as very low, low, medium, hot and very hot. Depending on degree of coldness, type and amount of food, the temperature is set automatically according to the feeded fuzzy rules in the rulebook. (ii) Heating Time: The time required for heating is classified as very short, short, medium, long and very long. Depending on degree of coldness, type and amount of food, the time is set automatically according to the embedded fuzzy rules.

4 Fuzzy Logic for Air Conditioner

The air conditionings systems are widely used in today's world and are found in homes and publicly capsulated areas to make snug surroundings. To control fin direction, compressor speed, mode of operation and fan speed in air conditioner according to the sensed temperature and humidity, we've designed a fuzzy logic or rules which not only control these but also save the electrical energy.

Input variables: Temperature—It is the temperature of the room on which the output of system depends. Humidity—It plays a significant role in controlling the air conditioner.

Output variables: (i) Compressor Speed—It is the speed of rotation of compressor which rotates according to temperature and humidity values. (ii) Fin direction—Fins are part of the condenser that assists heat in moving away from the AC so that the heat disperses more quickly. (iii) Fan Speed—It affects the ratio of dehumidification to cooling. The work of fan in AC is to circulate cool or hot air in the room. (iv) Operation Mode—It decides whether the AC would work like a dehumidifier or normal, i.e., AC (Fig. 2).

5 Fuzzy Logic for Fan

Fans are basic requirement of our day-to-day life. For fan, the fuzzy logic control has been developed to control the speed of rotation of fan according to air temperature. In this, air temperature is taken as the input variable, whereas the output variable is fan speed.

Input variables: Air Temperature—The value of air temperature plays a very magnificent role in determining the speed of rotation of fan in smart home appliances.

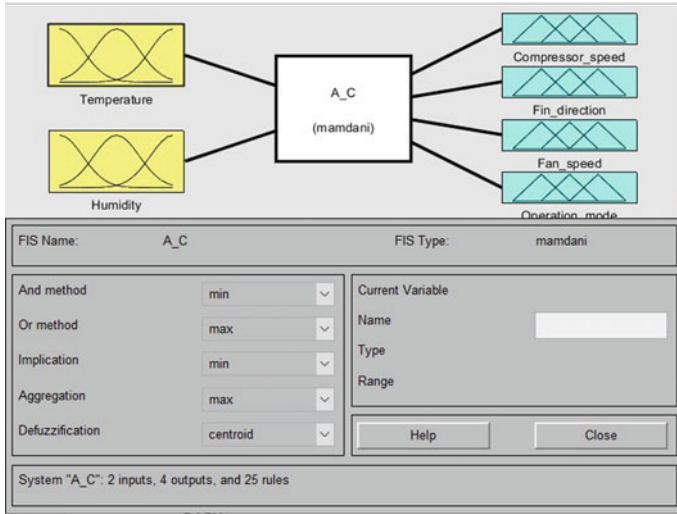


Fig. 2 Design for air conditioner on MATLAB

The requirement of fan’s speed depends on temperature of the room which changes according to the weather. For example, if there is winter season, then the fan will be OFF. But, if the weather changes, it becomes hot then the sensor senses the temperature and sends the data and speed of fan is controlled accordingly.

Output variables: Fan Speed—The fan speed is the output which is given during fuzzy control of fan. The fan speed depends on the room temperature. According to the temperature, the speed of rotation of fan becomes slow, normal, fast and very fast, respectively (Fig. 3).

6 Results

The results obtained after implementing fuzzy logic on microwave oven, air conditioner and fan are presented below.

6.1 For Microwave Oven

If the degree of coldness of food is 2.6, i.e. very cold, amount of food is 170, i.e. little and type of food is 88.2, i.e. solid then according to the logic, the heating temperature will be very hot and of range [170 200 230] and heating time is medium and of range [20 27 35]. And from graph, we can see that heating time is 27.4 min and heating temperature is 200 °C which lies in the given range and completely satisfies the logic (Fig. 4).

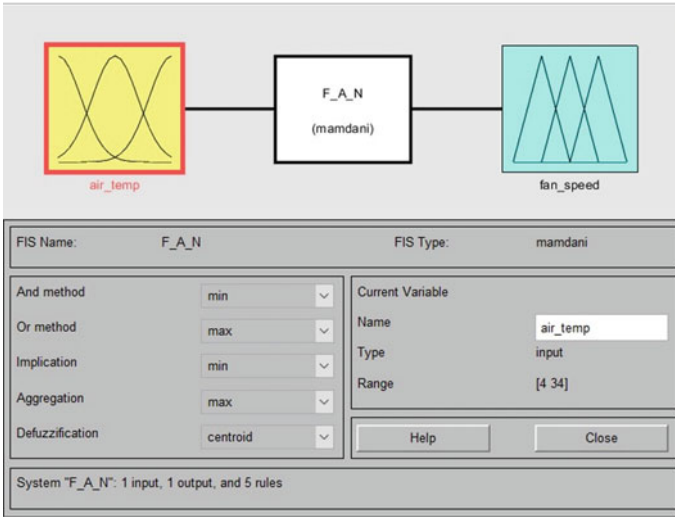
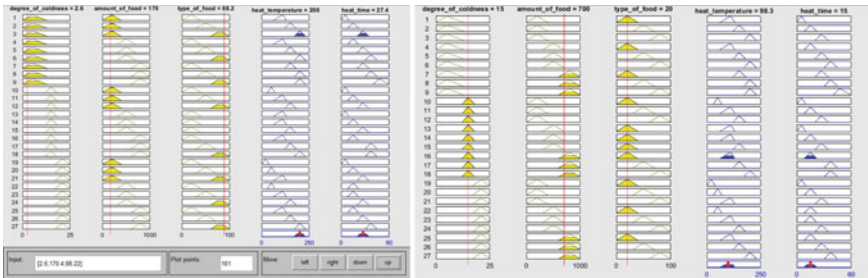
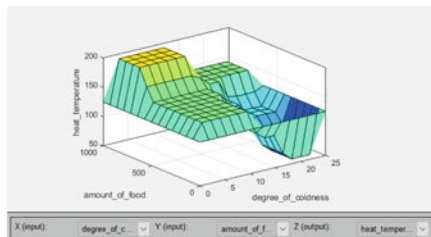


Fig. 3 Design for fan on MATLAB

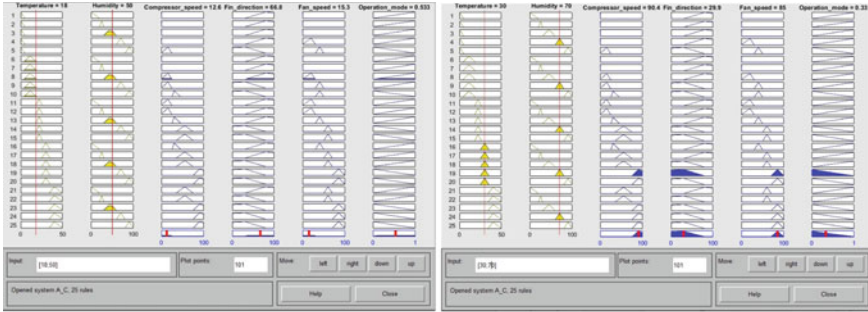


(a)

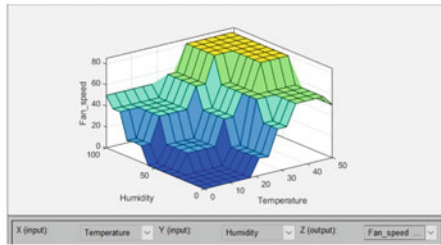


(b)

Fig. 4 a, b Show the inference results of microwave oven



(a)



(b)

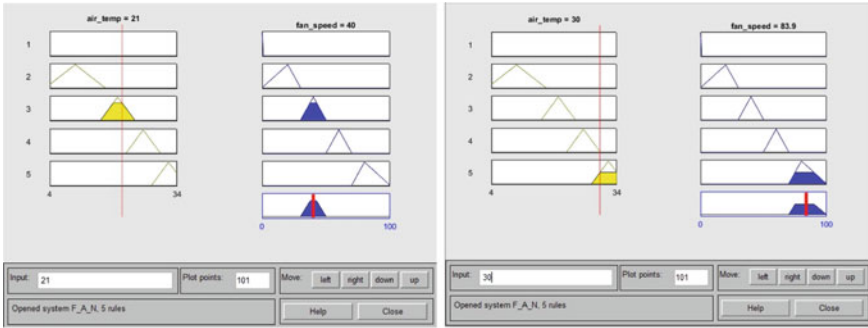
Fig. 5 a, b Inference results of air conditioner

6.2 For Air Conditioner

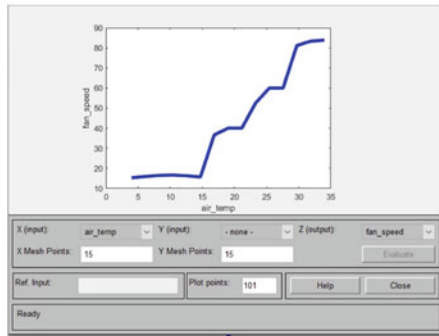
If the temperature of room is 18 °C, i.e. cold, humidity is 50%, i.e. comfortable and then according to the logic, the compressor speed will be very low and of range [0 15 25], fin direction is away and of range [30 80 200], the fan speed will be very low and of range [0 20 30], operation mode is AC and of range [0 1 2]. And from the graph, we can see that compressor speed is 12.6, fin direction is 66.8, fan speed is 15.3 and operation mode is 0.533 which lies in the given range and completely satisfies the logic (Fig. 5).

6.3 For Fan

If air temperature is 5 °C, i.e. cold, then according to the rules fan speed will be slow lying in the range [0 20 30]. And from graph, we can see the fan speed is 15.7 which completely satisfies our logic (Fig. 6).



(a)



(b)

Fig. 6 a, b Inference results of fan

7 Conclusion

Fuzzy logic has been introduced in the paper. The fuzzy logic and rules for home appliances like microwave oven, air conditioner and fan are explained in details and their results are verified through obtained graphs. On the basis of obtained result, it can be concluded that implementing the fuzzy logic in home appliance can helps in saving power consumptions.

References

1. Sobhy, S., Khedr, W.: Developing of fuzzy logic controller for air condition system. Int. J. Comput. Appl. (2015)
2. Deborah Lynn, N., Sourav, A.I., Santoso, A.J.: A fuzzy logic-based control system for microwave ovens. J. Phys., Conf. Ser., 01/01/2021
3. Shaik, A.A.R., Polaki, H.K., Bhavani, S.B., Mann, M.: Microwave oven controller using a fuzzy logic system. In: 10th International Conference on Cloud Computing, Data Science & Engineering, 2020

4. Bonato, J., Mrak, Z., Badurina, M.: Speed regulation in fan rotation using fuzzy inference system. *Sci. J. Maitime Res.* 23 June 2015
5. Kumari, R., Kumar, S., Sharma, V.: Air conditioning system with fuzzy logic and neuro-fuzzy algorithm. In: *Proceedings of the Second International Conference on Soft Computing for Problem Solving*, Jaipur, 2012

Portfolio Structure of Debt Mutual Funds in Indian Market



Soumya Banerjee, Banhi Guha, Amlan Ghosh, and Gautam Bandyopadhyay

Abstract This study aims at to determine the portfolio structure of Indian Debt Mutual Funds based on risk and return trade of parameters while allocating the weights dynamically. All the Debt Mutual Funds with inception before April 2015 are considered and monthly returns during April 2015–March 2020 are calculated. Some mutual funds are then eliminated due to negative return and/or negative skewness of return. The portfolio is, then, constructed by taking the funds with low standard deviation and high beta value. BSE 100 is taken as the market benchmark and their monthly returns for the same period are calculated. Then, appropriate weightage has been allocated among the funds belonging to the portfolio by using the generalized reduced gradient method. Finally, the results are validated by finding the efficiency scores of the selected funds based on market return parameters with the help of Data Envelopment Analysis. This research study will help the investors in formulating future investment decisions.

Keywords Debt mutual funds · Skewness · Portfolio optimization · Generalized reduced gradient method · Data envelopment analysis

1 Introduction

A mutual fund is a type of financial tool which pools money from investors to invest in securities like stocks, bonds, money market instruments and other assets. They are managed by professional's money managers, who allocate the assets appropriately to produce better returns for the investors. An investor's goal is to raise his/her wealth, which one may achieve by investing in stock markets, etc. However, these

S. Banerjee (✉) · A. Ghosh · G. Bandyopadhyay
Department of Management Studies, NIT Durgapur, Durgapur, West Bengal 713209, India
e-mail: souma.ban@gmail.com

S. Banerjee
Department of Statistics, Amity University Kolkata, Kolkata, West Bengal 700135, India

B. Guha
Department of Management Studies, St. Xavier's University, Kolkata, West Bengal 700160, India

investments involve high risk along with a certain level of return. In such a situation, mutual funds play an important role. They pool money from various investors and invest it in the stock market. They are professionally managed and uses advanced statistical tools to create a portfolio of their own striking a perfect balance between returns and risk, which is not possible for an individual investor. The wealth generated from these investments is then distributed to the shareholders. Earlier, people used to invest money in Fixed Deposits and various safe investment instruments. But nowadays people are inclining more and more toward investing in mutual funds owing to its advantages over Fixed Deposits. A Fixed Deposit offers a pre-decided return, whereas mutual funds offer better returns on long-term investments as they are market-linked. Raju et al. [1] observed that mutual funds provide higher returns than other investments. Fixed Deposits are not liquid as the amount invested is frozen for a certain time-period. Mutual funds possesses higher liquidity as they can be sold even in a short time-period. The interest earned from a Fixed Deposit is taxable depending on the tax slab of the individual, whereas the taxation on Mutual Funds depends on the time period of holding. The Equity Linked Savings Scheme (ELSS) Mutual Fund is considered an alternative to Fixed Deposits for saving tax with the lowest lock-in period and better returns as obtained from Agarwal and Mukhtar [2].

An investor always prefers the risk to be minimum given a certain level of return. A return is the change in price of an asset, investment or project over time, which may be represented in terms of price change or percentage change. Returns are often annualized for comparison purposes, while a holding period return calculates the gain or loss during the entire period an investment was held. Holding period return may be expressed nominally or as a percentage. When expressed as a percentage, it is known as rate of return. The real return accounts for the effects of inflation and other external factors, while the nominal return is only interested in price change. The total return for stocks includes price change as well as dividend and interest payments. Risk, on the other hand, is the measure of volatility that arises in a mutual fund's returns. According to the Modern Portfolio Theory, there are two components of risk associated with Mutual Funds—Unsystematic Risk and Systematic Risk. Unsystematic risk is stock specific and can be reduced through diversification. Systematic Risk is the possibility that the entire market or economy will show losses that affect the investments adversely. The Modern Portfolio Theory does not claim that it can moderate this type of risk. Markowitz [3] said that we can create an optimal portfolio of our investments by diversifying our assets with the aim of minimizing our risk and achieving a certain level of return.

Regarding portfolio optimization of mutual funds, the existing literatures suffer from the fact that they are considering funds from some selected houses or making portfolio on the basis of performance of the funds only. Debasish et al. [4] examined 23 mutual fund schemes, both private and public, and determined the ones which performed excellently. Gupta and Gupta [5], conducted a research on Comparative Performance Evaluation of Select Indian Mutual Fund Schemes. Kiliçman and Sivalingam [6] developed a hybrid model for optimization of assets in equity mutual funds offered by three banks from Malaysia. Prajapati and Patel [7] has made a comparative study on performance evaluation of some selected diversified equity

mutual fund schemes of Indian companies, and he conclude that most of the funds has given positive return during 2007–2011 and among them HDFC and Reliance mutual funds have performed well as compared to the Sensex return. Li and Chan [8] investigated the optimum portfolio of Real Estate Investment Trust (REIT) with the nonlinear Generalized Reduced Gradient method and showed that the performance of global optimal portfolio is better than using Asia–Pacific countries Thus, from the existing literatures, we found that not much work has been done regarding construction of portfolios of mutual funds in Indian market. This prompted us to work on the creation an optimal portfolio of mutual funds, by diversifying the fund’s assets as is given by the Modern Portfolio Theory with respect to Indian market.

This paper, thus, aims to find an optimum portfolio of Indian Debt Mutual Fund by allocating weights to the funds constituting the portfolio using nonlinear optimization technique so as to minimize risk subject to a specific return. The rest of the paper is structured as follows. The next section gives the data and the methodology used for the study. Section 3 presents the results and discuss on the findings while Sect. 4 concludes the study highlighting some future scope of research.

2 Data and Methodology

Debt funds invest in securities which generate fixed income like treasury bills, corporate bonds, commercial papers, government securities, and many other money market instruments. They are considered to be low-risk investment options and thus highly favored by the investors.

The period of study is from April 2015 to March 2020. We have considered the monthly Net Asset Value (NAV) of the Indian Debt funds (Association of Mutual Funds of India (AMFI)) [14] whose inception is before April 2015 and are still existing in the market. NAV represents the market value of the mutual fund. It is the price at which the investors buy and redeems the fund’s share. Unlike a stock price, it only changes at the end of any trading day, even though a mutual fund portfolio might include stock options which changes in real time. The monthly returns are calculated from the monthly NAVs by using the formula

$$\text{Return} = \text{Logarithm}(\text{NAV}_t / \text{NAV}_{t-1}); t = 2 \text{ to } 60$$

For this study, a three-stage analysis has been conducted.

In the 1st stage, some funds are eliminated which have a negative average monthly return and/or negative skewness of monthly return. The skewness coefficient is measured by the formula

$$Sk = [(Q_3 - Q_2) - (Q_2 - Q_1)] / 2Q_2$$

where

- Q_3 = Third quartile
- Q_2 = Second quartile or median
- Q_1 = First quartile.

Negatively skewed means that the average return of most of the funds is less than combined average return.

Then, we considered those funds whose Beta values are greater than the average beta value and standard deviation is less than the combined standard deviation.

Standard Deviation—Standard deviation measures the volatility of a fund’s returns. It is the deviation of the returns from their average value. It measures the total risk of an investor. It is calculated using the formula:

$$SD = \left[\sum (x - \bar{x})^2 / n \right]^{1/2}$$

Beta—Beta is used to measure the systematic risk. In a simple linear regression analysis of returns of the fund and the market, it represents the coefficient of the market returns. It measures the changes in the fund relative to the overall market.

$$\beta = \text{Cov}(X, Y) / \text{Var}(X);$$

where, X ’s are the market returns and Y ’s are the mutual fund’s returns.

In the second stage, we would allocate the weights of each of the selected funds by using suitable non-linear optimization technique. Our objective is to minimize the variance of the weighted returns subject to a certain level of return, i.e., we want to minimize

$$\sum \text{Var}(w_i R_i) = \sum w_i^2 \text{Var}(R_i) + \sum \sum w_i w_j \text{Cov}(R_i R_j);$$

where R_i is the return of the i th fund and w_i is the weight of the i th fund.

- subject to constraints $w_i > 0$ for all i ,
- $\sum w_i = 1$, and
- $\sum w_i E(R_i) \geq$ Combined average return.

For this, we use generalized reduced gradient method (GRG) to determine the weights. GRG is a most widely used and efficient methods for portfolio optimization. We have find the application of GRG in constructing the portfolio in many studies viz. Alrabadi [9], Gupta et al. [10], etc.

It is an algorithm to solve nonlinear problems of general structure. Lasdon defines this method as

$$\text{Minimize } f(X), \text{ subject to } g_i(X) = 0, \quad i = 1 \dots m \text{ and } l_i < X_i < u_i, \quad i = 1 \dots n,$$

where X is an n -vector and u_i and l_i are given upper and lower bounds. We assume $m < n$ since, in most cases, $m \geq n$ implies an infeasible problem or one with a unique solution. The fundamental idea of GRG is to use the equalities to express m of the variables, called basic variables, in terms of the remaining $n - m$ non-basic variables. This is also the way the simplex method of linear programming operates.

Finally, we measure the efficiency of the selected funds using Data Envelopment Analysis (DEA). DEA is a nonparametric mathematical programming model (linear) developed by Charnes et al. [11] to measure the relative efficiency of decision-making units (DMUs) in using multiple inputs to produce multiple outputs where the production form is unknown. Charnes et al. [11] defined a model which had an input orientation and assumed constant return to scale (CRS) known as a CCR model while Banker et al. [12] formulated a variable return to scale (VRS) approach and known as a BCC model.

The calculations for an input-oriented DEA for CRS and VRS model which are given below

CRS Model

Min θ

Subject to

$$\sum x_{ij}\lambda_j \leq \theta x_{it}, \quad i = 1, 2, \dots, m \text{ (input constraint)}$$

$$\sum y_{rj}\lambda_j \geq y_{rt}, \quad r = 1, 2, \dots, s \text{ (output constraint)}$$

$$\lambda_j \geq 0 \text{ for all } j = 1, 2, \dots, n$$

VRS Model

Min θ

Subject to

$$\sum x_{ij}\lambda_j \leq \theta x_{it}, \quad i = 1, 2, \dots, m \text{ (input constraint)}$$

$$\sum y_{rj}\lambda_j \geq y_{rt}, \quad r = 1, 2, \dots, s \text{ (output constraint)}$$

$$\lambda_j \geq 0 \text{ and } \sum \lambda_j = 1; \text{ for all } j = 1, 2, \dots, n$$

If $\theta = 1$ or 100% for two or more DMU's, then super-efficiency value needs to be calculated to distinguish them.

To perform DEA, the number of DMU to be equal to the greatest number of multiplications of input and output or three times of the summation of outputs and inputs [13].

In our study, we take mean and SD as the input variable and beta as the output variable.

3 Results and Discussions

There are altogether 172 mutual funds whose inception are before April 2015 and still exists in the market; out of which, 72 funds are there with positive average and positive skewness of monthly returns.

The combined SD is calculated by the formula:

$$\text{combined SD} = \left(\sum n_i \cdot s_i^2 + \sum n_i \cdot (\bar{x}_i - \bar{x})^2 / \sum n_i \right)^{1/2}$$

where s_i^2 = variance of monthly returns of each fund, \bar{x}_j = average monthly returns of each fund, \bar{x} = combined average and n_i = number of monthly returns for each fund (here $n_i = 59$ for all i) and the average beta = sum of the beta values/ n where n = number of funds. Using the formula as stated above, combined SD = 0.0446 and average beta = 0.1501. There are 11 funds whose SD is less than the combined SD and beta is greater than the average beta. These funds along with their SD and beta are given in Annexure.

Next, the weights of these funds are calculated using GRG (Table 1).

Table 1 Weights of the funds using GRG

Funds	Weights
IDFC Banking and PSU Debt Fund—Regular Plan	0
HDFC Corporate Bond Fund—Growth Option	17%
Kotak Corporate Bond Fund—Direct Plan—Growth	22%
SBI Magnum Ultra Short Duration fund—Direct Plan—Growth	29%
PGIM India Dynamic Bond Fund—Direct Plan	0
L&T Liquid Fund—Direct Plan—Growth	0
Edelweiss Money Market Fund—Direct Plan—Growth	8%
DSP Overnight Fund—Direct Plan—Growth	0
ICICI Prudential All Seasons Bond Fund—Direct Plan	0
UTI—Short Term Income Fund—Direct Plan—Growth	0
Baroda Overnight Fund—Direct Plan—Growth	24%

Thus, we find that five funds, namely SBI Magnum Ultra Short Duration fund, Baroda Overnight Fund, Kotak Corporate Bond Fund, HDFC Corporate Bond Fund and Edelweiss Money Market Fund constitutes the portfolio.

The optimum risk of the portfolio is 0.015% and the expected return on the portfolio is 0.34%.

The correlation matrix of the monthly returns of these five funds is given in Table 2.

The value of the determinant is $0.00014 \approx 0$, which implies that the portfolio is diversified, and thus unsystematic risk is eliminated.

Finally, we use DEA to measure the efficiency of the selected funds.

The efficiency of each of these funds by CRS and VRS technique and their rankings are given in Table 3.

Table 2 Correlation matrix

	HDFC Corporate Bond Fund—Growth Option	Kotak Corporate Bond Fund—Direct Plan—Growth	SBI Magnum Ultra Short Duration fund—Direct Plan—Growth	Edelweiss Money Market Fund—Direct Plan—Growth	Baroda Overnight Fund—Direct Plan—Growth
HDFC Corporate Bond Fund—Growth Option	1	0.914968	0.915082	0.909439	0.90125
Kotak Corporate Bond Fund—Direct Plan—Growth	0.914968	1	0.999998	0.853142	0.96752
SBI Magnum Ultra Short Duration fund—Direct Plan—Growth	0.915082	0.999998	1	0.85364	0.89865
Edelweiss Money Market Fund—Direct Plan—Growth	0.90439	0.853142	0.85634	1	0.97542
Baroda Overnight Fund—Direct Plan—Growth	0.90125	0.96752	0.89865	0.97542	1

Table 3 Ranking of the funds using DEA

Funds	CRS_Input_Orient	VRS_Input_Orient	Scale efficiency	Super efficiency	Ranking
IDFC Banking and PSU Debt Fund—Regular Plan	0.9503	0.9507	0.99958	0.99958	3
HDFC Corporate Bond Fund—Growth Option	0.9487	0.9509	0.99769	0.99769	4
Kotak Corporate Bond Fund—Direct Plan—Growth	1	1	0.9992	1.0021	2
SBI Magnum Ultra Short Duration fund—Direct Plan—Growth	1	1	0.99838	1.0467	1
PGIM India Dynamic Bond Fund—Direct Plan	0.8837	0.9116	0.96939	0.96939	5
L&T Liquid Fund—Direct Plan—Growth	0.8824	0.9123	0.96723	0.96723	6
Edelweiss Money Market Fund—Direct Plan—Growth	0.8794	0.9215	0.95431	0.95431	8
DSP Overnight Fund—Direct Plan—Growth	0.9256	Very large	NA	NA	NA
ICICI Prudential All Seasons Bond Fund—Direct Plan	0.8751	0.9224	0.94872	0.94872	9
UTI—Short Term Income Fund—Direct Plan—Growth	0.9041	0.9392	0.96263	0.96263	7
Baroda Overnight Fund—Direct Plan—Growth	0.9838	1.0493	0.93758	0.93758	10

4 Conclusion

Mutual funds are a means of wealth creation. The Asset Management Companies manage these funds using professional expertise, which common individuals lack, to keep a balance between returns and risks. In this study, we have assigned weights to the funds forming the portfolio and in doing that, firstly we have applied some criteria to select the efficient funds and then we have used a nonlinear optimizing technique, GRG for allocating weights to these funds with an objective to minimize the portfolio risk and finally, measure the efficiency of the selected funds using DEA. In the study, we see that an investor should invest maximum in SBI Magnum Ultra Short Duration fund followed by Baroda Overnight Fund, Kotak Corporate Bond Fund, HDFC Corporate Bond Fund and Edelweiss Money Market Fund. The optimal portfolio weight obtained by using GRG suggests investing the largest portfolio proportion in SBI Magnum Ultra Short Duration fund (29%) which is also the most efficient fund as obtained from the DEA. Baroda Overnight Fund and Kotak Corporate Bond Fund, which constitutes the portfolio with 24 and 17% weightage, respectively, also ranks very high in terms of their efficiency scores. The other two funds constituting the portfolio, namely HDFC Corporate Bond Fund and Edelweiss Money Market Fund have also a high efficiency score of more than 90%. The optimum risk of the portfolio is 0.015%, which is fairly low.

Hence this study will help an investor to determine the proportion of diversifying his/her assets to minimize risk and obtain a specified level of return and thereby will allow them to make prudent selection of mutual funds and distribution of appropriate investment weightage which will help them in formulating future investment decisions. This study paves the way for further areas of research. The monthly returns of these five funds can be forecasted for the next year, thereby seeing the effect of COVID-19 an also using this approach we can try to make portfolio taking different types of mutual funds and not restricting to Debt Mutual Funds.

References

1. Raju, J., Manjunath, B.R., Nagaraja, G.M.: Performance evaluation of Indian Equity Mutual Fund Schemes. *J. Bus. Manag. Soc. Sci. Res. (JBM&SSR)* (2015), ISSN (2319-5614)
2. Agarwal, R., Mukhtar, W.: Critical analysis of stock selection strategies of growth mutual funds in India: application of Sharpe optimization technique. Available at SSRN 1537037 (2010)
3. Markowitz, H.: Portfolio selection. *J. Finance* **7**(1), 77–91 (1952)
4. Debasish, S.S., Shil, N.C.: Performance evaluation of selected growth oriented mutual funds. *Adarsh J. Manag. Res.* **2**(2), 1 (2009)
5. Gupta, O.P., Gupta, A.: Performance evaluation of select Indian mutual fund schemes: an empirical study. *The ICFAI J. Appl. Finance* **10**(12), 81–98 (2004)
6. Kiliçman, A., Sivalingam, J.: Portfolio optimization of equity mutual funds—Malaysian case study. *Adv. Fuzzy Syst.* (2010)
7. Prajapati, K.P., Patel, M.K.: Comparative study on performance evaluation of mutual fund schemes of Indian companies. *Res. World* **3**(3), 47 (2012)

8. Li, R.Y.M., Chan, A.: REITs portfolio optimization: a nonlinear generalized reduced gradient approach. *DEStech Trans. Comput. Sci. Eng.* (mso) (2018)
9. Alrabadi, D.W.H.: Portfolio optimization using the generalized reduced gradient nonlinear algorithm. *Int. J. Islamic Middle Eastern Finance Manag.* (2016)
10. Gupta, S., Bandyopadhyay, G., Biswas, S., Upadhyay, A.: A hybrid machine learning and dynamic nonlinear framework for determination of optimum portfolio structure. In: *Innovations in Computer Science and Engineering*, pp. 437–448. Springer, Singapore (2019)
11. Charnes, A., Cooper, W.W., Rhodes, E.: Measuring the efficiency of decision-making units. *Eur. J. Oper. Res.* **2**(6), 429–444 (1978)
12. Banker, R.D., Charnes, A., Cooper, W.W.: Some models for estimating technical and scale inefficiencies in data envelopment analysis. *Manage. Sci.* **30**(9), 1078–1092 (1984)
13. Raab, R.L., Lichty, R.W.: Identifying subareas that comprise a greater metropolitan area: the criterion of county relative efficiency. *J. Reg. Sci.* **42**(3), 579–594 (2002)
14. www.amfiindia.com

Fixed Point Theorems for Digital Images Using Path Length Metric



R. Om Gayathri and R. Hemavathy

Abstract Till recent years, the Euclidean metric has been used to study the results pertaining to the digital images. Boxer [4] had suggested the use of path length metric for digital images as it is more appropriate compared to any other metric. In this paper, we define the path length metric in a different manner. Also, we derive the fixed point results for mappings satisfying quasi-contraction constraint using the same. As an application to the digital image processing, the path length metric is used to obtain the boundary of a given digital image and an algorithm for the same is demonstrated.

Keywords Digital image · Path length metric · Quasi-contraction · Boundary following algorithm

JEL Classification 47H10, 54H25, 68U10

1 Introduction

Digital topology focuses on the topological properties of two-dimensional or three-dimensional digital pictures. A rectangular array of nonnegative numbers forms a digital picture. A digital image is a subset of the digital picture and is composed of picture elements known as pixels.

Digital topology was first studied and coined by Rosenfeld [25] in the year 1979 and emphasizes the need to study some basic topological properties which forms the foundation for the algorithms of digital pictures such as thinning, border following,

R. Om Gayathri (✉)

Department of Mathematics, Meenakshi College for Women (Affiliated to University of Madras), Chennai, Tamil Nadu, India
e-mail: omgayathri.r@gmail.com

R. Hemavathy

Department of Mathematics, Queen Mary's College (Affiliated to University of Madras), Chennai, Tamil Nadu, India
e-mail: hemaths@gmail.com

etc. The concept of continuity for digital pictures and the fixed point property of the continuous functions was discussed in the paper [26] by Rosenfeld in the year 1986. In the year 1989, Kong [21] defined a group analogous to the fundamental group for digital pictures based on the adjacency relation. An extensive study in digital topology has been exhibited by Kong and Rosenfeld in [22, 23]. The concept of continuity for digital pictures introduced by Rosenfeld was extended to homeomorphism, retractions and homotopies by Boxer [2] in the year 1994 and constructed the digital fundamental group based on digitally continuous functions in Boxer [3].

The non-product property of digital fundamental group was demonstrated by Han [14] in the year 2005. The study of blending of fixed point theory and digital images was initiated by Ege and Karaca [12] in the year 2013 by proving the Lefschetz fixed point theorem for digital images. They also proved the Banach fixed point theorem for digital images [13] which was later enhanced by Han [16]. Various contraction principles were studied by different authors [11, 17, 19, 20, 24] in digital metric spaces, where the metric used is the Euclidean metric. In 2015, Han [15] had reviewed many papers including [12, 13] related to the fixed point theorems in digital images.

Boxer [4–8] had pointed out the flaws in the research papers [11, 17, 19, 20, 24] on fixed point theorems in digital metric space and specifically in [7] remarked that the major shortcoming is the disconnection between the metric and the adjacency of points of the digital images. Most of the authors, till now, have used either the Euclidean metric or the Manhattan metric. Boxer [4] suggested that the natural metric which could be used for a connected digital image is the path length metric. The idea of using the path length metric is more relevant since a graph can be easily extracted from the digital image by replacing the pixels by the vertices of the graph and the edges can be defined using the adjacency between the pixels.

In this paper, the path length metric is defined for a connected digital image. Also the graph associated with the digital image is defined which in turn makes a digital image into a digital metric space endowed with graph under the path length metric. Basic definitions namely, convergence, Cauchy sequence in the digital metric space are defined using the path length metric. Fixed point theorem for mappings satisfying quasi-contraction constraint is established for the digital metric space endowed with graph. An example supporting the main theorem is also exhibited. The most celebrated Banach contraction mapping theorem is obtained as a particular case of the main theorem. An application of the path length metric in digital image processing is discussed. Our results opens a new path in digital metric space endowed with graph.

2 Preliminaries

Let Z notate the set of all integers. Then Z^n is the set of lattice points in the n dimensional Euclidean space. A digital image X is a subset of Z^n and is denoted by the ordered pair (X, k) where k represents the adjacency between points in the digital image (For elaborate discussions, one may refer [21, 25]).

Definition 2.1 ([21]) Two points $p = (p_1, p_2, \dots, p_n)$ and $q = (q_1, q_2, \dots, q_n)$ in Z^n are said to be k_l -adjacent if there are at most l indices i such that $\|p_i - q_i\| = 1$ and for all other indices j , $p_j = q_j$. We give below the definition of adjacency relation.

- In Z , the points 2-adjacent to p are $p \pm 1$.
- In Z^2 , the two points p and q are either 4-adjacent or 8-adjacent according as the two points differ exactly in one co-ordinate (either one more or one less than the co-ordinate) or by at most one in each co-ordinate respectively.
- In Z^3 , the two points p and q are either 6-adjacent or 18-adjacent or 26-adjacent according as the two points differ in exactly one co-ordinate(one more or one less than the co-ordinate) or differ by one in at most two co-ordinates or by at most one in each co-ordinate respectively.

A digital image can be treated as a graph with the vertices of the graph are the pixels and the edges of the graph are the line segments between the adjacent points of the digital image, where the adjacency relation is defined as in Definition 2.1. The following figure shows a digital image in Z^2 and its graph (Fig. 1).

Definition 2.2 ([18]) A path of length n in a graph is a sequence $\{x_i\}_{i=0}^n$ of $n + 1$ distinct vertices such that x_i, x_{i+1} are adjacent points of the graph. If there exists a path between any two vertices of a graph, then the graph is said to be connected. The shortest path between any two vertices in a graph denotes the path containing minimum number of edges connecting the two vertices of the graph.

The following definitions and results are pre-requisites for proving the main theorem. Ciric[10] has proved the fixed point theorem for mappings satisfying quasi-contraction constraint in the metric space and a similar result is proved for a digital metric space in the next section.

Definition 2.3 ([10]) Suppose S is a mapping of a metric space X into itself. Let $B \subset X$. Define $\delta(B)$ as $\delta(B) = \text{Sup}\{d(a, b)/a, b \in B\}$. For each $y \in X$ and for any positive integer n , let $O(y, n) = \{y, Sy, S^2y, \dots, S^n y\}$. $O(y, \infty) = \{y, Sy, S^2y, \dots \dots \dots\}$ is called the orbit of S at y .

Definition 2.4 If every Cauchy sequence in $O(y, \infty)$ is convergent in X , then the metric space X is known to be S -orbitally complete.

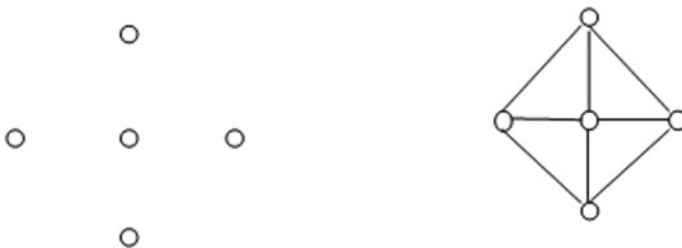


Fig. 1 A digital image in Z^2 and its graph

Remark 2.1 *If a metric space X is complete, then it is S -orbitally complete.*

Definition 2.5 ([10]) A mapping $S : X \rightarrow X$ of a metric space X onto itself is said to be a quasi-contraction if and only if there exists a number $q', 0 \leq q' < 1$, such that

$$d(S(x), S(y)) \leq q' \max\{d(x, y), d(x, Sx), d(y, Sy), d(x, Sy), d(y, Sx)\}$$

holds for every $x, y \in X$.

Lemma 2.1 ([10]) *Let S be a quasi-contraction on X and let n be any positive integer. Then for each $x \in X$ and all positive integers i and $j, i, j \in \{1, 2, \dots, n\}$ implies*

$$d(S^i x, S^j x) \leq q' \delta[O(x, n)].$$

Remark 2.2 ([10]) *From the above lemma it follows that if S is a quasi-contraction and $y \in X$, then for every positive integer n there exists a positive integer $k \leq n$, such that $d(y, S^k y) = \delta[O(y, n)]$.*

Lemma 2.2 ([10]) *Let S be a quasi-contraction on X , then $O(y, \infty) \leq \frac{1}{1 - q'} d(y, Sy)$.*

Theorem 2.1 ([10]) *Let S be a quasi-contraction on a metric space X and let X be S -orbitally complete. Then*

- (a) S has a unique fixed point u in X ;
- (b) $\lim_n S^n x = u$ and
- (c) $d(S^n x, u) \leq \frac{q'}{1 - q'} d(x, Sx)$ for all $x \in X$.

In the next section, the path length metric is defined for digital images and the fixed point results for mappings satisfying quasi-contraction constraint is proved. An example supporting the main theorem is also given. The Banach contraction principle for digital metric space is obtained as a corollary to the main result. An application of the path length metric in border following is discussed and the algorithm for the same is provided.

3 Main Results

We now define the path length metric in a different way for digital images. This metric is defined using the adjacency of points of digital images.

Definition 3.1 Let Y^* denote a digital image. i.e. $Y^* \subset I_2$. We define a graph induced on Y^* as $G^* = (V^*, E^*)$ where $V^* = Y^*$ and $E^* = \{e^*/e^*$ is an edge between two

points p^* and q^* if p^*, q^* are either 4–adjacent or 8–adjacent in Y^* (adjacency between points defined as in Definition 2.1). Define a function d on Y^* as follows:

$$d(p^*, q^*) = \text{length of the shortest path between } p^* \text{ and } q^*.$$

The distance function defined as above is a metric on Y^* , as it satisfies all the three properties of a metric, namely, non-negativity, symmetry and triangle inequality (one can refer [9]). This function is called the path length metric.

Remark 3.1 *The significance of the path length metric is that it is defined using the adjacency of points and it is integer valued compared to the usual Euclidean metric. Also it is more appropriate when graph theoretic approach is used to study the properties of the digital images.*

Remark 3.2 *The set Y^* together with the metric d defined as in Definition 3.1 is called a digital metric space endowed with a graph G^* .*

The following definitions are given for the digital metric space (Y^*, d) .

Definition 3.2 If a sequence $\{x_n\}_{n=1}^\infty$ of points in the digital metric space (Y^*, d) satisfies the following condition, then it is said to be a Cauchy sequence. For each $\epsilon > 0$, there exists $\alpha \in \mathbb{N}$ such that for all $m, n > \alpha$, $d(x_n, x_m) < \epsilon$.

Definition 3.3 In a digital metric space (Y^*, d) , a sequence $\{x_n\}_{n=1}^\infty$ of points is said to converge to a limit $L \in Y^*$, if it satisfies the following condition: For each $\epsilon > 0$, there exists $\alpha \in \mathbb{N}$ such that for all $n > \alpha$, $d(x_n, L) < \epsilon$.

Definition 3.4 If every Cauchy sequence in the digital metric space (Y^*, d) converges to a point in Y^* , then (Y^*, d) is said to be complete.

Remark 3.3 *In a digital metric space, the terms of a Cauchy sequence or convergent sequence coincides after a certain stage and this is proved in the following theorems.*

Theorem 3.1 *In a digital metric space (Y^*, d) , if a sequence $\{x_n\}_{n=1}^\infty$ is a Cauchy sequence, then $x_n = x_m$ for all $m, n > \alpha$, where $\alpha \in \mathbb{N}$.*

Proof According to the Definition 3.2, a sequence $\{x_n\}_{n=1}^\infty$ of points is Cauchy if for all $m, n \geq \alpha$, we have $d(x_n, x_m) < \epsilon$. Since ϵ is a very small positive quantity, $d(x_n, x_m) < \epsilon$ implies the length of the shortest path between x_n, x_m is 0. i.e. $x_n = x_m$ for $m, n \geq \alpha$.

Theorem 3.2 *In a digital metric space (Y^*, d) , if a sequence $\{x_n\}_{n=1}^\infty$ converges to a limit $L \in Y^*$, then there is an $\alpha \in \mathbb{N}$ such that for all $m, n > \alpha$, $x_n = L$ i.e. $x_n = x_{n+1} = x_{n+2} = \dots = L$ for $n \geq \alpha$.*

Proof According to the Definition 3.3, a sequence $\{x_n\}_{n=1}^\infty$ of points converges to a limit $L \in Y^*$, if for all $\epsilon > 0$, $d(x_n, L) < \epsilon$, for all $n > \alpha$, $\alpha \in \mathbb{N}$. From the definition of d , it follows that if $d(x_n, L) < \epsilon$, then $x_n = L$ for all $n \geq \alpha$. i.e. after a certain stage the points of the sequence coincides with L . Therefore $x_n = x_{n+1} = x_{n+2} = \dots = L$ for $n \geq \alpha$.

Theorem 3.3 A digital metric space (Y^*, d) is complete.

Proof Let $\{x_n\}_{n=1}^\infty$ be a Cauchy sequence of points in the digital metric space (Y^*, d) . Then by Theorem 3.1, there exists $\alpha \in \mathbb{N}$ such that $x_n = x_m$ for all $m, n \geq \alpha$. i.e. $x_n = x_{n+1} = x_{n+2} = \dots = L$ for $n \geq \alpha$. Hence by Theorem 3.2, $\{x_n\}_{n=1}^\infty$ is convergent. i.e. The digital metric space (Y^*, d) is complete.

Remark 3.4 From Remark 2.1 and Theorem 3.3 it follows that the digital metric space Y^* is S -orbitally complete.

Following are the definitions of digital contraction and quasi-contraction mappings in a digital metric space.

Definition 3.5 A self mapping S on a digital metric space (Y^*, d) is said to be a digital contraction mapping if and only if there exists a non-negative number $q' < 1$ such that

$$d(Sx^*, Sy^*) \leq q' d(x^*, y^*) \quad \forall x^*, y^* \in Y^*.$$

Definition 3.6 A self mapping S on a digital metric space (Y^*, d) is said to be a quasi-contraction if and only if there exists a non negative number $q' < 1$ such that

$$d(Sx^*, Sy^*) \leq q' \max\{d(x^*, y^*), d(x^*, Sx^*), d(y^*, Sy^*), d(x^*, Sy^*), d(y^*, Sx^*)\} \quad \forall x^*, y^* \in Y^*.$$

An example of a mapping satisfying the quasi-contraction constraint in a digital metric space endowed with graph is given below:

Example 3.1 Let Y^* be a digital image in the Euclidean 2-dimensional space given by $Y^* = \{x_1, x_2, x_3, \dots, x_8\}$ where $x_1 = (2, 1), x_2 = (3, 1), x_3 = (3, 2), x_4 = (3, 3), x_5 = (2, 3), x_6 = (1, 3), x_7 = (1, 2), x_8 = (1, 1)$.

Let G^* be the graph as defined in Definition 3.1 and let d be the path length metric on Y^* (Fig. 2).

Then (Y^*, d) is a digital metric space. Y^* together with the graph G^* is a digital metric space endowed with graph G^* .

Let $S : Y^* \rightarrow Y^*$ be defined as below:

$$S : x_i \rightarrow x_1, \quad i = 1, 2, 3, 5, 7, 8,$$

$$S : x_4 \rightarrow x_2,$$

$$S : x_6 \rightarrow x_8.$$

Now the image of Y^* under S is a straight line as shown in Fig. 3.

The maximum distance (using path length metric) between any two points in Y^* is 3 units whereas in the image of Y^* , the maximum distance between any two points is 2 units. Hence S satisfies

$$d(Sx^*, Sy^*) \leq q' \max\{d(x^*, y^*), d(x^*, Sx^*), d(y^*, Sy^*), d(x^*, Sy^*), d(y^*, Sx^*)\} \quad \forall x^*, y^* \in Y^*.$$

i.e. S is a quasi-contraction on Y^* .

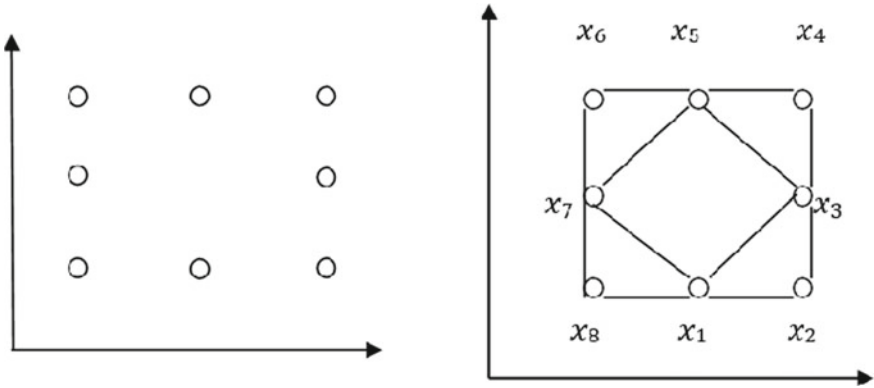
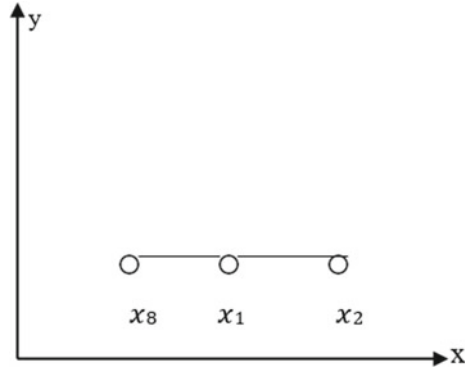


Fig. 2 The digital image Y^* and its graph G^*

Fig. 3 Image of Y^* under the mapping S



Following is the fixed point theorem for mappings satisfying quasi-contraction constraint in a digital metric space.

Theorem 3.4 *Let (Y^*, d) be a digital metric space endowed with graph G^* , where d is the path length metric. Let $S : Y^* \rightarrow Y^*$ be a quasi-contraction on Y^* . Then we have,*

- (a) $\lim_{l \rightarrow \infty} S^l x = u_1$;
- (b) S has a unique fixed point u_1 in Y^* and
- (c) $d(S^l x, u_1) \leq \frac{(q')^l}{(1 - q')} d(x, Sx)$ for all $x \in Y^*$ and $l \in \mathbb{N}$.

Proof Let $x \in Y^*$ be arbitrary. Consider the sequence of iterates $\{x, Sx, S^2x, \dots\}$. Let $x_1 = x, x_l = S^{l-1}x, l = 1, 2, 3, \dots$. We shall prove that $\{x_l\}_{l=1}^\infty$ is a cauchy sequence in Y^* . Let l be any positive integer.

$$d(S^l x, S^{l+1} x) = d(S.S^{l-1} x, S^2.S^{l-1} x).$$

From Lemma 2.1, we have

$$d(S^l x, S^{l+1} x) \leq q' \delta[O(S^{l-1} x, 2)].$$

It follows from the Remark 2.2 of Lemma 2.1, there exists an integer $k', 1 \leq k' \leq 2$ such that

$$\delta[O(S^{l-1} x, 2)] = d(S^{l-1} x, S^{k'} S^{l-1} x).$$

Hence we have,

$$\begin{aligned} \delta[O(S^{l-1} x, 2)] &= d(S.S^{l-2} x, S^{k'+1}.S^{l-2} x) \\ &\leq q' \delta[O(S^{l-2} x, k' + 1)] \\ &\leq q' \delta[O(S^{l-2} x, 3)]. \end{aligned}$$

Therefore, $d(S^l x, S^{l+1} x) \leq q' q' \delta[O(S^{l-2} x, 3)]$.

Proceeding in this way, we have,

$$d(S^l x, S^{l+1} x) \leq (q')^l \delta[O(x, l + 1)] \leq (q')^l \delta[O(x, \infty)].$$

From Lemma 2.2 we have,

$$d(S^l x, S^{l+1} x) \leq \frac{(q')^l}{(1 - q')} d(x, Sx). \tag{1}$$

Since $\lim_{l \rightarrow \infty} (q')^l = 0$, we have $\{S^l x\}_{l=1}^\infty$ is a Cauchy sequence in Y^* . From Theorem 3.1, there exists $\alpha \in \mathbb{N}$ such that $x_m = x_n, m, n \geq \alpha$.

$$\text{Let } x_m = u_1, m \geq \alpha \tag{2}$$

Then $\{x_l\}_{l=1}^\infty$ converges to u_1 .

The uniqueness of the fixed point of S is proved below:

Let $v_1 \neq u_1$ be any other fixed point of S such that $Sv_1 = v_1$. Since S is a quasi-contraction, it follows that

$$\begin{aligned} d(Su_1, Sv_1) &\leq q' \max\{d(u_1, v_1), d(u_1, Su_1), d(v_1, Sv_1), d(u_1, Sv_1), d(v_1, Su_1)\} \\ d(u_1, v_1) &\leq q' d(u_1, v_1) \end{aligned}$$

As $d(u_1, v_1) > 0$, we have, $1 < q'$ which is a contradiction, since $q' \leq 1$. Hence the fixed point of S is unique. Both (a) and (b) is proved.

We now proceed to prove (c).

In (1), choose l so that $l + 2 = \alpha$. Then from (2), $x_{l+2} = u_1$ which implies $S^{l+1} x = u_1$. Hence (1) reduces to

$$d(S^l x, u_1) \leq \frac{(q')^l}{(1 - q')} d(x, Sx).$$

This proves inequality (c) of the theorem.

The following is the Banach contraction theorem for digital images which is a particular case of Theorem 3.4.

Corollary 3.1 *Let (Y^*, d) be a digital metric space endowed with graph G^* , where d is the path length metric. Let $S : Y^* \rightarrow Y^*$ be a digital contraction map on Y^* . Then S has a unique fixed point.*

Proof In our Theorem 3.4, if

$\max \{d(x_1, y_1), d(x_1, Sx_1), d(y_1, Sy_1), d(x, Sy_1), d(y_1, Sx_1)\} = d(x_1, y_1)$ then

$$d(Sx_1, Sy_1) \leq q' d(x_1, y_1), \text{ where } 0 \leq q' < 1.$$

Example 3.2 *Consider the digital image*

$$X = \{(x_i, y_j), 1 \leq x_i \leq 4, 1 \leq y_j \leq 4, i, j = 1, 2, 3, 4\} \subset \mathbb{Z}^2.$$

Define a mapping $S : X \rightarrow X$ as below:

$$S(x_i, y_j) = \begin{cases} (x_1, y_1), & \text{if } x_i = y_j \text{ (or) } d(x_i, y_j), (x_1, y_1) = 1, \\ (x_1 + 1, y_1 + 1), & \text{if } d((x_i, y_j), (x_1, y_1)) = 2, \\ \left. \begin{matrix} (x_1, y_1 + 1), & \text{if } x_i < y_j \\ (x_1 + 1, y_1), & \text{if } x_i > y_j \end{matrix} \right\} & \text{and } d((x_i, y_j), (x_1, y_1)) = 3. \end{cases}$$

for all $i, j = 1, 2, 3, 4$.

The digital pictures of the images corresponding to the mapping S and its graph are depicted below (Figs.4 and 5):

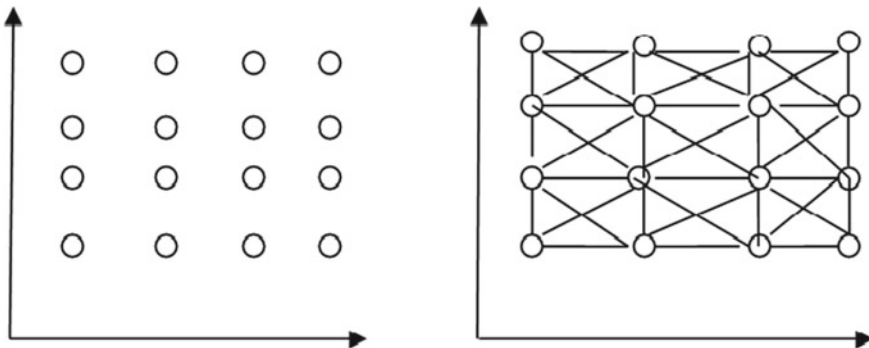


Fig. 4 The digital image in X and Graph of X

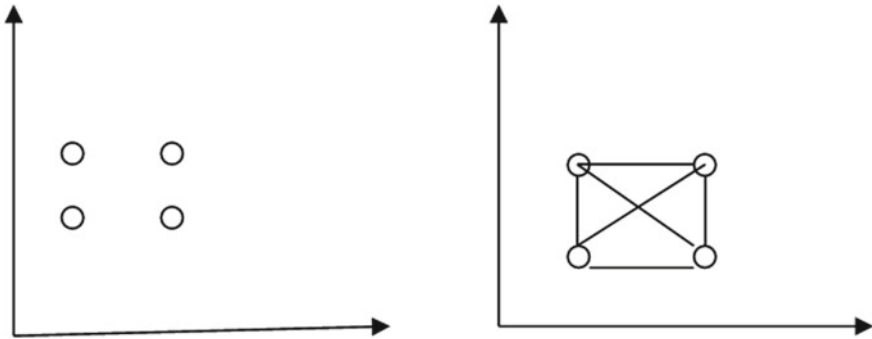
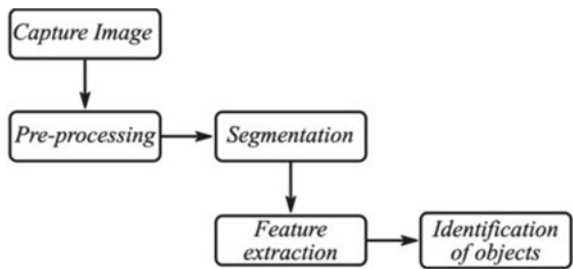


Fig. 5 The digital image in SX and Graph of SX

Fig. 6 Stages of digital image processing



In the above example, the digital image having 16 points is reduced in size to a digital image having 4 points. The distance of other points is measured from the point (x_1, y_1) . The points whose x and the y co-ordinate are equal are mapped to (x_1, y_1) . The points which are at a distance 1 unit from (x_1, y_1) are also mapped to (x_1, y_1) . The points which are at distance two units and three units are respectively mapped to $(x_1 + 1, y_1 + 1)$ and $(x_1, y_1 + 1)$ or $(x_1 + 1, y_1)$ according as $x_i < y_j$ (or) $x_i > y_j$. The mapping S so defined as above satisfies the quasi-contraction constraint and (x_1, y_1) is the fixed point of the mapping.

4 Application to Digital Image Processing

The method of processing a digital image using algorithms is known as digital image processing. A digital image is stored as a set of numbers in a digital computer. The picture is divided into small areas called as pixels. These pixels represent some characteristics of the image such as the brightness or its color. The numbers are arranged in rows and columns similar to a matrix representation. There are various operations performed on a digital image to improve its quality and to extract useful information from the image. The schematic diagram of the steps involved in the digital image processing is given in Fig. 6:

1. **Capture Image:** Image capturing (OR) Image acquisition is the process of acquiring an image from some source usually a hardware-based source. The unprocessed image is acquired using Image capturing.
2. **Pre-processing:** Pre-processing methods deals with suppression of undesired distortions and enhancement of some features required for future operations with the image. The aim of pre-processing is to improve the quality of the image so that it can be analyzed in a better way.
3. **Segmentation:** The process of dividing the digital image into segments (like set of pixels) is called as image segmentation. The main objective of image segmentation is to locate objects and boundaries in images.
4. **Feature Extraction:** Feature extraction is the process by which the dimensionality of the initial set of raw data can be reduced to manageable groups for further processing.
5. **Identification of objects:** This is the last step in Image processing. It involves identifying objects in an image or video sequence.

As an application to digital image processing, the path length metric is used to extract the boundary of a digital image and the algorithm is discussed below. The process of identifying the boundary or extracting the boundary of a digital image is called boundary following or contour tracing in digital image processing. A digital image consists of two types of pixels namely, black or white. A black pixel will be denoted by value '1' and an white pixel is denoted by the value '0'. White pixel forms the background of the image. A boundary of the digital image consists of the set of border points of the image. Border following or contour tracing is necessary to obtain information regarding the shape of the objects of the digital image.

Boundary following Algorithm

There are various algorithms already proposed for boundary following or contour tracing of a digital image. According to the Moore's algorithm, one has to start with the top leftmost pixel and move in the clockwise direction to find the boundary points. In the present algorithm, we find the pair of points having the maximum distance using the path length metric. Start with one of the points in the pair and move clockwise direction to find the remaining boundary points.

Consider a digital image Y^* . Let G^* be the graph associated with Y^* . Let d be the path length metric defined on Y^* . Before giving the algorithm, the various possible 4-adjacency and 8-adjacency of a point in Y^* is illustrated in Fig. 7 (always look for adjacent point in the clockwise direction of the point):

Algorithm 4.1 Let $m = \max\{d(a, b) \mid a, b \in Y^*\}$.

Step 1 Choose pair of points u, v in Y^* such that $m = d(u, v)$.

Step 2 Find a closed path starting from u in the clockwise direction. Set $b_0 = u$.
Let $i = 1$.

Step 3 Let b_i be the next boundary point. Then b_i is either

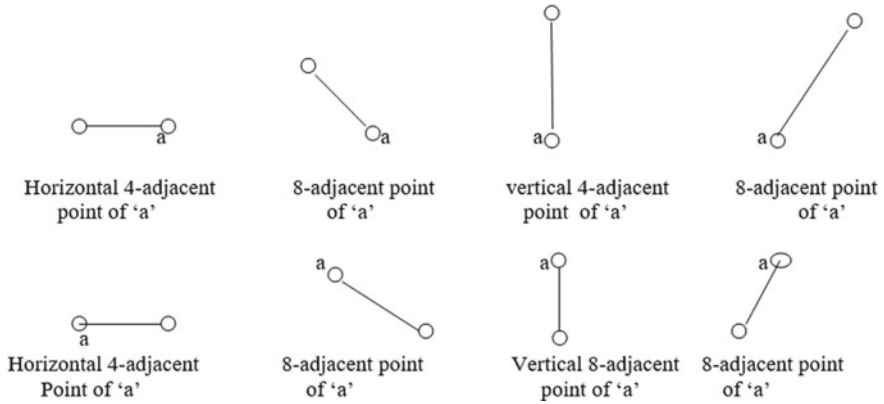
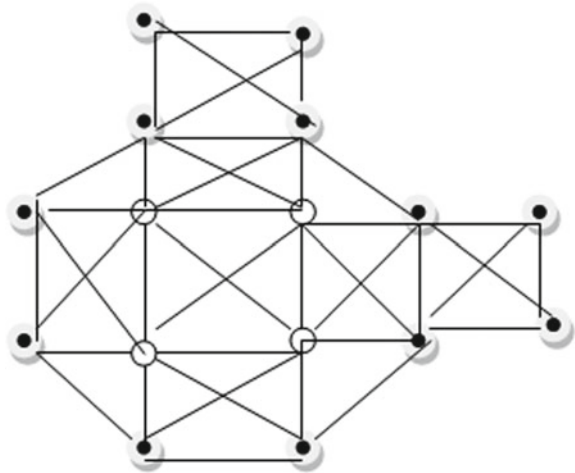


Fig. 7 4-adjacent and 8-adjacent positions of a point

Fig. 8 Graph of digital picture Y^*



- Case (i): b_i is a horizontal 4-adjacent point of b_{i-1} . (OR)
- Case (ii): b_i is a 8-adjacent point of b_{i-1} . (OR)
- Case (iii): b_i is a vertical 4-adjacent point of b_{i-1} . $i = i + 1$.

Step 4 Repeat step 3 until for some l , Either b_l is a 8-adjacent point of u (or) b_l is a horizontal 4-adjacent point of u .

Step 5 The sequence of points $u = b_0, b_1, b_2, \dots, b_l, b_{l+1} = u$ is the boundary of Y^* .

Example 4.1 Consider the following digital picture Y^* .

	1	1		
	1	1		
1	1	1	1	1
1	1	1	1	1
	1	1		

The graph of the above digital picture is given in Fig. 8.

The points shaded black forms the boundary of the given digital picture and is obtained using Algorithm 4.1.

5 Conclusion

In this paper, the digital image is remodeled into a digital metric space using the path length metric. The preliminaries such as convergence of sequence and Cauchy sequence in the digital metric space are defined using path length metric. The behavior of these sequences in the digital metric space are also established by proving related results. The digital version of the fixed point theorem for mappings satisfying quasi-contraction constraint is proved and the Banach Contraction theorem for digital metric space is obtained as a particular case. An example supporting the main theorem is also given. An application of the path length metric in the boundary following or contour tracing in digital image processing is discussed and the algorithm for the same is provided.

Acknowledgements The authors are grateful to the reviewers for their valuable suggestions in improving the quality of the paper.

References

1. Banach, S.: Sur les operations dans les ensembles abstraits et leurs applications aux equations integrales. *Fund. Math.* **3**, 133–181 (1922)
2. Boxer, L.: Digitally continuous functions. *Pattern Recogn. Lett.* **15**, 833–839 (1994)
3. Boxer, L.: A classical construction for the digital fundamental group. *J. Math. Imaging and Vision* **10**, 51–62 (1999)
4. Boxer, L., Staecker, P.C.: Remarks on fixed point assertions in digital topology. *Appl. Gen. Topol.* **20**(1), 135–153 (2019)
5. Boxer, L.: Remarks on fixed point assertions in digital topology, 2. *Appl. Gen. Topol.* **20**(1), 155–175 (2019)
6. Boxer, L.: Remarks on fixed point assertions in digital topology, 3. *Appl. Gen. Topol.* **20**(2), 349–361 (2019)
7. Boxer, L.: Remarks on fixed point assertions in digital topology, 4. *Appl. Gen. Topol.* **21**(2), 265–284 (2020)

8. Boxer, L.: Remarks on Fixed Point Assertions in Digital Topology, 5, <https://arxiv.org/abs/2111.07813v1>[math.GT] (2021)
9. Chartrand, G., Tian, S.: Distance in digraphs. *Comput. Math. App.* **34**(11), 15–23 (1997)
10. Ćirić Lj., A generalization of Banach's contraction principle, *Proc. Amer. Math. Soc.* **45**, 267–273 (1974)
11. Dolhare, U.P., Nalawade, V.V.: Fixed point theorems in digital images and applications to fractal image compression. *Asian J. Math. Comp. Res.* **25**(1), 18–37 (2018)
12. Ege, O., Karaca, I.: The Lefschetz Fixed Point Theorem for Digital Images, *Fixed Point Theory and App.*, p. 253 (2013)
13. Ege, O., Karaca, I.: Banach fixed point theorem for digital images. *J. Nonlinear Sci. App.* **8**, 237–245 (2015)
14. Han, S.E.: Non-product property of the digital fundamental group. *Inf. Sci.* **171**, 73–91 (2005)
15. Han, S.E.: Fixed point theorems for digital images. *Honam Math. J.* **37**, 595–608 (2015)
16. Han, S.E.: Banach fixed point theorem from the viewpoint of digital topology. *J. Nonlinear Sci. App.* **9**, 895–905 (2016)
17. Hossain, A., Ferdausi, R., Mondal, S., Rashid, H.: Banach and Edelstein fixed point theorems for digital images. *J. Math. Sci. App.* **5**(2), 36–39 (2017)
18. Tremblay, J.P., Manohar, R.: *Discrete Mathematical structures with Applications to Computer Science*. McGraw Hill Education (2017)
19. Jyoti, K., Rani, A.: Digital expansions endowed with fixed point theory. *Turk. J. Anal. Number Theory* **5**(5), 146–152 (2017)
20. Jyoti, K., Rani, A.: Fixed point theorems for $\beta - \psi - \varphi$ expansive type mappings in digital metric spaces. *Asian J. Math. Comp. Res.* **24**(2), 56–66 (2018)
21. Kong, T.Y.: Digital fundamental group. *Comput. Graph.* **13**, 159–166 (1989)
22. Kong, T.Y., Rosenfeld, A.: Digital topology: introduction and survey. *Comput. Vis. Graph. Image Process.* **48**, 357–393 (1989)
23. Kong, T.Y., Roscoe, A.W., Rosenfeld, A.: Concepts of digital topology. *Top. App.* **46**, 219–262 (1992)
24. Park, C., Ege, O., Kumar, S., Jain, D., Lee, J.R.: Fixed point theorems for various contraction conditions in digital metric spaces. *J. Comp. Anal. App.* **26**(8), 1451–1458 (2019)
25. Rosenfeld, A.: Digital topology. *Amer. Math. Monthly* **86**(8), 621–630 (1979)
26. Rosenfeld, A.: 'Continuous' functions on digital pictures. *Pattern Recognit. Lett.* **4**, 177–184 (1986)

Emergency Help for Road Accidents



Raman Prakash Verma, Nishchaya Kumar, Santosh Kumar, and S. Sruthi

Abstract Road accidents are very destructive phenomena, because of which India loses almost 1.5 lakh lives every year. Nowadays, number of accidents has increased rapidly. There are many reasons behind the accidents and many programs are running to minimize the accidents. However, when an accident meets, sometimes due to delay in emergency service, people die and local people do not help because of accident case. So here in this work, we propose a system, with the help of latest Internet of Things (IoT) technology that would help in emergency accidents. The system can effectively help in preventing any kind of mishaps and when such conditions occur then it detects and informs the concerned authorities and people, so that the situation can be taken care immediately and victim should not undergo critical health situation. The system is associated with the vehicles and the mobile of user that immediately sends a recorded message with location to the specified contacts selected by the user when accidents happens. This work has been completed using Arduino Board—Node MCU ESP 8266 and sensors like—Ultrasonic Sensor for calculating distance between vehicle and other object, GPS for getting the exact location, Force Sensitive Resistor for sensing the force applied on vehicle.

Keywords Emergency help · IoT · Road accidents help · Force sensitive resistor · Arduino · Ultrasonic sensor

R. P. Verma (✉) · N. Kumar · S. Kumar · S. Sruthi
Department of Computer Applications, Cochin University of Science and Technology, Kochi,
India
e-mail: cusat.ramanprakashverma@gmail.com

N. Kumar
e-mail: knishchaya@gmail.com

S. Kumar
e-mail: cusat.santoshkumar@gmail.com

S. Sruthi
e-mail: sruthisreeku89@gmail.com

1 Introduction

The title of this work, i.e., “Emergency Help for Road Accidents” quietly clears our idea behind this work that we just want to provide Emergency help for the victims of road accidents meanwhile the system will inform to Hospital, Family, Friend, etc. with exact location.

There is no such existing research on the relevant topic using force sensitive register (FSR).

This work is completely devoted to provide the emergency help to those who got injured in any road accident. To help people this work is completely based on IoT. The system will definitely reduce the death rate of accidents because many people dies just in lack of emergency help and by means of this work, the system will provide emergency help with exact location.

This paper elaborates things in very simple way that is our system will run on distance and measuring force. It means that if you are at very close to other vehicle then our system will indicate the driver and by chance vehicle collide with each other, then our system will generate a message/loud notification to nearby hospital, police station and to emergency contact also, i.e., his/her family contact that was set by user in initial stage.

For developing such a useful system we have worked with some IoT components like, we have used here Force sensitive resistor (FSR), GPS sensor, Ultra sonic distance finder sensor, and obviously one Arduino board, i.e., ESP8266 NODE MCU.

The ultrasonic sensor is used for finding the distance between two devices, Force sensitive register that measure the force after collision, this is the hero of this system, GPS that will navigate the current location and due to this, and we can send the location to the hospital and police station and family contact.

It is easy to use for users as there is an application where user can login and can set emergency number and turn on the GPS mode, and it will be activate the system and system will gently work.

2 Literature Review

Some recent works related to emergency help for road accidents are as follows.

Kattukaran et al. [1] This work is basically based for two wheeler vehicles and has used accelerometer sensor for detecting weather the vehicle has fallen down, GPS for detecting the live location, using the concept of sending accident alert to nearest hospital friends and family. Mainly this work has two modules one detects both vehicles has fallen down or not and other modules consist of heartbeat sensor MSP430 microcontroller, buzzer and Bluetooth module. The second module will only active when the vehicle falls.

Sharma and Sebastian [2] the work is based on identifying the accident based on car acceleration, retardation, heartbeat of driver. For this work, they have used

accelerometer, vibration sensor, heartrate sensor. Since the heartrate sensor is in seat belt so there is need of special care for seat belt and if with some reasons (like if car is static and driver is just sitting) user do not tie seat belt then system will fail to respond. In addition, the system is dedicated only for cars.

Habib et al. [3] In this work the sensors used are temperature, gas, flame, piezo, accelerometer, gps. It asks services from nearest hospital only, It do not informs family and friend. There is need to register much hospital on map.

Gokul Lal et al. [4] The work is completely dependent upon mobile application so there is much chance of human or machine error. Because of being dependent upon mobile application user should confirm the availability of telecommunication network all time. As per the work user phone should be in vehicle which will capture images in every 10 seconds if accident happens but if in accident user phone is damaged then there will be problem for user's phone to function.

Vishnupriyan et al. [5] In this work author has used ultrasonic sensor, gps, gsm, crash sensor, IR sensor, lcd, dc motor, zigbee, pic controller, and a system to monitor driver's eyes for drowsiness checking. To reduce the vehicle speed DC motor is used and PIC controller controls the DC motor.

In all the works [6–11] that has been studied, accelerometer sensor is used for detecting vibrations, inclination, speed, seismic activity. Therefore, there is much chance of machine error with this sensor and with application of this sensor; the model will be helpful only when vehicle is moving. Therefore, instead of this in our work we have used Force sensitive resistor, which can be effective in both cases the vehicle is either moving or static. In addition, FSR measures the force applied to it, so it will be helpful to identify the collision of vehicles.

3 Research Problem

Accidents are one of the major problem over the world. There are many development have been done to cure accidents. However, sometimes people dies because of unavailability of immediate help also. Therefore, this paper presents a method how to provide immediate help to people who are victim of road accidents.

4 Proposed System

As we know according to the data of WHO approximately 1.35 million peoples loses their life in road crashes. We can say that there can be many people who dies just because of the unavailability of immediate help. There can be many reasons behind an accident, i.e., over speeding, drunken driving, violation of traffic rules, improper weather, defects of road, etc. but beyond of this after any accident we can save some life just by providing them the emergency help at right time.

Here we are presenting a solution for minimizing the death rate from road accidents. Our work is devoted to provide immediate help accidents on time for all the peoples who are injured in any road accidents, so that we can save some life and minimize the death rate.

Now let us talk about the model we have worked on, for this solution. The model on which we had worked is a combination of hardware and software with IoT features or even we can say that it is an IoT-based model. The main idea behind this model is that the hardware devices will be attached to user vehicle and by means of mobile interface, the user will turn on his WIFI, so that the user phone got connected with hardware device. Not only this but user can also put some contacts to their emergency contacts list by means of mobile interface so that when any mishap happens with user the user's emergency contacts will be notified by a loud alarm even if their phone is in silent mode with exact location. To develop our model we have used Arduino Board ESP8266 the main objective of using this board is this board comes with inbuilt WIFI feature. Let us come to the working of model all the sensors including GPS, FSR, and Ultrasonic sensor are attached with Arduino board using some jumper wires and breadboard. This device should be installed in user's vehicle and the application for this model should be installed in user's phone. In the mobile application user can add any contacts to his emergency contact list like family and friend. When user is driving the vehicle, the ultrasonic sensor keeps measuring the distance between user's vehicle and any other vehicle or object. If the distance is less than 10 m then the device will give notification to driver so that he/she can slow down vehicle and can prevent the accident. If unfortunately accident happens the FSR will detect the collision with magnitude, i.e., with what amount of force vehicle collide and if the magnitude is beyond the set threshold the module will be activated and with the help of GPS it will notify to nearest hospital for emergency help. Meanwhile the emergency contacts, which were added by user, will also be notified even if their phone is in silent mode with exact location. The idea behind adding family members and friends to emergency contact is that with some unfortunate reasons hospital may ignore the notification but friend and family will not ignore. So in that case they can also contact to nearest hospital or police station for immediate help and if possible they can also reach to accident victim as soon as possible.

Now let us describe the hardware device and its components in detail. For development of this device, we have used Arduino board ESP8266 NODE MCU, Force sensitive resistor (FSR), Ultra Sonic distance finder sensor, GPS, one 1.1 k resistor and some jumper wires.

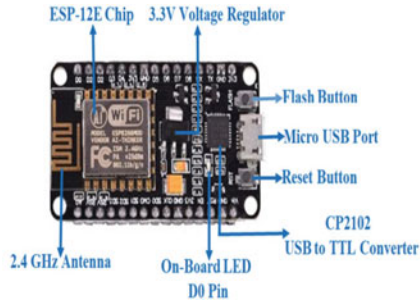
A. Hardware Components

A. Hardware Components

ESP8266 NODE MCU (Arduino Board)



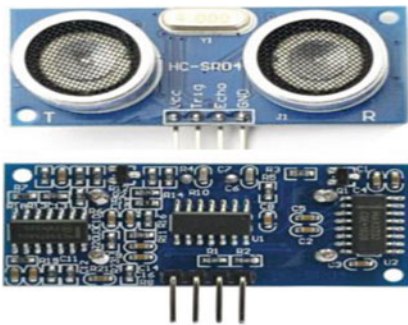
[13]



[14]

The NodeMCU ESP8266 Arduino board has Tensilica Xtensa 32 bit LX106 RISC microprocessor with 16 Digital input/output pins. It has 128 KB RAM and 4 MB of flash memory and it can operate with adjustable clock frequency between 80 and 160 MHz. The major advantage with this module is it has built in Wi-Fi/ Bluetooth and deep sleep operating features. Therefore, for making connection of mobile with device we need not any other sensors or device. The board have Micro USB jack and VIN pin (for external power supply) and it supports UART, SPI, and I2C interface.

Ultrasonic Sensor



[15]



[16]

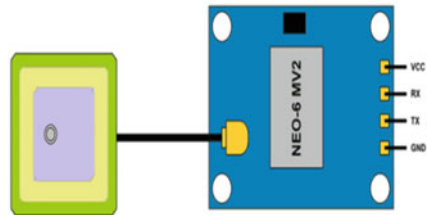
In this work, we have used Ultrasonic Sensor, which can efficiently measures the distance between any two vehicles or between any vehicle and other object. It is an electronic device that measures the distance by emitting sound waves. Ultrasonic waves travels faster than the speed human audible sound so it can quickly get the distance. This sensor has two main components the one is transmitter, which emits the sound using piezoelectric crystals, and the second is receiver, which encounters the returning sound after hitting the target. It converts reflected sound into an electrical

signal. In our work as the vehicles installed with this system have distance with other object less than 10 m the system will notify the driver.

GPS NEO 6M



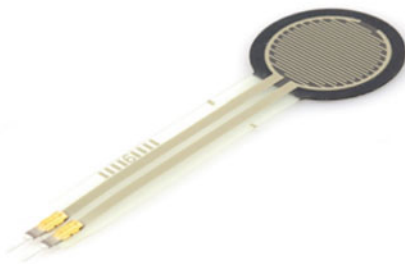
[17]



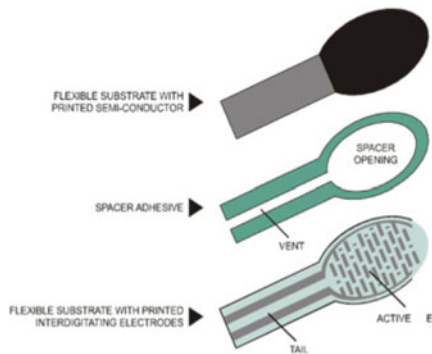
[18]

The NEO-6MV2 is a GPS module, which is used for finding exact location and can be used for navigation. In our work, the module simply checks the location of vehicle so that the notified persons can easily get the accident location. Its power, memory and overall compact design makes it more attractive and it is ideal for battery-operated device. This module has four pins.

Force Sensitive Resistor



[19]



[20]

FSR which stands for force sensitive register here in our work we used this to identify the collision and seriousness of accident. The FSR can measure with what force other body collide with vehicle and if the measurement is beyond the threshold then it will be considered as accident. We have developed just a model so for real life implementation we need this FSR all around the vehicles so that it can find collision from any collision. This resistor changes its resistive value depending on

how much it is pressed. It consist of two layers separated by a spaces as the more one presses the more of those active elements dots touch the semiconductor and makes the resistance go down. The sensing film consists of both electrically non-conducting and conducting particles suspended in matrix. When the force is applied, the sensing film causes particles to touch the conducting electrodes. As a result, the resistance of the film decreases. The resistance of the force sensitive resistor is depends on the amount of force applied.

B. Mobile Interface

The specified IoT system will be attached to user vehicle for detection of any mishaps but for the transmission of detection to notification for user specified contacts, we are using a mobile user-friendly interface. With the help of this interface user can add their emergency contacts so in case of accident those contacts will be notified immediately. User can add their friends and family member to emergency contacts, the major concept behind this is sometimes nearest hospital may not respond because of many reasons like hospital is closed, hospital is busy than usual, or hospital staff may ignore also but our friends and family members will not ignore and they can help by many ways. The specified IoT system will be attached to user vehicle for detection of any mishaps but for the transmission of detection to notification for user specified contacts, we are using a mobile user-friendly interface. For setting up the hardware system user need to install its mobile interface also (Driver software). After installing the software user need to turn on the hotspot by giving permission to app. As the hotspot, turns on the WIFI SSID and Password will be shard to Arduino and board will be connected to user phone. This mobile interface will be installed at user (who is driving vehicle) phone and if accidents happens the system will detect and connect to mobile interface and the app will generate the notification to user specified emergency contacts.

5 Result and Discussion

The complete module built using Arduino IDE where the coding of all sensors completed in C++ language. Further, the code uploaded to microcontroller for testing on output screen. We tested each sensor and all are working fine and giving correct output. As stated in research problem, this module can be very useful for saving someone's life by providing immediate help in road accidents.

An IoT device along with one mobile interface is considered in our work. The focus in this work is using Force sensitive resistor and Ultrasonic sensor. Up to my study, all the researchers had included Accelerometer sensor for detecting and measuring the seriousness of an accident. The accelerometer sensor works for vibration, seismic activity, inclination, speed, etc. Therefore, in these cases there is more chance of machine error. Instead of accelerometer sensor we have used here FSR which is not dependent on any above factors like vibration, speed, inclination, etc. the FSR will work only when there is collision of vehicles with any other vehicles or object.

Though just one FSR is not enough we need to wrap the vehicles all around with this resistor so that we can identify collision from any side and if the pressure is beyond the threshold device will consider it as accident. Surely, using FSR is minimizing the machine error or we can say the machine error is negligible and improves result that is more accurate.

The second focus is on Ultrasonic sensor, this sensor can surely minimize the accident rate also. This sensor is used to find the distance of target object using ultrasonic sound wave which faster than human audible sound. As per our work, we have implemented this with our module to notify the driver if there is less distance between the user vehicle and any other vehicle or object. For e.g., let the threshold distance be 10 m, so when there is any other vehicle or object between the range of 10 m from user's vehicles it the module will notify the user and user can easily decreasing the speed. This sensor can be very helpful in foggy climate where there is more chances of accidents.

Further, this work is designed for all types of vehicles and users can set their emergency contacts whosoever will be notified even if their phone will be in silent mode. With implementation of this we can identify the accident even if vehicle is static which we cannot do using accelerometer sensor.

6 Conclusion and Future Work

As the name of our work, Emergency Help for Road Accidents the major idea behind this work is to minimize the death rate of accidents. As we know many people dies in accident just because of unavailability of immediate medical help. With the implementation of this work, the developed module will identify the accident and the give alert to nearest hospital, family members and friends so that the victim can easy get emergency medical aid. FSR and Ultrasonic sensor are the main bone of this work, which will not only give the more precise result but can also minimize the accidents. With the help of mobile application, user's emergency contacts will be notified about accident even if their phone is in silent mode. The model is ideal for all types of vehicles and detection as well as measurement of seriousness does not depend upon mobile app or heartbeat sensor or accelerometer, so there is negligible chance of machine error. On the basis of above conclusion, we can say that this work is filling the gaps found in literature review.

For the future Enhancement with this work, we can add features like checking drowsiness of driver, prior notification of traffic light, speed breakers. We can also add features like when accidents happens there should be immediately automatic call forwarded to nearest hospital and police station.

References

1. Kattukkaran, N., George, A., Mithun Haridas, T.P.: Intelligent accident detection and alert system for emergency medical assistance. In: 2017 International Conference on Computer Communication and Informatics (ICCCI). IEEE, 2017
2. Sharma, S., Sebastian, S.: IoT based car accident detection and notification algorithm for general road accidents. *Int. J. Electri. Comput. Eng.* (2088-8708) **9**(5) (2019)
3. Habib, S., et al.: Design and development of IoT based accident detection and emergency response system. In: Proceedings of the 2020 5th International Conference on Cloud Computing and Internet of Things, 2020
4. Gokul Lal, K.V., et al.: Design and development of a smartphone-based application to save lives during accidents and emergencies. *Procedia Comput. Sci.* **167**, 2267–2275 (2020)
5. Vishnupriyan, J., et al.: Automated detection and rescue system for road accidents. *Int. J. Control Autom.* **13**(2), 564–568 (2020)
6. Leni, A.E.S.: Instance vehicle monitoring and tracking with internet of things using Arduino. *Int. J. Smart Sens. Intell. Syst.* **10**(5) (2017)
7. Reddy, M.S., Raghava Rao, K.: Fire accident detection and prevention monitoring system using wireless sensor network enabled android application. *Indian J. Sci. Technol.* **9**(17), 1–5 (2016)
8. Karmokar, P., et al.: A novel IoT based accident detection and rescue system. In: 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT). IEEE, 2020
9. Kumar, N., Acharya, D., Lohani, D.: An IoT-based vehicle accident detection and classification system using sensor fusion. *IEEE Internet Things J.* **8**(2), 869–880 (2020)
10. Uma, S., Eswari, R.: Accident prevention and safety assistance using IOT and machine learning. *J. Reliable Intell. Environ.* 1–25 (2021)
11. Patil, P.J., et al.: IOT protocol for accident spotting with Medical facility. *J. Artif. Intell.* **3**(02), 140–150 (2021)
12. https://www.robotrack.co.in/index.php?route=product/product&product_id=70&search=ESP8266+ESP-12E+Development+Board+WiFi+WLAN+Wireless+Module+CP2102+for+Nodemcu+for+ESP-12E+for+Arduino&description=true&gclid=CjwKCAjwoMSWBhAdEiwAVJ2ndnoTIqu1rG1Z78-5y5Cv7eh_wILmZZVeUNNEWgFH0xfPN_1Q_tyxtxocYo8QAvD_BwE
13. <https://components101.com/development-boards/nodemcu-esp8266-pinout-features-and-datasheet>
14. http://m.jselectronics.com.my/index.php?ws=showproducts&products_id=1929990
15. <https://components101.com/sensors/ultrasonic-sensor-working-pinout-datasheet>
16. <https://www.indiamart.com/proddetail/neo-6m-gps-module-with-eprom-23732513333.html>
17. <https://solicstores.in/product/gps-module-neo-6m-with-flight-control-eprom-mwc/>
18. <https://www.sparkfun.com/products/9375>
19. <https://learn.adafruit.com/force-sensitive-resistor-fsr?view=all>

Music Classification Based on Lyrics and Audio by Using Machine Learning



E. Arul, A. Punidha, S. Akash Kumar, R. Ragul, and V. B. Yuvaanesh

Abstract Music has the ability to cross regions and borders as it has no language. The line of the research problem is the development of music application that is capable of understanding the lyrics as well as audio and classifies it based on the emotions conveyed through the lyrics and audio. Once if we are able to classify the music genre based on lyrics and category based on audio, it would be easy to create playlists of that genre. This would ease the activity of hearing music and reduce the time for the listener in figuring out the genre of the music. This also reduces the search time of the listener when people are obsessed in listening to a particular genre of music. Our proposed system consists of modules, namely data preprocessing, classification, creation of playlist and recommendations. The lyrics module utilizes the ensemble vote classifier for classifying songs to different emotions based on the polarity value. The audio module utilizes support vector classifier for classifying songs into different genres based on the audio meta details extracted using librosa.

Keywords Google API · EnsembleVote classifier · Support vector classifier · Librosa · Data preprocessing · Playlist creation

E. Arul (✉) · S. Akash Kumar · R. Ragul · V. B. Yuvaanesh
Department of Information Technology, Coimbatore Institute of Technology, Coimbatore, Tamil Nadu, India
e-mail: arulcitit@gmail.com

S. Akash Kumar
e-mail: 1807007it@cit.edu.in

R. Ragul
e-mail: 1807042it@cit.edu.in

V. B. Yuvaanesh
e-mail: 1807060it@cit.edu.in

A. Punidha
Department of Computer Science and Engineering, Coimbatore Institute of Technology, Coimbatore, Tamil Nadu, India
e-mail: arulcitit@gmail.com; punitulip@gmail.com

1 Introduction

Everybody in this world loves music. Lyrics and the audio of the songs have greater information about the mood of the listener. Our research idea's main aim is to understand the lyric as well as audio and thereby classify it into different music categories and genres. The time taken for identifying the type of music is high. We propose a system which is capable of categorizing the music into different genres. An integrated system of both audio and lyric helps in creation of enhanced playlist and thereby enhances suggestions. We can thereby use the classification categories and thereby create playlists and eventually enhance the users experience in listening to music. The algorithms that we have used here are EnsembleVote Classifier for lyric module and support vector clustering for audio module.

In Sect. 3, the dataset that we have used is discussed. Our proposed classification methodology for both lyric and audio is discussed in Sect. 4. In Sect. 5, the results of the proposed system is discussed.

1.1 Lyric Module

Machine learning is subset of artificial intelligence (AI), is a popular and widely used technique, that has been applied in various domains including biology, medical, computer vision, speech recognition and others. The process of classification of music is divided into two parts, lyrics processing and audio processing. In starting of the module, the lyrics of music is extracted using Google API, and a csv file is created and extracted lyrics is stored in it. The polarity of the song is calculated using the TextBlob. After calculating the polarity, by utilizing the polarity value of the song, the emotion delivered is calculated using EnsembleVote Classifier. Ensemble classification models can be powerful machine learning tools capable of achieving excellent performance and generalizing well to new unseen datasets.

1.2 Audio Module

The dataset contains the audio metadata for a song and its corresponding category. The metadata of song is extracted using librosa module in Python. Librosa is a Python package for music and audio analysis. It provides the building blocks necessary to create music information retrieval system. These metadata are sent to the model to get classified. The model predicts the music category with the help of supplied metadata. Here the model is trained using support vector clustering algorithm.

2 Related Works

In [1], they assumed that the words of the lyrics are independent and had equal weight. The conditional probability of each category was found out of which the biggest one was the category of the sample. Naive Bayes classifier was used as it has high learning efficiency.

In [2], the main idea was to use text authorship identification to classify music. Each song has its own distinct text signatures that can distinguish to a large degree solely through reading. The algorithm used there is Naïve Bayes classifier.

In [3], the role of lyrics in music emotion recognition is discussed. The classification is done by using the Russels model over the manually annotated data.

In [4], they extract features from a lyric corpus to handle lyric specific expressions. It helps in estimating emotion from each phrase. The K-nearest neighbor algorithm with cosine summation is used for word summation. High degree of precision was obtained by using the concept of simple summation of vectors.

In [5], they have managed to decode the sound information that is very hard to specifically translate mood. The Naïve Bayes is used to classify the music. They classification was done into eighteen mood arrangements based upon the value of audio features and lyrics content. The consolidation of lyrics and audio features generally enhances exhibition on classes.

In [6], the main aim is to integrate the two methodologies of Chinese music classifications by sentiment dictionary-based method and the machine learning method. By doing so we could overcome the shortcomings of each method. In sentiment dictionary method, the text records are classified into corresponding feature vectors and its weight is calculated. This paper only uses lyric to classify the music.

In [7], they used the methodologies of Bag of words representations and black box models. The text features most related to specific emotions are chosen, and the algorithm that was used to predict the music emotion is decision tree classifier. In [8, 9], the general ways of how to do the sentimental analysis is discussed.

3 Dataset

We used two datasets one for lyrics and other for audio. The lyric module dataset consists of the polarity of the song along with the corresponding emotion category such as angry, sad, relaxed and depressed. The audio dataset consists of the meta details of the song such as rhythm and tempo and its corresponding genre. The lyrics dataset has 360 songs belonging to four different music emotions. The audio dataset consists of 1000 songs of 10 different genre (Figs. 1, 2 and 3).

	tempo	chroma_stft	rmse	spectral_centroid	spectral_bandwidth	rolloff	zero_crossing_rate	mfcc1	mfcc2	mfcc3
tempo	1.000000	0.030640	-0.043503	-0.014415	-0.039283	-0.020564	0.019935	0.011271	0.007200	-0.097491
chroma_stft	0.030640	1.000000	0.499577	0.615524	0.517482	0.597910	0.554735	0.659294	-0.598495	-0.054431
rmse	-0.043503	0.499577	1.000000	0.509984	0.529768	0.537718	0.332119	0.807260	-0.483765	0.139339
spectral_centroid	-0.014415	0.615524	0.509984	1.000000	0.904438	0.979633	0.874755	0.720831	-0.940171	0.187999
spectral_bandwidth	-0.039283	0.517482	0.529768	0.904438	1.000000	0.956194	0.612325	0.648373	-0.896660	0.494052
rolloff	-0.020564	0.597910	0.537718	0.979633	0.956194	1.000000	0.779809	0.721746	-0.934305	0.295590
zero_crossing_rate	0.019935	0.554735	0.332119	0.874755	0.612325	0.779809	1.000000	0.634139	-0.760127	-0.179883
mfcc1	0.011271	0.659294	0.807260	0.720831	0.648373	0.721746	0.634139	1.000000	-0.638911	-0.101078
mfcc2	0.007200	-0.598495	-0.483765	-0.940171	-0.896660	-0.934305	-0.760127	-0.638911	1.000000	-0.268841
mfcc3	-0.097491	-0.054431	0.139339	0.187999	0.494052	0.295590	-0.179883	-0.101078	-0.268841	1.000000
mfcc4	0.029352	0.264226	0.008695	-0.136839	-0.279636	-0.156114	-0.007672	0.108676	0.049740	-0.409602
mfcc5	-0.082223	-0.010116	0.108700	0.069032	0.302955	0.124208	-0.201045	-0.114746	-0.152303	0.601300
mfcc6	0.043894	0.416953	0.125782	0.017572	-0.066098	0.034629	0.069507	0.235657	-0.092587	-0.255766
mfcc7	-0.083500	0.067536	0.167592	0.215744	0.391085	0.249704	-0.030584	0.007181	-0.264085	0.519127
mfcc8	0.037268	0.509829	0.179000	0.134347	0.027736	0.137618	0.197509	0.331664	-0.192085	-0.317157
mfcc9	-0.073588	0.115721	0.216041	0.288783	0.409573	0.311073	0.075159	0.053417	-0.322575	0.438685
mfcc10	0.032041	0.489380	0.186025	0.192851	0.063432	0.184945	0.275797	0.340284	-0.216329	-0.347247

Fig. 1 Correlation of audio dataset

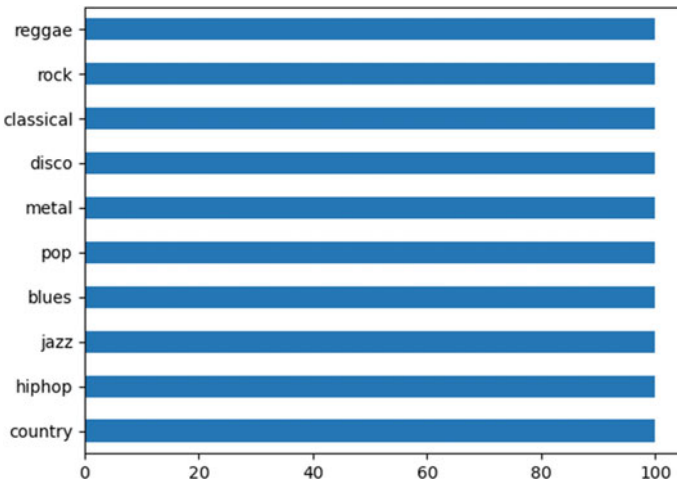


Fig. 2 Audio dataset

4 Methodology

The details of the proposed methodology consist of various steps. They are.

- EnsembleVote Classifier for lyrics extraction.
- Support vector classifier (SVC) for audio classification.

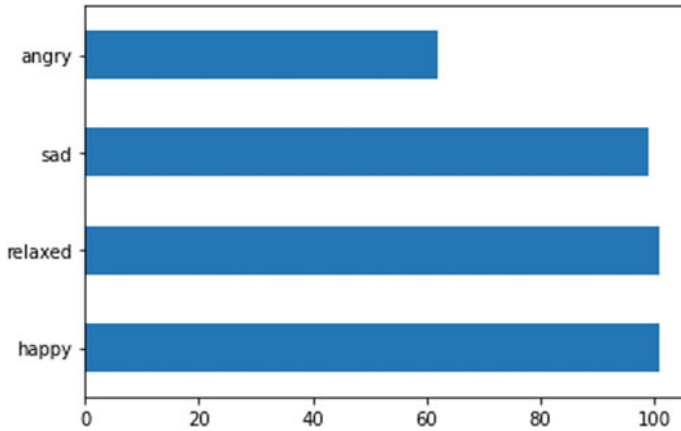


Fig. 3 Lyric dataset

4.1 Feature Vector Extraction

The feature vector for the lyric module is the polarity. The polarity is calculated from the lyrics of the song that we are interested in. The stopwords in the lyrics are removed and for the remaining words in the lyric, polarity is calculated using TextBlob. The feature vector for the audio module is the audio meta details such as rhythm and tempo. It is extracted using the librosa module that fetches the audio meta details for the song that we are interested in.

4.2 Training and Testing

The EnsembleVote Classifier utilizes the four algorithms, namely logistic regression, K-nearest neighbors, SVM and decision tree. 80% of the data is used for training and 20% is used for testing for both the lyrics and audio module. The testing parameters that we used are r2 score and accuracy score.

4.3 EnsembleVote Classifier for Lyrics Extraction

The prerequisite of this module is Engine_ID and API_KEY. First, we need to create a Google custom search engine. After creating the custom search engine by giving certain searching Web sites such as www.genius.com, the API key of that search engine should be generated to integrate with our code. By importing the SongLyrics package, we would be able to search for the lyrics of the given song name.

Once the lyrics is extracted from the `getLyrics()`, the data is subjected to preprocessing. Initially, the entire data is converted into lower case, then sentence tokenization and word tokenization are done. After that the presence of any stop words is identified, if identified then it is removed. After that, the rest of the words undergoes stemming and lemmatization. Once all these steps are done, the data is ready to find the sentimental value.

The dataset itself contains the lyrics of the songs. In building of the model, the first step is the calculation of the polarity of the songs in the dataset. Once the polarity is identified, then by utilizing the classification, algorithm model is built. Here, EnsembleVote Classifier is used to create the model by utilizing the polarity value of the song along with the emotion delivered through the song. The song file is given as an input in the UI, from that the lyric of the song is extracted and the data preprocessing is done and the polarity is calculated. This polarity value is passed to the model, and the result is obtained. Once the result is obtained, the song is moved to a folder of its emotion which is later displayed as a playlist.

4.4 Support Vector Classifier(SVC) for Audio Classification

Here the model is trained using support vector clustering algorithm. Once is audio file is selected, the audio preprocessing of that file to get the audio metadata is done. Here the metadata is obtained using the `librosa` module in Python. These metadata are sent to the model to get classified. The model predicts the music category with the help of supplied metadata.

The input file is given through the form tag's file input. Once the file is uploaded, the emotion of the song is identified using the above-mentioned steps. Then to make listening even simpler, a system voice is generated to exhibit the emotion of the song. Then another voice will be prompted asking whether you want to play that song. If one replies positively, then it is redirected to another page where the music controls would be present.

When the polarity of the song is calculated and the emotion is categorized, the uploaded music file location would be found and then it would be moved to a folder containing the categorized emotion. To have a better understanding a word, cloud is created representing the most commonly occurring words in the song.

5 Results and Discussion

The idea of incorporating both the lyrics and audio classification to give an enhanced music listening experience to the users is achieved by making use of machine learning [10]. The EnsembleVote Classifier algorithm's efficiency is finding the lyrical emotions and the support vector classifier algorithm's efficiency in finding

out the music genre is highly commendable. The result of our entire proposed work is explained below.

Figure 6 depicts the system architecture of our project. In the main page of our local server deployed project, you will be asked to select the song which you are willing to add to the playlist. Once selected, the name of the song is extracted from the file that we select. For that song, the lyrics would be extracted dynamically [11]. Once lyrics is extracted, then it is subjected to data preprocessing and then the preprocessed data’s polarity value is calculated using TextBlob. This polarity value is passed to the model to find out the music emotion that is conveyed. Then the audio of the song is processed using librosa module, and the extracted metadata is passed to the model. It would predict the music genre of the audio, and the song would be moved to their appropriate locations.

Figure 4 depicts how the data is manipulated in different stages. The song name is extracted from the file that we select. Using the name, the lyrics of the song is extracted by using the Google API. Then from the extracted lyrics, stopwords are removed and then the sentences are tokenized. Then these tokenized sentences are further reduced to tokens of single words. The tokens are lemmatized which means the correction of the grammar is done and then subjected to stemming which means reduction it to the root form of the word. Then the polarities of the remaining words are calculated and the music emotion classification is done by passing it to the model [12]. Similarly, the audio features are extracted using librosa module, and the extracted metadata is passed to the model. It would be used to predict the music genre of the audio that has been selected.

Figure 5 depicts the project flow diagram. In the main page of the application, the song that you need to add to the playlist is selected. Then two modules will start to run one after the other. The first module is the lyric module in which the lyric of the song that has been selected would be extracted and the lyrics would be subjected to data preprocessing, and the polarity of the song would be calculated. And based on the value obtained, the song’s emotion is identified [13]. Then based on the audio, the audio features are extracted using the librosa module. The songs genre is identified by passing the metadata to the model. Then if we wish to listen to the song, then we will be redirected to a page where we will be able to listen to the music along with the lyrics displayed below. The other songs in the playlist will be shown as recommendations below the lyrics.

Lyric Model Accuracy

Algorithm	Accuracy (%)
K-nearest neighbors	36
Logistic regression	35
Naïve Bayes	38
SVM	45
EnsembleVote classifier	65

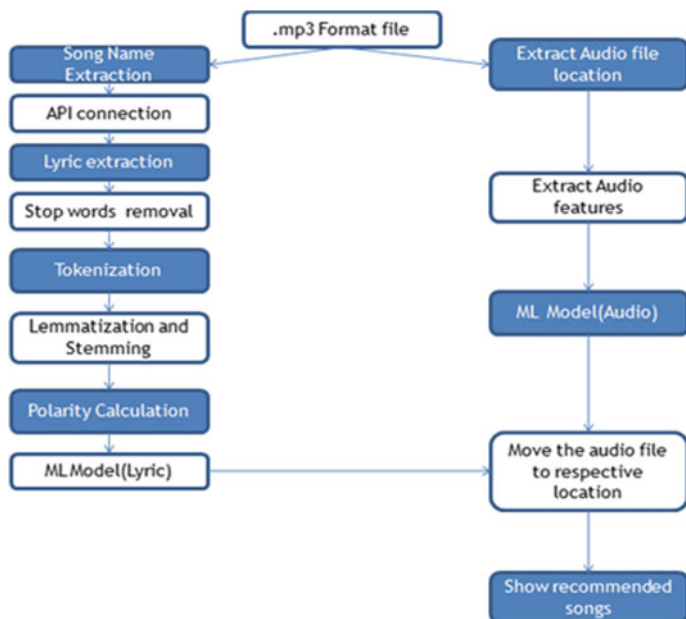


Fig. 4 Functional diagram

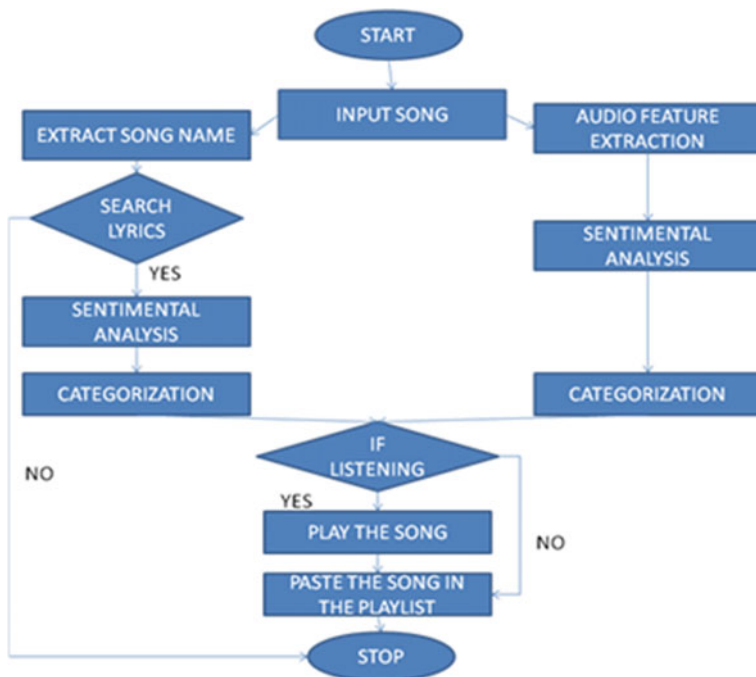


Fig. 5 Flowchart

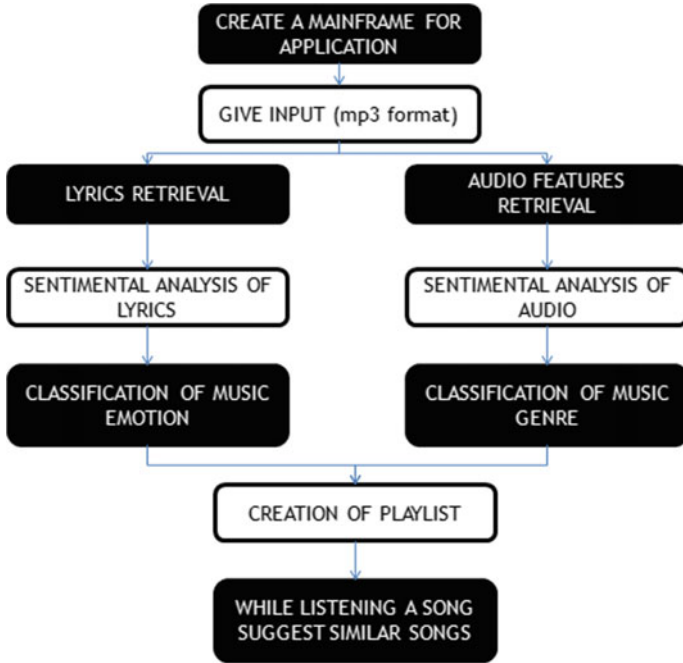


Fig. 6 Overall system architecture

Audio Model Accuracy

Algorithm	Accuracy
Support vector classifier	72%

6 Conclusion and Future Enhancements

The properties that makes our research distinct is that our model is capable of classifying four emotion categories. By carefully selecting out the feature vectors, we were able to classify the songs efficiently. The important audio features were selected using backward elimination, and hence, the model is simple and the training and testing time is reduced. The time taken for the model to classify the given song in real time is decreased greatly as only key features were utilized to categorize the song. The lyrics need not be given to the model as it is dynamically extracted during run time.

Our proposal has the ability to classify the audio file based on both lyric and audio and categorize it into different music category and genre. This would be very useful for all music lovers as there would no need of wasting time in searching similar music. It would be highly useful for the visually challenged people as well because they can't search song. The recommendations algorithms can be developed and strengthened by knowing what kind of the playlist the user often listens to. In the future, the number of classification categories for the music emotion detection can be increased. Other algorithms efficiency would also be tested for the lyric and audio module. Certain other features audio features would also be considered in the development of the model so that the accuracy of the model can be further enhanced.

References

1. An, Y., Sun, S., Wang, S.: Naive Bayes classifiers for music emotion classification based on lyrics. In: 2017 IEEE 4th International Conference on Computer and Communications (ICCC) (2017)
2. Bužić, D., Dobša, J.: Lyrics classification using Naive Bayes. In: 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO) (2018)
3. Malheiro, R., Panda, R., Gomes, P., Paiva, R.P.: Emotionally-relevant features for classification and regression of music. *IEEE Trans. Affect. Comput.* **12**(2), 173–184, 1028 (2019)
4. Matsumoto, K., Sasayama, M.: Lyric emotion estimation using word embedding learned from lyric corpus. In: 2018 IEEE 4th International Conference on Computer and Communications (ICCC) (2019)
5. Kashyap, N., Choudhury, T., Chaudhary, D.K., Lal, R.: Mood based classification of music by analyzing lyrical data using text mining. In: 2016 International Conference on Micro-Electronics and Telecommunication Engineering (ICMETE) (2016)
6. Wu, X., Cao, Y.: Research on song sentiment binary classification based on Chinese lyrics. In: 2018 IEEE/ACIS 17th International Conference on Computer and Information Science (ICIS) (2018)
7. Yang, D., Lee, W.-S.: Music emotion identification from lyrics. In: 2009 11th IEEE International Symposium on Multimedia (2009)
8. Surya, P.P.M., Subbulakshmi, B.: Sentimental analysis using naive Bayes Classifier. In: 2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN) (2019)
9. Pankaj, P.P., Muskan, N.S.: Sentiment analysis on customer feedback data: amazon product reviews. In: 2019 International conference on machine learning, big data, cloud and parallel computing (COMITCon) (2019)
10. Ahuja, M., Sangal, A.L.: Opinion mining and classification of music lyrics using supervised learning algorithms 2018. In: First International Conference on Secure Cyber Computing and Communication (ICSCCC), vol. 12 pp. 223–225 (2019)
11. Sharma, H., Gupta, S., Sharma, Y., Purwar, A.: A new model for emotion prediction in music. In: 2020 6th International Conference on Signal Processing and Communication (ICSC), vol. 12, no. 2, pp. 173–184 (2020)
12. Xiong, Y., Su, F., Wang, Q.: Automatic music mood classification by learning cross-media relevance between audio and lyrics. In: 2017 IEEE International Conference on Multimedia and Expo (ICME) (2017)
13. Chauhan, S., Chauhan, P.: Music mood classification based on lyrical analysis of Hindi songs. In: 2016 International Conference on Information Technology (InCITe)—The Next Generation IT Summit on the Theme—Internet of Things: Connect your Worlds (2016)

Multiobjective Optimization for Hospital Nurses Scheduling Problem Using Binary Goal Programming



Salman Khalil and Umar Muhammad Modibbo

Abstract Hospital nurse scheduling is a very challenging task before the management. It is several nurses' assignment to many wards or units with diverse demands and specialities. This paper studies the nurse scheduling problem and considers nurses' preferences such as rotation, length of stay, weekly cycle and daily shifts. There are four different conflicting objective functions. Two are related to nurses shifting on the duty roster. In contrast, the other two objectives are related to nurses weekends and the maximum possible working days within the scheduling cycle. The problem is modeled using the goal programming approach and solution obtained via a zero-one technique. The complex and soft constraints are satisfied optimally. The program extended to provide a one-year complete schedule, and the promising solution will help the management solve the lingering problem of scheduling in hospitals.

Keywords Binary goal programming · Nurse scheduling · Soft constraints · Hard constraints · Nurses preferences

1 Introduction

Scheduling of nurses in any healthcare facility is a complex optimization problem that requires the incorporation of many objective from different sources that are conflicting in nature. The nurses may have their preferences within which they want to exercise their duties such as time of the day, and the head of various units may

S. Khalil

Department of Community Medicine, Jawaharlal Nehru Medical College, A. M. U., Aligarh 202002, India

U. M. Modibbo (✉)

Department of Statistics and Operations Research, Aligarh Muslim University, Aligarh 202002, India

Department of Statistics and Operations Research, Modibbo Adama University, P.M.B. 2076, Yola, Nigeria

e-mail: umarmodibbo@mautech.edu.ng

have their own priority of assigning the nurses to different wards depending on their experience, qualifications and nature of bottleneck in those wards. The management authorities of hospitals or healthcare centers on the other hand have their own policies, objectives and requirement as to how the nurses should operate, for example, certain number of nurses may be required in certain unit of the system at certain period of time, a particular expertise and experience among the nurses may be required at a particular unit of the system. These are all conflicting interest from different parties involved each of which may numerous constraints that need to be satisfied, as a result of these, the problem is viewed as a complex multiobjective decision-making problem. Apart from these, nurses scheduling always have element of uncertainties which also need to be considered while drafting the duty roster by the heads. For example, a particular nurse on duty may felt seek, or pregnant nurse on duty may deliver which in any case hinder them from coming to discharge their duties as scheduled, because of these uncertainties, the head may not have a precise aspiration levels about the situation created by the system's uncertainty. Many researchers are working in this domain of real-life situation using different approaches based on the problem they perceived in the study area. The overall goal of any hospital or healthcare center is the delivery of services to humanity in an effective, efficient and qualitative manner; hence, the judicious utilization of human resources (nurses) cannot be over-emphasized.

The expectation of any patient is to be satisfactorily serviced. Therefore, health-care services should not be compromise, it suppose to be daily (24 hours per day), weekly (7 days per week) and yearly (365 days per year). Nurses play a vital role in ascertain this routine services because of their saddled responsibility of managing the patients throughout their stay in the system. Looking at these, nurse scheduling can be viewed as NP-hard problem as will be reviewed in the literature section. Decision-makers are faced with numerous constraints while determining the optimal nurses scheduling, these various constraints can be broadly categorized into soft and hard constraints. The hard constraints are those that has do with the policies of the system such as minimum staff level requirement in a unit, maximum shift for a nurse per day, maximum or minimum consecutive days for a particular shift, minimum off days, range of days per schedule per cycle, etc., while the soft constraints include among others avoidance to have a consecutive shifts by a nurse in a day, not to worked in the morning period (shift) then followed by evening and or night period (shift) the next day, having an unbias number of shift by all nurses within the scheduling period, etc.

2 Literature Review

Optimal allocation of personnel in healthcare systems has been studied by many researchers using different approach. Warner [1] used mathematical programming concepts to scheduled nursing personnel in accordance with their preferences concerning length of stay in duty, shifting pattern and off duty days. The study employed multi-choice programming with nurses preferences as the objective functions and minimum number of personnel as constraints with varying degree of expertise, the

model was implemented in different units of the hospitals. The multiobjective problem of preference nurse scheduling was solved by Bard and Purnomo [2], using a column generation approach which combines both integer programming and heuristics models. These approach were tested with 100 nurses from different hospitals in US, and the outcome shows that solutions of a desired-quality are possible to be attained. Wang et al. [3] use binary goal programming (GP) in solving outpatient “nurse scheduling problem.” The solution assigned optimal number of full-time nurses during the weekdays and a sensitivity analysis was also carried out on the result. Yelmaz [4] developed a mathematical programming model to optimize the idle waiting time for nurses during the planning frame. The model was illustrated with numerical example, and a sensitivity analysis was carried out to identify the efficiency of the model. Integer linear programming (ILP) with local search approach was used as hybrid method for scheduling of nursing personnel in a hospital as documented in Valouxis and Housos [5]. Weighted and sequential approach of multiobjective programming problem with binary variables was considered by Viviane et al. [6], optimal result for the flying squad nurses was obtained using the mixed solution approach and was illustrated with numeric example. A 0–1 linear GP used by Azaiez and Al Sharif [7], to develop a computerized scheduling model for nurses and their preferences. The model was experimented with a six-month scheduling period which indicated a performance improvement over the manual process.

Topaloglu and Selim [8] applied a fuzzy set theory on the nurse scheduling and developed a multiobjective integer programming model with three different fuzzy set to address the uncertainty in the scheduling problem. The models were validated with a real-life application, and sensitivity analysis was also carried out to reveal the efficiency of the model. Cetin and Sarucan [9] proposed a multiple objective integer program for modeling “nurse scheduling problem,” and they applied a binary fuzzy programming developed by Chang [10] on their model which was validated with a real-life hospital data to demonstrate the efficiency of the model. Sufahani et al. [11] use ILP to model a scheduling of an operating theater as a mathematical model with a strategy of an open scheduling, a numeric example was used to validate the model and GLPK/AMPL software was employed to obtain the optimal schedule for 5 working days (Monday–Friday). Modibbo et al. [12] studied supplier selection problem in pharmaceutical company using FuzzyTopsis—a multicriteria decision-making tool.

A heuristic based and hybrid genetic algorithm approach were used by Fei et al. [13] to plan and schedule a theater operating rooms in a hospital. The planning aspect was described as integer programming problem, while the scheduling part was viewed as a hybrid flow-shop problem. Operating rooms utilization were maximize and cost of overtime & idle unexpected time between surgical cases in the operating theater rooms were minimized. The results show an improvement over several actual schedules in the Belgian University hospital. Mixed-integer programming was used to optimize number of nursing personnel with different categories of skills in various wards and shifts in a hospital by Warner and Prawda [14]. They also developed a model which minimizes the nursing shortage cost and total personnel capacity satisfaction. Warner [15] classified healthcare investigations into staffing skills, allocation capacity, scheduling cycles and nurses assignment.

3 Nurse Scheduling Model

This section provides and defines the decision variables, the indices, the objectives and constraints used in model building. The constraints are further sub-divided into soft and hard constraints.

3.1 Indices

m : total number of nurses in the unit ($m = 18$)

n : number of days in the schedule ($n = 21$)

i : days index, $i = 1, 2, \dots, n$.

k : nurses index, $k = 1, 2, \dots, m$

P_i : number of personnel for morning shifts, $i = 1, 2, \dots, n$.

Q_i : number of personnel for evening shifts, $i = 1, 2, \dots, n$.

R_i : number of personnel for night shifts, $i = 1, 2, \dots, n$.

3.2 Decision Variables

These are standard variables formulation across assignment models

$$W_{i,k} =: \begin{cases} 1, & \text{if nurse } k \text{ is } \mathbf{off\ duty} \text{ on day } i, \quad i = 1, 2, 3, \dots, n \\ 0, & \text{Otherwise} \end{cases} \quad (1)$$

$$X_{i,k} =: \begin{cases} 1, & \text{if nurse } k \text{ is on } \mathbf{morning\ shift} \text{ on day } i, \quad i = 1, 2, 3, \dots, n \\ 0, & \text{Otherwise} \end{cases} \quad (2)$$

$$Y_{i,k} =: \begin{cases} 1, & \text{if nurse } k \text{ is on } \mathbf{evening\ shift} \text{ on day } i, \quad i = 1, 2, 3, \dots, n \\ 0, & \text{Otherwise} \end{cases} \quad (3)$$

$$Z_{i,k} =: \begin{cases} 1, & \text{if nurse } k \text{ is on } \mathbf{night\ shift} \text{ on day } i, \quad i = 1, 2, 3, \dots, n \\ 0, & \text{Otherwise} \end{cases} \quad (4)$$

3.3 Hard Constraints

Constraints set 1: Minimum staff level requirement must be satisfied

$$\sum_{k=1}^m X_{i,k} \geq P_i, \quad i = 1, 2, \dots, n$$

$$\sum_{k=1}^m Y_{i,k} \geq Q_i, \quad i = 1, 2, \dots, n \quad (5)$$

$$\sum_{k=1}^m Z_{i,k} \geq R_i, \quad i = 1, 2, \dots, n$$

Constraints set 2: Each nurse works only one shift a day

$$W_{i,k} + X_{i,k} + Y_{i,k} + Z_{i,k} = 1, \quad \forall i = 1, 2, \dots, n. \quad \forall k = 1, 2, \dots, m. \quad (6)$$

Constraints set 3: Avoid any isolated days of Off-ON-Off

$$W_{i,k} + X_{i+1,k} + Y_{i+1,k} + Z_{i,k} + W_{i+2,k} \leq 2, \quad i = 1, 2, \dots, n \quad k = 1, 2, \dots, m. \quad (7)$$

Constraints set 4: Each nurse works 3 consecutive days of night shift and followed by 3 days off. These implies that each nurse will be assign to their shifts and off days as follows:

$$\sum_{i=1}^3 Z_{i,k} + \sum_{i=4}^6 W_{i,k} + \sum_{i=19}^{21} Z_{i,k} = 9, \quad k = 1, 7, 13.$$

$$\sum_{i=1}^3 W_{i,k} + \sum_{i=16}^{18} Z_{i,k} + \sum_{i=19}^{21} W_{i,k} = 9, \quad k = 2, 8, 14,$$

$$\sum_{i=13}^{15} Z_{i,k} + \sum_{i=16}^{18} W_{i,k} = 6, \quad k = 3, 9, 15,$$

$$\sum_{i=10}^{12} Z_{i,k} + \sum_{i=13}^{15} W_{i,k} = 6, \quad k = 4, 10, 16;$$

$$\sum_{i=7}^9 Z_{i,k} + \sum_{i=10}^{12} W_{i,k} = 6, \quad k = 5, 11, 17$$

$$\sum_{i=4}^6 Z_{i,k} + \sum_{i=7}^9 W_{i,k} = 6, \quad k = 6, 12, 18.$$

Constrain 5: Each nurse must work between 12 and 14 days per schedule.

$$\begin{aligned} \sum_{i=1}^m (X_{i,k} + Y_{i,k} + Z_{i,k}) &\geq 12, \quad k = 1, 2, \dots, m \\ \sum_{i=1}^m (X_{i,k} + Y_{i,k} + Z_{i,k}) &\leq 14, \quad k = 1, 2, \dots, m \\ \sum_{i=n-v}^n W_{i,k} + \sum_{i=1}^{6-v} W_{i,k+1} &\geq 1, \quad v = 0, 1, 2, \dots, 5. \quad k = 1, 2, \dots, m-1, \quad n = 1, 2, \dots, 7, \\ \sum_{i=n-v}^n W_{i,m} + \sum_{i=1}^{6-v} W_{i,1} &\geq 1, \quad v = 0, 1, 2, \dots, 5, \quad n = 1, 2, \dots, 7. \end{aligned} \tag{9}$$

Constraints set 6: Each nurse works not more than 6 consecutive days

$$\begin{aligned} W_{i,k} + W_{i+1,k} + W_{i+2,k} + W_{i+3,k} + W_{i+4,k} + W_{i+5,k} + W_{i+6,k} &\geq 1, \\ i = 1, 2, \dots, n-4. \quad k = 1, 2, \dots, m. \end{aligned} \tag{10}$$

Constraints set 7: Evening shift constitutes atleast 25% workload

$$\sum_{i=1}^n Y_{i,k} \geq 3, \quad k = 1, 2, \dots, m \tag{11}$$

Constraints set 8: Morning shift constitutes atleast 30% of the total workload

$$\sum_{i=1}^n X_{i,k} \geq 4, \quad k = 1, 2, \dots, m \tag{12}$$

Constraints set 9: Night shift constitutes atleast 25% workload

$$\sum_{i=1}^n Z_{i,k} \leq 3, \quad k = 1, 2, \dots, m \tag{13}$$

3.4 Soft Constraints

Constraint set 10: Avoid working in evening shift followed by morning/night shift the next day

$$\begin{aligned} Y_{i,k} + X_{i+1,k} + Z_{i+1,k} &\leq 1, \quad i = 1, 2, \dots, n-1, \quad k = 1, 2, \dots, m \\ Y_{n,k} + X_{1,k+1} + Z_{1,k+1} &\leq 1, \quad k = 1, 2, \dots, m-1 \\ Y_{1,1} + X_{1,1} + Z_{1,1} &\leq 1, \end{aligned} \tag{14}$$

Constraint set 11: Avoid working in morning shift followed by evening/night shift the next day

$$\begin{aligned} X_{i,k} + Y_{i+1,k} + Z_{i+1,k} &\leq 1, \quad i = 1, 2, \dots, n-1, \quad k = 1, 2, \dots, m \\ X_{n,k} + Y_{1,k+1} + Z_{1,k+1} &\leq 1, \quad k = 1, 2, \dots, m-1 \\ X_{n,m} + Y_{1,1} + Z_{1,1} &\leq 1, \end{aligned} \quad (15)$$

Constraint set 12: Each nurse has atleast one weekend Off

$$W_{7,k} + W_{14,k} + W_{21,k} \geq 1, \quad k = 1, 2, \dots, m \quad (16)$$

Constrain 13: All nurses have the same amount of total workload.

$$\sum_{i=1}^n (X_{i,k} + Y_{i,k} + Z_{i,k}) \geq 13, \quad k = 1, 2, \dots, m \quad (17)$$

3.5 Goals of the Problem

There are four different goals under this section which the management wish to satisfy simultaneously. The goals can be seeing as conflicting in nature. However, our objective is minimize the deviations from under or over achieving of these goals. The first and second goals are related to shifting od duty roster, whereas the third and fourth goals are related to nurses weekend and maximum possible working days within the scheduling cycle respectively.

Goal 1: This objective is to avoid assigning a nurse to have an evening shift followed by a morning shift or night shift the following day. Here, we set D^- and D^+ as the amount of deviation from the goal for nurses. That is:

$$\begin{aligned} Y_{i,k} + X_{i+1,k} + Z_{i+1,k} + D_{n,k}^- - D_{n,k}^+ &\leq 1, \quad i = 1, 2, \dots, n-1, \quad k = 1, 2, \dots, m \\ Y_{n,k} + X_{1,k+1} + Z_{1,k+1} + D_{n,k}^- - D_{n,k}^+ &\leq 1, \quad k = 1, 2, \dots, m-1 \\ Y_{n,m} + X_{1,1} + Z_{1,1} + D_{n,m}^- - D_{n,m}^+ &\leq 1 \end{aligned} \quad (18)$$

Goal 2: This objective will try to avoid assigning a nurse to have a morning shift followed by an evening or night shift the following day. Here, we set D^- and D^+ as the amount of deviation from this goal which are to minimize. That is:

$$\begin{aligned}
 X_{i,k} + Y_{i+1,k} + Z_{i+1,k} + D_{i,k}^- - D_{i,k}^+ &\leq 1, \quad i = 1, 2, \dots, n - 1, \quad k = 1, 2, \dots, m \\
 X_{n,k} + Y_{1,k+1} + Z_{1,k+1} + D_{n,k}^- - D_{n,k}^+ &\leq 1, \quad k = 1, 2, \dots, m - 1 \\
 X_{n,m} + Y_{1,1} + Z_{1,1} + D_{n,k}^- - D_{n,k}^+ &\leq 1
 \end{aligned}
 \tag{19}$$

Goal3: This objective will ensure that each nurse have atleast one day Off weekend in the 3 week schedule. That is:

$$W_{7,k} + W_{14,k} + W_{21,k} + D_k^- - D_k^+ \geq 1 \quad k = 1, 2, \dots, m \tag{20}$$

Goal 4: Here, the goals will seek to adhere to the fact that all nurses are schedule to have 13 days as possible in 3 week schedule, i.e., to say all the nurses have the same amount of workload during the cyclic period of schedule.

$$\sum_{i=1}^m (X_{i,k} + Y_{i,k} + Z_{i,k}) + D_k^- - D_k^+ = 13, \quad k = 1, 2, \dots, m \tag{21}$$

Therefore, the overall goal will be to minimize the following deviations as

Minimize Z

$$= \left\{ \sum \sum (D_{i,k}^- + D_{i,k}^+) + \sum (D_{n,k}^- + D_{n,k}^+) + \sum (D_k^- + D_k^+) + (D_{n,m}^- + D_{n,m}^+) \right\} \tag{22}$$

$$\text{Subject to: } \begin{cases} i. \text{ Hard constraints equations 5–13} \\ ii. \text{ Soft constraints equations 14–17} \\ X, Y, Z \ \& \ W, \text{ assume the values } 0 \text{ or } 1 \\ D_{i,k}^-, D_{i,k}^+, D_{n,k}^-, D_{n,k}^+, D_{n,m}^-, D_{n,m}^+, D_k^-, D_k^+ \geq 0 \end{cases} \tag{23}$$

4 Complete Mathematical Model of the Problem

$$\begin{aligned}
 \text{Minimize } Z = & (D_{1,1}^- + D_{1,1}^+ + D_{1,2}^- + D_{1,2}^+ + \dots, + \\
 & D_{21,18}^- + D_{21,18}^+) + (D_{21,1}^- + D_{21,1}^+ + \dots + \\
 & D_{21,18}^- + D_{21,18}^+) + (D_{21,18}^- + D_{21,18}^+) + (D_1^- + D_1^+ \\
 & + D_2^- + D_2^+ + \dots + D_{18}^- + D_{18}^+)
 \end{aligned}
 \tag{24}$$

$$\text{Subject to: } \left\{ \begin{array}{l} \sum_{j=1}^{18} X_{ij} \geq 4, \quad \forall i = 1, 2, \dots, 21 \\ \sum_{j=1}^{18} Y_{ij} \geq 4, \quad \forall i = 1, 2, \dots, 21 \\ \sum_{j=1}^{18} Z_{i,j} \geq 3, \quad \forall i = 1, 2, \dots, 21 \\ \sum_{i=j=1} (X_{i,j} + Y_{i,j} + Z_{i,j} + W_{i,j}) = 1 \\ \sum_{i=1, j=2} (X_{i,j} + Y_{i,j} + Z_{i,j} + W_{i,j}) = 1 \\ \cdot \\ \cdot \\ \cdot \\ \sum_{i=21, j=18} (X_{i,j} + Y_{i,j} + Z_{i,j} + W_{i,j}) = 1 \\ W_{1,1} + X_{2,1} + Y_{2,1} + Z_{2,1} + W_{3,1} \leq 2 \\ W_{1,2} + X_{2,2} + Y_{2,2} + Z_{2,2} + W_{3,2} \leq 2 \\ \cdot \\ \cdot \\ \cdot \\ W_{19,18} + X_{20,18} + Y_{20,18} + Z_{20,18} + W_{21,18} \leq 2 \end{array} \right. \quad (25)$$

$$\sum_{i=1}^3 Z_{i,1} + \sum_{i=4}^6 W_{i,1} + \sum_{i=19}^{21} Z_{i,1} + \sum_{i=1}^{21} Z_{i,7} + \sum_{i=1}^{21} Z_{i,13} = 9$$

$$\sum_{i=13}^{15} Z_{i,3} + \sum_{i=16}^{18} W_{i,3} + \sum_{i=13}^{15} Z_{i,9} + \sum_{i=16}^{18} W_{i,9} + \sum_{i=13}^{15} Z_{i,15} + \sum_{i=16}^{18} W_{i,15} = 6$$

$$\sum_{i=10}^{12} Z_{i,4} + \sum_{i=13}^{15} W_{i,4} + \sum_{i=10}^{12} Z_{i,10} + \sum_{i=13}^{15} W_{i,10} + \sum_{i=10}^{12} Z_{i,16} + \sum_{i=13}^{15} W_{i,16} = 6$$

$$\sum_{i=7}^9 Z_{i,5} + \sum_{i=10}^{12} W_{i,5} + \sum_{i=7}^9 Z_{i,11} + \sum_{i=10}^{12} W_{i,11} + \sum_{i=7}^9 Z_{i,17} + \sum_{i=10}^{12} W_{i,17} = 6$$

$$\sum_{i=4}^6 Z_{i,6} + \sum_{i=7}^9 W_{i,6} + \sum_{i=4}^6 Z_{i,18} + \sum_{i=7}^9 W_{i,18} = 6$$

$$\sum_{i=j=1} (X_{i,j} + Y_{i,j} + Z_{i,j}) + \sum_{i=1,j=2} (X_{i,j} + Y_{i,j} + Z_{i,j}) + \dots + \sum_{i=21,j=18} (X_{i,j} + Y_{i,j} + Z_{i,j}) \geq 12$$

$$\sum_{i=j=1} (X_{i,j} + Y_{i,j} + Z_{i,j}) + \sum_{i=1,j=2} (X_{i,j} + Y_{i,j} + Z_{i,j}) + \dots + \sum_{i=21,j=18} (X_{i,j} + Y_{i,j} + Z_{i,j}) \leq 14$$

$$\sum_{j=1}^{17} W_{21,j} + \sum_{j=1}^{18} W_{20,j} + \sum_{j=1}^{18} W_{19,j} + \dots + \sum_{j=1}^{18} W_{1,j} \geq 1$$

$$\sum_{i=21}^{i=5} W_{i,18} + \sum_{i=1}^{21} W_{i,1} \geq 1$$

$$\sum_{j=1}^{18} Y_{1,j} \geq 3$$

$$\sum_{j=1}^{18} Y_{2,j} \geq 3$$

⋮

$$\sum_{j=1}^{18} Y_{21,j} \geq 3$$

$$\sum_{j=1}^{18} X_{1,j} + \sum_{j=1}^{18} X_{2,j} + \dots + \sum_{j=1}^{18} X_{21,j} \geq 4$$

$$Y_{1,1} + X_{2,1} + Z_{2,1} + D_{1,1}^- - D_{1,1}^+ + \dots + Y_{20,18} + X_{21,18} + Z_{21,18} + D_{20,18}^- - D_{20,18}^+ \leq 1$$

$$Y_{21,1} + X_{1,2} + Z_{1,2} + D_{2,1}^- - D_{2,1}^+ + \dots + Y_{21,18} + X_{1,18} + Z_{1,18} + D_{20,18}^- - D_{20,18}^+ \leq 1$$

$$Y_{21,18} + X_{1,1} + Z_{1,1} + D_{21,18}^- - D_{21,18}^+ \leq 1$$

$$X_{1,1} + Y_{2,1} + Z_{2,1} + D_{1,1}^- - D_{1,1}^+ + \dots + X_{20,18} + Y_{21,18} + Z_{21,18} + D_{20,18}^- - D_{20,18}^+ \leq 1$$

$$X_{21,1} + Y_{1,2} + Z_{1,2} + D_{2,1}^- - D_{2,1}^+ + \dots + X_{21,17} + Y_{1,18} + Z_{1,18} + D_{21,17}^- - D_{21,17}^+ \leq 1$$

$$X_{21,18} + Y_{1,1} + Z_{1,1} + D_{21,18}^- - D_{21,18}^+ \leq 1$$

$$W_{7,1} + W_{14,1} + W_{21,1} + D_1^- - D_1^+ + \dots + W_{7,18} + W_{14,18} + W_{21,18} + D_{18}^- - D_{18}^+ \geq 1$$

$$X_{1,1} + Y_{1,1} + Z_{1,1} + X_{21,18} + Y_{21,18} + Z_{21,18} + D_1^- - D_1^+ + D_2^- - D_2^+ + \dots + D_{18}^- - D_{18}^+ = 13 \quad (26)$$

5 Results and Discussion

The above model Eqs. (24)–(26) implemented using a zero-one GP, and considered for the hospital wards with eighteen total number of nurses. Further, it considers a situation whereby four nurses are required in both morning and evening shifts and only 3 nurses in the night shift. Preemptive GP with priority ordering $1 \rightarrow 2 \rightarrow 3 \rightarrow 4$, is used to solve the model. The solution considered one goal at a time starting with a higher priority goal down to the lower. A LINGO optimization package version 16.0 used to generate the nurses scheduling problem. The zero-one GP scheduling pattern of the nurses is presented in Table 1.

It can be observed from Table 1 that throughout the scheduling pattern, no single staff nurse has a morning shift duty followed the next day by either evening or night shift duty. Similarly, no evening shift duty followed the next day by either morning or night shift duty. Here, both soft and hard constraints are fully satisfied during the planning horizon. Also, goal 1 and 2 [Model (18) and (19)] are satisfied fully as no overlapping of consecutive shifting throughout the scheduling.

Table 2 presents the summary of nurses number of days on duty at each shift of the planning horizon. For example, staff nurse 1 spends four days working during the morning shift in the first week of the schedule, three days on duty during the evening shift in the second week of the schedule and six days on duty during the night shift in the last week of the schedule period. Therefore, the staff nurse has completed thirteen days on duty at various shifts within the three week scheduling period (21 days). A similar analysis goes to all other staff nurses as given in Table 2. It can be noticed from Table 2 that during these 21 scheduling days, the staff nurses have enjoyed at least a weekend off duty, and all of them have worked for a total of thirteen days at various shifts. Hence, the objective or goal 3 and 4 [Model (20) and (21)] have been fully satisfied.

Table 3 presents the daily distribution of nurses in morning, evening and night shifts during the scheduling period of 21 days. It can be seen that there is a uniform distribution, except on day 8 and 18, and there is a variation of 1 nurse. It can be concluded that the distribution varies between 11 and 12 nurses per day throughout the planning horizon.

Table 4 presents the cyclical scheduling where nurses rotated for a period of 18 cycles (54 weeks). Each scheduling circle consists of 3 weeks (21 days). It is interesting to notice that the nurses are rotated within these periods equally and systematically. This schedule will require not much effort from the management or nurses head in designing the duty roster or assigning various nurses to respective shifts in the desired wards.

Table 1 Nurses scheduling pattern using zero-one goal programming

Days	SN_1	SN_2	SN_3	SN_4	SN_5	SN_6	SN_7	SN_8	SN_9	SN_{10}	SN_{11}	SN_{12}	SN_{13}	SN_{14}	SN_{15}	SN_{16}	SN_{17}	SN_{18}
1	NS	-	ES	MS	ES	MS	NS	-	-	ES	-	MS	NS	-	-	-	ES	MS
2	NS	-	ES	MS	-	MS	NS	-	-	-	ES	-	NS	-	ES	MS	ES	MS
3	NS	-	ES	-	MS	-	NS	-	MS	MS	ES	-	NS	-	ES	MS	ES	-
4	-	ES	-	-	MS	NS	-	MS	MS	MS	ES	NS	-	ES	-	-	ES	NS
5	-	ES	MS	MS	-	NS	-	MS	MS	-	ES	NS	-	ES	ES	ES	-	NS
6	-	ES	MS	MS	-	NS	-	MS	MS	-	ES	NS	-	ES	ES	ES	-	NS
7	ES	ES	-	MS	NS	-	MS	MS	-	MS	-	-	-	ES	-	ES	NS	-
8	ES	-	ES	-	NS	-	MS	MS	-	MS	NS	-	MS	ES	MS	ES	NS	-
9	ES	MS	ES	-	NS	-	MS	-	ES	-	NS	-	MS	ES	MS	-	NS	-
10	-	MS	-	NS	-	MS	MS	ES	ES	NS	NS	ES	-	-	MS	NS	-	ES
11	-	MS	-	NS	-	MS	-	ES	ES	NS	-	ES	-	MS	MS	NS	-	ES
12	MS	MS	-	NS	-	MS	-	ES	-	NS	-	ES	ES	MS	-	NS	-	ES
13	MS	MS	NS	-	MS	-	-	ES	NS	-	-	ES	ES	MS	NS	-	-	ES
14	-	MS	NS	-	MS	MS	-	ES	NS	-	-	ES	ES	MS	NS	-	-	ES
15	MS	-	NS	-	MS	MS	ES	-	NS	-	ES	ES	-	-	NS	-	MS	ES
16	MS	NS	-	ES	MS	-	ES	NS	-	ES	ES	-	MS	NS	-	-	MS	-
17	-	NS	-	ES	-	ES	ES	NS	-	ES	-	MS	MS	NS	-	MS	-	MS
18	-	NS	-	ES	ES	ES	-	NS	-	ES	MS	MS	-	NS	-	MS	MS	MS
19	NS	-	NS	ES	ES	ES	NS	-	ES	-	MS	MS	NS	-	-	-	MS	-
20	NS	-	NS	-	ES	-	NS	-	ES	MS	MS	-	NS	-	ES	ES	MS	-
21	NS	-	NS	ES	-	-	NS	-	ES	MS	MS	-	NS	-	ES	ES	MS	-

SN_i Staff Nurse i ; $i = 1, 2, \dots, 18$, *MS* Morning Shift; *ES* Evening Shift; *NS* Night Shift

Table 2 Nurses number of days at each shift in the scheduling cycle

Shifts	SN_1	SN_2	SN_3	SN_4	SN_5	SN_6	SN_7	SN_8	SN_9	SN_{10}	SN_{11}	SN_{12}	SN_{13}	SN_{14}	SN_{15}	SN_{16}	SN_{17}	SN_{18}
MS	4	6	5	5	6	7	4	5	4	6	4	4	4	4	4	4	6	4
ES	3	4	5	5	4	3	3	5	6	4	6	6	6	6	6	6	4	6
NS	6	3	3	3	3	3	6	3	3	3	3	3	3	3	3	3	3	3
Total	13	13	13	13	13	13	13	13	13	13	13	13	13	13	13	13	13	13

SN_i Staff Nurse i ; $i = 1, 2, \dots, 18$, MS Morning Shift; ES Evening Shift; NS Night Shift

Table 3 Daily number of nurses at each shift in the scheduling cycle

Days	Morning shift	Evening shift	Night shift	Total
1	4	4	3	11
2	4	4	3	11
3	4	4	3	11
4	4	4	3	11
5	4	4	3	11
6	4	4	3	11
7	4	4	3	11
8	5	4	3	12
9	4	4	3	11
10	4	4	3	11
11	4	4	3	11
12	4	4	3	11
13	4	4	3	11
14	4	4	3	11
15	4	4	3	11
16	4	4	3	11
17	4	4	3	11
18	5	4	3	12
19	4	4	3	11
20	4	4	3	11
21	4	4	3	11

6 Conclusion and Future Work

Nursing is an integral part of any hospital or healthcare service delivery system. In most hospitals globally, nurses are the cardinal functionality of the wards. They take care of triaging activities and render initial first aid to patients before the arrivals of consultants and doctors. In most cases, they are overstretched due to their increasing demand in the various units of the hospitals. In this study, a 0–1 GP technique is used to develop a nurse scheduling model for JNMC, Aligarh, India. The current operand is manual and based on the intuition of the head nurse, which is costly, time-consuming and inefficient in terms of output, duplication of assignment and human errors. Moreover, some critical criteria such as schedule balancing, the nurses' preferences and fairness consideration may not be achieved manually.

The model of this study has improvements and efficiency over the existing system. It has satisfied balancing in the scheduling period; most soft and hard constraints are well incorporated. The nurses are distributed optimally over the three available shifts. The quantity of nurses required in every shift is also met. There is proportionality among all the nurses throughout the shifts and days in the scheduling cycle. The

Table 4 Nurses cycling scheduling pattern for 54 weeks

Cycles(weeks)	J_1	J_2	J_3	J_4	J_5	J_6	J_7	J_8	J_9	J_{10}	J_{11}	J_{12}	J_{13}	J_{14}	J_{15}	J_{16}	J_{17}	J_{18}
1(1-3)	SN ₁	SN ₂	SN ₃	SN ₄	SN ₅	SN ₆	SN ₇	SN ₈	SN ₉	SN ₁₀	SN ₁₁	SN ₁₂	SN ₁₃	SN ₁₄	SN ₁₅	SN ₁₆	SN ₁₇	SN ₁₈
2(4-6)	SN ₂	SN ₃	SN ₄	SN ₅	SN ₆	SN ₇	SN ₈	SN ₉	SN ₁₀	SN ₁₁	SN ₁₂	SN ₁₃	SN ₁₄	SN ₁₅	SN ₁₆	SN ₁₇	SN ₁₈	SN ₁
3(7-9)	SN ₃	SN ₄	SN ₅	SN ₆	SN ₇	SN ₈	SN ₉	SN ₁₀	SN ₁₁	SN ₁₂	SN ₁₃	SN ₁₄	SN ₁₅	SN ₁₆	SN ₁₇	SN ₁₈	SN ₁	SN ₂
4(10-12)	SN ₄	SN ₅	SN ₆	SN ₇	SN ₈	SN ₉	SN ₁₀	SN ₁₁	SN ₁₂	SN ₁₃	SN ₁₄	SN ₁₅	SN ₁₆	SN ₁₇	SN ₁₈	SN ₁	SN ₂	SN ₃
5(13-15)	SN ₅	SN ₆	SN ₇	SN ₈	SN ₉	SN ₁₀	SN ₁₁	SN ₁₂	SN ₁₃	SN ₁₄	SN ₁₅	SN ₁₆	SN ₁₇	SN ₁₈	SN ₁	SN ₂	SN ₃	SN ₄
6(16-18)	SN ₆	SN ₇	SN ₈	SN ₉	SN ₁₀	SN ₁₁	SN ₁₂	SN ₁₃	SN ₁₄	SN ₁₅	SN ₁₆	SN ₁₇	SN ₁₈	SN ₁	SN ₂	SN ₃	SN ₄	SN ₅
7(19-21)	SN ₇	SN ₈	SN ₉	SN ₁₀	SN ₁₁	SN ₁₂	SN ₁₃	SN ₁₄	SN ₁₅	SN ₁₆	SN ₁₇	SN ₁₈	SN ₁	SN ₂	SN ₃	SN ₄	SN ₅	SN ₆
8(22-24)	SN ₈	SN ₉	SN ₁₀	SN ₁₁	SN ₁₂	SN ₁₃	SN ₁₄	SN ₁₅	SN ₁₆	SN ₁₇	SN ₁₈	SN ₁	SN ₂	SN ₃	SN ₄	SN ₅	SN ₆	SN ₇
9(25-27)	SN ₉	SN ₁₀	SN ₁₁	SN ₁₂	SN ₁₃	SN ₁₄	SN ₁₅	SN ₁₆	SN ₁₇	SN ₁₈	SN ₁	SN ₂	SN ₃	SN ₄	SN ₅	SN ₆	SN ₇	SN ₈
10(28-30)	SN ₁₀	SN ₁₁	SN ₁₂	SN ₁₃	SN ₁₄	SN ₁₅	SN ₁₆	SN ₁₇	SN ₁₈	SN ₁	SN ₂	SN ₃	SN ₄	SN ₅	SN ₆	SN ₇	SN ₈	SN ₉
11(31-33)	SN ₁₁	SN ₁₂	SN ₁₃	SN ₁₄	SN ₁₅	SN ₁₆	SN ₁₇	SN ₁₈	SN ₁	SN ₂	SN ₃	SN ₄	SN ₅	SN ₆	SN ₇	SN ₈	SN ₉	SN ₁₀
12(34-36)	SN ₁₂	SN ₁₃	SN ₁₄	SN ₁₅	SN ₁₆	SN ₁₇	SN ₁₈	SN ₁	SN ₂	SN ₃	SN ₄	SN ₅	SN ₆	SN ₇	SN ₈	SN ₉	SN ₁₀	SN ₁₁
13(37-39)	SN ₁₃	SN ₁₄	SN ₁₅	SN ₁₆	SN ₁₇	SN ₁₈	SN ₁	SN ₂	SN ₃	SN ₄	SN ₅	SN ₆	SN ₇	SN ₈	SN ₉	SN ₁₀	SN ₁₁	SN ₁₂
14(40-42)	SN ₁₄	SN ₁₅	SN ₁₆	SN ₁₇	SN ₁₈	SN ₁	SN ₂	SN ₃	SN ₄	SN ₅	SN ₆	SN ₇	SN ₈	SN ₉	SN ₁₀	SN ₁₁	SN ₁₂	SN ₁₃
15(43-45)	SN ₁₅	SN ₁₆	SN ₁₇	SN ₁₈	SN ₁	SN ₂	SN ₃	SN ₄	SN ₅	SN ₆	SN ₇	SN ₈	SN ₉	SN ₁₀	SN ₁₁	SN ₁₂	SN ₁₃	SN ₁₄
16(46-48)	SN ₁₆	SN ₁₇	SN ₁₈	SN ₁	SN ₂	SN ₃	SN ₄	SN ₅	SN ₆	SN ₇	SN ₈	SN ₉	SN ₁₀	SN ₁₁	SN ₁₂	SN ₁₃	SN ₁₄	SN ₁₅
17(49-51)	SN ₁₇	SN ₁₈	SN ₁	SN ₂	SN ₃	SN ₄	SN ₅	SN ₆	SN ₇	SN ₈	SN ₉	SN ₁₀	SN ₁₁	SN ₁₂	SN ₁₃	SN ₁₄	SN ₁₅	SN ₁₆
18(52-54)	SN ₁₈	SN ₁	SN ₂	SN ₃	SN ₄	SN ₅	SN ₆	SN ₇	SN ₈	SN ₉	SN ₁₀	SN ₁₁	SN ₁₂	SN ₁₃	SN ₁₄	SN ₁₅	SN ₁₆	SN ₁₇

SN₁ Staff Nurse 1; SN₂ Staff Nurse 2; ... SN₁₈ Staff Nurse 18

nurses' preferences are well captured; hence, there is healthiness in the process as every nurse have some weekend off to enable them to have enough rest and social interaction with friends and relatives. Also, evident from the 0–1 GP model, there is a continuity in the crew of nursing shifts, as no overlapping in the assignment of duties and no consecutive shifts for the nurses. The limitation of the study is that a fixed number of nurses considered based on data availability. Another one is, the expertise of nurses is not considered in this model. Future work may include a broader range of nurses and the expertise and experience of the nurses, which might improve the current model if it is incorporated.

References

1. Warner, D.M.: Scheduling nursing personnel according to nursing preference: a mathematical programming approach. *Oper. Res.* **24**(5), 842–856 (1976)
2. Bard, J.F., Purnomo, H.W.: Preference scheduling for nurses using column generation. *Eur. J. Oper. Res.* **164**(2), 510–534 (2005)
3. Wang, S.P., Hsieh, Y.K., Zhuang, Z.Y., Ou, N.C.: Solving an outpatient nurse scheduling problem by binary goal programming. *J. Ind. Prod. Eng.* **31**(1), 41–50 (2014)
4. Yilmaz, E.: A mathematical programming model for scheduling of nurses' labor shifts. *J. Med. Syst.* **36**(2), 491–496 (2012)
5. Valouxis, C., Housos, E.: Hybrid optimization techniques for the workshift and rest assignment of nursing personnel. *Artif. Intell. Med.* **20**(2), 155–175 (2000)
6. Gascon, V., Villeneuve, S., Michelon, P., Ferland, J.A.: Scheduling the flying squad nurses of a hospital using a multi-objective programming model. *Ann. Oper. Res.* **96**(1–4), 149 (2000)
7. Azaiez, M.N., Al Sharif, S.S.: A 0–1 goal programming model for nurse scheduling. *Comput. Oper. Res.* **32**(3), 491–507 (2005)
8. Topaloglu, S., Selim, H.: Nurse scheduling using fuzzy modeling approach. *Fuzzy Sets Syst.* **161**(11), 1543–1563 (2010)
9. Cetin, E., Sarucan, A.: Nurse scheduling using binary fuzzy goal programming. In 2015 6th International Conference on Modeling, Simulation, and Applied Optimization (ICMSAO), pp. 1–6. IEEE (2015)
10. Chang, C.T.: Binary fuzzy goal programming. *Eur. J. Oper. Res.* **180**(1), 29–37 (2007)
11. Sufahani, S.F., Razali, S N., Ismail, Z.: A scheduling problem for hospital operating theatre. <http://arxiv.org/abs/1205.2108> [arXiv:1205.2108](https://arxiv.org/abs/1205.2108) (2012)
12. Modibbo, U.M., Hassan, M., Ahmed, A., Ali, I.: Multi-criteria decision analysis for pharmaceutical supplier selection problem using fuzzy TOPSIS. In: *Management Decision*, vol. 60, No. 3, pp. 806–836. <https://doi.org/10.1108/MD-10-2020-1335> (2022)
13. Fei, H., Meskens, N., Chu, C.: A planning and scheduling problem for an operating theatre using an open scheduling strategy. *Comput. Ind. Eng.* **58**(2), 221–230 (2010)
14. Warner, D., Prawda, J.: A mathematical programming model for scheduling nursing personnel in a hospital. *Manage. Sci.* **19**(4), 411–422 (1972)
15. Warner, D.M.: Nurse staffing, scheduling, and reallocation in the hospital. *Hosp. Health Services Admin.* **21**(3), 77–90 (1976)

On the Performance of a Flow Energy Harvester Using Time Delay



Zakaria Ghoul

Abstract This paper describes vibration-based periodic energy harvesting (EH) during a delayed device consisting of a nonlinear oscillator excited at gallop under time-periodic delay and coupled to a circuit via a piezoelectric coupling mechanism. It is assumed that the amplitude of the time delay is modulated so that the modulation frequency is close to double the intrinsic frequency of the oscillation. The application of multiple scales tactics will give an estimate of the amplitude of the periodic oscillation, and therefore, an estimate of the corresponding power extracted by the acquisition device. The results show that the presence of amplitude of delay modulation in the mechanical component greatly increases the amplitude of vibration and therefore the power output in a particular wind speed range. Numerical simulations are performed to support analytical predictions.

Keywords Energy harvesting · Delayed oscillator · Galloping excitation · Piezoelectric coupling

1 Introduction

Harvesting electrical energy from mechanical vibrations using a delayed system with time-periodic delay amplitude has been amply investigated [1–4]. It was considered that in a van der Pol oscillator with modulated delay amplitude introduced in the mechanical subsystem and coupled to an electromagnetic energy harvester that the quasi-periodic (QP) vibrations can be used to harvest energy with good performance [1]. The rest of this study presented in [1] was considered in [2] where the time delay is introduced in addition in the electrical circuit.

Another study reported that modulating the delay amplitude in a Duffing-type oscillator produced higher amplitude QP oscillations, which were used for energy recovery and were efficient [3]. Recently, it has been reported that the use of a QP oscillator improves the energy acquisition efficiency in a delayed Duffing-van der

Z. Ghoul (✉)

Polydisciplinary Faculty of Taroudant, University Ibn Zohr, Taroudant, Morocco
e-mail: goulizakaria@gmail.com

Pol oscillator with a delayed piezoelectric coupling of appropriate delay frequency and amplitude value [4].

However, in certain harvester systems under aerodynamic and fundamental excitations, it has been shown that QP vibration will cause a significant reduction in harvesting power exceeding the flutter speed [5, 6]. Therefore, for such systems, extracting energy should be avoided in the QP area. Nevertheless, it turns out that in the presence of a time delay, QP vibration can produce beneficial effects, which can then be used to harvest energy [7–9].

In [7], time delay has been used for Duffing-type collector devices that are excited by harmonics and coupled to piezoelectric circuits. The results show that for appropriate delay parameter values, QP vibration can be used to extract energy with good efficiency at a broadband excitation frequency far from resonance, thereby avoiding hysteresis and bistability problems near resonance.

The energy harvesting based on QP vibration in a delayed nonlinear MEMS device composed of a delayed Mathieu-van der Pol-Duffing oscillator coupled to a delayed piezoelectric coupling mechanism has been studied in [8]. It has also been shown that the QP vibration can be used for energy extraction with higher efficiency compared to periodic vibration.

Recently, we investigated the energy recovery performance of a Van der Pol oscillator under harmonic excitation and combined with a delayed piezoelectric recuperator [9]. It has been shown that if the circuit has a time delay, there is an optimum frequency at which the power emitted by the QP oscillation is maximized.

Taking advantages of the time delay effect, this work investigates harvesting energy from the periodic vibrations in a nonlinear delayed electromechanical harvester device with modulated delay amplitude in cross flow and coupled to an electric circuit through a piezoelectric mechanism. This study is useful in many applications where time is specific to the mechanical part of the harvester [10–12]. In this case, the time delay is not considered as additional input power introduced into the system.

The rest of this paper is organized as follows: Sect. 2 describes the recovery system and uses a multiple scales approach to provide an estimate of the cyclic response and the amount of energy obtained near the delay parametric resonance. Section 3 analyses the influence of the delay amplitude introduced in the mechanical components of the energy harvesting device on the energy harvesting efficiency and summarizes the results in the conclusion.

2 Model Description and Periodic Energy Harvesting

The energy harvester device we consider consists of a nonlinear electromechanical oscillator in cross flow as shown in the schematic presented in Fig. 1. The mechanical oscillator consists of a bluff body of mass m coupled to an electrical circuit through a piezoelectric device. Assuming that the mechanical parts of the harvester receive slow feedback, the system's dimensionless adjustment equation can be written as:

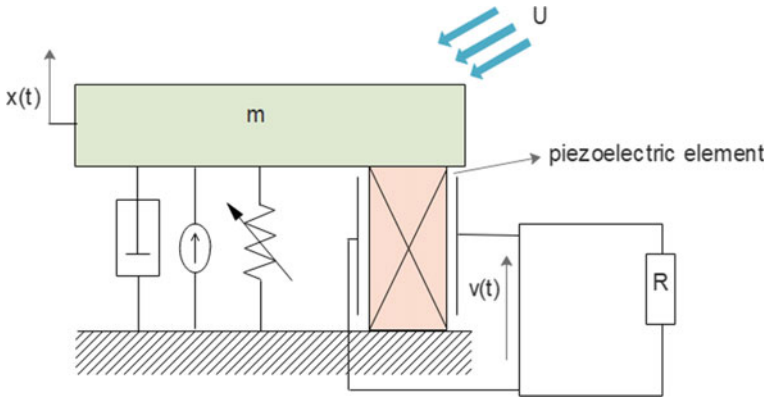


Fig. 1 Schematic of the EH system

$$\ddot{x}(t) + \mu x(t) + \xi_1 \dot{x}(t) + \xi_3 \dot{x}(t)^3 + \gamma x(t)^3 - \kappa v(t) = \lambda(t)x(t - \tau) \quad (1)$$

$$\dot{v}(t) + \alpha v(t) + \dot{x}(t) = 0 \quad (2)$$

Here, $x(t)$ is the transverse displacement of the mass m and $v(t)$ is the voltage across the load resistance. The coefficients ξ_1 , ξ_3 are the mechanical damping components, μ the natural frequency of the harvester, γ the stiffness, κ the piezoelectric coupling term in the mechanical attachment, and α the reciprocal of the time constant of the electrical circuit. The parameter $\lambda(t)$ is the feedback gains in the mechanical attachment and τ is the time delay. The current research is based totally on the linearized piezoelectric coupling in Eq. (1) although the piezoelectric constitutive equations are nonlinear [13–15]. We assume that the delay amplitude $\lambda(t)$ is modulated around a mean value such that

$$\lambda(t) = \lambda_1 \cos(\omega t) \quad (3)$$

where λ_1 and ω are, respectively, the amplitude and the frequency of the modulation. Specifically [1–4] shows that adjusting the delay amplitude of the mechanical part in the collector system is beneficial to the performance of the collected energy. Note that this time delay in the mechanical part of the harvester is useful in many applications because it is inherently present in the system [10–12]. In this case, the time delay is not considered as additional input power introduced into the system. The coefficients ξ_1 and ξ_3 in Eq. (1) are defined by

$$\xi_1 = 2[\xi_m - ma_1U], \quad \xi_3 = \frac{2ma_3}{U} \quad (4)$$

where ξ_m is the mechanical damping ratio, m is the flow to harvester mass ratio, U is the reduced wind speed, and the coefficients a_1 and a_3 account for the different geometries and aspect ratios of the bluff body. As given in [16], the damping coeffi-

cients are modelled in Eq. (4) in this form to study the effect and performance of the flying energy harvester. Note that the case of undelayed system ($\lambda_1 = \tau = 0$) was studied in [16].

To study the response of the harvester near the delay parametric resonance, we assume the resonance condition $\omega = 2\sqrt{\mu} + \sigma$ where σ is a detuning parameter. Approximation of solutions are obtained using the method of multiple scales [17] by introducing a bookkeeping parameter ε and scaling as $\xi_1 = \varepsilon\tilde{\xi}_1, \xi_3 = \varepsilon\tilde{\xi}_3, \gamma = \varepsilon\tilde{\gamma}, \kappa = \varepsilon\tilde{\kappa}, \lambda_1 = \varepsilon\tilde{\lambda}_1, \sigma = \varepsilon\tilde{\sigma}$. Equations (1) and (2) become

$$\ddot{x}(t) + \mu x(t) = \varepsilon[-\tilde{\xi}_1\dot{x} - \tilde{\xi}_3\dot{x}(t)^3 - \tilde{\gamma}x(t)^3 + \tilde{\kappa}v(t) + \tilde{\lambda}_1 \cos(\omega t)x(t - \tau)] \tag{5}$$

$$\dot{v}(t) + \alpha v(t) + \dot{x}(t) = 0 \tag{6}$$

We seek a solution of Eqs. (5) and (6) in the form

$$x(t) = x_0(T_0, T_1) + \varepsilon x_1(T_0, T_1) + O(\varepsilon^2) \tag{7}$$

$$v(t) = v_0(T_0, T_1) + \varepsilon v_1(T_0, T_1) + O(\varepsilon^2) \tag{8}$$

where $T_0 = t$ and $T_1 = \varepsilon t$. Using the time derivatives $\frac{d}{dt} = D_0 + \varepsilon D_1 + O(\varepsilon^2)$ and $\frac{d^2}{dt^2} = D_0^2 + \varepsilon^2 D_1^2 + 2\varepsilon D_0 D_1 + O(\varepsilon^2)$ where $D_i^j = \frac{\partial^j}{\partial T_i^j}$, substituting (7), (8) into (5), (6) and balancing terms of like powers of ε , we obtain up to the second order

$$D_0^2 x_0 + \mu x_0 = 0 \tag{9}$$

$$D_0 v_0 + \alpha v_0 + D_0 x_0 = 0 \tag{10}$$

and

$$D_0^2 x_1 + \mu x_1 = -2D_0 D_1 x_0 + \tilde{\kappa} v_0 - \tilde{\gamma} x_0^3 - \tilde{\xi}_3 (D_0 x_0)^3 - \tilde{\xi}_1 (D_0 x_0) + \tilde{\lambda}_1 \cos(2\sqrt{\mu}T_0 + \tilde{\sigma}T_1)x_{0\tau} \tag{11}$$

$$D_0 v_1 + \alpha v_1 = -D_1 v_0 - D_0 x_1 - D_1 x_0 \tag{12}$$

The first-order solution generated from Eqs. (9), (10) analytically is given by

$$x_0(T_0, T_1) = A(T_1)e^{i\sqrt{\mu}T_0} + \bar{A}(T_1)e^{-i\sqrt{\mu}T_0} \tag{13}$$

$$v_0(T_0, T_1) = \frac{-i\sqrt{\mu}A(T_1)}{\alpha + i\sqrt{\mu}}e^{i\sqrt{\mu}T_0} + \frac{i\sqrt{\mu}\bar{A}(T_1)}{\alpha - i\sqrt{\mu}}e^{-i\sqrt{\mu}T_0} \tag{14}$$

where $A(T_1)$ and $\bar{A}(T_1)$ are unknown complex conjugate functions. Substituting Eqs. (13), (14) into (11), (12) and eliminating the secular terms, gives

$$\begin{aligned}
 & -2i\sqrt{\mu}(D_1A) - 3\tilde{\gamma}A^2\bar{A} + 3\tilde{\xi}_3(i\sqrt{\mu})^3A^2\bar{A} \\
 & -i\sqrt{\mu}\tilde{\xi}_1A - \frac{\tilde{\kappa}i\sqrt{\mu}A}{\alpha + i\sqrt{\mu}} + \frac{\tilde{\lambda}_1\bar{A}}{2}e^{i\sqrt{\mu}\tau}e^{i\tilde{\sigma}T_1} = 0
 \end{aligned} \tag{15}$$

Assuming $A = \frac{1}{2}ae^{i\theta}$ where a and θ are the amplitude and the phase we obtain the modulation equations

$$\begin{cases} \frac{da}{dt} = S_1a + S_2a^3 + S_5a \cos(\varphi) + S_6a \sin(\varphi) \\ a \frac{d\varphi}{dt} = S_3a + S_4a^3 + 2S_6a \cos(\varphi) - 2S_5a \sin(\varphi) \end{cases} \tag{16}$$

in which $S_i (i = 1, \dots, 6)$ are given in Appendix and $\varphi = \tilde{\sigma}T_1 - 2\theta$. The solution given by (13), (14) reads

$$\begin{cases} x_0(T_0, T_1) = a \cos(\sqrt{\mu}t + \theta) \\ v_0(T_0, T_1) = V \cos(\sqrt{\mu}t + \theta + \arctan \frac{\alpha}{\sqrt{\mu}}) \end{cases} \tag{17}$$

with the voltage amplitude V is given by

$$V = \frac{\sqrt{\mu}}{\sqrt{\alpha^2 + \mu}}a \tag{18}$$

The steady-state response of the periodic solutions for the original system [Eqs. (5), (6)] is obtained by setting $\frac{da}{dt} = \frac{d\varphi}{dt} = 0$ in (16) and squaring, adding, and eliminating the phase, the amplitude a is obtained by the equation

$$(S_1a + S_2a^3)^2 + \left(\frac{S_3}{2}a + \frac{S_4}{2}a^3\right)^2 = (S_5^2 + S_6^2)a^2 \tag{19}$$

Then the Eq. (19) corresponding to the steady-state response of the periodic solutions for the original system. The condition for Eq. (19) to have two real roots is

$$\left(2S_1S_2 + \frac{S_3S_4}{2}\right)^2 - 4\left(S_2^2 + \frac{S_4^2}{4}\right)\left(S_1^2 + \frac{S_3^2}{4} - S_5^2 - S_6^2\right) > 0 \tag{20}$$

and the conditions for the stability of the steady-state response are

$$2S_1S_2 + \frac{S_3S_4}{2} > 0 \tag{21}$$

$$S_1^2 + \frac{S_3^2}{4} - S_5^2 - S_6^2 > 0 \tag{22}$$

Note that the case of undelayed system ($\lambda_1 = \tau = 0$) where ($S_5 = S_6 = 0$) the amplitude a is obtained by the equation

$$a = \sqrt{-\frac{S_1}{S_2}} = \sqrt{-\frac{S_3}{S_4}} \tag{23}$$

Figure 2a shows the stability chart in the parameter plane (λ_1, τ) for $\omega = 2$ indicating the *grey* regions where stable QP (SQP) solutions take place and the *white* region where the conditions (21) and (22) are satisfied corresponding to stable periodic (SP) solutions. Figure 2b shows time histories and the corresponding output power responses related to crosses labelled 1, 2, 3 in Fig. 2a. From crossover point 1 or 3 to crossover point 2, the dichotomous response of the SP oscillation to SQP via the secondary Hopf bifurcation reduces the amplitude response and output power at crossover point 2, respectively.

The average power obtained by integrating a dimensionless form of instantaneous power $P(t) = \alpha v(t)^2$ over a period T . This is given by

$$P_{av} = \frac{1}{T} \int_0^T \alpha v^2 dt \tag{24}$$

where $T = \frac{4\pi}{\omega}$. Hence, the average power expressed by $P_{av} = \frac{\alpha V^2}{2}$ takes the form

$$P_{av} = \frac{1}{2} \left(\frac{\alpha \mu}{\alpha^2 + \mu} \right) a^2 \tag{25}$$

where the amplitude a is given by Eq. (19) in the case of delayed system or by Eq. (23) in the case of undelayed one. Applying the maximization procedure by using the global optimal value method, the maximum power reads

$$P_{max} = \frac{\alpha \mu a^2}{\alpha^2 + \mu} \tag{26}$$

Equations (19), (23), and (26) will be used to study the effect of parameters on the steady-state response and on the output power of the harvester.

Hereafter, we fix the parameters as $\mu = 1.175, \gamma = 0.024, \kappa = 10^{-2}, m = 6.83 \times 10^{-4}, \xi_m = 3.10^{-3}, a_1 = 2.5,$ and $a_3 = 130.$

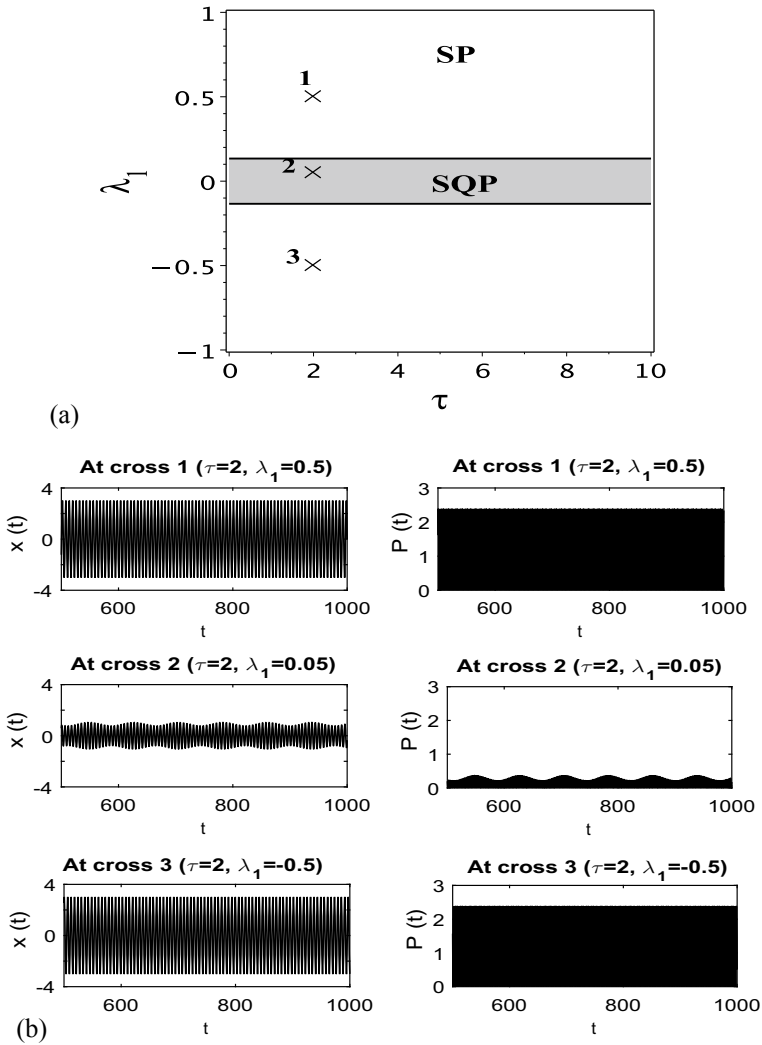


Fig. 2 a Stability chart in the plane (λ_1, τ) , b time and power histories corresponding to different regions picked from (a). SP: stable periodic, SQP: stable QP; $h = 0.25, \omega = 2, \alpha_2 = 0.05, \tau_2 = 4.2, \chi = 0.05,$ and $\kappa = 0.5$

3 Main Results

Next, examine the effects of various system parameters such as the reduced wind speed U , the frequency of the modulation ω , the amplitude of the modulation λ_1 , and the reciprocal of the time constant of the electrical circuit α on amplitude and power. Equations (19) and (23) are used for periodic solutions to delayed and non-delayed system, respectively, and Eq. (26) is exploited to calculate the power response. Numerical simulations are performed to support analytical forecasting using `dde23` algorithm [18].

Figure 3 shows the variation of the amplitudes of periodic response (Fig. 3a) as well as the maximum output power amplitudes (Fig. 3b) versus the reduced wind speed U for $\lambda_1 = \tau = 0$ (undelayed circuit, *grey* lines) and for $\lambda_1 = 0.3$, $\omega = 2.1$, $\tau = 2\pi$ (delayed circuit, *black* lines). The analytical prediction is compared to numerical simulation (circles) obtained by using `dde23` algorithm [18]. It can be observed from Fig. 3 that the presence of modulated delay amplitude in mechanical components can significantly increase vibration and power output within certain limits of U (Fig. 3, *black* lines).

The frequency response curve for amplitude and power response is shown in Fig. 4 when there is a delay in the mechanical subsystem ($\lambda_1 = 0.3$, $\tau = 2\pi$). From Fig. 4, we can see that the presence of low delay amplitude in mechanics part of the harvester has a significant effect on energy recovery from periodic oscillations.

Figure 5 shows the variation of the amplitude of the responses (Fig. 5a) and the powers (Fig. 5b) versus the mechanical delay amplitude λ_1 for $\omega = 2.1$ and $\tau = 2\pi$. The analytical prediction is compared to numerical simulation (circles) obtained by using `dde23` algorithm [18]. One can observe from Fig. 5b that for small values of

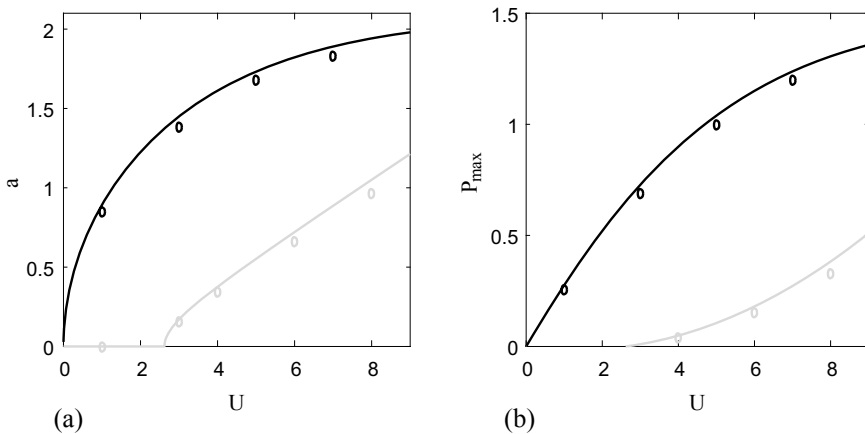


Fig. 3 Vibration and power amplitudes versus U . *Black* lines for delayed mechanical component ($\lambda_1 = 0.3$, $\omega = 2.1$, $\tau = 2\pi$.) and *grey* lines for undelayed system ($\lambda_1 = \tau = 0$)

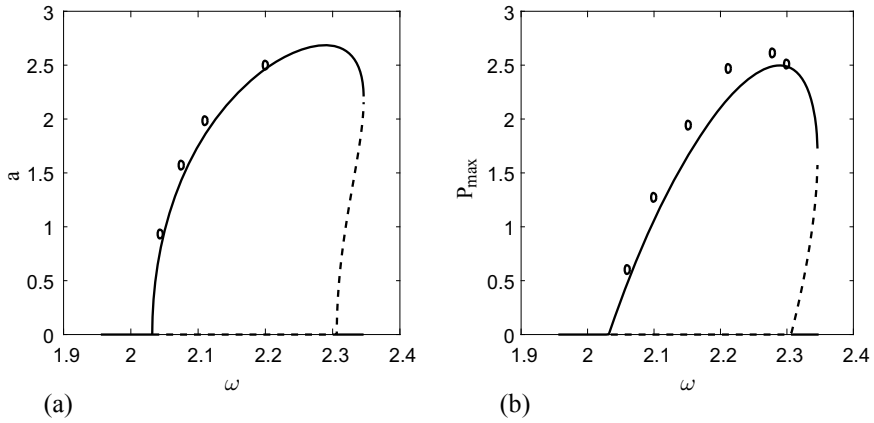


Fig. 4 Vibration and power amplitudes versus ω for $(\lambda_1 = 0.3, \tau = 2\pi)$

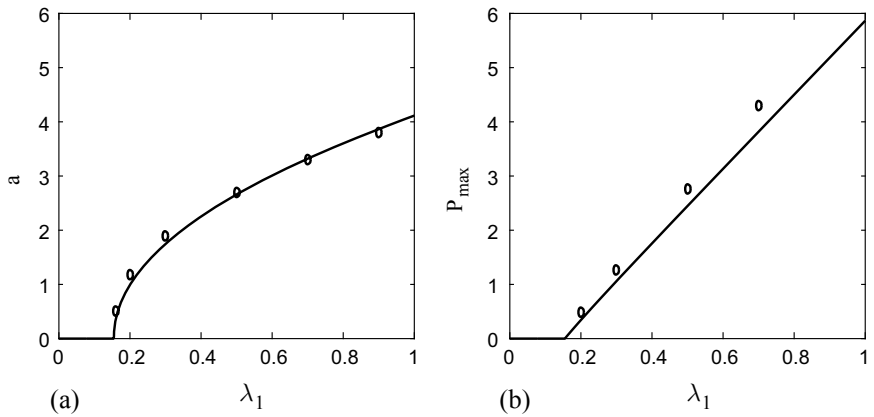


Fig. 5 Vibration and power amplitudes versus λ_1 for $(\omega = 2.1, \tau = 2\pi)$

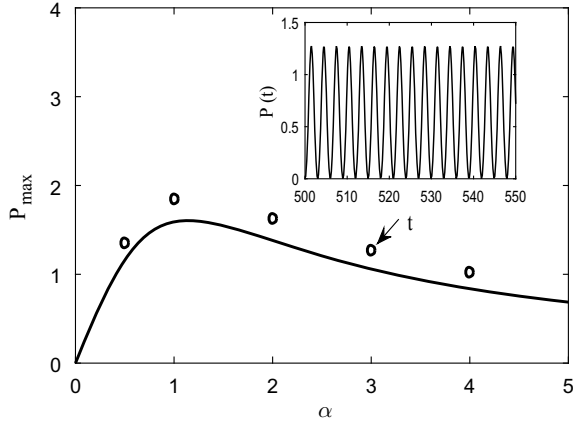
λ_1 , energy cannot be extracted from periodic vibrations. Beyond a certain value of λ_1 , a periodic solution is displayed, providing better performance energy recovery.

Finally, we show in Fig. 6 the influence of the reciprocal of the time constant of the electrical circuit α on the powers. It can be observed that for a particular value of parameter, the range of α where significant energy capture efficiency is achieved.

4 Conclusions

Energy recovery based on periodic vibrations has been studied with a delayed non-linear oscillator which receives uncontrolled excitation and is coupled to the circuit via a piezoelectric mechanism when the amplitude of the delay is modulated in fre-

Fig. 6 Power amplitudes versus α for ($\omega = 2.1$, $\tau = 2\pi$ and $\lambda_1 = 0.3$)



quency, close to the resonance parameter of the system. The tactical application of multiple scales allows an estimate of the amplitude of the vibrations and therefore of the output power. It can be seen that the amplitude modulation of the delay, which is close to the parametric resonance, greatly increases the amplitude of the oscillation which produces the output with better efficiency.

Appendix

$$S_1 = -\frac{\xi_1}{2} - \frac{\kappa\alpha}{2\alpha^2 + 2\mu}, \quad S_2 = -\frac{3\xi_3}{8}, \quad S_4 = -\frac{3\gamma}{4\sqrt{\mu}}$$

$$S_3 = \sigma - \frac{\kappa\sqrt{\mu}}{\alpha^2 + \mu}, \quad S_5 = \frac{\lambda_1}{4\sqrt{\mu}} \sin(\sqrt{\mu}\tau), \quad S_6 = \frac{\lambda_1}{4\sqrt{\mu}} \cos(\sqrt{\mu}\tau)$$

References

1. Belhaq, M., Hamdi, M.: Energy harversting from quasi-periodic vibrations. *Nonlinear Dyn.* **86**, 2193–2205 (2016)
2. Ghouli, Z., Hamdi, M., Belhaq, M.: Energy harvesting from quasi-periodic vibrations using electromagnetic coupling with delay. *Nonlinear Dyn.* **89**, 1625–1636 (2017)
3. Ghouli, Z., Hamdi, M., Belhaq, M.: Energy harvesting in a Duffing oscillator with modulated delay amplitude. In: Kovacic, I., Lenci, S. (eds.) *IUTAM Symposium on Exploiting Nonlinear*

- Dynamics for Engineering Systems. ENOLIDES 2018. IUTAM Bookseries, vol. 37. Springer, Cham (2020)
4. Ghouli, Z., Belhaq, M.: Energy harvesting in a delay-induced parametric van der Pol-Duffing oscillator. *Eur. Phys. J. Spec. Top.* **230**, 3591–3598 (2021)
 5. Abdelkefi, A., Nayfeh, A.H., Hajj, M.R.: Design of piezoaeroelastic energy harvesters. *Nonlinear Dyn.* **68**, 519–530 (2012)
 6. Bibo, A., Daqaq, M.F.: Energy harvesting under combined aerodynamic and base excitations. *J. Sound Vibr.* **332**, 5086–5102 (2013)
 7. Ghouli, Z., Hamdi, M., Lakrad, F., Belhaq, M.: Quasiperiodic energy harvesting in a forced and delayed Duffing harvester device. *J. Sound Vibr.* **407**, 271–285 (2017)
 8. Belhaq, M., Ghouli, Z., Hamdi, M.: Energy harvesting in a Mathieu-van der Pol-Duffing MEMS device using time delay. *Nonlinear Dyn.* **94**, 2537–2546 (2018)
 9. Ghouli, Z., Hamdi, M., Belhaq, M.: The delayed van der Pol oscillator and energy harvesting. In: Belhaq, M. (ed.) *Topics in Nonlinear Mechanics and Physics*. Springer Proceedings in Physics, vol. 228. Springer, Singapore (2019)
 10. Stepan, G., Kalmar-Nagy, T.: Nonlinear regenerative machine tool vibrations, In: *Proceedings of the 1997 ASME Design Engineering Technical Conferences, 16th ASME Biennial Conference on Mechanical Vibration and Noise (Sacramento, 1997)*, DETC97/VIB-4021, pp. 1–11 (1997)
 11. Kalmar-Nagy, T., Stepan, G., Moon, F.C.: Subcritical Hopf bifurcation in the delay equation model for machine tool vibrations. *Nonlinear Dyn.* **26**, 121–142 (2001)
 12. Rusinek, R., Weremczuk, A., Warminski, J.: Regenerative model of cutting process with nonlinear duffing oscillator. *Mech. Mech. Eng.* **15**, 129–143 (2011)
 13. Erturk, A., Inman, D.J.: On mechanical modeling of cantilevered piezoelectric vibration energy harvesters. *J. Intell. Mater. Syst. Struct.* **19**, 1311–1325 (2008)
 14. Erturk, A., Inman, D.J.: An experimentally validated bimorph c antilever model for piezoelectric energy harvesting from base excitations. *Smart Mater. Struct.* **18**, 025009 (2009)
 15. Abdelkefi, A., Nayfeh, A.H., Hajj, M.R.: Modeling and analysis of piezoaeroelastic energy harvesters. *Nonlinear Dyn.* **67**, 925–939 (2011)
 16. Bibo, A., Alhadidi, A.H., Daqaq, M.F.: Exploiting a nonlinear restoring force to improve the performance of flow energy harvesters. *J. Appl. Phys.* **117**, 045103 (2015)
 17. Nayfeh, A.H., Mook, D.T.: *Nonlinear Oscillations*. Wiley, New York (1979)
 18. Shampine, L.F., Thompson, S.: Solving delay differential equations with dde23. PDF available on-line at <http://www.radford.edu/~thompson/webddes/tutorial.pdf> (2000)

Identification of Some Spatial Coefficients in Some Engineering Topics



A. Badran

Abstract There are many physical and engineering problems which can be reduced to the moment problems. Some of these problems are the identification of the source term in some differential equation. In tomography, the unknown attenuation coefficient of the tissue can be determined by solving the related moment problem, and also in synthetic-aperture radar (SAR) imaging, the reflectivity function can be identified by solving the related moment problem. In this paper, we present a method, which gives a basis that acquires the same a priori information of the unknown function (signal). Many applications in differential equations, tomography and (SAR) are presented.

Keywords Differential equations · Moment problem · Backus and Gilbert method · Tomography · Synthetic-aperture radar

1 Introduction

The classical applied mathematics consist mainly of problems, in which a given set of inputs is mapped continuously to a unique output and the problem is solved when the unique output is computed, exactly or approximately, from the inputs. Problems of this type come to be known as direct or forward problems. Relatively recently, a new type of problem has arisen in applied mathematics. These problems seek to determine some of the input from observations of the corresponding output and are referred to as inverse problems. Inverse problems for differential equations [1–3] are a topic of increasing interest not just to mathematicians but to scientists and engineers as well. The method is based on an old idea that goes back to Backus and Gilbert [4], who applied it in connection with problem arising in Geophysics. Maas and Louis [5] provided a mathematical context which can be summarized as follows. Suppose N measurements $(\mu_1, \mu_2, \dots, \mu_N)$ are available which represent $\mu_i = \langle f, \psi_i \rangle := \int_0^1 f(x) \cdot \psi_i(x) dx$ on the unknown function f . Then

A. Badran (✉)

Department of Mathematics, Faculty of Science, Damietta University, Damietta, Egypt
e-mail: Badran@du.edu.eg

$$f(s) \approx f_N(s) = (\vec{a}, \vec{\mu}) := \sum_{i=1}^N a_i(s) \cdot \mu_i,$$

where

$$\begin{aligned} \vec{a}(s) &= \frac{G^{-1}\vec{\sigma}}{\langle G^{-1}\vec{\sigma}, \vec{\sigma} \rangle}, \\ \vec{\sigma} &= [\langle 1, \psi_1 \rangle_{L^2}, \dots, \langle 1, \psi_N \rangle_{L^2}]^T, G = [\langle \psi_i, \psi_j \rangle_s]; \langle \psi_i, \psi_j \rangle_s \\ &:= \int_0^1 \psi_i(x)\psi_j(x) \cdot (s-x)^2 dx \end{aligned}$$

There are many problems which can be reduced to the moment problems. In differential equations [6], assume an unknown function f represents the source in some given Dirichlet boundary value problem (GDBVP). The (direct) solution “ u ” of this (GDBVP) is given by $u(x) = \int_{\Omega} f(\xi) \cdot G(x, \xi) \cdot d\xi$, where $G(x, \xi)$ is the Green function corresponding to the (GDBVP). Since the flux (μ) at the boundary ($\partial\Omega$) can be measured, a moment problem in the unknown function f can be deduced as follows.

$$\frac{\partial u}{\partial n} \Big|_{\partial\Omega} = \int_{\Omega} \left(\frac{\partial G}{\partial n} \Big|_{\partial\Omega} \right) f(\xi) \cdot d\xi, n \text{ is the normal direction to the boundary.}$$

In tomography [7–9], assume the unknown attenuation coefficient of the tissue at $x \in \Omega \subset \mathbb{R}^2$ is $f \in H_0^m(\Omega)$, since $\int_{L_{ij}} f(x) dx$ can be measured, say μ_{ij} where the thin disjoint (for i) strips L_{ij} are around the lines $\tilde{L}_{ij} = \{x \in \Omega, x \cdot \theta_i = s_j; \theta_i \equiv (\cos \varphi_i, \sin \varphi_i)\}$.

Its moment problem can be stated as follows.

$$\int_{\Omega} f(x) \cdot \chi_{ij}(x) dx = \mu_{ij}, \text{ where the sampling functions are:}$$

$$\chi_{ij}(x) = \begin{cases} 1 & , x \in L_{ij} \cap \Omega \\ 0 & , x \notin L_{ij} \cap \Omega \end{cases}$$

In synthetic-aperture radar (SAR) [10–12], (SAR) image is the spatial (and frequency) distribution of the reflectivity function $f(w, x)$ corresponding to the target. $f(w, x)$ is a part in the source term of the Helmholtz differential equation, after using the Born approximation:

$$(\nabla^2 + k^2)E^{sc} = -f(\omega, x) \cdot E^{in}$$

where $E^{sc}(w, x)$ is the electric field due to target scattering which can be measured at the antenna, and $E^{in}(\omega, x)$ is the well-known incident field. So we get the moment problem:

$$\mu_{ij} \equiv E_{x_i}^{sc}(\omega_j) = \int \psi_{x_i}(\omega_j, y) \cdot f(\omega_j, y) dy;$$

$$\psi_{x_i}(\omega_j, y) \equiv \frac{e^{ik|x_i-y|}}{4\pi|x_i-y|} E^{\text{in}}(\omega_j, y) ; i = 1, 2, \dots, N; j = 1, 2, \dots, M$$

In the next section, we introduce a new method to deal with and to solve these kinds of the moment problems.

2 Section (1)

For $m \geq 0$, consider the operator

$$A : H_0^m(\Omega) \rightarrow \mathbb{R}^N$$

$$v \rightarrow (\langle v, g_1 \rangle, \langle v, g_2 \rangle, \dots, \langle v, g_N \rangle)$$

where $H_0^m(\Omega)$, according to the trace theorem [13, 14], is the space of functions, in the Sobolev space $(H^m(\Omega), \langle \langle \cdot, \cdot \rangle \rangle_m)$, all of whose derivatives of order inferior to $m - 1$ vanish on the boundary of the bounded open set $\Omega \subseteq \mathbb{R}^N$. While $\{g_i\}_{i=1}^N \subseteq (H^{-m}(\Omega), \langle \langle \cdot, \cdot \rangle \rangle_m)$, the dual of $H_0^m(\Omega)$.

$$\langle \langle f_1, f_2 \rangle \rangle_m := \sum_{p=0}^m (D^p f_1, D^p f_2) = \sum_{p=0}^m \int_{\Omega} [D^p f_1][D^p f_2]$$

Now, consider $\langle v, g_i \rangle := \int_{\Omega} v \cdot g_i ; i = 1, \dots, N$ for given $\{g_i\}_{i=1}^N \subseteq H^{-m}(\Omega)$, our goal is to find $v \in H_0^m$ such that $A(v) = \vec{\mu} \equiv (\mu_1, \mu_2, \dots, \mu_N)$. Assume that the function v at a particular point z_j is defined by

$$v(z_j) = \sum_{i=1}^N \mu_i \Phi_i(z_j) \quad (*)$$

$$\Rightarrow v(z_j) = \sum_{i=1}^N \mu_i \Phi_i(z_j) = \int \left[\sum_{i=1}^N \Phi_i(z_j) g_i(y) V(y) \right] dy$$

$$\Rightarrow v(z_j) = \int K(z_j; y) \cdot V(y) dy$$

; $K(z_j; y) = \sum_{i=1}^N \Phi_i(z_j) g_i(y)$ is the point-spread function.

Our goal is to find the unknowns $\Phi_i(z)$ such that; $K(z_j; y) \approx \delta(y - z_j) \in H^{-m}(\Omega)$.

Since

$$A^T : \mathbb{R}^N \rightarrow H^{-m}(\Omega)$$

$$\vec{b} \rightarrow \sum_{i=1}^N b_i g_i$$

So,

$$\begin{aligned} A^T(\overrightarrow{\Phi(z_j)}) &= \delta(y - z_j) \\ \Rightarrow (A^T)^* A^T(\overrightarrow{\Phi(z_j)}) &= (A^T)^* \delta(y - z_j) \end{aligned} \quad (1)$$

where

$(A^T)^* : H^{-m}(\Omega) \rightarrow \mathbb{R}^N$; the adjoint of A^T is defined by

$$\langle A^T \vec{b}, F \rangle_{H^{-s}} = (\vec{b}, (A^T)^* F)_{\mathbb{R}^N}$$

which implies that

$$(A^T)^* F = (\langle g_1, F \rangle_{H^{-m}}, \dots, \langle g_N, F \rangle_{H^{-m}}) \in \mathbb{R}^N$$

The left-hand side of (1) is given by

$$\begin{aligned} (A^T)^* A^T(\overrightarrow{\Phi(z_j)}) &= (A^T)^* \left[\sum_{i=1}^N \Phi_i(z_j) g_i \right] = \sum_{i=1}^N \Phi_i(z_j) (A^T)^* g_i \\ &= \sum_{i=1}^N \Phi_i(z_j) \cdot (\langle g_1, g_i \rangle_{H^{-m}}, \langle g_2, g_i \rangle_{H^{-m}}, \dots, \langle g_N, g_i \rangle_{H^{-m}}) \\ &= \sum_{i=1}^N \Phi_i(z_j) \cdot (\langle J_m^{-1} g_1, g_i \rangle_{H_0^m \times H^{-m}}, \dots, \langle J_m^{-1} g_N, g_i \rangle_{H_0^m \times H^{-m}}) \end{aligned}$$

while the right-hand side of (1) is given by

$$\begin{aligned} (A^T)^* \delta(y - z_j) &= (\langle g_1, \delta_{z_j} \rangle_{H^{-m}}, \dots, \langle g_N, \delta_{z_j} \rangle_{H^{-m}}) \\ &= (\langle J_m^{-1} g_1, \delta_{z_j} \rangle_{H_0^m \times H^{-m}}, \dots, \langle J_m^{-1} g_N, \delta_{z_j} \rangle_{H_0^m \times H^{-m}}) \end{aligned}$$

where $J_m : H_0^m(\Omega) \rightarrow H^{-m}(\Omega)$ denotes the duality operator $J_m f := \sum_{j=0}^m (-1)^j D^{2j} f$; $D := \frac{d}{dx}$, $f \in H_0^m$

So, $J_m^{-1} : H^{-m} \rightarrow H_0^m$ is defined by: $J_m^{-1}(g_i) = f_i$, where f_i is the unique solution of the boundary value problem:

$$\sum_{p=0}^m (-1)^p D^{2p} f_i = g_i,$$

$$f_i^{(p)} \Big|_{\partial\Omega} = 0; p = 0, 1, \dots, m - 1$$

Then, (1) can be stated as follows:

$$\begin{aligned} \sum_{i=1}^N \Phi_i(z_j) \cdot (\langle J_m^{-1} g_1, g_i \rangle_{H_0^m \times H^{-m}}, \dots, \langle J_m^{-1} g_N, g_i \rangle_{H_0^m \times H^{-m}}) \\ = (\langle J_m^{-1} g_1, \delta_{z_j} \rangle_{H_0^m \times H^{-m}}, \dots, \langle J_m^{-1} g_N, \delta_{z_j} \rangle_{H_0^m \times H^{-m}}) \end{aligned} \tag{2}$$

Then

$$\begin{pmatrix} \Phi_1(z_j) \\ \Phi_2(z_j) \\ \vdots \\ \Phi_N(z_j) \end{pmatrix} = \begin{pmatrix} \langle J_m^{-1} g_1, g_1 \rangle & \langle J_m^{-1} g_1, g_2 \rangle & \dots & \langle J_m^{-1} g_1, g_N \rangle \\ \langle J_m^{-1} g_2, g_1 \rangle & \langle J_m^{-1} g_2, g_2 \rangle & \dots & \langle J_m^{-1} g_2, g_N \rangle \\ \vdots & \vdots & \vdots & \vdots \\ \langle J_m^{-1} g_N, g_1 \rangle & \langle J_m^{-1} g_N, g_2 \rangle & \dots & \langle J_m^{-1} g_N, g_N \rangle \end{pmatrix}^{-1} \begin{pmatrix} \langle J_m^{-1} g_1, \delta(z_j) \rangle \\ \langle J_m^{-1} g_2, \delta(z_j) \rangle \\ \vdots \\ \langle J_m^{-1} g_N, \delta(z_j) \rangle \end{pmatrix}$$

Remarks: (1) For computer use, if the functions are defined by sequences, $v := (v(1), v(2), \dots, v(N))$, and if we choose $g_i = \delta(i), i = 1, 2, \dots, N$, then $\Phi_i(j) = 0 \forall i \neq j, \Phi_j(j) = 1$.

(2) If $m \geq k$, then

$$H_0^m(\Omega) \subset H_0^k(\Omega) \subset H_0^1(\Omega) \subset L^2(\Omega) \subset H^{-1}(\Omega) \subset H^{-k}(\Omega) \subset H^{-m}(\Omega)$$

$$\begin{pmatrix} \Phi_1(z_j) \\ \Phi_2(z_j) \\ \vdots \\ \Phi_N(z_j) \end{pmatrix} = \begin{pmatrix} \langle f_1, g_1 \rangle & \langle f_1, g_2 \rangle & \dots & \langle f_1, g_N \rangle \\ \langle f_2, g_1 \rangle & \langle f_2, g_2 \rangle & \dots & \langle f_2, g_N \rangle \\ \vdots & \vdots & \vdots & \vdots \\ \langle f_N, g_1 \rangle & \langle f_N, g_2 \rangle & \dots & \langle f_N, g_N \rangle \end{pmatrix}^{-1} \begin{pmatrix} \langle f_1, \delta(z_j) \rangle \\ \langle f_2, \delta(z_j) \rangle \\ \vdots \\ \langle f_N, \delta(z_j) \rangle \end{pmatrix}$$

$$\langle f_i, g_j \rangle = \int_{\Omega} f_i g_j; \langle f_i, \delta(z_j) \rangle = f_i(z_j); i, j = 1, 2, \dots, N$$

Finally,

$$v(z_j) = \mu_1 \Phi_1(z_j) + \mu_2 \Phi_2(z_j) + \dots + \mu_N \Phi_N(z_j); z_j \in \Omega$$

For $m = 1, \Omega := (0, 1)$,

$$f_i(x) = \int_0^1 G(x, \xi) g_i(\xi) d\xi;$$

where $G(x, \xi)$ is the Green function of this system which is defined by

$$G(x, \xi) = \begin{cases} \sinh(\xi) \sinh(1 - x); & 0 < \xi < x \\ \sinh(1 - \xi) \sinh(x); & x < \xi < 1 \end{cases}$$

$$\Rightarrow f_i(x) = \sinh(1 - x)G_{i1}(x) + \sinh(x)G_{i2}(x)$$

where

$$G_{i1}(x) = \int_0^x \sinh(\xi)g_i(\xi)d\xi; \quad G_{i2}(x) = \int_x^1 \sinh(1 - \xi)g_i(\xi)d\xi$$

Then

$$\langle f_i, g_j \rangle = \int_0^1 [\sinh(1 - x)G_{i1}(x) + \sinh(x)G_{i2}(x)]g_j(x)dx,$$

$$\langle f_i, \delta(z_j) \rangle = f_i(z_j)$$

If the grid points of the domain (0, 1) are $0 = x_1, x_2, \dots, x_J = 1$, then. $\Phi_{N \times J} = M_{N \times N}^{-1} F_{N \times J}$, where

$$\Phi_{N \times J} := \begin{bmatrix} \Phi_1(x_1) & \Phi_1(x_2) & \dots & \dots & \Phi_1(x_J) \\ \Phi_2(x_1) & \Phi_2(x_2) & \dots & \dots & \Phi_2(x_J) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \Phi_N(x_1) & \Phi_N(x_2) & \dots & \dots & \Phi_N(x_J) \end{bmatrix}$$

$$M_{N \times N}^{-1} = \left(\begin{matrix} \langle f_1, g_1 \rangle & \langle f_1, g_2 \rangle & \dots & \langle f_1, g_N \rangle \\ \langle f_2, g_1 \rangle & \langle f_2, g_2 \rangle & \dots & \langle f_2, g_N \rangle \\ \vdots & \vdots & \vdots & \vdots \\ \langle f_N, g_1 \rangle & \langle f_N, g_2 \rangle & \dots & \langle f_N, g_N \rangle \end{matrix} \right)^{-1}$$

$$F_{N \times J} := \begin{bmatrix} f_1(x_1) & f_1(x_2) & \dots & \dots & f_1(x_J) \\ f_2(x_1) & f_2(x_2) & \dots & \dots & f_2(x_J) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ f_N(x_1) & f_N(x_2) & \dots & \dots & f_N(x_J) \end{bmatrix}$$

Finally, our unknown function (sequence) is given by

$$[v(x_1) \ v(x_2), \dots, v(x_J)]_{1 \times J} = [\mu_1 \ \mu_2, \dots, \mu_N]_{1 \times N} \cdot \Phi_{N \times J}$$

Remarks:

1. A is a continuous linear operator

2. $\{g_i\}_{i=1}^N$ must be independent, not only for independent data, but also to guarantee the matrix $M_{N \times N}$ to be nonsingular. $M_{N \times N}$ is also symmetric.
3. $\langle \Phi_i, g_j \rangle = 0 \forall i \neq j, \langle \Phi_i, g_i \rangle = 1 \forall i, j = 1, 2, \dots, N$
4. The method described here exploits the approach in a way that is different from either of these earlier works and is also different from other current approaches to inverse problems in this class. A particularly attractive feature of this approach is the fact that in the identification of spatially variable physical properties, refining the spatial accuracy of the reconstruction can be carried out in a simple and efficient manner by using suitable bases.

Algorithm

1. Make a partition to the interval
2. $0.h = x_1, 1.h = x_2, 2h = x_3, \dots, jh = x_{(j+1)}, \dots, (J - 1)h = x_J = 1; h = 1/(J - 1)$
3. Consider $\{g_i\}_{i=1}^N \subset H^{-1}(0, 1)$; assume the unknown function $v \in H_0^1(0, 1)$
4. Measure $\{\mu_i\}_{i=1}^N$
5. Calculate $G_{i1}(x_k); G_{i2}(x_k); i = 1 : N, k = 1 : J$
6. Evaluate $f_i(x_k); i = 1 : N; k = 1 : J$, obtain the matrix $F_{N \times J}$
7. Get $M_{N \times N}; M_{N \times N}^{-1}$
8. Calculate the matrix $\Phi_{N \times J}$
9. Reconstruct the unknown function $v \equiv [v(x_1) v(x_2) \dots v(x_J)]$

Example (1): To get the ‘unknown’ function $\sinh(x). \sinh(x - 1)$.

The following is MATLAB program (for any integer number N) (Fig. 1).

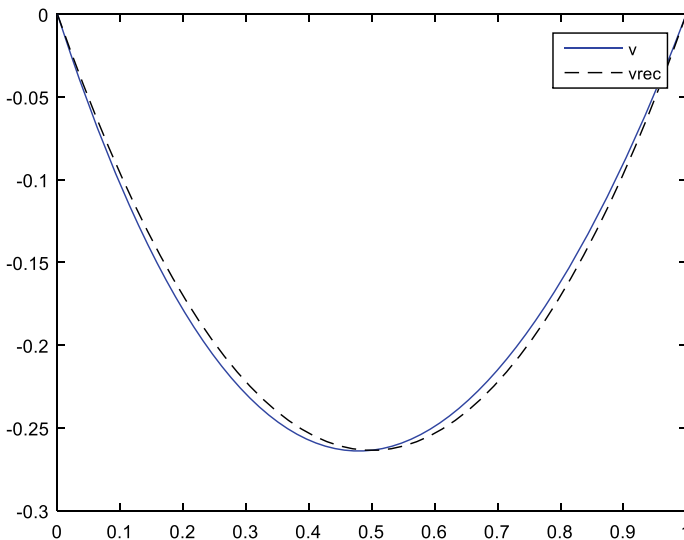


Fig. 1 Graph of the original function $v(x) = \sinh x. \sinh(x - 1)$ and its reconstructed

```

% for any value of N, say N=1
N=1
J=50
h=1/(J-1)
for ii=1:J
    x(ii)=(ii-1)*h;
    % for any unknown function , say

    v(ii)=sinh(x(ii))*sinh((x(ii)-1));

end
for ss=1:N
    for ii=1:J/ss
        g(ss,ii)=1
    end
end
for ii=1:J
    for gg=1:N
        vg(gg,ii)=v(ii)*g(gg,ii);
    end
end
for gg=1:N
    s1(gg)=0;
end
for ii=2:J
    for gg=1:N
        s1(gg)=s1(gg)+vg(gg,ii);
    end
end
for gg=1:N
    Ivg(gg)=(h/2)*(vg(gg,1)+vg(gg,J)+2*s1(gg));
end
for gg=1:N
    Mu(gg)=Ivg(gg)
end

for kk=1:J
    for gg=1:N
        sg1(gg,kk)=sinh(x(kk))*g(gg,kk)
        sg2(gg,kk)=sinh(1-x(kk))*g(gg,kk)
    end
end
for kk=1:J
    for gg=1:N
        s2(gg,kk)=0
        s3(gg,kk)=0

    end
end

```

```

    for kk=1:J
    for ii=2:kk
        for gg=1:N
            s2(gg, kk)=s2(gg, kk)+sg1(gg, ii)

        end
    end
end
    for kk=1:J
        for gg=1:N
G1(gg, kk)=(h/2) * (sg1(gg, 1)+sg1(gg, kk)+2*s2(gg, kk));
        end
    end

    for kk=1:J
        for ii=kk:J
            for gg=1:N
                s3(gg, kk)=s3(gg, kk)+sg2(gg, ii)
            end
        end
    end
    for kk=1:J
        for gg=1:N
G2(gg, kk)=(h/2) * (sg2(gg, kk)+sg2(gg, J)+2*s3(gg, kk));
        end
    end

    for ii=1:J
        for gg=1:N

            f(gg, ii)=sinh(1-x(ii))*G1(gg, ii)+sinh(x(ii))*G2(gg, ii)
        end
    end

    for ii=1:J
        for gg=1:N
            for jj=1:N

                fg(jj, gg, ii)=f(gg, ii)*g(jj, ii)

            end
        end
    end
    for gg=1:N
        for jj=1:N
s4(jj, gg)=0
        end
    end
end

```

```

for ii=2:J
    for gg=1:N
        for jj=1:N
            s4(jj,gg)=s4(jj,gg)+fg(jj,gg,ii)
        end
    end
end
for gg=1:N
    for jj=1:N
        Ifg(jj,gg)=(h/2)*(fg(jj,gg,1)+fg(jj,gg,J)+2*s4(jj,gg))
    end
end

for gg=1:N
    for jj=1:N

M(jj,gg)=Ifg(jj,gg)
    end
end
IM=inv(M)
FI=IM*f

vrec=Mu*FI
plot(x,v,'b-');
hold on;
plot(x,vrec,'k--');
hold off;
legend('v','vrec');

```

In the next section, we apply the previous method to the inverse problems in the differential equations, tomography and (SAR).

3 Section (II)

First, we present, via some examples, the abstract formulation of inverse problems in differential equations and the use of the abstract Green formula to obtain integral identities relating changes in inputs to change in outputs.

1. Example (1): Consider:

$$\left. \begin{aligned} -\frac{d^2 u}{dx^2} &= v(x), 0 < x < 1, \\ u(0) &= 0, u(1) = 0 \end{aligned} \right\}$$

where $v(x)$ is the source density. It is known that the (direct) solution is given

$$\text{by } u(x) = (1-x) \int_0^x y v(y) dy + x \int_x^1 (1-y) v(y) dy$$

So

$$u'(x) = (1 - x)[xv(x)] - \int_0^x yv(y)dy - x(1 - x)v(x) + \int_x^1 (1 - y)v(y)dy$$

Then, the corresponding moment problem is:

$$\mu_1 = u'(0) = \int_0^1 (1 - x)v(x)dx,$$

$$\mu_2 = -u'(1) = \int_0^1 x v(x)dx$$

This example can be stated, by using the Green function as follows: Consider

$$\left. \begin{aligned} -\frac{d^2u}{dx^2} &= f(x), 0 < x < 1; \\ u(0) &= 0, u(1) = 0 \end{aligned} \right\}$$

The Green function is:

$$G(x, \xi) = \begin{cases} (1 - \xi)x & , 0 \leq x < \xi \\ (1 - x)\xi & , \xi < x \leq 1 \end{cases}$$

Then, the moment problem is:

$$u'(0) = \int_0^1 \left(\frac{dG}{dx} \Big|_{x=0} \right) f(\xi)d\xi = \int_0^1 (1 - x)f(x)dx$$

$$-u'(1) = \int_0^1 \left(-\frac{dG}{dx} \Big|_{x=1} \right) f(\xi)d\xi = \int_0^1 x f(x)dx$$

So, the sampling functions are:

$$g_1(x) = 1 - x, g_2(x) = x$$

Solve the (FDBVP):

$$\left. \begin{aligned} -\frac{d^2 f_1}{dx^2} + f_1 &= 1 - x, 0 < x < 1; \\ f_1(0) = 0, f_1(1) &= 0 \end{aligned} \right\}$$

Then $f_1(x) = \left(\frac{-1}{e^2-1}\right)e^x + \left(\frac{e^2}{e^2-1}\right)e^{-x} + 1 - x$
 Also, solve the (FDBVP):

$$\left. \begin{aligned} -\frac{d^2 f_2}{dx^2} + f_2 &= x, 0 < x < 1; \\ f_2(0) = 1, f_2(1) &= 0 \end{aligned} \right\}$$

Then, $f_2(x) = \left(\frac{e}{e^2-1}\right)e^x - \left(\frac{e}{e^2-1}\right)e^{-x} + x$.

Figure 2 shows the graph of the original function $f(x) = x(1 - x)$ and the graph of the reconstructed function (for $(\mu_1, \mu_2) = (0.0833, 0.0833)$).

2. Example (2): Consider

$$\left. \begin{aligned} -\frac{d^2 u}{dx^2} + u &= f(x), 0 < x < 1; \\ u(0) = 0, u(1) &= 0 \end{aligned} \right\}$$

The Green function is given by:

$$G(x, \xi) = \begin{cases} \sinh(1 - \xi) \cdot \sinh(x) & , 0 \leq x \leq \xi \\ \sinh(\xi) \sinh(1 - x) & , \xi \leq x \leq 1 \end{cases}$$

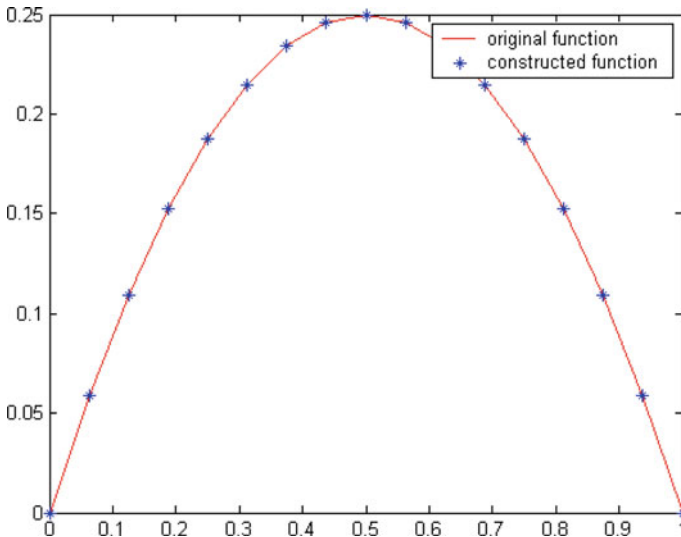


Fig. 2 Graph of the original function $f(x) = x(1 - x)$ and its reconstructed

The moment problem is:

$$\mu_1 \equiv u'(0) = \int_0^1 \left(\frac{dG}{dx} \Big|_{x=0} \right) \cdot f(\xi) d\xi = \int_0^1 (\sinh(1-x)) f(x) dx$$

$$\mu_2 \equiv -u'(1) = \int_0^1 \left(-\frac{dG}{dx} \Big|_{x=1} \right) f(\xi) d\xi = \int_0^1 \sinh(x) f(x) dx$$

So, the sampling functions are:

$$g_1(x) = \sinh(1-x), g_2(x) = \sinh x$$

By solving the (DBVP):

$$\left. \begin{aligned} -\frac{d^2 f_1}{dx^2} + f_1 = \sinh(1-x), 0 < x < 1; f_1(0) = 0, f_1(1) = 0 \end{aligned} \right\}$$

We get

$$f_1(x) = A_1 e^x - A_1 e^{-x} + \frac{e}{4} x e^x + \frac{1}{4e} x e^{-x}, \text{ where}$$

$$A_1 = \frac{-1 - e^4}{4e(e^2 - 1)}$$

Also, solve the (DBVP)

$$\left. \begin{aligned} -\frac{d^2 f_2}{dx^2} + f_2 = \sinh(x), 0 < x < 1; f_2(0) = 0, f_2(1) = 0 \end{aligned} \right\}$$

To get

$$f_2(x) = \frac{\coth(1)}{2} \sinh x - \frac{1}{2} x \cosh x$$

Figure 3 shows the graph of the original function $f(x) = x^2(1-x)$ and the graph of the reconstructed function (for $(\mu_1, \mu_2) = (0.0298, 0.1709)$).

Second, the method can be applied also in tomography [15]; assume the unknown attenuation coefficient of the tissue at $x \in \Omega \subset \mathbb{R}^2$ is $v \in H_0^m(\Omega)$, since $\int_{L_{ij}} v(x) dx$ can be measured, say μ_{ij} where the thin disjoint (for i) strips L_{ij} are around the lines $\tilde{L}_{ij} = \{x \in \Omega, x \cdot \theta_i = s_j; \theta_i \equiv (\cos \varphi_i, \sin \varphi_i)\}$.

Its moment problem can be stated as follows:

$$\int_{\Omega} v(x) \cdot \chi_{ij}(x) dx = \mu_{ij}, \text{ where the sampling functions are}$$

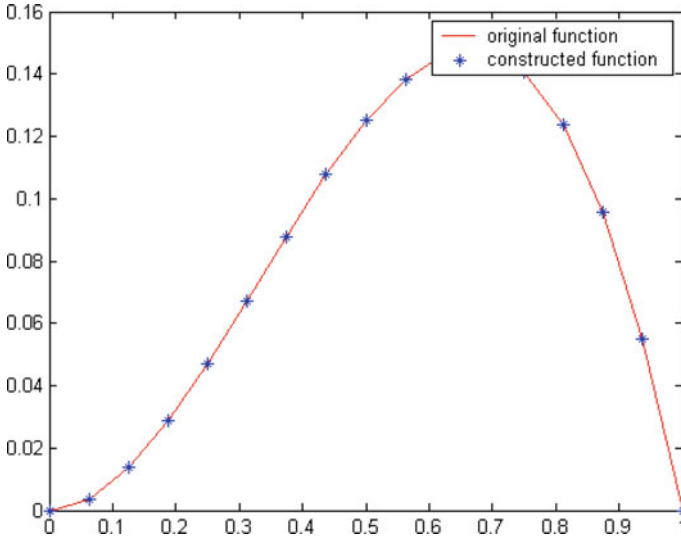


Fig. 3 Graph of the original function $f(x) = x^2(1 - x)$ and its reconstructed

$$\chi_{ij}(x) = \begin{cases} 1, & x \in L_{ij} \cap \Omega \\ 0, & x \notin L_{ij} \cap \Omega \end{cases}$$

So, we can solve the (FDBVP):

$$\sum_{k=0}^m (-1)^k D^{2k} f_{ij} = \chi_{ij}$$

$$f_{ij}^{(k)} \Big|_{\partial\Omega} = 0; k = 0, 1, \dots, m - 1$$

To get $f_{ij}(x)$, and to apply (2) in order to construct the attenuation coefficient approximation. Assume some unknown function $v \in H_0^1(0, 1)$ satisfies $\int_0^1 v(x)dx = \mu_1$ and $\int_0^{0.5} v(x)dx = \mu_2$. This means that its moment problem is $\int_0^1 v(x) \cdot \chi_{[0,1]} dx = \mu_1$ and $\int_0^1 v(x) \cdot \chi_{[0,0.5]} dx = \mu_1$

So, solve the {FDBVP}

$$\left. \begin{aligned} -\frac{d^2 f_1}{dx^2} + f_1 &= \chi_{[0,1]}, 0 < x < 1; \\ f_1(0) &= 0, f_1(1) = 0 \end{aligned} \right\}$$

So,

$$g_1(x) = \int_0^1 G(x, \xi).1.d\xi ; \quad g_2(x) = \int_0^{0.5} G(x, \xi).1d\xi .$$

Assume $f_j(y) \equiv f(\omega_j, y) \in H_0^m(\Omega) ; j = 1, 2, \dots, M$, we can now apply the previous procedure to get the reconstructed attenuation coefficient.

Finally, in synthetic-aperture radar, as we mentioned before (SAR) image is the spatial (and frequency) distribution of the reflectivity function $v(w, x)$ corresponding to the target. $v(w, x)$ is a part in the source term of the Helmholtz differential equation, after using the Born approximation: $(\nabla^2 + k^2)E^{sc} = -v(\omega, x).E^{in}$, where $E^{sc}(w, x)$ is the electric field due to target scattering which can be measured at the antenna, and $E^{in}(\omega, x)$ is the incident field. So we get the moment problem:

$$\mu_{ij} \equiv E_{x_i}^{sc}(\omega_j) = \int \psi_{x_i}(\omega_j, y).v(\omega_j, y)dy ;$$

$$\psi_{x_i}(\omega_j, y) \equiv \frac{e^{ik|x_i-y|}}{4\pi|x_i-y|} E^{in}(\omega_j, y) ; i = 1, 2, \dots, N ; j = 1, 2, \dots, M$$

Assume $v_j(y) \equiv v(\omega_j, y) \in H_0^m(\Omega) ; j = 1, 2, \dots, M$, we can now apply the previous procedure to get the reconstructed reflectivity functions for each frequency.

4 Section (3): Conclusion

In this work, we developed the Backus–Gilbert technique to identify the unknown function from its finite number of moments. In fact, the enclosed method is accurate when the unknown function belongs to the specific spaces H_0^m . MATLAB programs and related algorithms are enclosed. In the future work, we will try to extend this method to deal with H^m spaces.

References

1. Groetsch, C.W.: Inverse Problems in the Mathematical Sciences. Vieweg, Braunschweig, Wiesbaden (1993)
2. Colton, D., Ewing, R., Rundell, W.: Inverse Problems in Partial Differential Equations. SIAM, Philandadelphia, PA (1990)
3. DuChateau, P., Badran, A.: Identification of spatially dependent coefficients in partial differential equations. In: Dynamics of Continuous, Discrete and Impulsive Systems, Series A, Mathematical Analysis, vol. 10a Issue 5 (2003)
4. Backus, G., Gilbert, F.:The resolving power of gross earth data , Geophys. J. R. Astr. Soc. **16**, 169–205 (1968)
5. Louis, A.K., Maass, P.: Smoothed projection method for the moment problem. Numer. Math. **59**, 277–294 (1991)
6. Evans, L.C.: Partial Differential Equations (1999)

7. Natterer, F.: *The Mathematics of Computerized Tomography*, vol 4, Wiley (1986)
8. Natterer, F., Faridani: *Basic Algorithms in Tomography: Signal Processing Part II: Control Theory and Applications*, pp. 321–334. Springer (1990)
9. Natterer, F.: *The Mathematics of Computerized Tomography*. SIAM, Philadelphia, PA (2001)
10. Bleszynski, E. (ed.): *Autofocus algorithm for synthetic aperture radar imaging with large curvilinear apertures*. *Inverse Prob.* **29** (2013)
11. Cheney M.: *Imaging frequency—Dependent reflectivity from synthetic—Aperture radar*. *Inverse Prob* **29**, 054002 (2013)
12. Gilman, M. (ed.): *Reduction of ionospheric distortion for spaceborne synthetic aperture radar with the help of image registration*. *Inverse Prob* **29**, 054005 (2013)
13. Adams, R.A.: *Sobolev Spaces*. Academic Press , New York (1975)
14. Jean-Pierre, A.: *Applied Functional Analysis*. A wiley-Interscience Series of Text, Monographs, and Tracts (1979)
15. Faridani, A.: *Introduction to the mathematics of computed tomography*. *Inverse Prob* **47** (2003)

Automatic Venue Allocation for Varying Class Sizes Using Scoring and Heuristic Hill-Climbing



Uvir Bhagirathi and Ritesh Ajoodha

Abstract The following paper addresses an approach to the problem of venue allocation. Currently, classes are assigned to venues that are inadequate in terms of size and are inconveniently situated from the respective faculty building of the class. This results in classes from different schools being allocated venues across a university. Therefore, the aim of this research is to design an algorithm to aid appropriate and effective class allocation. In doing so, a Bayesian network was implemented to generate synthetic data, and Bayesian estimation was used to learn the parameters of the network. Thereafter, the network was scored using a greedy hill-climbing approach with random restart to ensure a global optimal solution. The program was tested with various datasets of different sizes, displaying the scalability of the program. This approach was more efficient than a brute force solution to the problem as the system produced the best allocations of venues to classes in time complexity of $O(kn^2)$. It also favors closer venues over venues with slightly better accommodation. Ultimately, this solution will allow universities and other academic institutions to allocate the best venue, in terms of size and distance, to each class, improving the overall functioning of a university or academic institution.

Keywords Scoring function · Heuristic search · Allocation · Bayesian network · Greedy hill-climbing

1 Introduction

Universities attract large amounts of students for which each faculty has to facilitate. Within each faculty, students are registered for various courses under different schools and require appropriate learning environments for each course. While these

U. Bhagirathi (✉) · R. Ajoodha
School of Computer Science and Applied Mathematics, The University of the Witwatersrand,
Johannesburg, South Africa
e-mail: uvir.bhagirathi@wits.ac.za

R. Ajoodha
e-mail: ritesh.ajoodha@wits.ac.za

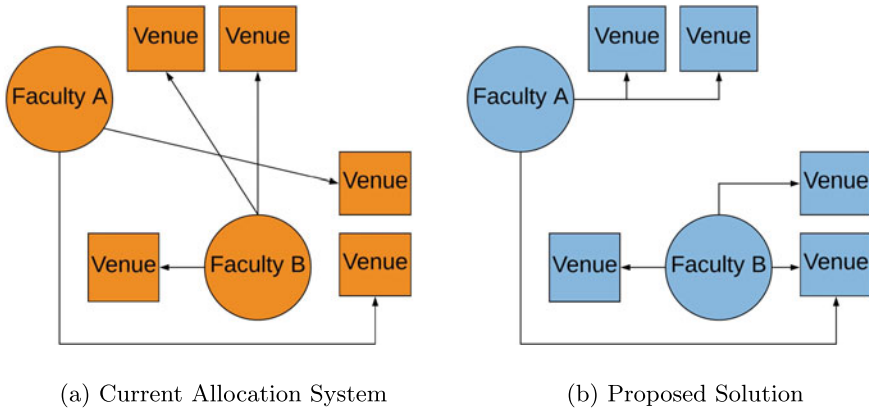


Fig. 1 Graphical representation of how the current allocation system works versus how the proposed system works

universities do offer such environments, there is often difficulty in allocating classes to these venues based on the needs required by different schools. The goal of this research project is to construct a system that allocates venues to classes of various sizes by taking into account the distance between the respective faculty buildings and the venues being used, the capacity of the venue as well as the number of students in the course.

In the current system of class venue allocation, classes are often assigned to venues that are too small or big and far from the courses' faculty building. This results in classes being allocated to venues that are far away from each other. The issue comes about when classes from each faculty is spread across different venues throughout the entire campus (e.g., courses from humanities get allocated venues near the science faculty) when there are suitable venues close to the respective faculty building. This type of allocation leads to an unnecessary waste of time as students and staff have to walk to these distant venues. Figure 1 illustrates how the current allocation system assigns venues to classes versus the proposed systems way of allocating venues.

In solving this problem, a Bayesian network (BN) model, an example of probabilistic models, was used to generate data of classes and venues. Probabilistic models utilize probability distributions as well as random variables in order to model an event. These models can, however, express uncertainty by modeling influence between variables. Variables of this network include attributes of the class and venue (size and type) as well as the distance to the respective faculty. After the data was generated, a greedy hill-climbing algorithm was used to find an optimal network. A network of randomly allocated classes and venues is first created. A scoring function was then used to evaluate the effectiveness of the network model. The score of an allocation corresponds to how well the venue can accommodate the class assigned to it. Afterward, a number of different operators are applied to the network. The algorithm stops once the score of the network does not better the current best score that it has found. This produces an optimal network.

The program developed in this research aims to allocate suitable venues to classes taking into consideration the distance between the venue and the respective class faculty location as well as the size of the venue and the number of students in the class. The system is expected to produce a list of these allocations in at least $O(kn^2)$ time with each venue being best suited for the class that is assigned to it. This system will allow universities and other academic institutions to allocate the best venue to each class in a short amount of time, resulting in better functioning universities and institutions. Overall, this program is the stepping stone in developing a more robust scheduling system in universities and other academia. The following sections of the research will address the background of the research, related work, methodology and results as well as future improvements.

2 Background

In this section BNs, Bayesian estimation and the greedy hill-climbing algorithm will be discussed. Probabilistic graphical models (PGMs), such as BNs, are used in various real world applications. Greedy hill-climbing is a heuristic search algorithm that is used to traverse through the many combinations of allocations in the network. It produces a solution that is not necessarily the optimal solution.

Currently, class timetables are created highly inefficiently. This leads to many problems including overcrowding in classrooms, inadequate equipment to cater for lectures within these venues, overbooking of venues and venues being too distant between consecutive lectures. Teachers have expressed that effective teaching is not possible in overcrowded classrooms [6]. Allocating venues far apart from each other can cause an inconvenience as students will feel the need to rush out of the current class they are in to make it in time for the next class. This can also lead to students arriving late to class and disrupting lectures.

The program that has been developed in this research automates this system. The automated system allocates each course to the most suited venue according to a predetermined scoring function. This system minimizes the amount of extra seats in each venue and the distance between the venue and the respective faculty building of the class assigned to it. This allows for students and staff members to be more efficient when attending lectures as all venues will be close by.

The following subsections address BNs, Bayesian estimation as well as scoring functions.

2.1 Bayesian Network

A Bayesian network (BN) is a probabilistic graphical model in the form of a Directed Acyclic Graph (DAG) where each node is a variable and each edge is the conditional dependence between two variables [7]. BNs map the relationship between events in terms of probability. It shows how the occurrence of certain events influences the

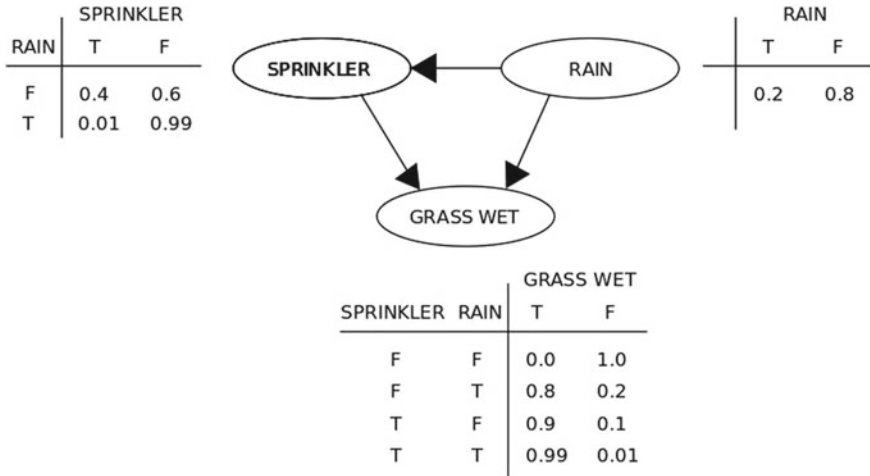


Fig. 2 Graphical representation of the “sprinkler problem”

probability of other events occurring. A BN is used to generate the data. Figure 2 is an example of a BN [9].

It can be seen in Figure 2, each node has a probability of its occurrence and probability of its occurrence depending on the occurrence of other nodes. This kind of diagram is very useful when we have inter-dependent nodes and want to model outcomes and make decisions. In order to make these decisions, the probability of each nodes’ occurrence needs to be known. These probabilities can be estimated using Bayesian estimation.

2.2 Bayesian Estimation

Bayesian estimation (BE) is a parameter learning method which is fairly similar to Maximum Likelihood Estimation (MLE), commonly used learning tool for observable data [1, 2]. BE views any event that has uncertainty as a random variable with a distribution over it [1, 2]. BE uses Bayes’ Theorem and other functions to calculate the parameter values. The Bayes formula is given as:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}, \tag{1}$$

where A and B are events. Bayesian analysis is a convenient setting for many models including hierarchical models. However, it can sometimes get computationally intensive when models involve many variables. Once the parameters of the model have been estimated, an assessment of the model has to be conducted. This can be done using a scoring function.

2.3 *Scoring Function*

A scoring function is used to evaluate the appropriateness of a network. Depending on the rules of the scoring function, each network is allocated a score which shows how well the model fits the observed training data [1, 2].

There are many scoring functions; however, the preferred method that is used is the greedy hill-climbing algorithm.

2.4 *Greedy Hill-Climbing*

The greedy hill-climbing algorithm is used in optimization problems where we need to minimize or maximize a function [10]. It is a heuristic search algorithm, which means that it might not find the optimum solution. However, it will provide a local minimum or maximum in reasonable time. The greedy part of the algorithm implies that it moves in which ever direction optimizes the function. The problem with this algorithm is that it may not always output the optimum network structure but this can be solved using backtracking or random restarts.

3 **Related Work**

Score-based structure learning defines the basis to evaluate how well a BN fits the data, then traverses over the network for a more optimal network which produces a better score. A score-based function is seen as an optimization problem where the produced result is not necessarily the optimal. However, the search space is super-exponential in size so we resort to heuristic search techniques.

A score-based structure is chosen as it allows for viable computations as it preserves basic score properties and indicates the effect of changes in the network structure. In this section, we will discuss the greedy hill-climbing heuristic search algorithm, and the scoring function that is used to score the BN in this research.

3.1 *Heuristic Search*

A heuristic search algorithm is a method that finds a suboptimal solution in reasonable time. It increases efficiency by sacrificing completeness. A classic example is the traveling salesman problem [3]. Greedy hill-climbing is an example of a heuristic search used in optimization problems.

In an optimization problem, we pursue some optimum solution to a network model. Any given state of a network can be a solution. Solutions can be evaluated for

it to be comparable. In greedy hill-climbing, an initial solution is then expanded into multiple neighboring solutions. The best network from those solutions is selected. This process continues until there are no better neighboring solutions. The final solution can be a sub-optimum solution as it is based on the generation of the neighboring networks, and the scoring function used to evaluate those networks.

3.2 *Score-Based Learning*

When a network is generated, there is no way of knowing its worth until it is scored. A scoring function is used to do this. The greedy hill-climbing algorithm is heavily dependent on the scoring of each network to produce a suitable solution. The score of the network indicates how well the network is suited for the task required [8].

4 Methodology

The following section addresses the methods utilized in solving the venue allocation problem. The structure of the BN utilized is also discussed as well as the way in which each node in the model interacts with one another. Additionally, a description of the data in the network will be given.

4.1 *Motivation*

The problem of venue allocation is difficult to solve as checking the compatibility of every class with every venue is an NP-hard problem. For example, assume that there is a class c that has to be allocated to one of the n number of venues available. The class is firstly allocated to this first venue and then checked to see if it meets the necessary requirements. It is then allocated to the second venue and checked, then the third and so on until it is checked against the n th venue. For a single c , this can lead to a complexity of $O(n!)$ which is problematic for a large number of venues and classes where the complexity can be $O(c!n!)$.

Despite this, there are other downfalls to this method. The output of this method does guarantee an optimal solution to this problem; however, it is an NP-hard problem that will take a long time to run. The attempt to attain an adequate venue on short notice is then very time-consuming especially for staff members and students, thus affecting the overall ability of a university to function productively and efficiently. The goal of this research project is to then find a more reasonable approach to solve this problem.

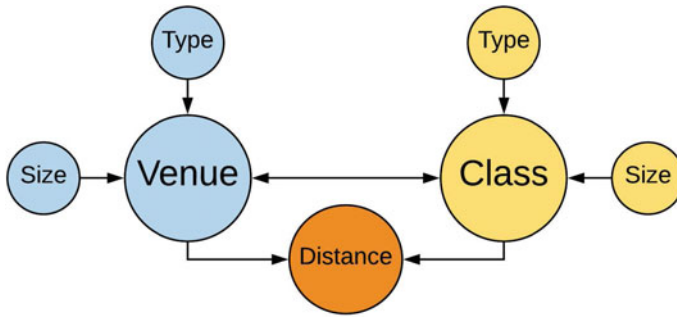


Fig. 3 Representation of the allocation algorithm

In my research, I propose to produce a more feasible approach to the venue allocation problem by using a heuristic search to find a solution. This method can lower the complexity of the problem to at least $O(kn^2)$ which is a major improvement to the original complexity of $O(c!n!)$.

4.2 Network Structure

The program allocates an adequate venue for each lecture depending on the following variables: the size of the class and venue, the type of class and venue and the distance of the venue from the courses' corresponding faculty. This is illustrated in Figure 3 where it can be seen that the venue chosen is dependent on the aforementioned variables. The distance between the class and venue, the size difference between the class and venue as well as the type of them, all influence the allocation score between that venue and class.

There is also a hidden variable, projector, which states if a venue or class has or requires a projector. This variable was added to influence better venue-class allocations. This is the only hidden variable in the program. It was used to test how easily the program can be influence to allocate venues to classes that require a projector. In the future, other equipment and facilities can be added to the program to make sure the correct kind of venues are allocated to classes.

4.3 Data

The variables of the data used to train this network are represented by Table 1.

From Table 1, it can be seen that both the class and venue variables have a size and type attribute as well as a distance matrix. These attributes will be used to evaluate the probability of a venue being used for a class. The course code and venue variables are a form of meta-data that allows distinguish-ability between different classes and venues, respectively.

Table 1 Data representation

Variable	Description	Example
Course code	Identification code used to distinguish between the different classes	C2
Faculty	The faculty that the course belongs to	S
Class size	The number of students enrolled for the course	200
Class type	Classifies the classes into 3 class types: Tut, Lecture, Lab	Lecture
Venue	Name of the Venue	V1
Venue size	The number of students that the venue can accommodate	350
Venue type	Classifies the venue into 3 venue types: Tut, Lecture, Lab	Lecture
Distance	Matrix of distances from each class to each venue	243

4.4 Scoring

The greedy hill-climbing algorithm is a heuristic search algorithm which is used to find a local minimum solution to the problem. Since the algorithm does not produce the optimum solution all the time, random restarts are used to find several solutions, then the best of those solutions is chosen as the final solution.

The greedy hill-climbing algorithm starts with any random network. From this network, it iteratively tries to improve the network by performing different search operators. Greedy hill-climbing always chooses the network which provides the best score until the score cannot be improved any more. Due to the structure of the network, four search operators are used to find a solution to this problem. These search operators are discussed later on.

4.5 Scoring Function

The actual scoring of the class to venue allocation will depend on the following criteria:

- The size of the class and venue
- The type of class and venue
- The attributes of the class and venue
- The distance of the venue from the faculty of the class.

This allows the score to be decomposable resulting in a more efficient algorithm [4]. The scoring function treats each allocation as a minimization problem. A reasonable positive score is given unless any of the following conditions are met. If a class is allocated to a venue that does not have enough seats, then a high positive score is given as the class cannot take place in that venue. However, if the size of the venue allocated surpasses the size of the class, then a low positive score is given

with respect to the ratio between the class and venue size as the class can continue, but there is waste of space that can be utilized by other classes.

A low positive score is given for any class to venue allocation where the types do not match. This is because certain classes may require specific facilities such as computers or work stations and therefore, cannot be allocated to a venue that does not accommodate these features. If the distance of a venue to the faculty of an allocated class is further than a certain threshold, then a low positive score is given. This will ensure that the closest venue is selected when possible. Lastly, if a venue clash occurs then a high positive score is given to that allocation, allowing for more clash-free allocations in the future. The pseudocode in Algorithm 1 illustrates these scoring features.

The best score of an allocation can be 0. This means that the class and venue type match as well as their sizes and the lecture/tut/lab, takes place in its respective faculty building. Algorithm 1 is the pseudocode of how the scoring function in the program works. It can be seen that no negative values are given and the lowest attainable score is, in fact, 0.

Algorithm 1: CalculateScore():

```

sizeDifference = VenueSize - ClassSize
if sizeDifference == 0 then
  | score = 0
else if sizeDifference > 0 then
  | score += sizeDifference * 0.2
else
  | score += -sizeDifference * 5
if VenueType != ClassType then
  | score *= 1.4
  | satisfiedType = False
else
  | satisfiedType = True
if VenueProjector != ClassProjector then
  | score *= 1.1
score += distance**1.2

```

The score of an allocation is highly affected by the distance between the venue and the class' relative faculty building as well as the size difference between the allocated venue and class. This influences the score to favor allocations that efficiently accomodate students and is close to the class' respective faculty building.

4.6 Operators

There are four operators used in the scoring function. Each operator effects a different aspect of the network and involves swapping the class and venues of allocations in order to get a better score.

Random Swaps between lower percent of scores: This offers a sense of randomness to the network which can allow the network to gain a better score. This swap is essential as the venues of the lower scored allocations can be fitting for the classes of the lower scored allocations.

Randomly Swap Highest and Lowest scored allocations: Swapping the venues and classes of the highest and lowest scored allocations allows for randomness in the high and low scored allocations. This can result in a better scoring network as a venue in the higher allocations can better accommodate a lower scored allocation resulting in a better scoring network.

Swap allocations where there is extra spaces with ones that are over booked: This swap allows venues with classes that possess extra space to be swapped with a venue that does not have enough space for the allocated class. This produces a network that better accommodates venues with classes. Here it can be seen that the venue and class type compromised as class venue size accommodation is more valuable.

Swap allocations with venues that distant: This swap tries to allocate a venue that is closer to the class' faculty building. This is done by swapping the venue with another to allocate courses closer to their respective faculties. This allocation has an exponential weight allocated to it as the distance between classes and the respective faculty is highest priority.

4.7 Random Restarts

As mentioned before, the algorithm starts off with randomized allocations of venues and classes and then proceeds to perform operators on those allocations until the network score cannot be bettered. The algorithm produces a sub-optimum solution if not the optimum. To ensure that the best solution is found, the program utilizes random restarts which basically re-runs the program with a new random allocation list at the beginning [5]. These random restarts run a set amount of times. The best solution from all of these runs is the final solution returned by the program.

4.8 Limitations

The closest, most suitable venue may not always be allocated to a class as it may already be occupied. In the situation where all nearby venues are booked and the only venue for this class is far away (on another campus), the algorithm will select this venue as it is the only current and available solution.

4.9 Testing

Correctness: To test if the program produces a suitable allocation list, a small data-set of five elements will be input into the system. This will allow for easy checking if the output is correct, i.e. checking if the best venue is allocated to each class.

Analysis: The program is run on six datasets, each increasing with size. The first dataset contains fifty venues and classes, the second contains a hundred and so on. Each dataset is run on the program with a thousand random restarts. The entire process is timed and also averaged over the one thousand random restarts. The results are recorded and analyzed thereafter.

5 Results

In this section, the results of the testing of the program as well as the future improvements of the system are discussed.

5.1 Program Correctness

The program was first run on a small dataset to examine how well the program performs on allocating suitable venues to classes. A dataset of five venues and classes were used to test this. Table 2 represents the data input into the program. Table 3 represents the best allocation produced from the program. It can be seen from Table 3 that the algorithm allocates the most suitable venue to each class.

It can also be seen (In Table 3 rows 2 and 5) that the algorithm favored the closer venue rather than the one that would accommodate the five more students. This can be easily fixed by adjusting the weights of the distance and size attributes in the scoring function; however, for this research, a closer venue is more favorable.

Table 2 Random allocations input to the system

Venue	Venue size	Course	Course size	Distance	Score
V5	100	C2	245	87	1450
V4	80	C1	250	41	1395
V3	180	C4	80	93	493
V1	250	C3	180	90	464
V2	245	C5	100	36	220

Table 3 Recommended allocations output from the system

Venue	Venue size	Course	Course size	Distance	Score
V4	80	C4	80	82	410
V2	245	C1	250	45	252
V3	180	C3	180	40	200
V5	100	C5	100	27	135
V1	250	C2	245	14	71

5.2 Analysis

The program was run and tested as mentioned previously. The results are graphed in Figure 4. From Figure 4, it can be seen that there is an exponential trend. This was expected as there are more classes and venues with different attributes and sizes as well as distances to take into consideration. This means that there are many possible combinations of classes and venues. This solution of complexity $O(kn^2)$ is a significant improvement over a brute force approach to the problem which is $O(c!n!)$.

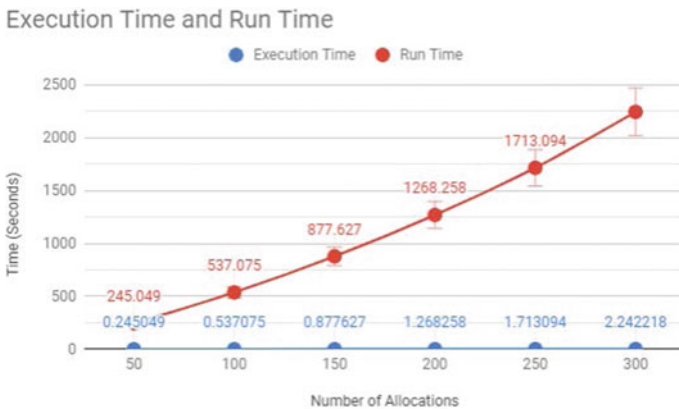


Fig. 4 Two line graphs illustrating the average time of a single execution and the total run time of each run respectively.

Using these results, it can be estimated that the generation of all the best allocations in a university will take the best of a few hours. This is a reasonable amount of time since the output generated will be the most appropriate allocations in terms of distance and size. This result will allow other academic institutions to apply the same program in their own systems, allowing their institution to be working at an optimum rate.

5.3 Future Work

Currently, the system only takes in the distance of the venue to the class' respective faculty and how well the venue can accommodate the class as the main influences in the score calculation. This can be adjusted to take into consideration the class times, if there are night classes and day classes, as well as generate time tables that best suit students and staff members.

6 Conclusion

The aim of this research project was to produce a more reasonable approach to the problem of venue allocation. This was accomplished using a BN to generate data and Bayesian estimation to learn the parameters of the network. A custom scoring function was developed to score the allocation of a venue to a class in terms of distance to its respective faculty building and accommodation level of the venue for that class.

Thereafter, a greedy hill-climbing heuristic search was implemented to traverse the various alternate networks of allocations to find the network with the best allocation score. The developed system efficiently produced ideal lists of allocations, favoring closer venues over venues with better accommodation for its allocated class. This can be adjusted to favor better accommodation if needs be. The algorithm runs in a time complexity of $O(kn^2)$ as predicted. This time complexity is a major improvement over the previous complexity of $O(c!n!)$.

This system can be used by universities and other academia to allocate the best venues to classes in a short amount of time. In the future, the algorithm can be adapted to produce more intuitive time tables for universities and students alike. An alternate solution to the venue allocation problem is to use a graph-coloring algorithm to allocate the best venue to a class.





Acknowledgements This work is based on the research supported in part by the National Research Foundation of South Africa (Grant number: 121835).

References

1. Ajoodha, R., Rosman, B.: Learning the influence between partially observable processes using score-based structure learning. *Structure* **21**(22), 23–24 (2020)
2. Ajoodha, R., Rosman, B.S.: Learning the influence structure between partially observed stochastic processes using Iot sensor data. *AAAI* (2018)
3. Christofides, N.: Worst-case analysis of a new heuristic for the travelling salesman problem. Technical Reports. Carnegie-Mellon University Management Sciences Research Group, Pittsburgh PA (1976)
4. Heckerman, D., Geiger, D., Chickering, D.M.: Learning Bayesian networks: the combination of knowledge and statistical data. *Mach. Learn.* **20**(3), 197–243 (1995)
5. Hu, X., Shonkwiler, R., Spruill, M.C.: *Random Restarts in Global Optimization* (2009)
6. Khan, P., Iqbal, M.: Overcrowded classroom: a serious problem for teachers. *Univ. Sci. Inf. Technol.* **49**, 10162–10165 (2012)
7. Koller, D., Friedman, N.: *Probabilistic Graphical Models: Principles and Techniques*. MIT Press (2009)
8. Munteanu, P., Cau, D.: Efficient score-based learning of equivalence classes of Bayesian networks. In: *European Conference on Principles of Data Mining and Knowledge Discovery*, pp. 96–105. Springer (2000)
9. Pearl, J.: *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Elsevier (2014)
10. Tsamardinos, I., Brown, L.E., Aliferis, C.F.: The max-min hill-climbing Bayesian network structure learning algorithm. *Mach. Learn.* **65**(1), 31–78 (2006)

DWT and Quantization Based Digital Watermarking Scheme Using Kernel OS-ELM



Neeraj Kumar Sharma , Subodh Kumar , Ankit Rajpal ,
and Naveen Kumar 

Abstract In recent years, an extensive amount of digital content is being shared over the Internet. This has increased the copyright, authenticity, and security concerns. Digital watermarking is a well-known mechanism which ensures the digital rights. This paper proposes a discrete wavelet transform (DWT) and quantization-based semi-blind watermarking scheme. The proposed scheme comprises of three phases, namely model generation, watermark embedding, and watermark extraction. The host image is subjected to four-level DWT, and the approximation coefficient vector (LL4) thus obtained is quantized using a quantization parameter (Q). The approximation coefficients and their quantized values are used as the target output and as an input for training the kernel online sequential extreme learning machine (KOS-ELM). The generated model is then utilized in watermark extraction to recover the binary watermark from the signed image in a semi-blind manner. Experimental results exhibit that the proposed scheme has high degree of robustness and high level of imperceptibility with an average peak signal-to-noise ratio (PSNR) of 42.54 dB. We have also performed statistical analysis using Wilcoxon signed rank test to compare our work with the recently published works. The statistical comparison as well as experimental results demonstrate that this work outperforms other recently published works in terms of imperceptibility as well as robustness.

Keywords Discrete wavelet transform · Semi-blind watermarking · Quantization · Kernel OS-ELM

N. K. Sharma

Department of Computer Science, Ram Lal Anand College, University of Delhi, New Delhi 110021, India

S. Kumar (✉) · A. Rajpal · N. Kumar

Department of Computer Science, University of Delhi, New Delhi 110007, India
e-mail: skumar.cs.du@gmail.com

A. Rajpal

e-mail: arajpal@cs.du.ac.in

1 Introduction

With the emergence of digital era, an extensive amount of digital content is being shared over the Internet. This has increased the copyright, authenticity, and security concerns. Digital watermarking is a well-known and the widely used mechanism to protect the digital rights. In digital watermarking, a secret information called as watermark is embedded into the host signal which may be recovered later to confirm its ownership or authenticity.

Imperceptibility, robustness, and capacity are some key concerns of a watermarking scheme. As there exists a trade-off between imperceptibility and robustness, striking a balance between the two is a challenge [9, 11]. Usually watermarking may be carried out either in frequency domain [8, 14, 16] or spatial domain [21, 27]. In the spatial domain watermarking, the spatial values of the signal are directly modified using a watermark, while in the frequency domain watermarking, the transformed signal is modulated using a watermark. It is well established in the literature that the frequency domain watermarking is more resilient to image processing and geometric attacks [4, 14, 28]. In the literature, several transform domain watermarking techniques utilizing various transforms such as discrete cosine transform (DCT) and discrete wavelet transform (DWT) are presented [8, 12, 13, 15–17]. Among them, DWT is most widely used in watermarking because of its multi-resolution analysis ability and energy compression qualities. DWT provides resilience against various interferences such as JPEG compression, Gaussian filtering, and salt pepper noise. The low-frequency components of DWT preserve the contour information of an image. On the other hand, the high-frequency components preserve the finer details [14].

Ray et al. [23] summarized the state-of-the-art watermarking schemes in which they mentioned that robustness varies with the selection of sub-bands used for watermarking. They recommended the use of lower frequency sub-band for better robustness and also listed common watermarking challenges such as imperceptibility robustness trade-off, watermarking strength, and capacity. Mohammed et al. [20] presented an image watermarking scheme using DWT-SVD. One-level DWT was applied on the watermark, and the resultant LL sub-band was subsequently applied with singular value decomposition (SVD). The host image was further subjected to level-level DWT and in the resultant HH sub-band, singular values of the watermark were embedded. The authors demonstrated the robustness of their scheme against various attacks such as image compression and median filtering. A DWT-DCT-based color image watermarking scheme was proposed by Abdulrahman et al. [1]. Each layer of the host RGB image was subjected to DCT, and the resultant coefficients were subsequently decomposed using DWT. Before embedding, the Arnold scrambled watermark was subjected to DCT and divided into four equal portions. Subsequently, each sub-band of the host picture was embedded with each portion of the DCT coefficients. They demonstrated that embedding DCT coefficients of the watermark into the DWT sub-band matrices of the host image resulted in high imperceptibility and robustness. A DWT-SVD-based multi-watermarking scheme was proposed by Anand et al. [3]. The host image was applied with two-level DWT,

and the resultant HL, LH sub-bands were subsequently decomposed into S_{HL} and S_{LH} using SVD. A binary watermark was divided into two-halves which were embedded into the S_{HL} and S_{LH} . Next, using Hamming code, a text watermark is encoded and inserted into the HH sub-band. They demonstrated that their scheme achieved high degree of imperceptibility and robustness.

Several research groups in recent years utilized various machine learning techniques such as support vector machine (SVM), artificial neural network (ANN), and extreme learning machine (ELM) in designing a watermarking scheme [5, 6, 10, 24, 25]. Agarwal et al. [2] reviewed several watermarking schemes in transform and spatial domains. They discussed basic concepts and important characteristics of the watermarking scheme. Further, they listed different techniques such as spread spectrum and neural network used for improved robust watermarking. They also noted that host image of size 512×512 and watermark of 32×32 , 64×64 and 128×128 were commonly used in literature. Dabas and Singh [5] proposed a DWT-SVD-based watermarking scheme utilizing kernel ELM (K-ELM) and reduced kernel ELM (RK-ELM). The grayscale host image was subjected to integer wavelet transform (IWT) and the resultant HL sub-band was split into 4×4 blocks. Subsequently, each block was decomposed using SVD and the binary watermark bits were inserted into the first singular value of each block. Further, K-ELM and RK-ELM models were trained to discover the association between the original and watermarked image coefficients. During watermark extraction, the trained model was used to extract the watermark without having the requirement of the original watermark. Sunesh et al. [25] put forward a histogram modification-based watermarking scheme. First of all, the histogram of the host image was generated and grouped into a number of bins by considering the two neighborhood pixels. The qualified bins were then selected based on a threshold value, and a binary watermark's bit value was embedded into it. They used ANN and backpropagation neural network (BPNN) to optimize the recovered bits of the watermark which ensured high degree of robustness of the presented scheme against attacks such as median filtering, rotation, and histogram equalization.

The above-mentioned research works have one or more of the following drawbacks:

1. Some of these works [6, 21, 25, 27] have used spatial domain to embed a watermark, resulting in lack of robustness [2].
2. These watermarking methods [3, 13, 26] have either poor imperceptibility or low robustness. The primary reason is that they did not find the optimal embedding strength to balance the trade-off between imperceptibility and robustness. Another reason is that they used higher frequency sub-bands for watermarking [23].
3. The time complexity in watermarking is another characteristic which is either not considered at all or is higher in other works [3, 5, 20].

The research gaps listed above have motivated us to propose a time-efficient watermarking scheme using KOS-ELM in order to effectively strike a balance between imperceptibility and the robustness.

1.1 Objective and Contribution

The first objective of this work is to propose a watermarking scheme having high imperceptibility. This means that the signed image obtained after insertion of watermark should look perceptually similar to host image. The second objective is that a watermarking scheme should also be robust against common image processing operations such as filtering and JPEG compression to protect the copyright information. Another objective is to maintain a low time complexity, so that the proposed watermarking can be utilized in real time. Moreover, in the absence of the host image at the receiver end, a semi-blind watermarking scheme is usually preferred over the informed watermarking schemes.

In this work, we have proposed a novel semi-blind image watermarking scheme using kernel online sequential extreme learning machine (KOS-ELM) and DWT. The major contribution of this work is as follows:

1. The watermark embedding strength is optimized to achieve a good balance between the imperceptibility and the robustness.
2. To ensure high robustness, the watermark is embedded in the perceptually significant region of the host image.
3. The time complexity of the proposed work is relatively less than the others [3, 20].
4. The KOS-ELM model is utilized to semi-blindly recover the watermark from the signed image without having the requirement of host image.

To the best of our knowledge, KOS-ELM is being used for the first time in the DWT-based semi-blind watermarking scheme. The choice of KOS-ELM is particularly apt as it need not to be trained afresh whenever new data becomes available. To train the KOS-ELM model, the host image is subjected to four-level DWT. The resultant approximation coefficients (LL4) and their quantized values act as target and input for KOS-ELM. The generated model is then utilized in the watermark extraction process to predict the original coefficients of the signed image. We have evaluated our work using some commonly used quality evaluation metrics like peak signal-to-noise ratio (PSNR), structural similarity index (SSIM), normalized correlation coefficient (NCC) and bit-error rate (BER). It is observed that proposed work has high imperceptibility and is also robust to common image processing attacks. The proposed work has depicted the high level of imperceptibility and superior robustness over the recently published works. Also, the time complexity of our work is relatively less. The proposed novel watermarking shows promising results which are expected to be useful in the diverse applications such as copyright protection.

The remaining of the paper is laid out as follows: In Sect. 2, we have presented the proposed watermarking scheme, and Sect. 3 discusses details of the experimental and the outcomes. In the last Sect. 4, the conclusion of the paper is presented.

2 Methodology

In this section, we first present the system model of the proposed watermarking scheme. Subsequently, a brief description of the model generation procedure, followed by the watermark embedding and extraction, procedure is presented in the following sub-sections.

2.1 Proposed Watermarking Scheme

The proposed watermarking scheme has three phases, namely model generation, watermark embedding, and extraction. Figure 1 depicts the overall system work-flow.

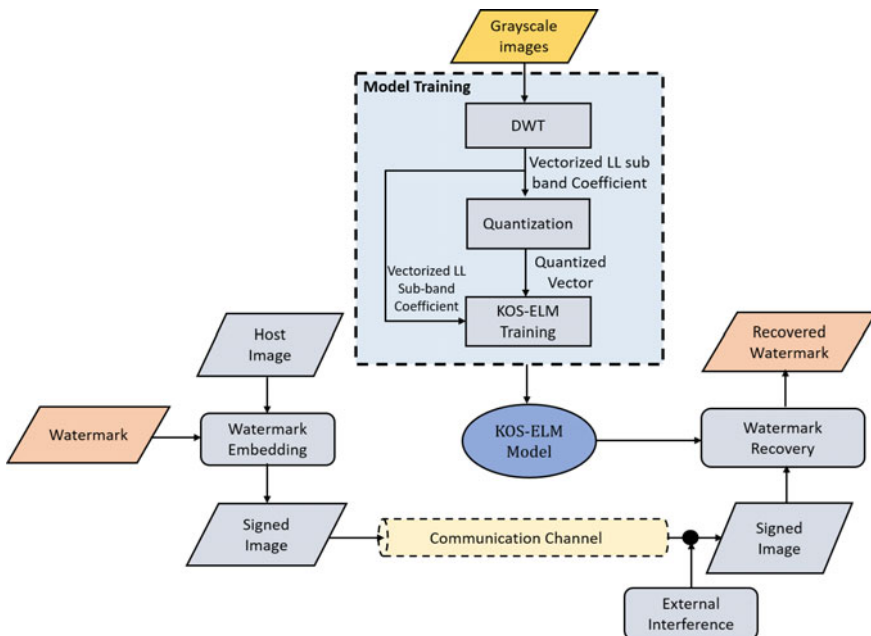


Fig. 1 Proposed watermarking scheme: A set of grayscale images are used to train the KOS-ELM model, which is subsequently used to recover the watermark from signed image in a semi-blind manner

At first, a kernel online sequential extreme learning machine (KOS-ELM) model is generated using the set of training images. The choice of KOS-ELM is particularly apt as it need not to be trained afresh whenever new data becomes available. The trained model is utilized to recover the watermark from the received image in a semi-blind manner, without needing the host image. To the best of our knowledge, KOS-ELM has been used for the first time in DWT-based semi-blind watermarking scheme. In the second phase reflecting the watermark embedding, a binary watermark is embedded into the host image using an embedding strength (γ). Further, in the extraction phase, the watermark is recovered from the signed image using the KOS-ELM model and the same embedding strength was used during embedding.

2.2 Model Generation Algorithm

In the proposed work, we have used kernel online sequential extreme learning machine (KOS-ELM). KOS-ELM is a variant of ELM which does not need to be trained afresh whenever new data becomes available, in contrast to the batch learning algorithms such as ELM. KOS-ELM is capable of learning the training data instance-by-instance or chunk-by-chunk. It consists of two phases: initialization and sequential learning. An initial model is generated during the initialization phase, which may be further updated in the sequential learning phase [22].

To train the KOS-ELM, each host image in the training set is subjected to four-level DWT to produce the LL_4 sub-band coefficient matrix. The LL_4 sub-band is further reshaped to a vector x which is subsequently quantized using the quantization parameter Q to get vector t . The vectors x and t are iteratively appended row-wise to form the matrices X and T which act as input and target for the KOS-ELM model. Using these matrices, the KOS-ELM model is trained to learn the approximation coefficient value corresponding to the quantized value. The trained model is later utilized in watermark recovery to predict the original coefficients of the signed image. The Algorithm 1 lays out the steps of model generation.

Algorithm 1: KOS-ELM Model Generation

Input: Set of Training Images (each having size 512×512)
 k : DWT levels (k is set to 4)
 H : Hidden neuron size
 KF : Kernel function ('sigmoid' kernel is used in this work)
 Q : Quantization parameter ($Q = 32$ based on our experimentation)
 KP : Kernel Parameters
 CS : Chunk Size

Output: $KOS-ELM_Model$

1. Initialize $X \leftarrow []$, $T \leftarrow []$.
2. **For** each grayscale image in training set **do**:
 - (a) Apply k -level DWT on the host image (I).

$$[LL_k, LH_k, HL_k, HH_k] \leftarrow DWT(I, \text{"Haar"}) \quad (1)$$

where, "Haar" is the wavelet filter, LL_k is the approximation sub-band and LH_k, HL_k, HH_k are the detailed sub-bands. Each sub-band is of size $n/2^k \times n/2^k$, where n is the dimension of the squared host image of size 512×512 .

- (b) Reshape the LL_k sub-band to create a vector x .
 $x \leftarrow \text{reshape}(LL_k, ((n/2^k \times n/2^k), 1))$
 - (c) Quantize the vector x using a quantization parameters Q .
 $t \leftarrow \text{round}(\frac{x}{Q})$
 - (d) $X \leftarrow X.append(x)$
 - (e) $T \leftarrow T.append(t)$
- end**

3. Train the KOS-ELM model using X as feature and T as target.

$KOS-ELM_Model \leftarrow \text{Model training}(X, T, H, KF, KP, CS)$

return $KOS-ELM_Model$

2.3 Watermark Embedding

We have presented the watermark embedding process in this section. In watermarking scheme, the choice of optimal embedding strength (γ) is a challenging task to maintain the balance between robustness and imperceptibility. During experimentation, we observed that with an increase in the value of embedding strength the robustness increases but imperceptibility decreases, whereas an opposite trend is observed when embedding strength is decreased. Similarly, the choice of number

of level of DWT decomposition is an another challenge. In this work, we computed an optimal embedding strength of 0.12 and employed four-level DWT decomposition using ‘‘Haar’’ wavelet to achieve a suitable balance between robustness and imperceptibility.

For watermarking, the host image I is subjected to four-level DWT and the resultant approximation coefficient LL_4 matrix is reshaped into vector P . Subsequently, the watermark is embedded into the vector P using the embedding strength γ . In this work, we have embedded the watermark in LL_4 sub-band as it balance out the imperceptibility and robustness trade-off. Then, inverse four-level DWT is applied to get the signed image. The watermark embedding steps are given in Algorithm 2.

Algorithm 2: Watermark Embedding Algorithm

Input: Trained $KOS-ELM_Model$, Host image (having size 512×512)

k : DWT levels (k equals 4)

$Wm_{original}$: Original Watermark (equals 32×32)

γ : Value of Embedding strength (equals 0.12 for this work)

Q : Quantization parameter (set to 32)

Output: Signed image (I_{signed})

1. Apply k -level DWT on the host image (I).

$$[LL_k, LH_k, HL_k, HH_k] \leftarrow DWT(I, \text{‘‘Haar’’}) \quad (2)$$

2. Select LL_k for watermark embedding and reshape it to a vector P .

$$P \leftarrow \text{reshape}(LL_k, ((n/2^k \times n/2^k), 1))$$

3. Reshape the watermark ($Wm_{original}$) in the form of a vector.

$$Wm \leftarrow \text{reshape}(Wm_{original}, ((n/2^k \times n/2^k), 1))$$

4. Embed the watermark vector (Wm) into vector P using the embedding strength γ .

$$C \leftarrow P + \gamma \cdot Wm \quad (3)$$

where, γ is chosen experimentally as 0.12.

5. Reshape C to get signed approximation sub-band LL'_k .

$$LL'_k \leftarrow \text{reshape}(C, (n/2^k \times n/2^k))$$

6. Apply k - level inverse DWT to get the signed image.

$$I_{signed} \leftarrow DWT(LL'_k, LH_k, HL_k, HH_k, \text{‘‘Haar’’}) \quad (4)$$

2.4 Watermark Extraction

In this section, we have described the watermark recovery process. The received image (I_{signed}) is subjected to four-level DWT, and the resultant LL'_4 sub-band is reshaped to a vector C' . The vector C' is then quantized using the quantization parameter Q . Using the trained KOS-ELM_Model, the unsigned approximation coefficient vector P' is predicted. Thereafter, P' is subtracted from C' to recover the watermark. The watermark extraction steps are given in Algorithm 3.

Algorithm 3: Watermark Extraction Algorithm

Input: Trained *KOS-ELM_Model*, Signed image (having size 512×512)

k : DWT levels

Q : Quantization parameter

Output: Extracted Watermark ($Wm_{\text{extracted}}$)

1. Apply k -level DWT on the signed image (I_{signed}).

$$[LL'_k, LH'_k, HL'_k, HH'_k] \leftarrow \text{DWT}(I_{\text{signed}}, \text{"Haar"}) \quad (5)$$

2. Reshape LL'_k approximation sub-band to a vector C' .

$$C' \leftarrow \text{reshape}(LL'_k, ((n/2^k \times n/2^k), 1))$$

3. Quantize C' using the quantization parameter Q that was used during embedding procedure.

$$Z' \leftarrow \text{round}\left(\frac{C'}{Q}\right)$$

4. Predict the original approximation coefficients using the trained KOS-ELM model while taking Z' as input.

$$P' \leftarrow \text{Predict}(Z', \text{KOS-ELM_Model})$$

5. Extract the watermark using C' , P' and γ .

$$W' \leftarrow \frac{|C' - P'|}{\gamma}$$

where, γ is the embedding strength used during embedding procedure.

6. Reshape W' to recover the watermark.

$$Wm_{\text{extracted}} \leftarrow \text{reshape}(W', (n/2^k \times n/2^k)) \quad (6)$$

3 Experimental Details and Results

The experimental details and outcomes of the proposed work are presented in this section. All the experiments have been performed on MATLAB R2021a Intel i7 tenth-generation processor with clock speed of 2.59 GHz.

3.1 Evaluation Metrics

With the aim to evaluate the proposed work, we have used four quality metrics, namely structural similarity index (SSIM), peak signal-to-noise ratio (PSNR), normalized correlation coefficient (NCC), and bit-error rate (BER). Out of these, SSIM and PSNR are used to assess imperceptibility, while NCC and BER are used to assess the robustness. The mathematical description of these metrics is as follows:

SSIM: SSIM is formulated as follows:

$$\text{SSIM}(I, I_{\text{signed}}) = \frac{(2\mu_I \mu_{I_{\text{signed}}} + D_1) (2\sigma_{I \cdot I_{\text{signed}}} + D_2)}{(\mu_I^2 + \mu_{I_{\text{signed}}}^2 + D_1) (\sigma_I^2 + \sigma_{I_{\text{signed}}}^2 + D_2)} \quad (7)$$

where σ and μ represent the standard deviation and mean, respectively. $\sigma_{I \cdot I_{\text{signed}}}$ is the covariance between images I and I_{signed} , and D_1, D_2 are positive constants.

PSNR: PSNR is defined as follows:

$$\text{PSNR}(I, I_{\text{signed}}) = 10 \log_{10} \frac{pq \cdot I_{\text{MAX}}^2}{\sum_{p=1, q=1}^{m, n} [I(p, q) - I_{\text{signed}}(p, q)]^2} \quad (8)$$

where I represent the host image, I_{signed} represent the signed image, and mn is dimension of the host and signed images. Usually, a value of PSNR above 32 dB is considered to be good [26].

NCC: NCC is defined as follows:

$$\text{NCC}(Wm_{\text{original}}, Wm_{\text{extracted}}) = \frac{\sum_{p=1}^r \sum_{q=1}^s Wm_{\text{original}}(p, q) Wm_{\text{extracted}}(p, q)}{\sum_{p=1, q=1}^{r, s} Wm_{\text{original}}(p, q)^2} \quad (9)$$

BER: BER is a measure of dissimilarity between original and extracted watermarks. It is defined as follows:

$$\text{BER}(Wm_{\text{original}}, Wm_{\text{extracted}}) = \sum_{k=1}^{r \times s} \frac{Wm_{\text{original}} \oplus Wm_{\text{extracted}}}{r \times s} \quad (10)$$

where \oplus symbol signifies the XOR operation, $Wm_{\text{extracted}}$ and Wm_{original} denote the extracted and original watermarks each having size $r \times s$.

The values of SSIM, BER, and NCC lie in the interval [0,1]. For SSIM and NCC, higher value is preferred, while for BER lower value is preferred.

3.2 Choice of Hyper-parameters

The kernel function (KF) and size of hidden layer neurons (H) are the two hyper-parameters of the KOS-ELM. Based on our experimentation, we have set H to 50 and used the “sigmoid” kernel function. We have observed that with $H = 50$ a good model can be generated. Figure 2 shows the training error. It is observed that the minimum mean-squared error (MSE) obtained is 0.0097.

3.3 Evaluation of Imperceptibility

We have presented the results of imperceptibility of the proposed scheme in this section. Figure 3 exhibits the binary watermark of size 32×32 that is used in embedding.

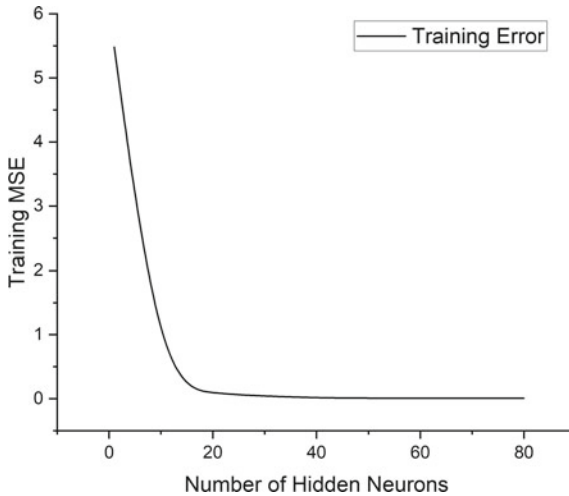


Fig. 2 Training error decreases with an increase in hidden neurons and then stabilizes at $H = 50$



Fig. 3 Binary watermark used in the proposed work

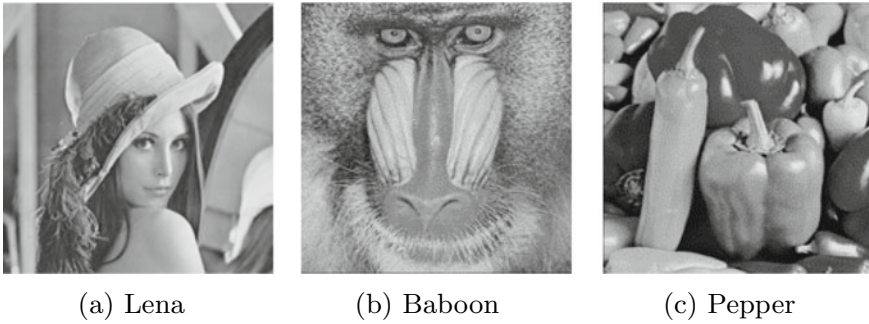


Fig. 4 Host standard images

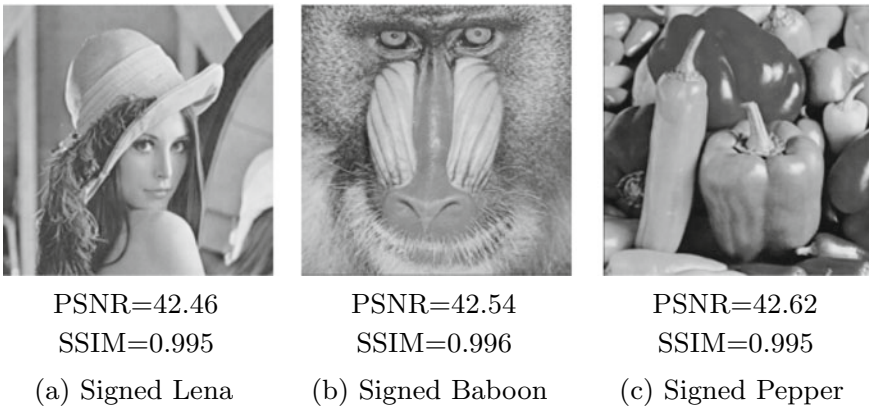


Fig. 5 Signed standard images. The high values of PSNR and SSIM show the signed images have high level of resemblance with respect to host image

To evaluate the visual quality of the signed image with respect to the host image, we have used three standard test images, namely Lena, Baboon, and Pepper. The size of these grayscale images are 512×512 . Figures 4 and 5 depict the host standard images and the corresponding signed images. It is observed that for all the signed images the value of PSNR is above 42.54 dB and SSIM is above 0.99. This clearly shows that the signed images have high visual quality with respect to the host images.

3.4 Evaluation of Robustness

To evaluate the robustness of our work, we have exposed the signed image to various attacks such as Gaussian filtering (filter size = 3×3), salt-and-pepper noise (noise density = 0.001), scaling (signed image is scaled down to 1/2 and then restored to

Table 1 Robustness results against common image processing attacks

	Metrics	Lena	Baboon	Peppers
No attack	BER	0.00	0.00	0.00
	NCC	1.00	1.00	1.00
Salt & pepper noise (noise density = 0.001)	BER	0.01	0.02	0.02
	NCC	0.95	0.92	0.93
Speckle noise (noise density = 0.001)	BER	0.02	0.03	0.02
	NCC	0.93	0.91	0.91
Gaussian filtering (Filter size = 3×3)	BER	0.00	0.00	0.00
	NCC	0.99	1.00	0.99
Median filtering (Filter size = 3×3)	BER	0.04	0.05	0.04
	NCC	0.85	0.83	0.87
Resizing (512→256→512)	BER	0.00	0.00	0.00
	NCC	0.99	0.99	1.00
JPEG compression (QF = 95)	BER	0.00	0.00	0.00
	NCC	1.00	1.00	1.00
JPEG compression (QF = 85)	BER	0.02	0.03	0.02
	NCC	0.98	0.97	0.99

its original dimensions), JPEG compression ($QF = 95$ and 85), and speckle noise (noise density = 0.001). Table 1 shows the BER and NCC values for different attacks applied. It is observed that the NCC values lie in the range $[0.91, 1.00]$ and BER values lie in the range $[0.00, 0.03]$. The values of NCC and BER show that the recovered watermarks have high level of resemblance with respect to the original watermarks.

3.5 Comparative Study

In this section, we have compared our work with some of the recently published works [3, 20] in terms of imperceptibility, robustness, and time complexity. In Table 2, we have given the comparative results of the proposed work with others [3, 20]. It may be noted from this table that the average value of PSNR for the proposed work is 42.54 dB which outperforms Anand et al.'s work [3] and is comparable to Mohammed et al.'s work [20]. In terms of NCC, the proposed work outperforms others in the presence of median filtering (filter size = 3×3), resizing (512 → 256 → 512), salt-

Table 2 Comparison results between the proposed work and some recently published works [3, 20]

	Metric	Proposed work	Mohammed et al. [20]	Anand et al. [3]
Visual quality of signed images	PSNR	42.54	42.14	36.10
No attack	NCC	1.00	–	0.99
Salt-and-pepper attack (density = 0.001)	NCC	0.95	0.84	0.92
Resizing (512→256→512)	NCC	0.99	0.93	0.98
JPEG compression (QF = 90)	NCC	0.99	0.98	0.97
Median filtering (Filter size = 3 × 3)	NCC	0.85	0.69	–
Time complexity		$O(n)$	$O(n^3)$	$O(n^3)$

Our work has higher PSNR and NCC which show our work has high imperceptibility and robustness. Also, the time requirement of our work is less than other works

and-pepper noise (density = 0.001). In the presence of JPEG compression ($QF = 90$), the proposed work achieves robustness at par with Mohammed et al. and Anand et al. [3, 20]. In terms of time complexity, the proposed work has time complexity of $O(n)$, where n is size of squared image. This time complexity is superior to Mohammed et al.'s and Anand et al.'s work which takes $O(n^3)$ time due to the use of SVD-based scheme. Moreover, on an average the proposed work takes only 1.92 s in watermarking, thus making it suitable for different real-time applications.

Statistical Analysis In this section, we statistically compare our work with [3, 20]. We have used Wilcoxon signed rank test [7, 18] to evaluate the significance of the difference between our work, and Mohammed et al. [20]/Anand et al. [3]. The Wilcoxon test was chosen because the experimentation results did not indicate a normal distribution [18]. For comparison, we have used a significance level α of 0.05. The null hypothesis H_0 in the test indicates no notable difference between the two works. The alternate hypothesis H_1 indicates that our work is better. When P -value ≤ 0.05 , the H_0 is rejected with 95% confidence level. In contrast, when P -value > 0.05 , we do not have sufficient evidence to reject the null hypothesis H_0 .

Let w be the Wilcoxon test statistics and w^* be the critical value. If $w < w^*$, we can interpret that the proposed scheme is better than other compared works; otherwise, we do not have any significant evidence that our work is better. In both the cases, w^* , w , and p -value are computed as 1, 0 and 0.0313, respectively. As $w < w^*$ and p -value is lesser than α , we reject the null hypothesis and interpret that our work is superior than the others [3, 20] in terms of imperceptibility and robustness. We have also verified the computed results of the Wilcoxon test using “signrank(.)” function [19] of the MATLAB software.

4 Conclusion

Digital watermarking is a well-known and widely used mechanism to ensure authenticity and copyright protection. In this paper, we have proposed a DWT- and quantization-based semi-blind watermarking scheme. For KOS-ELM training, the host image was subjected to four-level DWT, and the approximation coefficient thus obtained is reshaped to a vector and subsequently quantized. The quantized vector and the approximation vector act as input and target output for KOS-ELM. The trained KOS-ELM model is subsequently used in watermark extraction to semi-blindly extract the watermark. On evaluation of robustness and imperceptibility, the proposed work is found to be superior than the recently published and is also time-efficient. We have also performed Wilcoxon signed rank test to statistically compare our work with others which also demonstrate the superiority of our work. One of the limitations of the proposed work is that it does not provide an explicit scheme for securing the watermark. In future, we propose to enhance this work for color images along with enhanced security using encryption or scrambling techniques.

References

1. Abdulrahman, A.K., Ozturk, S.: A novel hybrid DCT and DWT based robust watermarking algorithm for color images. *Multimedia Tools Appl.* **78**(12), 17027–17049 (2019)
2. Agarwal, N., Singh, A.K., Singh, P.K.: Survey of robust and imperceptible watermarking. *Multimedia Tools Appl.* **78**(7), 8603–8633 (2019)
3. Anand, A., Singh, A.K.: An improved DWT-SVD domain watermarking for medical information security. *Comput. Commun.* **152**, 72–80 (2020)
4. Ansari, I.A., Pant, M., Ahn, C.W.: Robust and false positive free watermarking in IWT domain using SVD and ABC. *Eng. Appl. Artif. Intell.* **49**, 114–125 (2016)
5. Dabas, N., Singh, R.P.: Elm-Kernel and reduced Kernel ELM based watermarking scheme. *J. Inf. Secur. Appl.* **46**, 173–192 (2019)
6. Deeba, F., Kun, S., Dharejo, F.A., Memon, H.: Digital image watermarking based on ANN and least significant bit. *Inf. Secur. J. Global Perspect.* **29**(1), 30–39 (2020)
7. Derrac, J., García, S., Molina, D., Herrera, F.: A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. *Swarm Evol. Comput.* **1**(1), 3–18 (2011)
8. Fares, K., Amine, K., Salah, E.: A robust blind color image watermarking based on Fourier transform domain. *Optik* **208**, 164562 (2020)
9. Hien, T.D., Nakao, Z., Chen, Y.W.: Robust multi-logo watermarking by RDWT and ICA. *Signal Process* **86**(10), 2981–2993 (2006)
10. Islam, M., Mallikharjunudu, G., Parmar, A., Kumar, A., Laskar, R.: SVM regression based robust image watermarking technique in joint DWT-DCT domain. In: 2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICT), pp. 1426–1433. IEEE (2017)
11. Kumar, S., Singh, B.K., Yadav, M.: A recent survey on multimedia and database watermarking. *Multimedia Tools Appl.* **79**(27), 20149–20197 (2020)
12. Lai, C.C., Tsai, C.C.: Digital image watermarking using discrete wavelet transform and singular value decomposition. *IEEE Trans. Instrum. Meas.* **59**(11), 3060–3063 (2010)

13. Lei, B., Zhao, X., Lei, H., Ni, D., Chen, S., Zhou, F., Wang, T.: Multipurpose watermarking scheme via intelligent method and chaotic map. *Multimedia Tools Appl.* **78**(19), 27085–27107 (2019)
14. Li, Y.M., Wei, D., Zhang, L.: Double-encrypted watermarking algorithm based on cosine transform and fractional Fourier transform in invariant wavelet domain. *Inf. Sci.* **551**, 205–227 (2021)
15. Lin, S.D., Chen, C.F.: A robust DCT-based watermarking for copyright protection. *IEEE Trans. Consum. Electron.* **46**(3), 415–421 (2000)
16. Luo, Y., Li, L., Liu, J., Tang, S., Cao, L., Zhang, S., Qiu, S., Cao, Y.: A multi-scale image watermarking based on integer wavelet transform and singular value decomposition. *Expert Syst. Appl.*, 114272 (2020)
17. Makbol, N.M., Khoo, B.E.: Robust blind image watermarking scheme based on redundant discrete wavelet transform and singular value decomposition. *AEU-Int. J. Electron. Commun.* **67**(2), 102–112 (2013)
18. Makbol, N.M., Khoo, B.E.: A new robust and secure digital image watermarking scheme based on the integer wavelet transform and singular value decomposition. *Digital Signal Process.* **33**, 134–147 (2014)
19. Matlab: Wilxon signed rank test function. Retrieved from <https://in.mathworks.com/help/stats/signrank.html>
20. Mohammed, A.A., Salih, D.A., Saeed, A.M., Kheder, M.Q.: An imperceptible semi-blind image watermarking scheme in DWT-SVD domain using a zigzag embedding technique. *Multimedia Tools Appl.* **79**(43), 32095–32118 (2020)
21. Pal, P., Jana, B., Bhaumik, J.: Watermarking scheme using local binary pattern for image authentication and tamper detection through dual image. *Secur. Privacy* **2**(2), e59 (2019)
22. Pandeewari, N., Vignesh, D., Pushpalakshmi, R., et al.: Online sequential extreme learning algorithm with kernels for bigdata classification. In: 2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS), pp. 1–5. IEEE (2017)
23. Ray, A., Roy, S.: Recent trends in image watermarking techniques for copyright protection: a survey. *Int. J. Multimedia Inf. Retrieval*, pp. 1–22 (2020)
24. Sharma, N.K., Kumar, S., Rajpal, A., Kumar, N.: Mantaraywmark: an image adaptive multiple embedding strength optimization based watermarking using manta ray foraging and bi-directional ELM. *Expert Syst. Appl.* **200**, 116860 (2022). <https://doi.org/10.1016/j.eswa.2022.116860>
25. Sunesh, Kishore, R.R., Saini, A.: Optimized image watermarking with artificial neural networks and histogram shape. *J. Inf. Optim. Sci.* **41**(7), 1597–1613 (2020)
26. Swaraja, K., Meenakshi, K., Kora, P.: An optimized blind dual medical image watermarking framework for tamper localization and content authentication in secured telemedicine. *Biomed. Signal Process. Control* **55**, 101665 (2020)
27. Tuncer, T., Kaya, M.: A novel image watermarking method based on center symmetric local binary pattern with minimum distortion. *Optik* **185**, 972–984 (2019)
28. Wang, Y., Doherty, J.F., Van Dyck, R.E.: A wavelet-based watermarking algorithm for ownership verification of digital images. *IEEE Trans. Image Process.* **11**(2), 77–88 (2002)

Stress Level Analysis Using Bipolar Picture Fuzzy Set



P. Muralikrishna, P. Hemavathi, and K. Palanivel

Abstract The concept of bipolar picture fuzzy set is introduced, and few of its basic properties were provided. Also, presented the modal operators and arithmetic operators of picture fuzzy set. The relations and product on picture fuzzy set and its associated outcomes are explored. Further, the application is carried out in the stress level of the people who are working by using online platforms.

Keywords Fuzzy set · Bipolar fuzzy · Picture fuzzy · Stress level · Bipolar

1 Introduction

Zadeh [1] initiated fuzzy sets, which are characterized by a membership function that specifies a degree of membership between 0 and 1. Atanassov [2] introduced the concept of intuitionistic fuzzy sets as a generalization of fuzzy sets in 1986. Smarandache et al. [3, 4] studied the interval valued bipolar fuzzy weighted neutrosophic sets and its application to those that provide uncertain information, which is easier to form in the decision-making process. The author also discussed the concept and properties of bipolar neutrosophic sets and their applications based on multi-criteria decision-making problems. Muthumeenakshi et al. [5] presented the algorithm for decision-making using bipolar Q-fuzzy set and explained its applications towards choosing the cement with the influenced characters in civil engineering.

Coung [6–9] proposed the concept of picture fuzzy set, which is a direct extension of fuzzy set and intuitionistic fuzzy set. Based on these notations and properties, the

P. Muralikrishna

Department of Mathematics, Muthurangam Government Arts College (Autonomous),
Vellore 632002, India

P. Hemavathi (✉)

Division of Mathematics, Saveetha School of Engineering, SIMATS, Thandalam 602105, India
e-mail: hemavathip.sse@saveetha.com

K. Palanivel

Department of Mathematics, School of Advanced Sciences, Vellore Institute of Technology,
Vellore 632014, India

authors have made a deliberate attempt in presenting the engrossing results. In addition, the author also established fuzzy logic operators, such as negation, conjunction, disjunction, and the implications of picture fuzzy sets. In [10, 11], Palash Dutta strives to use distance measurement in diffuse reflectance image collection for medical diagnosis, and introduces the technology through appropriate case studies. In addition, the decomposition theorem of picture fuzzy set and the picture fuzzy algorithm based on extension principle are also given. Ashraf et al. [12, 13] proposed the concept of picture fuzzy linguistic set and its application in multi-attribute groups. In addition, the author describes a new concept of cubic picture fuzzy sets.

Zhang [14] established the resolutions of two types of level sets and examined the relations between them. Bipolar fuzzy set theory combines polarity and fuzzy into a unified model. In [15], the concept of a bipolar value fuzzy set and its applications are reviewed. Thereafter, bipolar fuzzy partial algebra and bipolar fuzzy ideals of BCK/BCI algebra were proposed and their relationship was studied by Lee [16]. Farhat [17] began the idea of the bipolar valued fuzzy K -subalgebra and came to the conclusion that the bipolar valued fuzzy K -subalgebra can be generated from the anode bipolar valued fuzzy set. The notion of N -structured ideal of a BF-algebra presented by Muralikrishna et al. [18] in 2013. Prince Williams [19] dealt with a coupled N -structure which is the generalization of N -structure have applied in a subtraction algebra. Jun et al. [20] analysed the relations of coupled N -ideal and coupled NC-ideal and discussed its applications in BCK/BCI algebras. In [21], the author explored the representation of the neutrosophic N -subalgebra and the neutrosophic N -ideal, and provided the conditions for closed N -neutrosophic ideal.

Deli et al. [22] introduced the idea of a bipolar neutrosophic set and its application based on multi-criteria decision problems. Phong et al. [23] have explained the composition of some picture fuzzy relationships. Abdullah Saleem et al. discussed the weighted average of the picture fuzzy information and the generalized form of the geometric aggregation operators. In 2017, Wei [24] proposed the idea of the picture fuzzy aggregation operator and its application in multi-attribute decision-making. By using all the above concepts, this paper intends to study the notions and properties of bipolar picture fuzzy sets. This work is classified into the following sections: Sect. 1 gives the introductory part and Sect. 2 presents the basic definitions and examples related to current work. Section 3 provides the interesting properties and the model operators of bipolar picture fuzzy set. Section 4 illustrates the product on bipolar picture fuzzy sets. Further, in Sect. 5 an application for the bipolar picture fuzzy set has been integrated and Sect. 6 gives the conclusion.

2 Preliminaries

This section presents the following necessary basic definitions and results.

Definition 2.1

Atanassov [2] The fuzzy set in \wp is defined as the function $\sqsupset : \wp \rightarrow [0, 1]$ For each element \wp in a universal set \wp , $\sqsupset(x)$ is called the membership value of $x \in \wp$.

Definition 2.2

Atanassov [2] The Intuitionistic fuzzy set (IFS) in the non-empty set \wp is defined by $\mathfrak{H} = \{(x, \sqsupset_{\mathfrak{H}}(x), \aleph_{\mathfrak{H}}(x) / x \in \wp)\}$ where $\sqsupset_{\mathfrak{H}} : \wp \rightarrow [0, 1]$ is a membership function of \mathfrak{H} and $\aleph_{\mathfrak{H}} : \wp \rightarrow [0, 1]$ is a non membership function of \mathfrak{H} which satisfies $0 \leq \sqsupset_{\mathfrak{H}}(x) + \aleph_{\mathfrak{H}}(x) \leq 1$.

Definition 2.3

Muthumeenakshi et al. [5] A bipolar valued fuzzy set \mathfrak{B} in a non-empty set \wp is an object with the form $\mathfrak{B} = \{(x, \sqsupset^+(x), \aleph^-(x)) : x \in \wp\}$ where $\sqsupset^+ : \wp \rightarrow [0, 1]$ and $\aleph^- : \wp \rightarrow [-1, 0]$ are assigned values $-1 \leq \sqsupset^+(x) + \aleph^-(x) \leq 1$. Here, $\mu^+(x)$ represents the positive membership degree which is also known as the satisfaction degree of an element x to the property corresponding to a bipolar valued fuzzy set $\mathfrak{B} = \{(x, \sqsupset^+(x), \aleph^-(x)) : x \in \wp\}$. And $\aleph^-(x)$ denotes the negative membership degree is also known as the satisfaction degree of an element x to some implicit counter-property corresponding to a bipolar valued fuzzy set $\mathfrak{B} = \{(x, \sqsupset^+(x), \aleph^-(x)) : x \in \wp\}$. If $\sqsupset^+(x) \neq 0$ and $\aleph^-(x) = 0$, x is considered to have only positive satisfaction for $\mathfrak{B} = \{(x, \sqsupset^+(x), \aleph^-(x)) : x \in \wp\}$. If $\sqsupset^+(x) = 0$, $\aleph^-(x) \neq 0$, it means that x does not satisfy the property of $\mathfrak{B} = \{(x, \sqsupset^+(x), \aleph^-(x)) : x \in \wp\}$ but does partially satisfy the counter-property of $\mathfrak{B} = \{(x, \sqsupset^+(x), \aleph^-(x)) : x \in \wp\}$. When the membership function of the property overlaps its counter property over some portion of \wp , an element x can be such that $\sqsupset^+(x) \neq 0$ and $\aleph^-(x) \neq 0$. For the sake of simplicity, the bipolar valued fuzzy set $\mathfrak{B} = \{(x, \sqsupset^+(x), \aleph^-(x)) : x \in \wp\}$ shall be denoted by the symbol the symbol $\mathfrak{B} = (\wp; \sqsupset^+, \aleph^-)$. For $x \in \wp$, the pair $\mathfrak{B}_x = (\sqsupset^+_x, \aleph^-_x)$ is known as bipolar valued fuzzy number (BVFN).

Definition 2.4

Dutta and Ganju [11, Abdullah and Ashraf 25] A Picture fuzzy set (PFS) \mathfrak{H} on a universe \wp is an object of the form $\mathfrak{H} = \{(x, \sqsupset_{\mathfrak{H}}(x), \eta_{\mathfrak{H}}(x), \aleph_{\mathfrak{H}}(x) / x \in \wp)\}$ where $\sqsupset_{\mathfrak{H}}(x) \in [0, 1]$ is called the degree of positive membership, $\eta_{\mathfrak{H}}(x) \in [0, 1]$ is called the degree of neutral membership and $\aleph_{\mathfrak{H}}(x) \in [0, 1]$ is called the degree of negative membership of $x \in \wp$ with the condition $\sqsupset_{\mathfrak{H}}(x) + \eta_{\mathfrak{H}}(x) + \aleph_{\mathfrak{H}}(x) \leq 1$. Then, $\forall x \in \wp$, $1 - (\sqsupset_{\mathfrak{H}}(x) + \eta_{\mathfrak{H}}(x) + \aleph_{\mathfrak{H}}(x))$ is called the degree of refusal membership.

Definition 2.5

A Bipolar valued Q -Fuzzy Decision set of \wp denoted by $BVQF_{\wp}D$ and is defined by $BVQF_{\wp}D = \{(\sqsupset^+_{QF_{\wp}D}(x), \aleph^-_{QF_{\wp}D}(x)) : x \in \wp\}$ which is bipolar fuzzy set over

\wp where $\sqsupset_{Q_{F_\wp D}}^+(\mathcal{x}) : \wp \rightarrow [0, 1]$ and $\aleph_{Q_{F_\wp D}}^-(\mathcal{x}) : \wp \rightarrow [-1, 0]$ and the positive and the negative membership degrees, respectively, such that

$$\sqsupset_{Q_{F_\wp D}}^+(\mathcal{x}) : \frac{1}{|N|} \sum_{j=1}^n \sqsupset_{\wp}^+(\mathcal{x}, q_j)$$

And

$$\aleph_{Q_{F_\wp D}}^-(\mathcal{x}) : \frac{1}{|N|} \sum_{j=1}^n \aleph_{\wp}^-(\mathcal{x}, q_j)$$

Hence $q_j \in Q$ and N is number of characterizing which influence the particular population.

Definition 2.6

Cuong [6, Cu^ong 9] For every two picture fuzzy sets \mathfrak{H} and \mathfrak{B} , the union and intersection are defined as follows:

- (i) $\mathfrak{H} \subseteq \mathfrak{B}$ iff $\forall \mathcal{x} \in \wp, \sqsupset_{\mathfrak{H}}(\mathcal{x}) \leq \sqsupset_{\mathfrak{B}}(\mathcal{x})$ and $\eta_{\mathfrak{H}}(\mathcal{x}) \leq \eta_{\mathfrak{B}}(\mathcal{x})$ and $\aleph_{\mathfrak{H}}(\mathcal{x}) \geq \aleph_{\mathfrak{B}}(\mathcal{x})$
- (ii) $\mathfrak{H} = \mathfrak{B}$ iff $(\mathfrak{H} \subseteq \mathfrak{B} \ \& \ \mathfrak{B} \subseteq \mathfrak{H})$
- (iii) $\mathfrak{H} \cup \mathfrak{B} = \{x, \max(\sqsupset_{\mathfrak{H}}(\mathcal{x}), \sqsupset_{\mathfrak{B}}(\mathcal{x})), \min(\eta_{\mathfrak{H}}(\mathcal{x}), \eta_{\mathfrak{B}}(\mathcal{x})), \min(\aleph_{\mathfrak{H}}(\mathcal{x}), \aleph_{\mathfrak{B}}(\mathcal{x})) : \mathcal{x} \in \wp\}$
- (iv) $\mathfrak{H} \cap \mathfrak{B} = \{x, \min(\sqsupset_{\mathfrak{H}}(\mathcal{x}), \sqsupset_{\mathfrak{B}}(\mathcal{x})), \max(\eta_{\mathfrak{H}}(\mathcal{x}), \eta_{\mathfrak{B}}(\mathcal{x})), \max(\aleph_{\mathfrak{H}}(\mathcal{x}), \aleph_{\mathfrak{B}}(\mathcal{x})) : \mathcal{x} \in \wp\}$

Definition 2.7

Muralikrishna and Chandramouleeswaran [18] (N -Function and N -Structure) For a non-empty Set S , $F(S, [-1, 0])$ represents the collection of functions from S to $[-1, 0]$. A member of $F(S, [-1, 0])$ is a negative valued function from S to $[-1, 0]$, also known as an N -function, and an N -structure (NS) on S is an ordered pair (S, η) of S and an N -function on S .

3 Bipolar Picture Fuzzy Set

This section provides a basic definition of the bipolar picture fuzzy set as well as its results.

Definition 3.1

For a non-empty set \wp , define an object of the form $\mathfrak{H} = \{\langle \mathcal{x}, \sqsupset^s_{\mathfrak{H}}(\mathcal{x}), \eta^s_{\mathfrak{H}}(\mathcal{x}), \aleph^s_{\mathfrak{H}}(\mathcal{x}), \sqsupset^c_{\mathfrak{H}}(\mathcal{x}), \eta^c_{\mathfrak{H}}(\mathcal{x}), \aleph^c_{\mathfrak{H}}(\mathcal{x}) \rangle : \mathcal{x} \in \wp\}$ is known as a bipolar picture fuzzy set (BPFS) where $\sqsupset^s(\mathcal{x}), \eta^s(\mathcal{x}), \aleph^s(\mathcal{x}) : \wp \rightarrow [0, 1]$ and $\sqsupset^c_{\mathfrak{H}}(\mathcal{x}), \eta^c_{\mathfrak{H}}(\mathcal{x}), \aleph^c_{\mathfrak{H}}(\mathcal{x})$

$(\varkappa) : \wp \rightarrow [-1, 0]$. The notations $\beth^s_{\wp}(\varkappa)$, $\eta^s_{\wp}(\varkappa)$, $\aleph^s_{\wp}(\varkappa)$ denotes the positive, intermediate, and negative satisfactory membership of an element $\varkappa \in \wp$ and $\beth^c_{\wp}(\varkappa)$, $\eta^c_{\wp}(\varkappa)$, $\aleph^c_{\wp}(\varkappa)$ denotes the positive, intermediate, and negative counter membership of the element $\varkappa \in \wp$, corresponding to a bipolar picture fuzzy set where $-1 \leq \beth^s_{\wp} + \beth^c_{\wp} \leq 1$, $-1 \leq \eta^s_{\wp} + \eta^c_{\wp} \leq 1$, and $-1 \leq \aleph^s_{\wp} + \aleph^c_{\wp} \leq 1$.

Example 3.2

Let $\wp = \{\varkappa_1, \varkappa_2, \varkappa_3\}$.

$$\text{Then } \wp = \left\{ \begin{array}{l} \langle \varkappa_1, 0.6, 0.4, 0.2, -0.5, -0.3, -0.02 \rangle \\ \langle \varkappa_2, 0.4, 0.3, 0.6, -0.03, -0.002, -0.4 \rangle \\ \langle \varkappa_3, 0.7, 0.04, 0.3, -0.2, -0.4, -0.05 \rangle \end{array} \right\} \text{ is a BPFs of } \wp.$$

Definition 3.3

In a non-empty set \wp , a BPF number (BPFN) is defined as an object of the form $\wp = \{(\beth^s_{\varkappa}, \eta^s_{\varkappa}, \aleph^s_{\varkappa}, \beth^c_{\varkappa}, \eta^c_{\varkappa}, \aleph^c_{\varkappa}) \mid \forall \varkappa \in \wp\}$.

Definition 3.4

A H-Bipolar fuzzy set (HBFS) \wp in a non-empty set \wp is an object with the form

$$\wp = \left\{ \left((\varkappa, h), \beth^s_{\wp}(\varkappa, h), \eta^s_{\wp}(\varkappa, h), \aleph^s_{\wp}(\varkappa, h), \beth^c_{\wp}(\varkappa, h), \eta^c_{\wp}(\varkappa, h), \aleph^c_{\wp}(\varkappa, h) \right) : (\varkappa, h) \in \wp \times H \right\}$$

where $\beth^s_{\wp}(\varkappa, h)$, $\eta^s_{\wp}(\varkappa, h)$, $\aleph^s_{\wp}(\varkappa, h) : \wp \times H \rightarrow [0, 1]$ and $\beth^c_{\wp}(\varkappa, h)$, $\eta^c_{\wp}(\varkappa, h)$, $\aleph^c_{\wp}(\varkappa, h) : \wp \times H \rightarrow [-1, 0]$ are mappings $-1 \leq \beth^s_{\wp}(\varkappa, h) + \beth^c_{\wp}(\varkappa, h) \leq 1$, $-1 \leq \eta^s_{\wp}(\varkappa, h) + \eta^c_{\wp}(\varkappa, h) \leq 1$, and $-1 \leq \aleph^s_{\wp}(\varkappa, h) + \aleph^c_{\wp}(\varkappa, h) \leq 1$ and H is any non-empty set.

Definition 3.5

A H-Bipolar picture fuzzy decision set (HBPF^D) set of \wp denoted by HBPF^D _{\wp} and is denoted by

$$H[D : \wp] = \left\{ \left(\varkappa, \beth^s_{HF^D_{\wp}}(\varkappa), \eta^s_{HF^D_{\wp}}(\varkappa), \aleph^s_{HF^D_{\wp}}(\varkappa), \beth^c_{HF^D_{\wp}}(\varkappa), \eta^c_{HF^D_{\wp}}(\varkappa), \aleph^c_{HF^D_{\wp}}(\varkappa) \right) : \varkappa \in \wp \right\}$$

which is bipolar picture fuzzy set over \wp where $\beth^s_{HF^D_{\wp}}(\varkappa)$, $\eta^s_{HF^D_{\wp}}(\varkappa)$, $\aleph^s_{HF^D_{\wp}}(\varkappa) : \wp \rightarrow [0, 1]$ and $\beth^c_{HF^D_{\wp}}(\varkappa)$, $\eta^c_{HF^D_{\wp}}(\varkappa)$, $\aleph^c_{HF^D_{\wp}}(\varkappa) : \wp \rightarrow [-1, 0]$. The notations $\mu^s_{HF^D_{\wp}}(\varkappa)$, $\eta^s_{HF^D_{\wp}}(\varkappa)$, $\nu^s_{HF^D_{\wp}}(\varkappa)$ denotes the positive, intermediate and negative satisfactory membership of an element $\varkappa \in \wp$ and $\beth^c_{HF^D_{\wp}}(\varkappa)$, $\eta^c_{HF^D_{\wp}}(\varkappa)$, $\aleph^c_{HF^D_{\wp}}(\varkappa)$ denotes the positive, intermediate and negative counter membership degrees respectively such that

$$\beth^s_{HF^D_{\wp}}(\varkappa) : \frac{1}{|N|} \sum_{j=1}^n \beth^s_{\wp}(\varkappa, h_j), \eta^s_{HF^D_{\wp}}(\varkappa)$$

$$\begin{aligned}
 &: \frac{1}{|N|} \sum_{j=1}^n \eta_{\wp}^s(\mathcal{Z}, h_j), \aleph^s_{H F_{\wp}^D}(\mathcal{Z}) : \frac{1}{|N|} \sum_{j=1}^n \aleph_{\wp}^s(\mathcal{Z}, h_j) \\
 \beth^c_{H F_{\wp}^D}(\mathcal{Z}) &: \frac{1}{|N|} \sum_{j=1}^n \beth_{\wp}^c(\mathcal{Z}, h_j), \eta^c_{H F_{\wp}^D}(\mathcal{Z}) \\
 &: \frac{1}{|N|} \sum_{j=1}^n \eta_{\wp}^c(\mathcal{Z}, h_j), \aleph^c_{H F_{\wp}^D}(\mathcal{Z}) : \frac{1}{|N|} \sum_{j=1}^n \aleph_{\wp}^c(\mathcal{Z}, h_j)
 \end{aligned}$$

Here, $h_j \in H$ and N is number of characteristics which influence the particular population.

Definition 3.6

The performance index (PI) of a BFN $\mathfrak{H} = (\beth_{\mathcal{Z}}^s, \eta_{\mathcal{Z}}^s, \aleph_{\mathcal{Z}}^s, \mu_{\mathcal{Z}}^c, \eta_{\mathcal{Z}}^c, \aleph_{\mathcal{Z}}^c)$ is defined as

$$\begin{aligned}
 \text{PI}(\mathfrak{H}_{\mathcal{Z}}) &= (\beth_{\mathcal{Z}}^s + \aleph_{\mathcal{Z}}^s - \eta_{\mathcal{Z}}^s) \left(1 + (\beth_{\mathcal{Z}}^s)^3 + (\eta_{\mathcal{Z}}^s)^3 + (\aleph_{\mathcal{Z}}^s)^3 - (\beth_{\mathcal{Z}}^s \cdot \eta_{\mathcal{Z}}^s \cdot \aleph_{\mathcal{Z}}^s) \right) + 1 \\
 &\quad - \left[(\beth_{\mathcal{Z}}^c + \aleph_{\mathcal{Z}}^c - \eta_{\mathcal{Z}}^c) \left(1 + (\beth_{\mathcal{Z}}^c)^3 + (\eta_{\mathcal{Z}}^c)^3 + (\aleph_{\mathcal{Z}}^c)^3 - (\beth_{\mathcal{Z}}^c \cdot \eta_{\mathcal{Z}}^c \cdot \aleph_{\mathcal{Z}}^c) \right) \right]
 \end{aligned}$$

This PI is used to identify an ideal solution from the various likewise objects from the given population.

Theorem 3.7

A BPFS is the generalization of a bipolar fuzzy set.

Proof Assume that \wp is a BPFS. Then by setting the satisfactory components η^s, \aleph^s equals to zero and the counter components η^c, \aleph^c equals to zero, the bipolar picture fuzzy set is reduced to bipolar fuzzy set.

Theorem 3.8

If BPFS \mathfrak{H} is defined as

$\mathfrak{H} = \{(\mathcal{Z}, \beth_{\mathfrak{H}}^s(\mathcal{Z}), \eta_{\mathfrak{H}}^s(\mathcal{Z}), \aleph_{\mathfrak{H}}^s(\mathcal{Z}), \beth_{\mathfrak{H}}^c(\mathcal{Z}), \eta_{\mathfrak{H}}^c(\mathcal{Z}), \aleph_{\mathfrak{H}}^c(\mathcal{Z})) : \mathcal{Z} \in \wp\}$ then the following are true:

- (i) $\beth_{\mathfrak{H}}^s(\mathcal{Z}), \eta_{\mathfrak{H}}^s(\mathcal{Z}), \aleph_{\mathfrak{H}}^s(\mathcal{Z})$ are fuzzy sets.
- (ii) $\beth_{\mathfrak{H}}^c(\mathcal{Z}), \eta_{\mathfrak{H}}^c(\mathcal{Z}), \aleph_{\mathfrak{H}}^c(\mathcal{Z})$ are N-functions.
- (iii) The pairs $(\beth_{\mathfrak{H}}^s(\mathcal{Z}), \aleph_{\mathfrak{H}}^s(\mathcal{Z}))$ and $(\eta_{\mathfrak{H}}^s(\mathcal{Z}), \aleph_{\mathfrak{H}}^s(\mathcal{Z}))$ are intuitionistic fuzzy sets.
- (iv) The triplet $(\beth_{\mathfrak{H}}^s(\mathcal{Z}), \eta_{\mathfrak{H}}^s(\mathcal{Z}), \aleph_{\mathfrak{H}}^s(\mathcal{Z}))$ is a picture fuzzy set.
- (v) The pairs $(\beth_{\mathfrak{H}}^c(\mathcal{Z}), \aleph_{\mathfrak{H}}^c(\mathcal{Z}))$ & $(\eta_{\mathfrak{H}}^c(\mathcal{Z}), \aleph_{\mathfrak{H}}^c(\mathcal{Z}))$ are coupled N-structures.
- (vi) The pairs $(\beth_{\mathfrak{H}}^c(\mathcal{Z}), \aleph_{\mathfrak{H}}^c(\mathcal{Z}))$ & $(\eta_{\mathfrak{H}}^c(\mathcal{Z}), \aleph_{\mathfrak{H}}^c(\mathcal{Z}))$ are bipolar fuzzy sets.

Proof The proof is straight forward by the definition of all fuzzy structures.

Definition 3.10

Let $\mathfrak{H} = \{ \langle \varkappa, \sqsupset_1^s(\varkappa), \eta_1^s(\varkappa), \aleph_1^s(\varkappa), \sqsupset_1^c(\varkappa), \eta_1^c(\varkappa), \aleph_1^c(\varkappa) \rangle, : \varkappa \in \wp \}$ and $\mathfrak{B} = \{ \langle \varkappa, \sqsupset_2^s(\varkappa), \eta_2^s(\varkappa), \aleph_2^s(\varkappa), \sqsupset_2^c(\varkappa), \eta_2^c(\varkappa), \aleph_2^c(\varkappa) \rangle, : \varkappa \in \wp \}$ be two BPFS. Then, union is defined as follows:

$$\mathfrak{H} \cup \mathfrak{B} = \left\{ \begin{array}{l} \varkappa, \max(\sqsupset_1^s(\varkappa), \sqsupset_2^s(\varkappa)), \min(\eta_1^s(\varkappa), \eta_2^s(\varkappa)), \min(\aleph_1^s(\varkappa), \aleph_2^s(\varkappa)); \\ \min(\sqsupset_1^c(\varkappa), \sqsupset_2^c(\varkappa)), \max(\eta_1^c(\varkappa), \eta_2^c(\varkappa)), \max(\aleph_1^c(\varkappa), \aleph_2^c(\varkappa)) : \varkappa \in \wp \end{array} \right\}$$

Example 3.11

Let $\wp = \{ \varkappa_1, \varkappa_2, \varkappa_3 \}$.

Then, $\mathfrak{H} = \left\{ \begin{array}{l} \langle \varkappa_1, 0.6, 0.4, 0.2, -0.5, -0.3, -0.02 \rangle \\ \langle \varkappa_2, 0.4, 0.3, 0.6, -0.03, -0.002, -0.4 \rangle \\ \langle \varkappa_3, 0.7, 0.04, 0.3, -0.2, -0.4, -0.05 \rangle \end{array} \right\}$ and

$\mathfrak{B} = \left\{ \begin{array}{l} \langle \varkappa_1, 0.4, 0.35, 0.1, -0.6, -0.4, -0.2 \rangle \\ \langle \varkappa_2, 0.3, 0.25, 0.5, -0.3, -0.145, -0.2 \rangle \\ \langle \varkappa_3, 0.4, 0.5, 0.2, -0.5, -0.45, -0.6 \rangle \end{array} \right\}$ are two BPFS in \wp .

Then their union is given as follows $\mathfrak{H} \cup \mathfrak{B} = \left\{ \begin{array}{l} \langle \varkappa_1, 0.6, 0.35, 0.1, -0.5, -0.3, -0.2 \rangle \\ \langle \varkappa_2, 0.4, 0.25, 0.5, -0.3, -0.002, -0.4 \rangle \\ \langle \varkappa_3, 0.7, 0.04, 0.2, -0.5, -0.45, -0.6 \rangle \end{array} \right\}$.

Definition 3.12

Let $\mathfrak{H} = \{ \langle \varkappa, \sqsupset_1^s(\varkappa), \eta_1^s(\varkappa), \aleph_1^s(\varkappa), \sqsupset_1^c(\varkappa), \eta_1^c(\varkappa), \aleph_1^c(\varkappa) \rangle, : \varkappa \in \wp \}$ and $\mathfrak{B} = \{ \langle \varkappa, \sqsupset_2^s(\varkappa), \eta_2^s(\varkappa), \aleph_2^s(\varkappa), \sqsupset_2^c(\varkappa), \eta_2^c(\varkappa), \aleph_2^c(\varkappa) \rangle, : \varkappa \in \wp \}$ be two BPFS. Then, intersection is then defined as follows:

$$\mathfrak{H} \cap \mathfrak{B} = \left\{ \begin{array}{l} \varkappa, \min(\sqsupset_1^s(\varkappa), \sqsupset_2^s(\varkappa)), \min(\eta_1^s(\varkappa), \eta_2^s(\varkappa)), \max(\aleph_1^s(\varkappa), \aleph_2^s(\varkappa)); \\ \max(\sqsupset_1^c(\varkappa), \sqsupset_2^c(\varkappa)), \max(\eta_1^c(\varkappa), \eta_2^c(\varkappa)), \min(\aleph_1^c(\varkappa), \aleph_2^c(\varkappa)) : \varkappa \in \wp \end{array} \right\}$$

Example 3.13

Let $\wp = \{ \varkappa, \varkappa_2, \varkappa \}$. Then, $\mathfrak{H} = \left\{ \begin{array}{l} \langle \varkappa_1, 0.1, 0.3, 0.2, -0.5, -0.2, -0.01 \rangle \\ \langle \varkappa_2, 0.4, 0.5, 0.6, -0.02, -0.004, -0.4 \rangle \\ \langle \varkappa_3, 0.6, 0.02, 0.3, -0.2, -0.4, -0.03 \rangle \end{array} \right\}$ and

$\mathfrak{B} = \left\{ \begin{array}{l} \langle \varkappa_1, 0.4, 0.25, 0.4, -0.5, -0.4, -0.1 \rangle \\ \langle \varkappa_2, 0.6, 0.35, 0.5, -0.3, -0.135, -0.2 \rangle \\ \langle \varkappa_3, 0.3, 0.5, 0.1, -0.4, -0.25, -0.6 \rangle \end{array} \right\}$ are two BPFS in \wp .

Then, their intersection is given as follows $\mathfrak{H} \cap \mathfrak{B} = \left\{ \begin{array}{l} \langle \varkappa_1, 0.1, 0.25, 0.4, -0.5, -0.2, -0.1 \rangle \\ \langle \varkappa_2, 0.4, 0.35, 0.6, -0.02, -0.004, -0.4 \rangle \\ \langle \varkappa_3, 0.3, 0.02, 0.3, -0.2, -0.25, -0.6 \rangle \end{array} \right\}$.

Proposition 3.14

For every BPFS $\mathfrak{H}, \mathfrak{B}, \mathfrak{C}$, the relation “ \subseteq ” is transitive.

Proposition 3.15

For every BPFS $\mathfrak{H}, \mathfrak{B}, \mathfrak{C}$, the operations \cap & \cup are commutative, associative, and distributive.

Definition 3.17

Let $\mathfrak{H} = \{ \langle \varkappa, \sqsupset^s(\varkappa), \eta^s(\varkappa), \aleph^s(\varkappa), \sqsupset^c(\varkappa), \eta^c(\varkappa), \aleph^c(\varkappa) \rangle : \varkappa \in \wp \}$ be a BPFS in X . The complement of \mathfrak{H} is then denoted by \mathfrak{H}^c and is defined by $\sqsupset_{\mathfrak{H}^c}^s(\varkappa) = 1 - \sqsupset_{\mathfrak{H}}^s(\varkappa)$, $\eta_{\mathfrak{H}^c}^s(\varkappa) = 1 - \eta_{\mathfrak{H}}^s(\varkappa)$, $\aleph_{\mathfrak{H}^c}^s(\varkappa) = 1 - \aleph_{\mathfrak{H}}^s(\varkappa)$ and $\sqsupset_{\mathfrak{H}^c}^c(\varkappa) = 1 - \sqsupset_{\mathfrak{H}}^c(\varkappa)$, $\eta_{\mathfrak{H}^c}^c(\varkappa) = 1 - \eta_{\mathfrak{H}}^c(\varkappa)$, $\aleph_{\mathfrak{H}^c}^c(\varkappa) = 1 - \aleph_{\mathfrak{H}}^c(\varkappa) \forall \varkappa \in \wp$.

Example 3.18

Let $\wp = \{ \varkappa_1, \varkappa_2, \varkappa_3 \}$.

Then, $\mathfrak{H} = \left\{ \begin{array}{l} \langle \varkappa_1, 0.6, 0.4, 0.2, -0.5, -0.3, -0.02 \rangle \\ \langle \varkappa_2, 0.4, 0.3, 0.6, -0.03, -0.002, -0.4 \rangle \\ \langle \varkappa_3, 0.7, 0.04, 0.3, -0.2, -0.4, -0.05 \rangle \end{array} \right\}$ is a BPFS in \wp . Then, the complement of \mathfrak{H} is given by $\mathfrak{H}^c = \left\{ \begin{array}{l} \langle \varkappa_1, 0.4, 0.6, 0.8, -0.5, -0.7, -0.98 \rangle \\ \langle \varkappa_2, 0.6, 0.7, 0.4, -0.97, -0.998, -0.6 \rangle \\ \langle \varkappa_3, 0.3, 0.96, 0.7, -0.8, -0.6, -0.95 \rangle \end{array} \right\}$.

Definition 3.19

Let $\mathfrak{H} = \{ \langle \varkappa, \sqsupset^s_{\mathfrak{H}}(\varkappa), \eta^s_{\mathfrak{H}}(\varkappa), \aleph^s_{\mathfrak{H}}(\varkappa), \sqsupset^c_{\mathfrak{H}}(\varkappa), \eta^c_{\mathfrak{H}}(\varkappa), \aleph^c_{\mathfrak{H}}(\varkappa) \rangle : \varkappa \in \wp \}$ is a BPFS of \wp . The model operators on \mathfrak{H} are denoted by $\boxtimes \mathfrak{H}$ and $\otimes \mathfrak{H}$, respectively, and are defined by

$\boxtimes \mathfrak{H} = \{ \langle \varkappa, \sqsupset^s_{\mathfrak{H}}(\varkappa), \eta^s_{\mathfrak{H}}(\varkappa), \aleph^s_{\mathfrak{H}}(\varkappa), -\sqsupset^s_{\mathfrak{H}}(\varkappa), -\eta^s_{\mathfrak{H}}(\varkappa), -\aleph^s_{\mathfrak{H}}(\varkappa) \rangle : \varkappa \in \wp \}$ and

$\otimes \mathfrak{H} = \{ \langle \varkappa, -\sqsupset^c_{\mathfrak{H}}(\varkappa), -\eta^c_{\mathfrak{H}}(\varkappa), -\aleph^c_{\mathfrak{H}}(\varkappa), \sqsupset^c_{\mathfrak{H}}(\varkappa), \eta^c_{\mathfrak{H}}(\varkappa), \aleph^c_{\mathfrak{H}}(\varkappa) \rangle : \varkappa \in \wp \}$.

Example 3.20

If $\mathfrak{H} = \{ (0,0.2,0.4,0.6,-0.3,-0.7,-0.1), (1,0.3,0.2,0.5,-0.1,-0.6,-0.7), (2,0.2,0.3,0.6,-0.2,-0.4,-0.5), (3,0.1,0.4,0.5,-0.7,-0.8, -0.2), (4,0.9,0.3,0.1,-0.6,-0.2,-0.4) \}$ is BPFS in a set $\wp = \{0, 1, 2, 3, 4\}$.

Then

$\boxtimes \mathfrak{H} = \{ (0,0.2,0.4,0.6,-0.2, -0.4,-0.6)(1,0.3,0.2,0.5,-0.3,-0.2,-0.5)(2,0.2,0.3,0.6,-0.2,-0.3,-0.6), (3,0.1,0.4,0.5,-0.1,-0.4,-0.5), (4,0.9,0.3,0.1,-0.9,-0.3,-0.1) \}$ and

$\otimes \mathfrak{H} = \{ (0,0.3,0.7,0.1,-0.3,-0.7,-0.1), (1,0.1,0.6,0.7,-0.1,-0.6, 0.7), (2,0.2,0.4,0.5,-0.2,-0.4,-0.5), (3,0.7,0.8,0.2,-0.7,-0.8,-0.2), (4, 0.6,0.2,0.4,-0.6,-0.2,-0.4) \}$

Definition 3.21

Let $\mathfrak{H}_1 = \{(\underline{\mathfrak{I}}^s_1, \eta^s_1, \aleph^s_1, \underline{\mathfrak{I}}^c_1, \eta^c_1, \aleph^c_1)\}$ and $\mathfrak{H}_2 = \{(\underline{\mathfrak{I}}^s_2, \eta^s_2, \aleph^s_2, \underline{\mathfrak{I}}^c_2, \eta^c_2, \aleph^c_2)\}$ be two BPFN. The operations for these numbers are then defined as follows:

- (i) $\alpha \mathfrak{H}_1 = (1 - (1 - \underline{\mathfrak{I}}^s_1)^\alpha, (\eta^s_1)^\alpha, (\aleph^s_1)^\alpha, -(-\underline{\mathfrak{I}}^c_1)^\alpha, -(-\eta^c_1)^\alpha, -(1 - (1 - (\aleph^c_1)^\alpha))^\alpha)$
- (ii) $\mathfrak{H}_1^\alpha = ((\underline{\mathfrak{I}}^s_1)^\alpha, 1 - (1 - \eta^s_1)^\alpha, 1 - (1 - \nu^s_1)^\alpha, - (1 - (1 - (-\underline{\mathfrak{I}}^c_1)^\alpha)^\alpha), -(-\eta^c_1)^\alpha, -(-\aleph^c_1)^\alpha)$
- (iii) $\mathfrak{H}_1 + \mathfrak{H}_2 = (\underline{\mathfrak{I}}^s_1 + \underline{\mathfrak{I}}^s_2 - \underline{\mathfrak{I}}^s_1 \cdot \underline{\mathfrak{I}}^s_2, \eta^s_1 \eta^s_2, \aleph^s_1 \aleph^s_2, -\underline{\mathfrak{I}}^c_1 \cdot \underline{\mathfrak{I}}^c_2, -(-\eta^c_1 - \eta^c_2 - \eta^c_1 \cdot \eta^c_2), -(-\aleph^c_1 - \aleph^c_2 - \aleph^c_1 \cdot \aleph^c_2))$
- (iv) $\mathfrak{H}_1 \mathfrak{H}_2 = (\underline{\mathfrak{I}}^s_1 \underline{\mathfrak{I}}^s_2, \eta^s_1 + \eta^s_2 - \eta^s_1 \cdot \eta^s_2, \aleph^s_1 + \aleph^s_2 - \aleph^s_1 \cdot \aleph^s_2, -(-\underline{\mathfrak{I}}^c_1 - \underline{\mathfrak{I}}^c_2 - \underline{\mathfrak{I}}^c_1 \cdot \underline{\mathfrak{I}}^c_2), (\eta^c_1 \eta^c_2), (\aleph^c_1 \aleph^c_2))$ where $\alpha > 0$.

4 Product on Bipolar Picture Fuzzy Set

In this section propose the product definition of BPFS and some operators on BPF sets.

Definition 4.1

Let X and Y be two universes and let

$\mathfrak{H} = \{(x, \underline{\mathfrak{I}}^s_{\mathfrak{H}}(x), \eta^s_{\mathfrak{H}}(x), \aleph^s_{\mathfrak{H}}(x), \underline{\mathfrak{I}}^c_{\mathfrak{H}}(x), \eta^c_{\mathfrak{H}}(x), \aleph^c_{\mathfrak{H}}(x)) : x \in \wp\}$ and $\mathfrak{B} = \{(y, \underline{\mathfrak{I}}^s_{\mathfrak{B}}(y), \eta^s_{\mathfrak{B}}(y), \aleph^s_{\mathfrak{B}}(y), \underline{\mathfrak{I}}^c_{\mathfrak{B}}(y), \eta^c_{\mathfrak{B}}(y), \aleph^c_{\mathfrak{B}}(y)) : y \in Y\}$ be two BPFS.

The Cartesian product of these two BPFS is defined as follows,

$$\mathfrak{H} \times \mathfrak{B} = \left\{ (x, y), \min(\underline{\mathfrak{I}}^s_{\mathfrak{H}}(x), \underline{\mathfrak{I}}^s_{\mathfrak{B}}(y)), \min(\eta^s_{\mathfrak{H}}(x), \eta^s_{\mathfrak{B}}(y)), \max(\aleph^s_{\mathfrak{H}}(x), \aleph^s_{\mathfrak{B}}(y)); \max(\underline{\mathfrak{I}}^c_{\mathfrak{H}}(x), \underline{\mathfrak{I}}^c_{\mathfrak{B}}(y)), \max(\eta^c_{\mathfrak{H}}(x), \eta^c_{\mathfrak{B}}(y)), \min(\aleph^c_{\mathfrak{H}}(x), \aleph^c_{\mathfrak{B}}(y)) : x \in \wp, y \in Y \right\}$$

Example 4.2

Assume the sets $\wp = \{x_1, x_2, x_3\}$ and $Y = \{y_1, y_2\}$. Then

$\mathfrak{H} = \{(x_1, 0.2, 0.4, 0.6, -0.3, -0.7, -0.1), (x_2, 0.3, 0.2, 0.5, -0.1, -0.6, -0.7), (x_3, 0.2, 0.3, 0.6, -0.2, -0.4, -0.5)\}$ and $\mathfrak{B} = \{(y_1, 0.1, 0.3, 0.5, -0.2, -0.6, -0.4), (y_2, 0.2, 0.4, 0.5, -0.1, -0.6, -0.3)\}$ be two BPFS in \wp . Then

$\mathfrak{H} \times \mathfrak{B} = \{((x_1, y_1), 0.1, 0.3, 0.6, -0.2, -0.6, -0.4), ((x_1, y_2), 0.2, 0.4, 0.6, -0.1, -0.6, -0.3), ((x_2, y_1), 0.1, 0.2, 0.5, -0.1, -0.6, -0.7), ((x_2, y_2), 0.2, 0.4, 0.5, -0.1, -0.6, -0.7), ((x_3, y_1), 0.1, 0.3, 0.6, -0.2, -0.4, -0.5), ((x_3, y_2), 0.2, 0.3, 0.6, -0.1, -0.4, -0.5)\}$

Theorem 4.3

Let \mathfrak{H} and \mathfrak{B} are any two BPFS of \wp . Then, $\mathfrak{H} \times \mathfrak{B}$ is a BPFS of $\wp \times \wp$.

Proof Take (x_1, y_1) & $(x_2, y_2) \in \wp \times \wp$,
Then

$$\begin{aligned} &(\sqsupset^s \mathfrak{H} \times \sqsupset^s \mathfrak{B})[(x_1, y_1).(x_2, y_2)] \\ &= (\sqsupset^s \mathfrak{H} \times \sqsupset^s \mathfrak{B})[(x_1.x_2), (y_1.y_2)] \\ &= \min \{ \sqsupset^s \mathfrak{H}(x_1.x_2), \sqsupset^s \mathfrak{B}(y_1.y_2) \} \\ &= \min \{ \min \{ \sqsupset^s \mathfrak{H}(x_1), \sqsupset^s \mathfrak{H}(x_2) \}, \min \{ \sqsupset^s \mathfrak{B}(y_1), \sqsupset^s \mathfrak{B}(y_2) \} \} \\ &= \min \{ \min \{ \sqsupset^s \mathfrak{H}(x_1), \sqsupset^s \mathfrak{B}(y_1) \}, \min \{ \sqsupset^s \mathfrak{H}(x_2), \sqsupset^s \mathfrak{B}(y_2) \} \} \\ &= \min \{ (\sqsupset^s \mathfrak{H} \times \sqsupset^s \mathfrak{B})(x_1, y_1), (\sqsupset^s \mathfrak{H} \times \sqsupset^s \mathfrak{B})(x_2, y_2) \} \end{aligned}$$

Similarly, $(\eta^s \mathfrak{H} \times \eta^s \mathfrak{B})[(x_1, y_1).(x_2, y_2)] = \min \{ (\eta^s \mathfrak{H} \times \eta^s \mathfrak{B})(x_1, y_1), (\eta^s \mathfrak{H} \times \eta^s \mathfrak{B})(x_2, y_2) \}$

$$\begin{aligned} &(\aleph^s \mathfrak{H} \times \aleph^s \mathfrak{B})[(x_1, y_1).(x_2, y_2)] = (\aleph^s \mathfrak{H} \times \aleph^s \mathfrak{B})[(x_1.x_2), (y_1.y_2)] \\ &= \max \{ \aleph^s \mathfrak{H}(x_1.x_2), \aleph^s \mathfrak{B}(y_1.y_2) \} \\ &= \max \{ \max \{ \aleph^s \mathfrak{H}(x_1), \aleph^s \mathfrak{H}(x_2) \}, \max \{ \aleph^s \mathfrak{B}(y_1), \aleph^s \mathfrak{B}(y_2) \} \} \\ &= \max \{ \max \{ \aleph^s \mathfrak{H}(x_1), \aleph^s \mathfrak{B}(y_1) \}, \max \{ \aleph^s \mathfrak{H}(x_2), \aleph^s \mathfrak{B}(y_2) \} \} \\ &= \max \{ (\aleph^s \mathfrak{H} \times \aleph^s \mathfrak{B})(x_1, y_1), (\aleph^s \mathfrak{H} \times \aleph^s \mathfrak{B})(x_2, y_2) \} \end{aligned}$$

$$\begin{aligned} &\max \sqsupset \times \sqsupset^c \mathfrak{B})[(x_1, y_1).(x_2, y_2)] = (\sqsupset^c \mathfrak{H} \times \sqsupset^c \mathfrak{B})[(x_1.x_2), (y_1.y_2)] \\ &= \max \{ \sqsupset^c \mathfrak{H}(x_1.x_2), \sqsupset^c \mathfrak{B}(y_1.y_2) \} \\ &= \max \{ \max \{ \sqsupset^c \mathfrak{H}(x_1), \sqsupset^c \mathfrak{H}(x_2) \}, \max \{ \sqsupset^c \mathfrak{B}(y_1), \sqsupset^c \mathfrak{B}(y_2) \} \} \\ &= \max \{ \max \{ \sqsupset^c \mathfrak{H}(x_1), \sqsupset^c \mathfrak{B}(y_1) \}, \max \{ \sqsupset^c \mathfrak{H}(x_2), \sqsupset^c \mathfrak{B}(y_2) \} \} \\ &= \max \{ (\sqsupset^c \mathfrak{H} \times \sqsupset^c \mathfrak{B})(x_1, y_1), (\sqsupset^c \mathfrak{H} \times \sqsupset^c \mathfrak{B})(x_2, y_2) \} \end{aligned}$$

$$(\eta^s \mathfrak{H} \times \eta^s \mathfrak{B})[(x_1, y_1).(x_2, y_2)] = \max \{ (\eta^s \mathfrak{H} \times \eta^s \mathfrak{B})(x_1, y_1), (\eta^s \mathfrak{H} \times \eta^s \mathfrak{B})(x_2, y_2) \}$$

$$\begin{aligned} &(\aleph^c \mathfrak{H} \times \aleph^c \mathfrak{B})[(x_1, y_1).(x_2, y_2)] = (\aleph^c \mathfrak{H} \times \aleph^c \mathfrak{B})[(x_1.x_2), (y_1.y_2)] \\ &= \min \{ \aleph^c \mathfrak{H}(x_1.x_2), \aleph^c \mathfrak{B}(y_1.y_2) \} \\ &= \min \{ \min \{ \aleph^c \mathfrak{H}(x_1), \aleph^c \mathfrak{H}(x_2) \}, \min \{ \aleph^c \mathfrak{B}(y_1), \aleph^c \mathfrak{B}(y_2) \} \} \\ &= \min \{ \min \{ \aleph^c \mathfrak{H}(x_1), \aleph^c \mathfrak{B}(y_1) \}, \min \{ \aleph^c \mathfrak{H}(x_2), \aleph^c \mathfrak{B}(y_2) \} \} \\ &= \min \{ (\aleph^c \mathfrak{H} \times \aleph^c \mathfrak{B})(x_1, y_1), (\aleph^c \mathfrak{H} \times \aleph^c \mathfrak{B})(x_2, y_2) \} \end{aligned}$$

Theorem 4.4

If $\mathfrak{H} \times \mathfrak{B}$ is a BPFS of $\wp \times \wp$, then either \mathfrak{H} or \mathfrak{B} is BPFS of \wp .

Proof Since $\mathfrak{H} \times \mathfrak{B}$ BPFS of $\wp \times \wp$,

$$(\sqsupset^s \mathfrak{H} \times \sqsupset^s \mathfrak{B})[(x_1, y_1).(x_2, y_2)] = \min \{ (\sqsupset^s \mathfrak{H} \times \sqsupset^s \mathfrak{B})(x_1, y_1), (\sqsupset^s \mathfrak{H} \times \sqsupset^s \mathfrak{B})(x_2, y_2) \} \quad (1)$$

Put $x_1 = x_2 = 0$ in (1) we get

$$(\sqsupset^s_{\mathfrak{H}} \times \sqsupset^s_{\mathfrak{B}})[(0, y_1).(0, y_2)] = \min\{(\sqsupset^s_{\mathfrak{H}} \times \sqsupset^s_{\mathfrak{B}})(0, y_1), (\sqsupset^s_{\mathfrak{H}} \times \sqsupset^s_{\mathfrak{B}})(0, y_2)\} \text{ and } (\sqsupset^s_{\mathfrak{H}} \times \sqsupset^s_{\mathfrak{B}})(0, (y_1.y_2)) = \min\{(\sqsupset^s_{\mathfrak{H}} \times \sqsupset^s_{\mathfrak{B}})(0, y_1), (\sqsupset^s_{\mathfrak{H}} \times \sqsupset^s_{\mathfrak{B}})(0, y_2)\}.$$

Thus, we have $\sqsupset^s_{\mathfrak{B}}(y_1.y_2) = \min\{\sqsupset^s_{\mathfrak{B}}(y_1), \sqsupset^s_{\mathfrak{B}}(y_2)\}$.

Similarly, $\eta^s_{\mathfrak{B}}(y_1.y_2) = \min\{\eta^s_{\mathfrak{B}}(y_1), \eta^s_{\mathfrak{B}}(y_2)\}$.

$$(\aleph^s_{\mathfrak{H}} \times \aleph^s_{\mathfrak{B}})[(x_1, y_1).(x_2, y_2)] = \max\{(\aleph^s_{\mathfrak{H}} \times \aleph^s_{\mathfrak{B}})(x_1, y_1), (\aleph^s_{\mathfrak{H}} \times \aleph^s_{\mathfrak{B}})(x_2, y_2)\} \quad (2)$$

Put $x_1 = x_2 = 0$ in (2) we get

$$(\aleph^s_{\mathfrak{H}} \times \aleph^s_{\mathfrak{B}})[(0, y_1).(0, y_2)] = \max\{(\aleph^s_{\mathfrak{H}} \times \aleph^s_{\mathfrak{B}})(0, y_1), (\aleph^s_{\mathfrak{H}} \times \aleph^s_{\mathfrak{B}})(0, y_2)\} \text{ and } (\aleph^s_{\mathfrak{H}} \times \aleph^s_{\mathfrak{B}})(0, (y_1.y_2)) = \max\{(\aleph^s_{\mathfrak{H}} \times \aleph^s_{\mathfrak{B}})(0, y_1), (\aleph^s_{\mathfrak{H}} \times \aleph^s_{\mathfrak{B}})(0, y_2)\}.$$

Thus we have $\aleph^s_{\mathfrak{B}}(y_1.y_2) = \max\{\aleph^s_{\mathfrak{B}}(y_1), \aleph^s_{\mathfrak{B}}(y_2)\}$.

$$(\sqsubset^c_{\mathfrak{H}} \times \sqsubset^c_{\mathfrak{B}})[(x_1, y_1).(x_2, y_2)] = \max\{(\sqsubset^c_{\mathfrak{H}} \times \sqsubset^c_{\mathfrak{B}})(x_1, y_1), (\sqsubset^c_{\mathfrak{H}} \times \sqsubset^c_{\mathfrak{B}})(x_2, y_2)\} \quad (3)$$

Put $x_1 = x_2 = 0$ in (3) we get.

$$(\sqsubset^c_{\mathfrak{H}} \times \sqsubset^c_{\mathfrak{B}})[(0, y_1).(0, y_2)] = \max\{(\sqsubset^c_{\mathfrak{H}} \times \sqsubset^c_{\mathfrak{B}})(0, y_1), (\sqsubset^c_{\mathfrak{H}} \times \sqsubset^c_{\mathfrak{B}})(0, y_2)\} \text{ and } (\sqsubset^c_{\mathfrak{H}} \times \sqsubset^c_{\mathfrak{B}})(0, (y_1.y_2)) = \max\{(\sqsubset^c_{\mathfrak{H}} \times \sqsubset^c_{\mathfrak{B}})(0, y_1), (\sqsubset^c_{\mathfrak{H}} \times \sqsubset^c_{\mathfrak{B}})(0, y_2)\}.$$

Thus, we have $\sqsubset^c_{\mathfrak{B}}(y_1.y_2) = \max\{\sqsubset^c_{\mathfrak{B}}(y_1), \sqsubset^c_{\mathfrak{B}}(y_2)\}$.

Similarly, $\eta^c_{\mathfrak{B}}(y_1.y_2) = \max\{\eta^c_{\mathfrak{B}}(y_1), \eta^c_{\mathfrak{B}}(y_2)\}$

$$(\aleph^c_{\mathfrak{H}} \times \aleph^c_{\mathfrak{B}})[(x_1, y_1).(x_2, y_2)] = \min\{(\aleph^c_{\mathfrak{H}} \times \aleph^c_{\mathfrak{B}})(x_1, y_1), (\aleph^c_{\mathfrak{H}} \times \aleph^c_{\mathfrak{B}})(x_2, y_2)\}. \quad (4)$$

Put $x_1 = x_2 = 0$ in (4) we get

$$(\aleph^c_{\mathfrak{H}} \times \aleph^c_{\mathfrak{B}})[(0, y_1).(0, y_2)] = \min\{(\aleph^c_{\mathfrak{H}} \times \aleph^c_{\mathfrak{B}})(0, y_1), (\aleph^c_{\mathfrak{H}} \times \aleph^c_{\mathfrak{B}})(0, y_2)\} \text{ and } (\aleph^c_{\mathfrak{H}} \times \aleph^c_{\mathfrak{B}})(0, (y_1.y_2)) = \min\{(\aleph^c_{\mathfrak{H}} \times \aleph^c_{\mathfrak{B}})(0, y_1), (\aleph^c_{\mathfrak{H}} \times \aleph^c_{\mathfrak{B}})(0, y_2)\}.$$

Thus, we have $\aleph^c_{\mathfrak{B}}(y_1.y_2) = \min\{\aleph^c_{\mathfrak{B}}(y_1), \aleph^c_{\mathfrak{B}}(y_2)\}$.

Definition 4.5

Let $\mathfrak{H} = \{\langle x, \sqsupset^s_{\mathfrak{H}}(x), \eta^s_{\mathfrak{H}}(x), \aleph^s_{\mathfrak{H}}(x), \sqsubset^c_{\mathfrak{H}}(x), \eta^c_{\mathfrak{H}}(x), \aleph^c_{\mathfrak{H}}(x) \rangle : x \in \wp\}$ and $\mathfrak{B} = \{\langle y, \sqsupset^s_{\mathfrak{B}}(y), \eta^s_{\mathfrak{B}}(y), \aleph^s_{\mathfrak{B}}(y), \sqsubset^c_{\mathfrak{B}}(y), \eta^c_{\mathfrak{B}}(y), \aleph^c_{\mathfrak{B}}(y) \rangle : y \in Y\}$ be two BPFs of \wp and Y . The modal operator defined on the product $\mathfrak{H} \times \mathfrak{B}$ of $\wp \times Y$,

- (i) $\boxtimes(\mathfrak{H} \times \mathfrak{B}) = ((x, y), \sqsupset^s_{\mathfrak{H} \times \mathfrak{B}}(x, y), \eta^s_{\mathfrak{H} \times \mathfrak{B}}(x, y), \aleph^s_{\mathfrak{H} \times \mathfrak{B}}(x, y), -\sqsupset^s_{\mathfrak{H} \times \mathfrak{B}}(x, y), -\eta^s_{\mathfrak{H} \times \mathfrak{B}}(x, y), -\aleph^s_{\mathfrak{H} \times \mathfrak{B}}(x, y))$
- (ii) $\otimes(\mathfrak{H} \times \mathfrak{B}) = ((x, y), -\sqsubset^c_{\mathfrak{H} \times \mathfrak{B}}(x, y), -\eta^c_{\mathfrak{H} \times \mathfrak{B}}(x, y), -\aleph^c_{\mathfrak{H} \times \mathfrak{B}}(x, y), \sqsubset^c_{\mathfrak{H} \times \mathfrak{B}}(x, y), \eta^c_{\mathfrak{H} \times \mathfrak{B}}(x, y), \aleph^c_{\mathfrak{H} \times \mathfrak{B}}(x, y))$

are BPFs of $\wp \times Y$.

Example 4.6

Assume the sets $\wp = \{z_1, z_2, z_3\}$ and $Y = \{y_1, y_2\}$. If

$\mathfrak{H} = \{(z_1, .0.2,0.4,0.6,-0.3,-0.7,-0.1),(z_2,0.3,0.2,0.5,-0.1,-0.6,-0.7),(z_3,0.2,0.3,0.6,-0.2,-0.4,-0.5)\}$ and $\mathfrak{B} = \{(y_1,0.1,0.3,0.5,-0.2,-0.6,-0.4),(y_2,0.2,0.4,0.5,-0.1,-0.6,-0.3)\}$ are two BPFS in \wp of Y , then

(i) $\boxtimes(\mathfrak{H} \times \mathfrak{B}) = \{((z_1, y_1),0.1,0.3,0.6,-0.1,-0.3,-0.6),((z_1, y_2),0.2,0.4,0.6,-0.2,-0.4,-0.6),((z_2, y_1),0.1,0.2,0.5,-0.1,-0.2,-0.5), ((z_2, y_2),0.2,0.4,0.5,-0.2,-0.4,-0.5),((z_3, y_1),0.1,0.3,0.6,-0.1,-0.3,-0.6),((z_3,y_2),0.2,0.3,0.6,-0.2,-0.3,-0.6)\}$

(ii) $\otimes(\mathfrak{H} \times \mathfrak{B}) = \{((z_1,y_1), 0.2,0.6,0.4,-0.2,-0.6,-0.4),((z_1,y_2),0.1,0.6,0.3,-0.1,-0.6,-0.3),((z_2,y_1),0.1,0.6,0.7,-0.1,-0.6,-0.7), ((z_2,y_2),0.1,0.6,0.7,-0.1,-0.6,-0.7),((z_3,y_1),0.2,0.4,0.5,-0.2,-0.4,-0.5),((z_3,y_2),0.1,0.4,0.5,-0.1,-0.4,-0.5)\}$

5 Application of Bipolar Picture Fuzzy Set

In this section, the stress level of the people in the Banking, E-commerce, Education, and IT sectors are to be measured. For our convenience, the above four sectors can be represented as $S_1, S_2, S_3,$ and S_4 . To assess the measure, the data have been collected from 100 respondents. As per the expert’s advice five major characteristics have been identified which are named as H_1, H_2, H_3, H_4, H_5 . The characteristics to examine the optimality of the process are service provider exposure (H_1), Mind level accessing of customer (H_2), Network feasibility (H_3), Communication platform (H_4), and Customer satisfaction (H_5). The researcher framed the questions which are more relevant to the people of $S_1, S_2, S_3,$ and S_4 . After receiving the responses from the people, grades have been assigned for their responses. The questions are precisely structured with three-point Likert scale which are agree, neutral, and disagree. The scale of agree takes the positive satisfactory, disagree takes the negative satisfactory (counter), and neutral takes the intermediate.

The following algorithm has been proposed for the purpose of selection.

- (i) Design HBPVF over \wp .
- (ii) Construct $H[D : \wp]$
- (iii) Find PI ($H[D : \wp]$)
- (iv) Inference

Step 1

The responses are enforced in the algorithm, and the values are estimated accordingly. Added all the positive satisfactory responses of each statement and divided by the number of respondents to get the average value. In the same way, the neutral responses of each statement were summed up and divided by the number of respondents to get the related value. Likewise, in each statement the negative satisfactory responses have been added and divided by the number of respondents to get the value.

Step 2

The following table shows the respective membership values towards the average value of respondents of the select sectors in the identification of most stress against their influenced characteristics.

H→ K ↓	H ₁	H ₂	H ₃	H ₄	H ₅
S ₁	(0.5,0.6,0.7, -0.4,-0.1, -0.01)	(0.4,0.5,0.6, -0.3,-0.2, -0.02)	(0.6,0.3,0.5, -0.4,-0.3, -0.03)	(0.3,0.2,0.4, -0.4,-0.1, -0.04)	(0.2,0.1,0.3, -0.1,-0.3, -0.05)
S ₂	(0.1,0.3,0.5, -0.1,-0.2, -0.07)	(0.2,0.4,0.1, -0.3,-0.1, -0.06)	(0.3,0.5,0.2, -0.2,-0.4, -0.05)	(0.4,0.6,0.3, -0.4,-0.2, -0.04)	(0.5,0.2,0.4, -0.1,-0.3, -0.02)
S ₃	(0.7,0.4,0.2, -0.5,-0.3, -0.03)	(0.6,0.3,0.1, -0.4,-0.2, -0.02)	(0.5,0.2,0.3, -0.5,-0.4, -0.01)	(0.4,0.1,0.5, -0.6,-0.1, -0.05)	(0.8,0.5,0.4, -0.6,-0.3, -0.06)
S ₄	(0.4,0.6,0.5, -0.1,-0.4, -0.06)	(0.5,0.7,0.6, -0.2,-0.5, -0.01)	(0.3,0.4,0.5, -0.1,-0.4, -0.04)	(0.2,0.1,0.6, -0.7,-0.4, -0.05)	(0.6,0.3,0.4, -0.1,-0.6, -0.03)

Step 3

The HBPF- set $H[D : \wp]$ has been attained using the Definition 3.5

$$H[D : \wp] = \left\{ \begin{array}{l} (0.4, 0.34, 0.5, -0.32, -0.2, -0.03) \\ (0.3, 0.4, 0.3, -0.22, -0.24, -0.048) \\ (0.6, 0.3, 0.3, -0.52, -0.26, -0.034) \\ (0.4, 0.42, 0.52, -0.24, -0.46, -0.03) \end{array} \right\}$$

Step 4

Find the PI for all PI's using the Definition 3.6

$$PI(S_1) = 1.7549$$

$$PI(S_2) = 1.2438$$

$$PI(S_3) = 2.0155$$

$$PI(S_4) = 1.4655$$

6 Conclusion

The concept of a bipolar picture fuzzy set is explored in this paper, and its properties are discussed with appropriate examples. Then, some operations like addition, subtraction, intersection, and union on bipolar picture fuzzy set were presented. Furthermore, the modal operators and its associated properties have been studied. Also, the H-Bipolar picture fuzzy set is defined and with the help of decision set, an algorithm is proposed to identify a decision-making model. This model is applied in this study to analyse the stress level of the people in the different sectors Banking (S_1), E-commerce (S_2), Education (S_3), and IT sectors (S_4). The calculated values of this analysis have shown that the stress level of the Education (S_3) sector is high compared to other sectors. In general, Education sector would be more effective if it has face-to-face interaction. In this scenario, the absence of such interaction may cause the high stress. The concept of Modal operators, BFN, and performance index of Bipolar Fuzzy Number was defined to elaborate the theoretical extension of the bipolar picture fuzzy set. In comparison with all the existing decision-making methodologies, so many matrix techniques and weighted operating techniques have been used. But we have provided a simplest algorithm with the help of performance index. So, the proposed algorithm may reduce the time consumption by taking a decision. In future, the concept of bipolar picture fuzzy set will be enforced into the other mathematical and management models.

References

1. Zadeh, L.A.: Fuzzy sets Inform. Control. **8**(3), 338–353 (1965)
2. Atanassov, K.T.: Intuitionistic fuzzy sets, fuzzy sets and systems. J. Math. Appl. **20**(1), 87–96 (1986)
3. Smarandache, F., Ali, M.: Interval valued bipolar fuzzy weighted neutrosophic sets and their application. A A. **10**(1) (2016)
4. Song, S.Z., Khan, M., Smarandache, F., Jun, Y.B.: A novel extension of neutrosophic sets and its application in BCK/BCI-algebras. Infinite Study (2018)
5. Muthumeenakshi, M., Muralikrishna, P., Sabarinathan, S.: Bipolar valued Q-fuzzy application in building sciences. Int. J. Civil Eng. Technol. **9**(5), 761–765 (2018)
6. Cuong, B.C.: Picture fuzzy set first results part-1, seminar neuro-fuzzy systems with applications. Preprint 03/2013, Institute of Mathematics, Hanoi
7. Cuong, B. C.: Picture fuzzy sets-first results. Part 2, seminar neuro-fuzzy systems with applications. Institute of Mathematics, Hanoi (2013)
8. Cuong, B.C., Van Hai, P.: Some fuzzy logic operators for picture fuzzy sets. In: 2015 Seventh International Conference on Knowledge and Systems Engineering (KSE), pp. 132–137 (2015)
9. Cu'ong, B.C.: Picture fuzzy sets. J. Comput. Sci. Cybernet. **30**(4), 409–420 (2014)
10. Dutta, P.: Medical diagnosis via distance measures on picture fuzzy sets. AMSE J.–AMSE IIETA Publ.-2017-Ser. Adv. A. **54**(2), 657–672 (2017)
11. Dutta, P., Ganju, S.: Some aspects of picture fuzzy set. Trans. A. Razmadze Math. Inst. **172**(2), 164–175 (2018)
12. Ashraf, S., Mehmood, T., Abdullah, S., Khan, Q.: Picture fuzzy linguistic sets and their applications for multi-attribute group. Nucleus **55**(2), 66–73 (2018)

13. Ashraf, S., Abdullah, S., Qadir, A.: Novel concept of cubic picture fuzzy sets. *J. New Theory*. **24**, 59–72 (2018)
14. Zhang, W.R.: Bipolar fuzzy sets. *Proc. FUZZ-IEEE*, 835–840 (1998)
15. Lee, K.M.: Bipolar valued fuzzy sets and their applications. In: *Proceedings of International Conference on Intelligent Technologies Bangkok, Thailand*, pp. 307–312 (2000)
16. Lee, K.J.: Bipolar fuzzy subalgebras and bipolar fuzzy ideals of BCK/BCI- algebras. *Bull. Malays. Math. Sci. Soc.* **32**(3), 361–373 (2009)
17. Nisar, F.: Bipolar valued fuzzy K-subalgebras. *World Appl. Sci. J.* **14**(12), 1914–1919 (2011)
18. Muralikrishna, P., Chandramouleeswaran, M.: Study on N-ideal of BF-algebras. *Int. J. Pure Appl. Math.* **83**(4), 607–612 (2013)
19. Prince Williams, D.R., Ahn, S.S., Jun, Y.B.: A coupled N-structure with an application in a subtraction algebra. *Honam Math. J.* **36**(4), 863–884 (2014)
20. Jun, Y.B., Ahn, S.S., Prince Williams, D.R.: Coupled N-structures and its application in BCK/BCI-algebras. *Iran. J. Sci. Technol.*, 133–140 (2013)
21. Jun, Y.B., Smarandache, F., Bordbar, H.: Neutrosophic N-structures applied to BCK/BCI-algebras. *Information* **8**(128), 1–12 (2017)
22. Deli, I., Ali, M., Smarandache, F.: Bipolar neutrosophic sets and their application based on multi-criteria decision-making problems. In: *2015 International Conference on Advanced Mechatronic Systems (ICAMechS)*, pp. 249–254 (2015)
23. Phong, P.H., Hieu, D.T., Ngan, R.T., Them, P.T.: Some compositions of picture fuzzy relations. In: *Proceedings of the 7th National Conference on Fundamental and Applied Information Technology Research, Thai Nguyen*, pp. 19–20 (2014)
24. Wei, G.: Picture fuzzy aggregation operators and their application to multiple attribute decision making. *J. Intell. Fuzzy Syst.* **33**(2), 713–724 (2017)
25. Abdullah, S., Ashraf, S.: Series of aggregation operators for picture fuzzy environments and their applications, aggregation operators for picture fuzzy sets. In: *Handbook of Research on Emerging Applications of Fuzzy Algebraic Structures*, pp. 328–351 (2020)

Asymptotic Behavior of Resolvents on Complete Geodesic Spaces With General Perturbation Functions



Yasunori Kimura and Keisuke Shindo

Abstract Asymptotic behavior of resolvents for a proper lower semicontinuous functions is studied under various settings. In particular, resolvents are defined as a lot of different forms depending on perturbation functions. In this paper, we consider asymptotic behavior of the resolvents on geodesic spaces with general perturbation functions.

Keywords Asymptotic behavior · Geodesic space · Convex function · Resolvent

Mathematics Subject Classification 52A41

1 Introduction

Asymptotic behavior of resolvents for a proper lower semicontinuous functions is an important problem of convex minimization and has been studied in various settings of spaces. For example, there are results in Hilbert spaces. Let $f : H \rightarrow]-\infty, \infty]$ be a function on a Hilbert space H and suppose that f is proper, lower semicontinuous, and convex. Define a resolvent $J_\lambda : H \rightarrow H$ of f with a parameter $\lambda \in]0, \infty[$ by

$$J_\lambda x = \operatorname{argmin}_{y \in H} \left\{ f(y) + \frac{1}{\lambda} \|x - y\|^2 \right\}$$

for all $x \in H$. Then the following hold; see [4].

Theorem 1.1 *Let H be a Hilbert space and $f : H \rightarrow]-\infty, \infty]$ a proper lower semicontinuous convex function. If $\{J_{\mu_n} x\}_{n \in \mathbb{N}}$ is bounded for some sequence $\{\mu_n\} \subset]0, \infty[$ such that $\mu_n \rightarrow \infty$, then $\operatorname{argmin} f \neq \emptyset$ and*

Y. Kimura (✉) · K. Shindo

Department of Information Science, Toho University, Miyama, 274-8510 Funabashi, Chiba, Japan
e-mail: yasunori@is.sci.toho-u.ac.jp

K. Shindo

e-mail: 7520001s@st.toho-u.jp

$$\lim_{\lambda \rightarrow \infty} J_\lambda x = P_{\operatorname{argmin} f} x.$$

Theorem 1.2 *Let H be a Hilbert space and $f : H \rightarrow] - \infty, \infty]$ a proper lower semicontinuous convex function. Then*

$$\lim_{\lambda \rightarrow +0} J_\lambda x = P_{\overline{\operatorname{dom} f}} x.$$

On the other hand, geodesic spaces are metric spaces on which the mapping called geodesic is defined. A CAT(0) space is an example of geodesic spaces and a generalization of Hilbert spaces. CAT(1) spaces are also similar. On geodesic spaces, there are different types of resolvents depending on the perturbation functions. For instance, two types of resolvents are studied intensively on CAT(1) space. One is a resolvent whose perturbation function is $\tan(\cdot) \sin(\cdot)$, and the other is a resolvent whose perturbation function is $-\log(\cos(\cdot))$; see [3] and [1]. Also, the authors studied asymptotic behavior of these resolvents. Although resolvents are different from each other depending on perturbation functions, they have some common properties. In this paper, we consider asymptotic behavior of the resolvent of general perturbation functions using these properties.

2 Preliminaries

Let X be a metric space. For $x, y \in X$, a geodesic with endpoints x and y is a mapping $c : [0, d(x, y)] \rightarrow X$ satisfying $c(0) = x, c(d(x, y)) = y$, and $d(c(u), c(v)) = |u - v|$ for $u, v \in [0, d(x, y)]$. X is called a geodesic space if a geodesic with endpoints x and y exists for any $x, y \in X$. In what follows, we assume that any pair of points in a geodesic space has a unique geodesic. Let x, y be points in a geodesic space X . We define a geodesic segment joining x, y by the image of a geodesic c with endpoints x, y , and it is denoted by $[x, y]$. Then, for $t \in [0, 1]$ and $x, y \in X$, there exists a unique point $z \in [x, y]$ such that $d(x, z) = (1 - t)d(x, y)$ and $d(z, y) = td(x, y)$, which is denoted by $z = tx \oplus (1 - t)y$. For $\kappa \in \mathbb{R}$, let M_κ^2 be a two dimensional model space with a curvature κ , and D_κ is its diameter. For three points x_1, x_2, x_3 in a geodesic space X such that $d(x_1, x_2) + d(x_2, x_3) + d(x_3, x_1) < 2D_\kappa$, we define a geodesic triangle $\Delta(x_1, x_2, x_3) \subset X$ with vertices x_1, x_2, x_3 as the union of three geodesic segments joining each pair of vertices. A comparison triangle $\overline{\Delta}(\bar{x}_1, \bar{x}_2, \bar{x}_3) \subset M_\kappa^2$ for $\Delta(x_1, x_2, x_3) \subset X$ is a triangle whose vertices satisfy

$$\begin{cases} d(x_1, x_2) = d_{M_\kappa^2}(\bar{x}_1, \bar{x}_2), \\ d(x_2, x_3) = d_{M_\kappa^2}(\bar{x}_2, \bar{x}_3), \\ d(x_3, x_1) = d_{M_\kappa^2}(\bar{x}_3, \bar{x}_1). \end{cases}$$

If for any $p, q \in \Delta(x_1, x_2, x_3)$ and their comparison points $\bar{p}, \bar{q} \in \bar{\Delta}(\bar{x}_1, \bar{x}_2, \bar{x}_3)$, the inequality

$$d(p, q) \leq d_{M_\kappa^2}(\bar{p}, \bar{q})$$

is satisfied for all triangle in X , then X is called a $CAT(\kappa)$ space, and the inequality is called a $CAT(\kappa)$ inequality. X is called admissible if any $x, y \in X$ satisfy

$$d(x, y) < \frac{D_\kappa}{2}.$$

A sequence $\{x_n\}$ is said to be spherically bounded if there exists $a \in X$ such that

$$\sup_{n \in \mathbb{N}} d(x_n, a) < \frac{D_\kappa}{2}.$$

We describe fundamental properties of complete $CAT(0)$ and $CAT(1)$ spaces. The following inequalities are direct results obtained from $CAT(\kappa)$ inequalities.

Theorem 2.1 *Let X be a complete $CAT(0)$ space, $x, y, z \in X$, and $0 < t < 1$. Then*

$$d(tx \oplus (1 - t)y, z)^2 \leq td(x, z)^2 + (1 - t)d(y, z)^2 - t(1 - t)d(x, y)^2.$$

Theorem 2.2 *Let X be a complete admissible $CAT(1)$ space, $x, y, z \in X$, and $0 < t < 1$. Then*

$$\cos d(tx \oplus (1 - t)y, z) \sin d(x, y) \geq \cos d(x, z) \sin td(x, y) + \cos d(y, z) \sin(1 - t)d(x, y).$$

In particular, for $t = 1/2$, it holds that

$$\cos d\left(\frac{1}{2}x \oplus \frac{1}{2}y, z\right) \cos \frac{d(x, y)}{2} \geq \frac{1}{2} \cos d(x, z) + \frac{1}{2} \cos d(y, z),$$

and

$$\begin{aligned} & -\log \cos d\left(\frac{1}{2}x \oplus \frac{1}{2}y, z\right) \\ & \leq \frac{1}{2}(-\log(\cos d(x, z))) + \frac{1}{2}(-\log(\cos d(y, z))) + \log \cos \frac{d(x, y)}{2}. \end{aligned}$$

Let X be a complete $CAT(0)$ space. Then for $x \in X$, P_C is a metric projection onto C if and only if $P : X \rightarrow C$ satisfies

$$d(x, P_C x)^2 + d(P_C x, y)^2 \leq d(x, y)^2$$

for all $y \in C$ and $x \in X$. Let X be a complete admissible CAT(1) space and C a closed convex subset of X . Then for $x \in X$, P is a metric projection onto C if and only if $P : X \rightarrow C$ satisfies

$$\cos d(x, Px) \cos d(Px, y) \geq \cos d(x, y)$$

for all $y \in C$ and $x \in X$.

3 Resolvent on CAT(0) spaces

We consider asymptotic behavior of the resolvent with a general perturbation function on CAT(0) space. Let X be a complete CAT(0) space and φ a function $[0, \infty[\rightarrow \mathbb{R}$. We say φ satisfies the condition (A) if the following hold.

- $\varphi(0) = 0$;
- φ is increasing;
- φ is continuous;
- $\varphi(d(\cdot, x))$ is strictly convex for all $x \in X$;
- $\varphi(t) - Ct \rightarrow \infty$ as $t \rightarrow \infty$, for all constants C .

For example, if $\varphi(t) = t^2$ or $\varphi(t) = \tanh t \sinh t$, φ satisfies the condition (A). Let $f : X \rightarrow]-\infty, \infty[$ be a proper lower semicontinuous function, x a point in X and λ a positive real number. If φ satisfies the condition (A), then the function $g(\cdot) = f(\cdot) + \frac{1}{\lambda}\varphi(d(\cdot, x))$ has a unique minimizer on X and the resolvent J_λ with a perturbation function φ is a mapping defined as follows;

$$J_\lambda x = \operatorname{argmin}_{y \in X} \left\{ f(y) + \frac{1}{\lambda}\varphi(d(y, x)) \right\}.$$

Now we describe the properties of this resolvent.

Lemma 3.1 *Let X be a complete CAT(0) space, $f : X \rightarrow]-\infty, \infty[$ a proper lower semicontinuous function and $x \in X$. Suppose $0 < \lambda \leq \mu$, then $d(J_\lambda x, x) \leq d(J_\mu x, x)$ and $f(J_\lambda x) \geq f(J_\mu x)$ for all $x \in X$.*

Proof First we show $d(J_\lambda x, x) \leq d(J_\mu x, x)$. By the definition of the resolvent, we have

$$f(J_\lambda x) + \frac{1}{\lambda}\varphi(d(J_\lambda x, x)) \leq f(J_\mu x) + \frac{1}{\lambda}\varphi(d(J_\mu x, x))$$

and

$$f(J_\mu x) + \frac{1}{\mu}\varphi(d(J_\mu x, x)) \leq f(J_\lambda x) + \frac{1}{\mu}\varphi(d(J_\lambda x, x)).$$

Adding both sides of these inequalities, we get

$$\left(\frac{1}{\lambda} - \frac{1}{\mu}\right) \varphi(d(J_\lambda x, x)) \leq \left(\frac{1}{\lambda} - \frac{1}{\mu}\right) \varphi(d(J_\mu x, x)).$$

Since $\lambda \leq \mu$, we obtain

$$\varphi(d(J_\lambda x, x)) \leq \varphi(d(J_\mu x, x)).$$

This is equivalent to $d(J_\lambda x, x) \leq d(J_\mu x, x)$ as φ is increasing. Next we show $f(J_\lambda x) \geq f(J_\mu x)$. By the definition of the resolvent, we have

$$f(J_\mu x) - f(J_\lambda x) \leq \frac{1}{\mu} \{ \varphi(d(J_\lambda x, x)) - \varphi(d(J_\mu x, x)) \}.$$

Since $d(J_\lambda x, x) - d(J_\mu x, x) \leq 0$, we get $f(J_\mu x) - f(J_\lambda x) \leq 0$. This is the desired result.

Lemma 3.2 *Let X be a complete CAT(0) space, $f : X \rightarrow]-\infty, \infty]$ a proper lower semicontinuous function and $x \in X$. Suppose $0 < \lambda \leq \mu$, then*

$$d(J_\lambda x, J_\mu x)^2 \leq d(J_\mu x, x)^2 - d(J_\lambda x, x)^2$$

for all $x \in X$.

Proof By the definition of the resolvent, for t with $0 < t < 1$, we have

$$\begin{aligned} & f(J_\lambda x) + \frac{1}{\lambda} \varphi(d(J_\lambda x, x)) \\ & \leq f(tJ_\lambda x \oplus (1-t)J_\mu x) + \frac{1}{\lambda} \varphi(d(tJ_\lambda x \oplus (1-t)J_\mu x, x)) \\ & \leq tf(J_\lambda x) + (1-t)f(J_\mu x) + \frac{1}{\lambda} \varphi(d(tJ_\lambda x \oplus (1-t)J_\mu x, x)) \\ & \leq f(J_\lambda x) + \frac{1}{\lambda} \varphi(d(tJ_\lambda x \oplus (1-t)J_\mu x, x)). \end{aligned}$$

Then we obtain $d(J_\lambda x, x) \leq d(tJ_\lambda x \oplus (1-t)J_\mu x, x)$ and therefore,

$$\begin{aligned} & d(J_\lambda x, x)^2 \\ & \leq d(tJ_\lambda x \oplus (1-t)J_\mu x, x)^2 \\ & \leq td(J_\lambda x, x)^2 + (1-t)d(J_\mu x, x)^2 - t(1-t)d(J_\lambda x, J_\mu x)^2. \end{aligned}$$

Hence we have

$$t(1-t)d(J_\lambda x, J_\mu x)^2 \leq (1-t)d(J_\mu x, x)^2 - (1-t)d(J_\lambda x, x)^2.$$

Dividing by $(1-t)$ and letting $t \rightarrow 0$, we get

$$d(J_\lambda x, J_\mu x)^2 \leq d(J_\mu x, x)^2 - d(J_\lambda x, x)^2,$$

which is the desired result.

Lemma 3.3 *Let X be a complete CAT(0) space, $f : X \rightarrow]-\infty, \infty]$ a proper lower semicontinuous function and $x \in X$. If $\{J_{\mu_n} x\}_{n \in \mathbb{N}}$ is bounded for some sequence $\{\mu_n\} \subset]0, \infty[$ such that $\mu_n \rightarrow \infty$, Then for every increasing sequence $\{\lambda_n\}$ such that $\lambda_n \rightarrow \infty$, $\{J_{\lambda_n} x\}_{n \in \mathbb{N}}$ is bounded.*

Proof Since $\{J_{\mu_n} x\}_{n \in \mathbb{N}}$ is bounded, there exists $M > 0$ such that for all $n \in \mathbb{N}$

$$d(J_{\mu_n} x, x) \leq M.$$

Let $\{\lambda_n\}$ be an increasing sequence such that $\lambda_n \rightarrow \infty$. Suppose $\{J_{\lambda_n} x\}_{n \in \mathbb{N}}$ is not bounded. Then there exists a subsequence $\{\lambda_{n_i}\}$ of $\{\lambda_n\}$ such that

$$d(J_{\lambda_{n_i}} x, x) > M$$

for all $i \in \mathbb{N}$. Then there exists $l \in \mathbb{N}$, $\lambda_{n_i} \leq \mu_l$. From Lemma 3.1, we have

$$M < d(J_{\lambda_{n_i}} x, x) \leq d(J_{\mu_l} x, x) \leq M.$$

This is a contradiction and we have the desired result.

Theorem 3.4 *Let X be a complete CAT(0) space, $f : X \rightarrow]-\infty, \infty]$ a proper lower semicontinuous function, and $x \in X$. Let $\varphi : [0, \infty[\rightarrow \mathbb{R}$ be a function satisfying the condition (A) and J_μ a resolvent with a perturbation function φ for $\mu \in]0, \infty[$. If $\{J_{\mu_n} x\}_{n \in \mathbb{N}}$ is bounded for some sequence $\{\mu_n\} \subset]0, \infty[$ such that $\mu_n \rightarrow \infty$, then $\operatorname{argmin} f \neq \emptyset$ and*

$$\lim_{\lambda \rightarrow \infty} J_\lambda x = P_{\operatorname{argmin} f} x.$$

Proof Let $\{\lambda_n\} \subset \mathbb{R}$ be an increasing sequence such that $\lambda_n \rightarrow \infty$ and denote $J_{\lambda_n} x$ by x_n . Since $\{d(x_n, x)\}_{n \in \mathbb{N}}$ is bounded and increasing by Lemma 3.1, it is a convergent sequence.

Suppose $n \leq m$. Then we have

$$\begin{aligned} & f(x_n) + \frac{1}{\lambda_n} \varphi(d(x_n, x)) \\ & \leq f\left(\frac{1}{2}x_n \oplus \frac{1}{2}x_m\right) + \frac{1}{\lambda_n} \varphi\left(d\left(\frac{1}{2}x_n \oplus \frac{1}{2}x_m, x\right)\right) \\ & \leq \frac{1}{2}f(x_n) + \frac{1}{2}f(x_m) + \frac{1}{\lambda_n} \varphi\left(d\left(\frac{1}{2}x_n \oplus \frac{1}{2}x_m, x\right)\right) \\ & \leq f(x_n) + \frac{1}{\lambda_n} \varphi\left(d\left(\frac{1}{2}x_n \oplus \frac{1}{2}x_m, x\right)\right). \end{aligned}$$

It implies $d(x_n, x) \leq d\left(\frac{1}{2}x_n \oplus \frac{1}{2}x_m, x\right)$ and therefore,

$$\begin{aligned} d(x_n, x)^2 &\leq d\left(\frac{1}{2}x_n \oplus \frac{1}{2}x_m, x\right)^2 \\ &\leq \frac{1}{2}d(x_n, x)^2 + \frac{1}{2}d(x_m, x)^2 - \frac{1}{4}d(x_n, x_m)^2. \end{aligned}$$

Then we have

$$d(x_n, x_m)^2 \leq 2\{d(x_m, x)^2 - d(x_n, x)^2\}.$$

Since $\{d(x_n, x)^2\}_{n \in \mathbb{N}}$ is convergent, $\{x_n\}_{n \in \mathbb{N}}$ is a Cauchy sequence. By the completeness of X , $\{x_n\}_{n \in \mathbb{N}}$ is convergent to some $p \in X$. Next we show $p = P_{\text{argmin}f}x$. By the definition of the resolvent, we have

$$f(x_n) + \frac{1}{\lambda_n}\varphi(d(x_n, x)) \leq f(y) + \frac{1}{\lambda_n}\varphi(d(y, x))$$

for all $y \in X$. By the lower semicontinuity of f , we get

$$f(p) \leq \lim_{n \rightarrow \infty} \left(f(x_n) + \frac{1}{\lambda_n}\varphi(d(x_n, x)) \right) \leq \lim_{n \rightarrow \infty} \left(f(y) + \frac{1}{\lambda_n}\varphi(d(y, x)) \right) = f(y)$$

for all $y \in X$. It follows that $p \in \text{argmin}f$, and hence $\text{argmin}f \neq \emptyset$. Let $a \in \text{argmin}f$. Then

$$\begin{aligned} &f(x_n) + \frac{1}{\lambda_n}\varphi(d(x_n, x)) \\ &\leq f\left(\frac{1}{2}x_n \oplus \frac{1}{2}a, x\right) + \frac{1}{\lambda_n}\varphi\left(d\left(\frac{1}{2}x_n \oplus \frac{1}{2}a, x\right)\right) \\ &\leq \frac{1}{2}f(x_n) + \frac{1}{2}f(a) + \frac{1}{\lambda_n}\varphi\left(d\left(\frac{1}{2}x_n \oplus \frac{1}{2}a, x\right)\right) \\ &\leq f(x_n) + \frac{1}{\lambda_n}\varphi\left(d\left(\frac{1}{2}x_n \oplus \frac{1}{2}a, x\right)\right). \end{aligned}$$

It implies $d(x_n, x) \leq d\left(\frac{1}{2}x_n \oplus \frac{1}{2}y, x\right)$, and therefore,

$$d(x_n, x)^2 \leq d\left(\frac{1}{2}x_n \oplus \frac{1}{2}a, x\right)^2 \leq \frac{1}{2}d(x_n, x)^2 + \frac{1}{2}d(a, x)^2.$$

Hence $d(x_n, x) \leq d(a, x)$. Letting $n \rightarrow \infty$, we obtain that $d(p, x) \leq d(a, x)$ for all $a \in \text{argmin}f$. It implies $p = P_{\text{argmin}f}x$. Since $\{J_{\lambda_n}x\}_{n \in \mathbb{N}}$ converges to p for any

increasing sequences diverging to ∞ , we conclude

$$\lim_{\lambda \rightarrow \infty} J_\lambda x = P_{\operatorname{argmin} f} x.$$

This completed the proof.

Theorem 3.5 *Let X be a complete CAT(0) space, $f : X \rightarrow]-\infty, \infty]$ a proper lower semicontinuous function and $x \in X$. Let $\varphi : [0, \infty[\rightarrow \mathbb{R}$ be a function satisfying the condition (A) and J_μ a resolvent with a perturbation function φ for $\mu \in]0, \infty[$. Then the function $\lambda \mapsto J_\lambda x$ is continuous on $]0, \infty[$, and*

$$\lim_{\lambda \rightarrow +0} J_\lambda x = P_{\operatorname{dom} f} x.$$

Proof First we show that $\lambda \mapsto J_\lambda x$ is continuous on $]0, \infty[$. Let $\{\lambda_n\} \subset \mathbb{R}$ be a monotone increasing or decreasing sequence such that $\lambda_n \rightarrow \lambda_0 \in]0, \infty[$. Let $n, m \in \mathbb{N}$ with $n \leq m$. From Lemma 3.2, we have

$$d(J_{\lambda_n} x, J_{\lambda_m} x)^2 \leq |d(J_{\lambda_n} x, x)^2 - d(J_{\lambda_m} x, x)^2|.$$

Since a sequence $\{d(J_{\lambda_n} x, x)^2\}_{n \in \mathbb{N}}$ is increasing (decreasing) and bounded above (below), respectively, it is convergent. Therefore, $\{J_{\lambda_n} x\}_{n \in \mathbb{N}}$ is a Cauchy sequence on X . By the completeness of X , $\{J_{\lambda_n} x\}_{n \in \mathbb{N}}$ is convergent. Assume $J_{\lambda_n} x \rightarrow u$. For all $y \in X$,

$$f(J_{\lambda_n} x) + \frac{1}{\lambda_n} \varphi(d(J_{\lambda_n} x, x)) \leq f(y) + \frac{1}{\lambda_n} \varphi(d(y, x)).$$

Since f is lower semicontinuous, letting $n \rightarrow \infty$, we have

$$\begin{aligned} f(u) + \frac{1}{\lambda_0} \varphi(d(u, x)) &\leq \liminf_{n \rightarrow \infty} \left(f(J_{\lambda_n} x) + \frac{1}{\lambda_n} \varphi(d(J_{\lambda_n} x, x)) \right) \\ &\leq \liminf_{n \rightarrow \infty} \left(f(y) + \frac{1}{\lambda_n} \varphi(d(y, x)) \right) \\ &= f(y) + \frac{1}{\lambda_0} \varphi(d(y, x)), \end{aligned}$$

for all $y \in X$, and therefore $u = \operatorname{argmin} \left\{ f(\cdot) + \frac{1}{\lambda_0} \varphi(d(\cdot, x)) \right\} = J_{\lambda_0} x$. This implies that the function $\lambda \mapsto J_\lambda x$ is continuous on $]0, \infty[$.

Next we show $J_\lambda x \rightarrow P_{\operatorname{dom} f} x$ as $\lambda \rightarrow +0$. Let $\{\lambda_n\} \subset]0, \infty[$ be a decreasing sequence such that $\lambda_n \rightarrow 0$ and $\{z_m\} \subset \operatorname{dom} f$ such that $z_m \rightarrow P_{\operatorname{dom} f} x$. Put $x_n = J_{\lambda_n} x$ and $p = P_{\operatorname{dom} f} x$. Then we have

$$f(x_n) + \frac{1}{\lambda_n} \varphi(d(x_n, x)) \leq f(z_m) + \frac{1}{\lambda_n} \varphi(d(z_m, x)).$$

By the monotonicity of $\{f(x_n)\}_{n \in \mathbb{N}}$, there exists $\alpha \in \mathbb{R}$ such that

$$\varphi(d(x_n, x)) \leq \lambda_n(f(z_m) - \alpha) + \varphi(d(z_m, x)).$$

In fact, since $\lambda_n \leq \lambda_1$ for all $n \in \mathbb{N}$ and $f(J_\lambda x)$ is decreasing for λ , we have $f(x_1) \leq f(x_n)$. Putting $\alpha = f(x_1)$, we obtain the inequality above. Letting $n \rightarrow \infty$, we have

$$\limsup_{n \rightarrow \infty} \varphi(d(x_n, x)) \leq \varphi(d(z_m, x)),$$

which is equivalent to

$$\limsup_{n \rightarrow \infty} d(x_n, x)^2 \leq d(z_m, x)^2.$$

On the other hand, from the property of metric projection, we have

$$d(x_n, p)^2 \leq d(x_n, x)^2 - d(p, x)^2.$$

Consequently,

$$\begin{aligned} \limsup_{n \rightarrow \infty} d(x_n, p)^2 &\leq \limsup_{n \rightarrow \infty} d(x_n, x)^2 - d(p, x)^2 \\ &\leq d(z_m, x)^2 - d(p, x)^2. \end{aligned}$$

Letting $m \rightarrow \infty$, we have

$$\limsup_{n \rightarrow \infty} d(x_n, p)^2 \leq 0,$$

and hence $d(x_n, p) \rightarrow 0$. Then we obtain

$$\lim_{n \rightarrow \infty} J_{\lambda_n} x = p.$$

Since $\{J_{\lambda_n} x\}_{n \in \mathbb{N}}$ converges to p for any decreasing sequences converging to 0, we conclude

$$\lim_{\lambda \rightarrow +0} J_\lambda x = p,$$

which complete the proof.

4 Resolvents on CAT(1) Spaces

We consider asymptotic behavior with a general perturbation function on CAT(1) spaces.

Let X be a complete admissible CAT(1) space. We say $\varphi : [0, \pi/2[\rightarrow \mathbb{R}$ satisfies the condition (B) if the following hold.

- $\varphi(0) = 0$;
- φ is increasing;
- φ is continuous;
- $\varphi(d(\cdot, x))$ is strictly convex for all $x \in X$;
- $\varphi(t) \rightarrow \infty$ as $t \rightarrow \pi/2$.

For example, if $\varphi(t) = -\log(\cos(t))$ or $\varphi(t) = \tan t \sin t$, then φ satisfies the condition (B). Let f be a proper lower semicontinuous convex function and $x \in X$. If φ satisfies the condition (B), then the function $g = f(\cdot) + \frac{1}{\lambda}\varphi(d(\cdot, x))$ has a unique minimizer. See [2]. We define the resolvent J_λ with a perturbation function φ by

$$J_\lambda x = \operatorname{argmin}_{y \in X} \left\{ f(y) + \frac{1}{\lambda} \varphi(d(y, x)) \right\}$$

for $x \in X$.

Lemma 4.1 *Let X be a complete admissible CAT(1) space, $f : X \rightarrow]-\infty, \infty]$ a proper lower semicontinuous function and $x \in X$. Suppose $0 < \lambda \leq \mu$. Then $d(J_\lambda x, x) \leq d(J_\mu x, x)$ and $f(J_\lambda x) \geq f(J_\mu x)$ for all $x \in X$.*

Proof First we show $d(J_\lambda x, x) \leq d(J_\mu x, x)$. By the definition of the resolvent, we have

$$f(J_\lambda x) + \frac{1}{\lambda} \varphi(d(J_\lambda x, x)) \leq f(J_\mu x) + \frac{1}{\lambda} \varphi(d(J_\mu x, x))$$

and

$$f(J_\mu x) + \frac{1}{\mu} \varphi(d(J_\mu x, x)) \leq f(J_\lambda x) + \frac{1}{\mu} \varphi(d(J_\lambda x, x)).$$

Adding both side of these inequalities, we get

$$\left(\frac{1}{\lambda} - \frac{1}{\mu} \right) \varphi(d(J_\lambda x, x)) \leq \left(\frac{1}{\lambda} - \frac{1}{\mu} \right) \varphi(d(J_\mu x, x)).$$

Since $\lambda \leq \mu$, we obtain

$$\varphi(d(J_\lambda x, x)) \leq \varphi(d(J_\mu x, x)).$$

and this is equivalent to $d(J_\lambda x, x) \leq d(J_\mu x, x)$ as φ is increasing. Next we show $f(J_\lambda x) \geq f(J_\mu x)$. By the definition of the resolvent, we have

$$f(J_\mu x) - f(J_\lambda x) \leq \frac{1}{\mu} \left\{ \varphi(d(J_\lambda x, x)) - \varphi(d(J_\mu x, x)) \right\}.$$

Since $d(J_\lambda x, x) - d(J_\mu x, x) \leq 0$, we get $f(J_\mu x) - f(J_\lambda x) \leq 0$. This is the desired result.

Lemma 4.2 *Let X be a complete admissible CAT(1) space, $f : X \rightarrow]-\infty, \infty]$ a proper lower semicontinuous function, and $x \in X$. Suppose $0 < \lambda \leq \mu$, Then for λ, μ with $\lambda \leq \mu$,*

$$-\log(\cos d(J_\lambda x, J_\mu x)) \leq -\log(\cos d(J_\mu x, x)) - (-\log(\cos d(J_\lambda x, x))).$$

Proof From the definition of the resolvent, for t with $0 < t < 1$, we have

$$\begin{aligned} & f(J_\lambda x) + \frac{1}{\lambda} \varphi(d(J_\lambda x, x)) \\ & \leq f(tJ_\lambda x \oplus (1-t)J_\mu x) + \frac{1}{\lambda} \varphi(d(tJ_\lambda x \oplus (1-t)J_\mu x, x)) \\ & \leq tf(J_\lambda x) + (1-t)f(J_\mu x) + \frac{1}{\lambda} \varphi(d(tJ_\lambda x \oplus (1-t)J_\mu x, x)) \\ & \leq f(J_\lambda x) + \frac{1}{\lambda} \varphi(d(tJ_\lambda x \oplus (1-t)J_\mu x, x)). \end{aligned}$$

Then we obtain $d(J_\lambda x, x) \leq d(tJ_\lambda x \oplus (1-t)J_\mu x, x)$ and therefore,

$$\begin{aligned} & \cos d(J_\lambda x, x) \sin d(J_\lambda x, J_\mu x) \\ & \geq \cos d(tJ_\lambda x \oplus (1-t)J_\mu x, x) \sin d(J_\lambda x, J_\mu x) \\ & \geq \cos d(J_\lambda x, x) \sin td(J_\lambda x, J_\mu x) + \cos d(J_\mu x, x) \sin(1-t)d(J_\lambda x, J_\mu x). \end{aligned}$$

Hence we have

$$\begin{aligned} & \cos d(J_\lambda x, x) \sin(1-t)d(J_\lambda x, J_\mu x) \\ & \leq \cos d(J_\mu x, x) (\sin d(J_\lambda x, J_\mu x) - \sin td(J_\lambda x, J_\mu x)) \\ & \leq \cos d(J_\mu x, x) \left(2 \cos \frac{(1+t)d(J_\lambda x, J_\mu x)}{2} \sin \frac{(1-t)d(J_\lambda x, J_\mu x)}{2} \right) \\ & \leq \cos d(J_\mu x, x) \cos \frac{(1+t)d(J_\lambda x, J_\mu x)}{2} \sin(1-t)d(J_\lambda x, J_\mu x). \end{aligned}$$

Dividing by $\sin(1-t)d(J_\lambda x, J_\mu x)$ and letting $t \rightarrow 1$, we have

$$\cos d(J_\mu x, x) \leq \cos d(J_\lambda x, x) \cos d(J_\lambda x, J_\mu x).$$

Then we have

$$-\log(\cos d(J_\lambda x, J_\mu x)) \leq -\log(\cos d(J_\mu x, x)) - (-\log \cos(d(J_\lambda x, x))).$$

This is the desired result.

Lemma 4.3 *Let X be a complete admissible CAT(1) space, $f : X \rightarrow]-\infty, \infty]$ a proper lower semicontinuous function, and $x \in X$. If $\sup_{n \in \mathbb{N}} d(J_{\mu_n} x, x) < \pi/2$ for*

some sequence $\{\mu_n\} \subset]0, \infty[$ such that $\mu_n \rightarrow \infty$, then for all increasing sequence $\{\lambda_n\}$ such that $\lambda_n \rightarrow \infty$, $\sup_{n \in \mathbb{N}} d(J_{\lambda_n} x, x) < \pi/2$.

Proof By the assumption,

$$M := \sup_{n \in \mathbb{N}} d(J_{\mu_n} x, x) < \frac{\pi}{2}.$$

Let $\{\lambda_n\}$ be an increasing sequence such that $\lambda_n \rightarrow \infty$. Suppose $\sup_{n \in \mathbb{N}} d(J_{\lambda_n} x, x) = \pi/2$. Thus there exists $\lambda_{n_0} \in \mathbb{N}$ such that

$$d(J_{\lambda_{n_0}} x, x) > M.$$

Then there exists $l \in \mathbb{N}$ such that $\lambda_{n_0} \leq \mu_l$. From Lemma 4.1, we have $d(J_{\lambda_{n_0}} x, x) \leq d(J_{\mu_l} x, x)$. Then we have

$$M < d(J_{\lambda_{n_0}} x, x) \leq d(J_{\mu_l} x, x) \leq M.$$

This is a contradiction and we have the desired result.

Theorem 4.4 *Let X be a complete admissible CAT(1) space, $f : X \rightarrow]-\infty, \infty]$ a proper lower semicontinuous function and $x \in X$. Let $\varphi : [0, \pi/2[\rightarrow \mathbb{R}$ be a function satisfying the condition (B) and J_μ a resolvent with a perturbation function φ for $\mu \in]0, \infty[$. If $\sup_{n \in \mathbb{N}} d(J_{\mu_n} x, x) < \pi/2$ for some sequence $\{\mu_n\} \subset \mathbb{R}$ such that $\mu_n \rightarrow \infty$, then $\operatorname{argmin} f \neq \emptyset$ and*

$$\lim_{\lambda \rightarrow \infty} J_\lambda x = P_{\operatorname{argmin} f} x.$$

Proof Let $\{\lambda_n\} \subset \mathbb{R}$ be an increasing sequence such that $\lambda_n \rightarrow \infty$ and denote $J_{\lambda_n} x$ by x_n . Since $\sup_{n \in \mathbb{N}} d(x_n, x) < \pi/2$ and increasing by Lemma 4.2, $\{-\log(\cos d(x_n, x))\}_{n \in \mathbb{N}}$ is a convergent sequence.

Suppose $n \leq m$. Then we have

$$\begin{aligned} & f(x_n) + \frac{1}{\lambda_n} \varphi(d(x_n, x)) \\ & \leq f\left(\frac{1}{2}x_n \oplus \frac{1}{2}x_m\right) + \frac{1}{\lambda_n} \varphi\left(d\left(\frac{1}{2}x_n \oplus \frac{1}{2}x_m, x\right)\right) \\ & \leq \frac{1}{2}f(x_n) + \frac{1}{2}f(x_m) + \frac{1}{\lambda_n} \varphi\left(d\left(\frac{1}{2}x_n \oplus \frac{1}{2}x_m, x\right)\right) \\ & \leq f(x_n) + \frac{1}{\lambda_n} \varphi\left(d\left(\frac{1}{2}x_n \oplus \frac{1}{2}x_m, x\right)\right). \end{aligned}$$

This implies $d(x_n, x) \leq d\left(\frac{1}{2}x_n \oplus \frac{1}{2}x_m, x\right)$ and therefore,

$$\begin{aligned}
 & -\log(\cos d(x_n, x)) \\
 & \leq -\log\left(\cos d\left(\frac{1}{2}x_n \oplus \frac{1}{2}x_m, x\right)\right) \\
 & \leq \frac{1}{2}(-\log \cos(d(x_n, x))) + \frac{1}{2}(-\log(\cos d(x_m, x))) - \frac{1}{4}(-\log(\cos d(x_n, x_m))).
 \end{aligned}$$

Then we have

$$-\log(\cos(d(x_n, x_m))) \leq 2\{-\log(\cos(d(x_m, x))) - (-\log(\cos d(x_n, x)))\}.$$

Since $\{-\log(\cos(d(x_n, x)))\}_{n \in \mathbb{N}}$ is convergent, $\{x_n\}_{n \in \mathbb{N}}$ is a Cauchy sequence. By the completeness of X , $\{x_n\}_{n \in \mathbb{N}}$ is convergent to some $p \in X$. Next we show $p = P_{\text{argmin}f}x$. From the definition of the resolvent, we have

$$f(x_n) + \frac{1}{\lambda_n}\varphi(d(x_n, x)) \leq f(y) + \frac{1}{\lambda_n}\varphi(d(y, x)),$$

for all $y \in X$. Considering lower semicontinuity of f , we get

$$f(p) \leq \lim_{n \rightarrow \infty} \left(f(x_n) + \frac{1}{\lambda_n}\varphi(d(x_n, x)) \right) \leq \lim_{n \rightarrow \infty} \left(f(y) + \frac{1}{\lambda_n}\varphi(d(y, x)) \right) = f(y)$$

for all $y \in X$. It follows that $p \in \text{argmin}f$, and hence $\text{argmin}f \neq \emptyset$. For $a \in \text{argmin}f$,

$$\begin{aligned}
 & f(x_n) + \frac{1}{\lambda_n}\varphi(d(x_n, x)) \\
 & \leq f\left(\frac{1}{2}x_n \oplus \frac{1}{2}a, x\right) + \frac{1}{\lambda_n}\varphi\left(d\left(\frac{1}{2}x_n \oplus \frac{1}{2}a, x\right)\right) \\
 & \leq \frac{1}{2}f(x_n) + \frac{1}{2}f(a) + \frac{1}{\lambda_n}\varphi\left(d\left(\frac{1}{2}x_n \oplus \frac{1}{2}a, x\right)\right) \\
 & \leq f(x_n) + \frac{1}{\lambda_n}\varphi\left(d\left(\frac{1}{2}x_n \oplus \frac{1}{2}a, x\right)\right).
 \end{aligned}$$

This implies $d(x_n, x) \leq d\left(\frac{1}{2}x_n \oplus \frac{1}{2}y, x\right)$, and therefore,

$$\begin{aligned}
 -\log(\cos(d(x_n, x))) & \leq -\log\left(\cos\left(d\left(\frac{1}{2}x_n \oplus \frac{1}{2}a, x\right)\right)\right) \\
 & \leq \frac{1}{2}(-\log(\cos d(x_n, x))) + \frac{1}{2}(-\log(\cos d(a, x))),
 \end{aligned}$$

and hence $d(x_n, x) \leq d(a, x)$. Letting $n \rightarrow \infty$, we obtain that $d(p, x) \leq d(a, x)$ for all $a \in \operatorname{argmin} f$. This implies $p = P_{\operatorname{argmin} f} x$. Since $\{J_{\lambda_n} x\}_{n \in \mathbb{N}}$ converges to p for any increasing sequences divergent to ∞ , we conclude

$$\lim_{\lambda \rightarrow \infty} J_\lambda x = P_{\operatorname{argmin} f} x,$$

which is the desired result.

Theorem 4.5 *Let X be a complete admissible CAT(1) space, $f : X \rightarrow]-\infty, \infty]$ a proper lower semicontinuous function and $x \in X$. Let $\varphi : [0, \pi/2[\rightarrow \mathbb{R}$ be a function satisfying the condition (B) and J_μ a resolvent with a perturbation function φ for $\mu \in]0, \infty[$. Then the function $\lambda \mapsto J_\lambda x$ is continuous on $]0, \infty[$ and*

$$\lim_{\lambda \rightarrow +0} J_\lambda x = P_{\operatorname{dom} f} x.$$

Proof First we show that $\lambda \mapsto J_\lambda x$ is continuous on $]0, \infty[$. Let $\{\lambda_n\} \subset]0, \infty[$ be a monotone increasing or decreasing sequence such that $\lambda_n \rightarrow \lambda_0 \in]0, \infty[$. We suppose $n, m \in \mathbb{N}$ and $n \leq m$. From Lemma 4.2, we have

$$-\log(\cos d(J_{\lambda_n} x, J_{\lambda_m} x)) \leq |-\log(\cos d(J_{\lambda_n} x, x)) - (-\log(\cos d(J_{\lambda_m} x, x)))|.$$

Since a sequence $\{-\log(\cos d(J_{\lambda_n} x, x))\}_{n \in \mathbb{N}}$ is increasing (decreasing) and bounded above (below) respectively, it is convergent. Therefore $\{J_{\lambda_n} x\}_{n \in \mathbb{N}}$ is a Cauchy sequence on X . By the completeness of X , $\{J_{\lambda_n} x\}_{n \in \mathbb{N}}$ is convergent. Assume $J_{\lambda_n} x \rightarrow u$. For all $y \in X$,

$$f(J_{\lambda_n} x) + \frac{1}{\lambda_n} \varphi(d(J_{\lambda_n} x, x)) \leq f(y) + \frac{1}{\lambda_n} \varphi(d(y, x)).$$

Since f is lower semicontinuous, letting $n \rightarrow \infty$, we have

$$\begin{aligned} & f(u) + \frac{1}{\lambda_0} \varphi(d(u, x)) \\ & \leq \liminf_{n \rightarrow \infty} \left(f(J_{\lambda_n} x) + \frac{1}{\lambda_n} \varphi(d(J_{\lambda_n} x, x)) \right) \\ & \leq \liminf_{n \rightarrow \infty} \left(f(y) + \frac{1}{\lambda_n} \varphi(d(y, x)) \right) \\ & = f(y) + \frac{1}{\lambda_0} \varphi(d(y, x)), \end{aligned}$$

for all $y \in X$, and therefore, $u = \operatorname{argmin} \left\{ f(\cdot) + \frac{1}{\lambda_0} \varphi(d(\cdot, x)) \right\} = J_{\lambda_0} x$. This implies that the function $\lambda \mapsto J_\lambda x$ is continuous.

Next we show $J_\lambda x \rightarrow P_{\overline{\text{dom}f}}x$ as $\lambda \rightarrow +0$. Let $\{\lambda_n\} \subset]0, \infty[$ be a decreasing sequence such that $\lambda_n \rightarrow 0$ and $\{z_m\} \subset \text{dom}f$ such that $z_m \rightarrow P_{\overline{\text{dom}f}}x$. Put $x_n = J_{\lambda_n}x$ and $p = P_{\overline{\text{dom}f}}x$. Then we have,

$$f(x_n) + \frac{1}{\lambda_n} \varphi(d(x_n, x)) \leq f(z_m) + \frac{1}{\lambda_n} \varphi(d(z_m, x)).$$

By the monotonicity of $\{f(x_n)\}_{n \in \mathbb{N}}$, there exists $\alpha \in \mathbb{R}$ such that

$$\varphi(d(x_n, x)) \leq \lambda_n (f(z_m) - \alpha) + \varphi(d(z_m, x)).$$

In fact, since $\lambda_n \leq \lambda_1$ for all $n \in \mathbb{N}$ and $f(J_\lambda x)$ is decreasing for λ , we have $f(x_1) \leq f(x_n)$. Putting $\alpha = f(x_1)$, we obtain the inequality above. Letting $n \rightarrow \infty$, we have

$$\limsup_{n \rightarrow \infty} \varphi d(x_n, x) \leq \varphi d(z_m, x),$$

which is equivalent to

$$\limsup_{n \rightarrow \infty} (-\log(\cos d(x_n, x))) \leq -\log(\cos(d(z_m, x))).$$

On the other hand, from the property of metric projection, we have

$$-\log(\cos d(x_n, p)) \leq -\log(\cos d(x_n, x)) - (-\log(\cos d(p, x))).$$

Consequently,

$$\begin{aligned} \limsup_{n \rightarrow \infty} (-\log(\cos d(x_n, p))) &\leq \limsup_{n \rightarrow \infty} (-\log(\cos d(x_n, x)) - (-\log(\cos d(p, x)))) \\ &\leq -\log(\cos d(z_m, x)) - \log(\cos d(p, x)). \end{aligned}$$

Letting $m \rightarrow \infty$, we have

$$\limsup_{n \rightarrow \infty} (-\log(\cos d(x_n, p))) \leq 0,$$

and hence $d(x_n, p) \rightarrow 0$. Then we obtain

$$\lim_{n \rightarrow \infty} J_{\lambda_n} x = p.$$

Since $\{J_{\lambda_n} x\}_{n \in \mathbb{N}}$ converges to p for any decreasing sequences converging to 0, we conclude

$$\lim_{\lambda \rightarrow +0} J_\lambda x = p,$$

which is desired result.

Acknowledgements This work was supported by JSPS KAKENHI Grant Number JP21K03316.

References

1. Kajimura, T., Kimura, Y.: A new resolvent for convex functions in complete geodesic spaces. *RIMS Kôkyûroku*, **2112**, 141–147 (2019)
2. Kajimura, T., Kimura, Y.: Resolvents of convex functions in complete geodesic metric spaces with negative curvature. *J. Fixed Point Theory Appl.* **21**(1), Art. 32, 15 pp (2019)
3. Kimura, Y., Kohsaka, F.: Spherical non spreading ness of resolvents of convex functions in geodesic space. *J. Fixed Point Theory Appl.* **18**(1), 93–115 (2016)
4. Takahashi, W.: *Nonlinear Functional Analysis Fixed Point Theory and Its Applications*. Yokohama Publishers, Yokohama (2000)