3.7.1 Number of functional MoUs with institutions of national, international importance, other universities, industries, corporate houses etc. during the last five years (only functional MoUs with ongoing activities to be considered)

2021-22

INDEX

Principal
Chaitanya Bharathi Institute of Technology
(Autonomous)
Gandipet, Hyderabad-500 075.

	T. V. M. Sairam			
957	MVSS Sastri, Dr K Jagannadha Rao, V Bhikshma	1 Year	Reserach Paper	214
958	Vani S, ChandraSekhar P, Ramanarayan Sankriti & Aparna G	1 Year	Reserach Paper	215
959	C. Jenifa Latha, Ramanarayan Sankriti, Sudha Arvind, Ali Baig Mohammad, MS.Mariam Jenabhai	1 Year	Reserach Paper	222
960	Dr Dakshina Murthy, N R, Srikanth, Nune Seshu, D. Rama	5 Years	Reserach Paper	236
961	Dr NR Dakshina Murthy, Dr K. Jagannadha Rao, Dr D. Rama Seshu	1 Year	Reserach Paper	237
962	M. V. Krishna Rao, G. Tarun & V. Hari Leela	1 Year	Reserach Paper	238
963	Dr Srinivas Vasam, Dr. K. Jaganaadha Rao,	1 Year	Reserach Paper	239
964	Kanchumurty Anusha, Poojari Yugendar & S. Moses Santhakumar	1 Year	Australian Patent	240

Principal
Chaitanya Bharathi Institute of Technology
(Autonomous)
Gandipet, Hyderabad-500 075.

Hindawi
Mobile Information Systems
Volume 2022, Article ID 1833507, 26 pages
https://doi.org/10.1155/2022/1833507

Review Article

A Review on Machine Learning Strategies for Real-World Engineering Applications

Rutvij H. Jhaveri,[1] **A. Revathi**,[2] **Kadiyala Ramana**,[3] **Roshani Raut**,[4] **and Rajesh Kumar Dhanaraj**[5]

[1]*Department of Computer Science & Engineering, Pandit Deendayal Energy University, Gandhinagar, India*
[2]*Department of Computer Science and Engineering, SRM Institute of Science and Technology, Chennai, India*
[3]*Department of Artificial Intelligence & Data Science, Annamacharya Institute of Technology and Sciences, Rajampet, Andhra Pradesh, India*
[4]*Department of Information Technology, Pimpri Chinchwad College of Engineering, Savitribai Phule Pune University, Pune, India*
[5]*School of Computing Science and Engineering, Galgotias University, Greater Noida, Uttarpradesh 201310, India*

Correspondence should be addressed to Kadiyala Ramana; ramana.it01@gmail.com

Received 2 May 2022; Revised 7 July 2022; Accepted 14 July 2022; Published 28 August 2022

Academic Editor: Saqib Hakak

Huge amounts of data are circulating in the digital world in the era of the Industry 5.0 revolution. Machine learning is experiencing success in several sectors such as intelligent control, decision making, speech recognition, natural language processing, computer graphics, and computer vision, despite the requirement to analyze and interpret data. Due to their amazing performance, Deep Learning and Machine Learning Techniques have recently become extensively recognized and implemented by a variety of real-time engineering applications. Knowledge of machine learning is essential for designing automated and intelligent applications that can handle data in fields such as health, cyber-security, and intelligent transportation systems. There are a range of strategies in the field of machine learning, including reinforcement learning, semi-supervised, unsupervised, and supervised algorithms. This study provides a complete study of managing real-time engineering applications using machine learning, which will improve an application's capabilities and intelligence. This work adds to the understanding of the applicability of various machine learning approaches in real-world applications such as cyber security, healthcare, and intelligent transportation systems. This study highlights the research objectives and obstacles that Machine Learning approaches encounter while managing real-world applications. This study will act as a reference point for both industry professionals and academics, and from a technical standpoint, it will serve as a benchmark for decision-makers on a range of application domains and real-world scenarios.

1. Introduction

1.1. Machine Learning Evolution. In this digital era, the data source is becoming part of many things around us, and digital recording [1, 2] is a normal routine that is creating bulk amounts of data from real-time engineering applications. This data can be unstructured, semi-structured, and structured. In a variety of domains, intelligent applications can be built using the insights extracted from this data. For example, as in [3] author used cyber-security data for extracting insights and use those insights for building intelligent application for cyber-security which is automated and driven by data. In the article [1], the author uses mobile data for extracting insights and uses those insights for building an intelligent smart application which is aware of context. Real-time engineering applications are based on tools and techniques for managing the data and having the capability for useful knowledge or insight extraction in an intelligent and timely fashion.

Machine Learning is a stream in Artificial Intelligence, which is gaining popularity in recent times in the field of computing and data analysis that will make applications behave intelligently [4]. In industry 4.0 (fourth industrial revolution) machine learning is considered one of the popular technologies which will allow the application to learn from experience, instead of programming specifically for the enhancement of the system [1, 3]. Traditional practices of industries and manufacturing are automated in Industry 4.0 [5] by using machine learning which is considered a smart technology and is used for exploratory data processing. So, machine learning algorithms are keys to developing intelligent real-time engineering applications for real-world problems by analyzing the data intelligently. All the machine learning techniques are categorized into the following types (a) Reinforcement Learning (b) Unsupervised Learning (c) Semi-Supervised Learning, and (d) Supervised Learning.

Based on the collected data from google trends [6], popularity of these techniques is represented in Figure 1. In Figure 1 the y-axis indicated the popularity score of the corresponding technique and the x-axis indicated the time period. As per Figure 1, the popularity score of the technique is growing day by day in recent times. Thus, it gives the motivation to perform this review on machine learning's role in managing Real Time Engineering Applications. We may use Google Trends to find out what the most popular web subjects are at any given time and location. This could help us generate material and give us suggestions for articles that will most likely appeal to readers. Just make sure the content is relevant to our company or industry. We can look into the findings a little more carefully and investigate the reasons that may have influenced such trends because Google Trends can supply us with data about the specific regions in which our keywords drew substantial interest. With this level of data, we can figure out what's working and what needs to be improved.

Machine learning algorithms' performance and characteristics and nature of the data will decide the efficiency and effectiveness of the solution based on machine learning. The data-driven systems [7, 8] can be effectively built by using the following ML areas like reinforcement learning, association rule learning, reduction of dimensionality and feature engineering, data clustering, regression, and classification analysis. From ANN, a new technology is originated from the family of machine learning techniques called Deep Learning which is used for analyzing data intelligently [9]. Every machine learning algorithm's purpose is different even various machine learning algorithms applied over the same category will generate different outcomes and depends on the nature and characteristics of data [10]. Hence, it's challenging to select a learning algorithm for generating solutions to a target domain. Thus, there is a need for understanding the applicability and basic principle of ML algorithms in various Real Time Engineering Applications.

A comprehensive study on a variety of machine learning techniques is provided in this article based on the potentiality and importance of ML that can be used for the augmentation of application capability and intelligence. For industry people and academia this article will be acting as a reference manual, to research and study and build intelligent systems which are data-driven in a variety of real-time engineering applications on the basis of machine learning approaches.

1.2. Types of Machine Learning Techniques. Figure 2 shows the Machine Learning Timeline chart. There are 4 classes of machine learning approaches (a) Reinforcement Learning, (b) Semi-Supervised Learning (c) Unsupervised, and (d) Supervised Learning as shown in Figure 3. With the applicability of every ML technique in Real Time Engineering applications, we put down a brief discussion on all the four types of ML approaches as follows:

(i) Reinforcement learning: in an environment-driven approach, RL allows machines and software agents to assess the optimal behavior automatically to enhance the efficiency in a particular context [11]. Penalty or rewards are the basis for RL, and the goal of this approach is to perform actions that minimize the penalty and maximize the reward by using the extracted insights from the environment [12]. RL can be used for enhancing sophisticated systems efficiency by doing operational optimization or by using automation with the help of the trained Artificial Intelligence models like supply chain logistics, manufacturing, driving autonomous tasks, robotics, etc.

(ii) Semi-supervised: as this method operates on both unlabeled and labeled data [3, 7] it is considered a hybrid approach and lies between "with supervision" and "without supervision" learning approach. The author in [12] concludes that the semi-supervised approach is useful in real-time because of numerous amounts of unlabeled data and rare amounts of labeled data available in various contexts. The semi-supervised approach achieves the goal of predicting better when compared to predictions based on labeled data only. Text classification, labeling data, fraud detection, machine translation, etc., are some of the common tasks.

(iii) Unsupervised: as in [7], the author defines that unsupervised approach as a process of data-driven, with minimum or no human interface, it takes datasets consisting of unlabeled data and analyzes them. The unsupervised approach is widely used for purpose of exploring data, results grouping, identifying meaningful structures and trends, and extracting general features. Detecting anomalies, association rules finding, reducing dimensionality, learning features, estimating density, and clustering are the most usual unsupervised tasks.

(iv) Supervised: as in [7], author defines the supervised approach as a process of making a function to learn to map output from input. A function is inferred by using training example collection and training data which is labeled. As in [3], the author states that a supervised learning approach is a task-driven approach, which is to be initiated when certain inputs

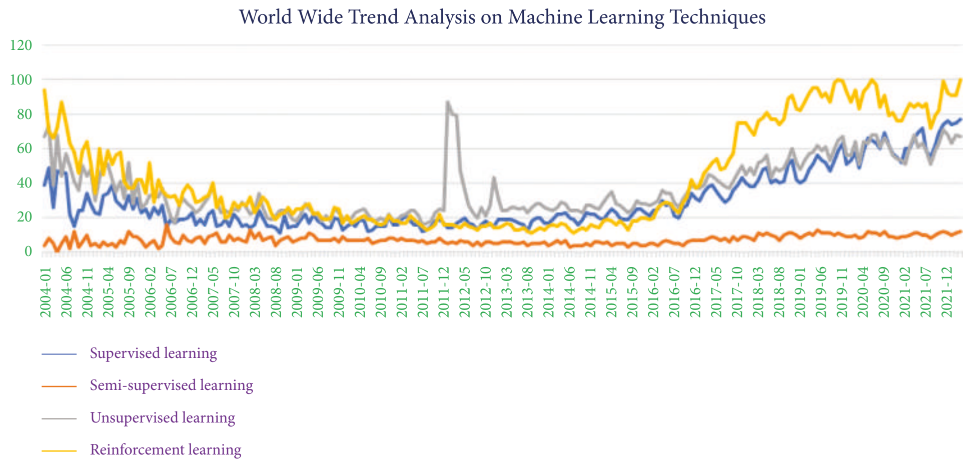


FIGURE 1: World wide trend analysis on machine learning techniques [6].

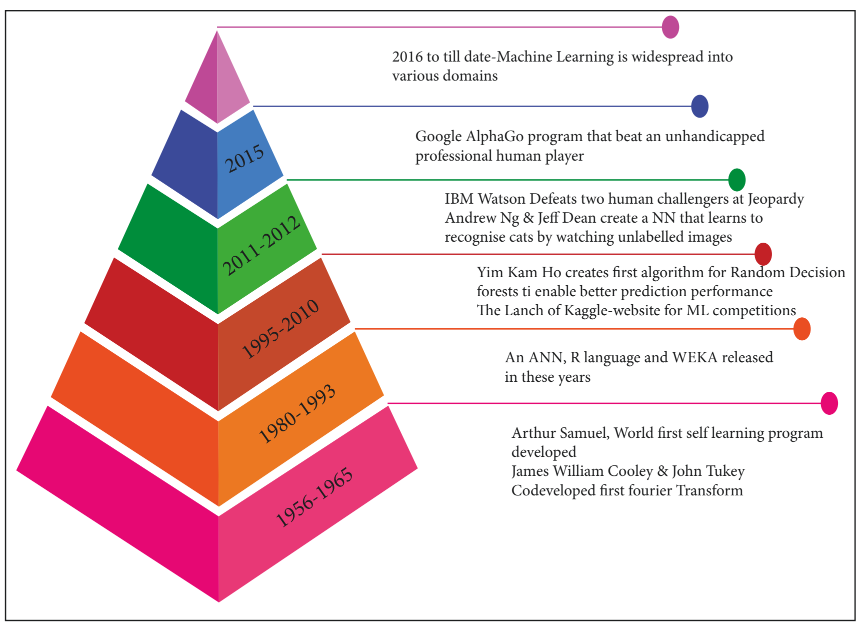


FIGURE 2: Machine learning time line.

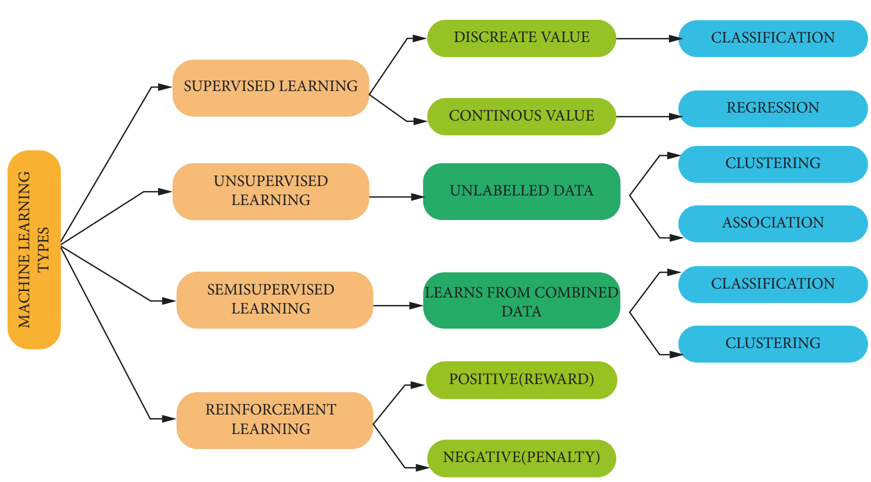


FIGURE 3: Machine learning techniques.

are capable to accomplish a variety of goals. The most frequently used supervised learning tasks are regression and classification.

In Table 1, we summarize various types of machine learning techniques with examples.

Table 2 summarizes the comparison between the current survey with existing surveys and highlights how it is different or enhanced from the existing surveys.

1.3. Contributions. Following are the key contributions to this article:

(i) A comprehensive view on variety of ML algorithms is provided which is applicable to improve data-driven applications, task-driven applications capabilities, and intelligence

(ii) To discuss and review the applicability of various solutions based on ML to a variety of real-time engineering applications

(iii) By considering the data-driven application capabilities and characteristics and nature of the data, the proposed study/review scope is defined

(iv) Various challenges and research directions are summarized and highlighted that fall within this current study scope

1.4. Paper Organization. The organization of the rest of the article is as follows: state of art is presented in the next section which explains and introduces real-time engineering applications and machine learning; in the next section, ML's role in real-time engineering applications is discussed; and in the coming section, challenges and lessons learned are presented; in the penultimate section, several future directions and potential research issues are discussed and highlighted; and in the final section conclude the comprehensive study on managing Real Time Engineering Applications using Machine Learning.

2. State of the Art

2.1. Real World Issues. Computer systems can utilize all client data through machine learning. It acts according to the program's instructions while also adapting to new situations or changes. Algorithms adapt to data and exhibit previously unprogrammed behaviors. Acquiring the ability to read and recognize context enables a digital assistant to skim emails and extract vital information. This type of learning entails the capacity to forecast future customer behaviors. This enables you to have a deeper understanding of your customers and to be proactive rather than reactive. Machine learning is applicable to a wide variety of sectors and industries and has the potential to expand throughout time. Figure 4 represents the real-world applications of machine learning.

2.2. Introduction to Cyber Security. For both, services and information, internet is most extensively exploited. In article [13], author summarizes that since 2017 as an information source Internet is utilized by almost 48% of the whole population in the world. As concluded in the article [14], this number is hiked up to 82% in advanced countries.

The interconnection of distinct devices, networks, and computers is called the Internet, whose preliminary job is to transmit information from one device to another through a network. Internet usage spiked due to the innovations and advancements in mobile device networks and computer systems. As internet is the mostly used by the majority of population as an information source so it's more prone to cyber criminals [15]. A computer system is said to be stable when it's offering integrity, availability, and confidentiality of information. As stated in the article [16], with intent to disturb normal activity, if an unauthorized individual enters into the network, then the computer system will be compromised with integrity and security. User assets and cyberspace can be secured from unauthorized individual attacks and access with the help of cyber security. As in article [17], the primary goal of cyber security is to keep information available, integral, and confidential.

2.3. Introduction to Healthcare. With advancements in the field of Deep Learning/Machine Learning, there are a lot of transformations happening in the areas like governance, transportation, and manufacturing. Extensive research is going on in the field of Deep Learning over the last decade. Deep Learning has been applied to lots of areas that delivered a state-of-the-art performance in variety of domains like speech processing, text analytics, and computer vision. Recently researchers started deploying Deep Learning/Machine Learning approaches to healthcare [18], and they delivered outstanding performances in the jobs like brain tumor segmentation [19], image reconstruction in medical images [20, 21] lung nodule detection [22], lung disease classification [23], identification of body parts [24], etc.

It is evident that CAD systems that provide a second opinion will help the radiologists to confirm the disease [25] and deep learning/machine learning will further enhance the performance of these CAD systems and other systems that will provide supporting decisions to the radiologists [26].

Advancement in the technologies like big data, mobile communication, edge computing, and cloud computing is also helping the deployment of deep learning/machine learning models in the domain of healthcare applications [27]. By combining they can achieve greater predictive accuracies and an intelligent solution can be facilitated which is human-centered [28].

2.4. Introduction to Intelligent Transportation Systems. In transit and transportation systems, after the deployment of sensing technologies, communication, and information, the resultant implementation is called an intelligent transportation system [29]. An intelligent transportation system is an intrinsic part of smart cities [30], which have the following services such as autonomous vehicles, public transit system management, traveler information systems, and road traffic management. These services are expected to contribute a lot to the society by curbing pollution, enhancing energy efficiency,

Table 1: ML Technique varieties with approaches and examples.

Examples	Model building	Learning types	Approach
Regression, classification	Models or algorithms that use labeled data for learning	Supervised	Task-driven approach
Dimensionality reduction, association, clustering	Models or algorithms that uses unlabeled data for learning	Unsupervised	Data-driven approach
Clustering and classification	Using combined data models are built	Semi-supervised	Unlabeled + labeled
Control and classification	Using the concept of penalty and reward as a basis models are built	Reinforcement	Environment-driven approach

transit and transportation efficiency is enhanced and finally, traffic and road safety is also improved.

Advances in technologies like wireless communication technology, computing, and sensing are enabling intelligent transportation systems applications and also bear a lot of challenges due to their capabilities to generate huge amounts of data, independent QoS requirements, and scalability.

Due to the recent traction in deep learning/machine learning models, approaches like RL and DL are utilized to exploit patterns and generate decisions and predictions accurately [31–33].

2.5. Introduction to Renewable Energy. Sustainable and alternative energy sources are in demand due to the effect created by burning fossil fuels in the environment and fossil fuel depletion. As in article [34], the energy market biomass, wind power, tidal waves, geothermal, solar thermal, and solar photovoltaic are growing as renewable energy resources. There will be instability in the power grids due to various reasons like when demand is more than the supply of the energy and when supply is more than the demand of the energy. Finally, environmental factors affect the energy output of the plants based on the renewable energy. To address the management and optimization of energy, machine learning is used.

2.6. Introduction to Smart Manufacturing. Manufacturing has been divided into a number of categories, one of the categories in which computer-based manufacturing is performed is called Smart Manufacturing, which performs workers' training, digital technology, and quick changes in the design and with high adaptability. Other responsibilities include recyclability of production effectively, supply chain optimization, and demand-based quick changes in the levels of production. Enabling technologies of Smart Manufacturing are advances in robotics, services and devices connectivity in the industry, and processing capabilities in the big data.

2.7. Introduction to Smart Grid. The basic structure of the electrical power grid has remained same over time, and it has been noticed that it has become outdated and ill-suited, unable to meet demand and supply in the twenty-first century. Even though we are in the twenty-first century, electrical infrastructure has remained mostly unaltered throughout time. However, as the population and consumption have grown, so requires power.

2.7.1. Drawbacks

(i) Analyzing the demand is difficult
(ii) Response time is slow

The new smart grid idea has evolved to address the issues of the old outdated electrical power system. SG is a large energy network that employs real time and intelligent monitoring, communication, control, and self-healing technologies to provide customers with a variety of alternatives while guaranteeing the stability and security of their electricity supply. By definition, SGs are sophisticated cyber-physical system. The functionality of this modern SG can be broken down into four parts.

This contemporary SG's functionality may be split down into four components:

(1) Consumption: electricity is used for a variety of reasons by various industries and inhabitants
(2) Distribution: the power so that it may be distributed more widely
(3) Transmission: electricity is transmitted over a high-voltage electronic infrastructure
(4) Generation: during this phase, electricity is generated in a variety of methods

ML and DL functionalities in the context of SG include predicting about

(1) Stability of the SG
(2) Optimum schedule
(3) Fraud detection
(4) Security breach detection
(5) Network anomaly detection
(6) Sizing
(7) Fault detection
(8) Energy consumption
(9) Price
(10) Energy generation

2.8. Introduction to Computer Networks. The usefulness of ML in networking is aided by key technological advancements in networking, such as network programmability via Software-Defined Networking (SDN). Though machine learning has been widely used to solve problems such as pattern recognition, speech synthesis, and outlier identification, its use in network operations and administration has

Table 2: Summary of important surveys on ML.

					Real time engineering applications															Machine learning				
S. no.	Reference no	Year	Citations	No. of references	Cyber security	Computer networks	Human machine interaction	Healthcare	Satellite imaging	Industry 4.0/5.0	Intelligent transportation system	Computer vision	Smart energy	Smart grids	Renewable energy systems	Electrical power systems	Civil engineering	Energy systems	Design and manufacturing	Techniques	Metrics	Tools	Trustworthiness	Adversarial ML
1	372	2014	47	42	x	x	x	x	x	x	x	x	x	x	√	x	x	x	x	√	√	x	x	x
2	373	2020	1	54	x	x	x	x	x	x	x	x	x	√	x	x	x	x	x	√	x	√	x	x
3	374	2020	12	668	√	x	x	x	x	x	x	x	x	x	x	x	x	x	x	√	√	√	√	√
4	375	2020	17	129	x	x	x	x	x	x	√	x	x	x	x	x	x	x	x	√	x	x	x	x
5	376	2020	1	259	x	x	x	x	x	x	√	x	x	x	x	x	x	x	x	√	x	x	x	x
6	377	2021	18	140	√	√	√	√	x	√	√	√	√	x	√	x	x	x	x	√	x	x	x	x
7	378	2020	46	208	x	x	x	√	x	x	x	x	x	x	x	x	x	x	x	√	x	x	x	x
8	376	2018	91	48	x	x	x	x	x	x	x	x	x	x	x	x	x	x	√	√	x	x	x	x
9	380	2020	6	14	x	x	x	x	x	x	x	√	x	x	x	x	x	x	x	√	x	x	x	x
10	381	2017	445	73	x	x	√	x	x	x	x	x	x	x	x	x	x	x	x	√	x	x	x	x
11	382	2018	40	235	x	x	x	x	x	x	x	√	x	x	x	x	x	x	x	√	x	√	x	x
12	383	2020	2	62	x	x	x	x	x	x	x	x	x	√	x	x	x	x	x	√	x	x	x	x
13	384	2021	4	180	x	x	x	x	x	x	x	x	x	x	x	x	x	x	√	√	x	x	x	x
14	385	2019	53	310	x	x	x	x	x	√	x	x	x	x	x	x	x	x	√	√	√	√	√	√
15	386	2019	35	210	x	x	x	x	x	x	x	√	x	x	x	x	x	x	x	√	x	x	x	x
16	387	2020	16	150	x	x	x	x	x	x	x	x	x	x	√	x	x	x	x	√	x	√	x	x
17	388	2019	1	25	x	x	x	x	√	x	x	x	x	x	x	x	x	x	x	√	x	x	x	x
18	389	2018	470	501	x	√	x	x	x	x	x	x	x	x	x	x	x	x	x	√	x	x	x	x
19	390	2019	63	149	x	x	x	x	x	x	x	x	√	x	x	√	x	x	x	√	√	x	x	x
20	391	2017	202	252	x	x	x	x	x	x	x	√	x	x	x	x	x	x	x	√	x	x	x	x
21	392	1997	97	97	x	x	x	x	x	x	x	x	x	x	x	x	√	x	x	√	x	x	x	x
22	393	2019	148	114	x	x	x	x	x	x	x	x	x	x	x	x	x	√	x	√	√	x	x	x
23	394	2021	1	193	x	x	x	x	x	√	x	x	x	x	x	x	x	x	√	√	√	x	x	x
24	395	2018	83	79	x	√	x	x	x	x	x	x	x	x	x	x	x	x	x	√	x	x	x	x
25	396	2020	47	103	x	x	x	x	x	√	x	x	x	x	x	x	x	x	x	√	x	x	x	x
26	Proposed survey	2021	—	396	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√

Figure 4: Applications of machine learning.

been limited. The biggest roadblocks are determining what data may be collected and what control actions can be taken on legacy network equipment. These issues are alleviated by the ability to program the network using SDN. ML-based cognition can be utilized to help automate network operation and administration chores. As a result, applying machine learning approaches to such broad and complicated networking challenges is both intriguing and challenging. As a result, ML in networking is a fascinating study area that necessitates a thorough understanding of ML techniques as well as networking issues.

2.9. Introduction to Energy Systems. A set of structured elements designed for the creation, control, and/or transformation of energy is known as an energy system [35, 36]. Mechanical, chemical, thermal, and electro-magnetical components may be combined in energy systems to span a wide variety of energy categories, including renewables and alternative energy sources [37–39]. The progress of energy systems faces difficult decision-making duties in order to meet a variety of demanding and conflicting objectives, such as functional performance, efficiency, financial burden, environmental effect, and so on [40]. The increasing use of data collectors in energy systems has resulted in an enormous quantity of data being collected. Smart sensors are increasingly widely employed in the production and consumption of energy [41–43]. Big data has produced a plethora of opportunities and problems for making well-informed decisions [44, 45]. The use of machine learning models has aided the deployment of big data technologies in a variety of applications [46–50]. Prediction approaches based on machine learning models have gained popularity in the energy sector [51–53] because they make it easier to infer functional relationships from observations. Because of their accuracy, effectiveness, and speed, ML models in energy systems are becoming crucial for predictive modeling of production, consumption, and demand analysis [54, 55]. In the context of complex human interactions, ML models provide give insight into energy system functioning [56, 57]. The use of machine learning models is in making traditional energy systems, as well as alternative and renewable energy systems.

3. Recent Works on Real-Time Engineering Applications

3.1. Machine Learning for ITS. Exposure to traffic noise, air pollution, road injuries, and traffic delays are only some of the key issues that urban inhabitants experience on a daily basis. Urban areas are experiencing severe environmental and quality-of-life difficulties as a result of rapid car expansion, insufficient transportation infrastructure, and a lack of road safety rules. For example, in many urban areas, large trucks violate the typical highways, resulting in traffic congestion and delays. In addition, many bikers have frequent near misses as a result of their clothes, posture changes, partial occlusions, and varying observation angles all posing significant challenges to the Machine Learning (ML) algorithms' detection rates.

Over the last decade, there has been a surge in interest in using machine learning and deep learning methods to analyze and visualize massive amounts of data generated from various sources in order to improve the classification and recognition of pedestrians, bicycles, special vehicles (e.g., emergency vehicles vs. heavy trucks), and License Plate Recognition (LPR) for a safer and more sustainable environment. Although deep models are capable of capturing a wide variety of appearances, adaption to the environment is essential.

Artificial neural networks form the base for deep learning success; in artificial neural networks to mirror an image, the human brain functioning interconnected node system sets are present. The neighboring layer's nodes will be consisting of connections with weights coming from nodes from other layers. The output value is generated by given input and weight to the activation function in a node. Figure 5 presents the ML mainstream approaches used in ITS.

Figure 6 shows the RL working in intelligent transportation system.

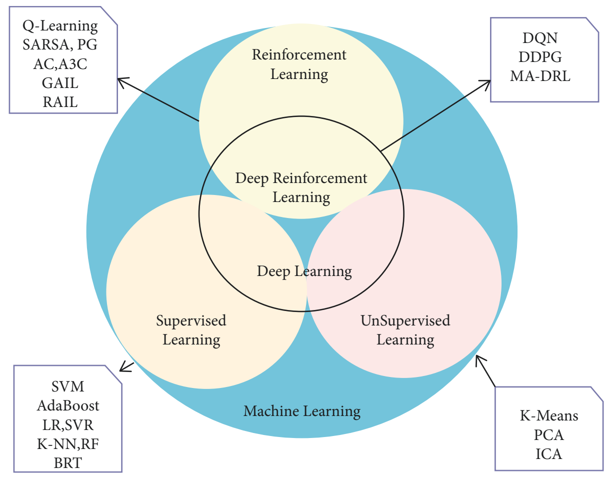


Figure 5: Mainstream ML approaches.

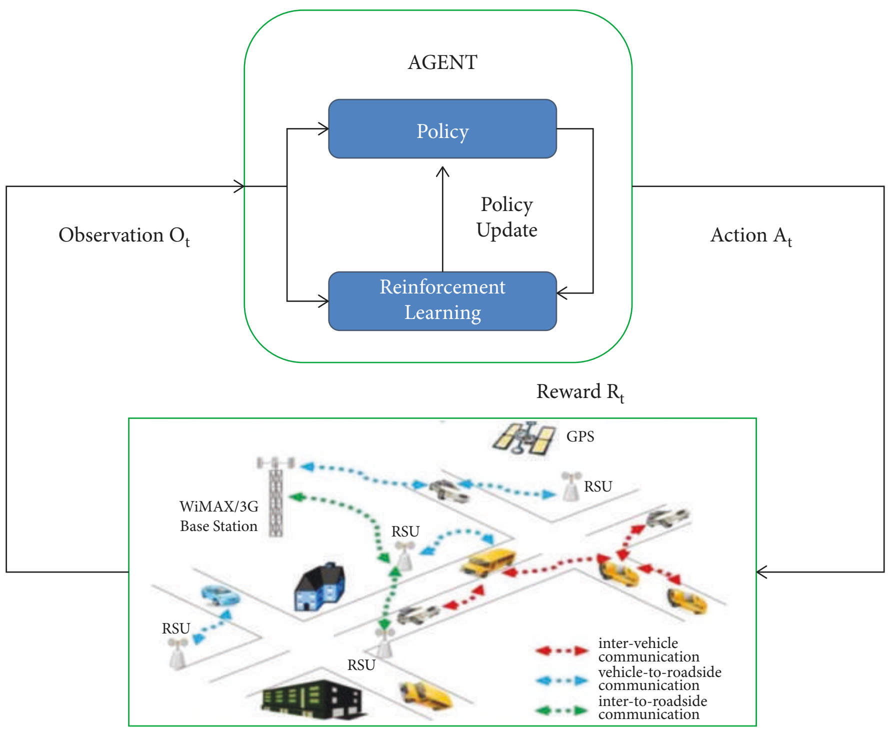


Figure 6: RL working in intelligent transportation system.

Figures 7–9 present the interaction between ITS and ML and Machine Learning Pipeline.

3.2. Machine Learning for HealthCare. Over time, for the actions performed as a response reward, actions and observations are given as input to policy functions, and the method that learns from this policy function is called RL [58]. There is a wide range of healthcare applications where RL can be used even RL can be used in the detection of disease based on checking symptoms ubiquitously [59]. Another potential use of RL in this domain is Gogame [60].

In semi-supervised learning, both unlabeled data and labeled data are used for training particularly greater doses of unlabeled data and little doses of labeled data are available, and then semi-supervised learning is suitable. Semi-supervised learning can be applied to a variety of healthcare applications like medical image segmentation [61, 62] using various sensors recognition of activity is proposed in [61], in [63] author used semi-supervised learning for healthcare data clustering.

In supervised learning, labeled information is used for training the model to map the input to output. In the regression output value is continuous and in classification output value is discrete. Typical application of supervised learning in the healthcare domain is the identification of organs in the body using various image modalities [19] and nodule classification in the lung images [21].

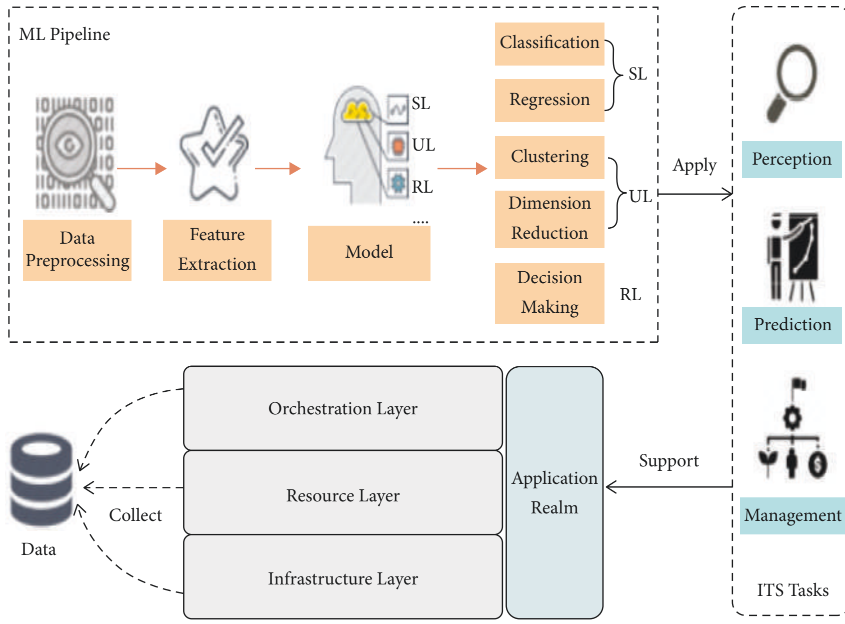


Figure 7: ML pipeline and interaction between ITS and ML.

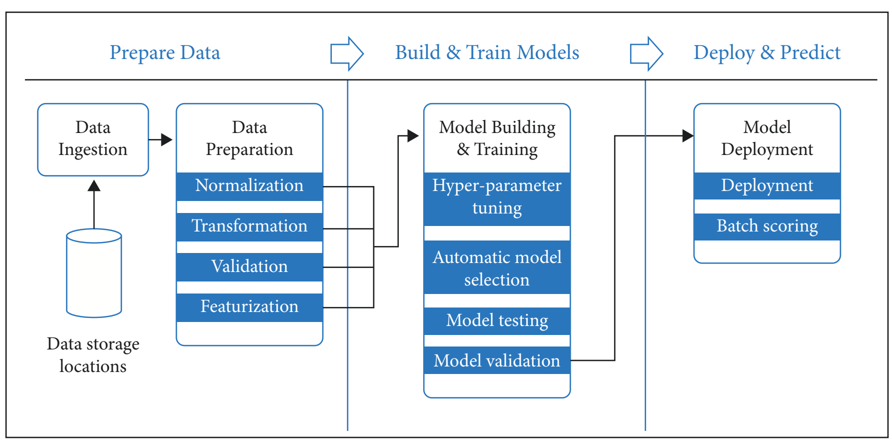


Figure 8: ML pipeline.

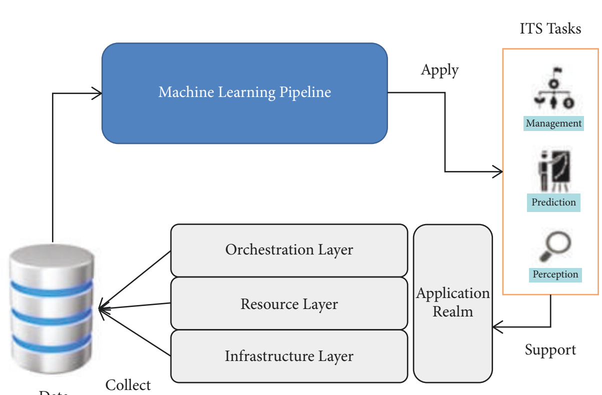


Figure 9: Interaction between ML and ITS.

In unsupervised learning, mapping of input to the output will be done by training the model using unlabeled data:

(i) Similarity is used for clustering

(ii) Feature selection/dimensionality reduction

(iii) Anomaly detection [64]

Unsupervised learning can be applied to a lot of healthcare applications like feature selection [65] using PCA and using Clustering [66] for heart disease prediction.

Various phases in an ML-based Healthcare system are shown in Figure 10.

The four major applications of healthcare that can benefit from ML/DL techniques are prognosis, diagnosis, treatment, and clinical workflow, which are described in Table 3.

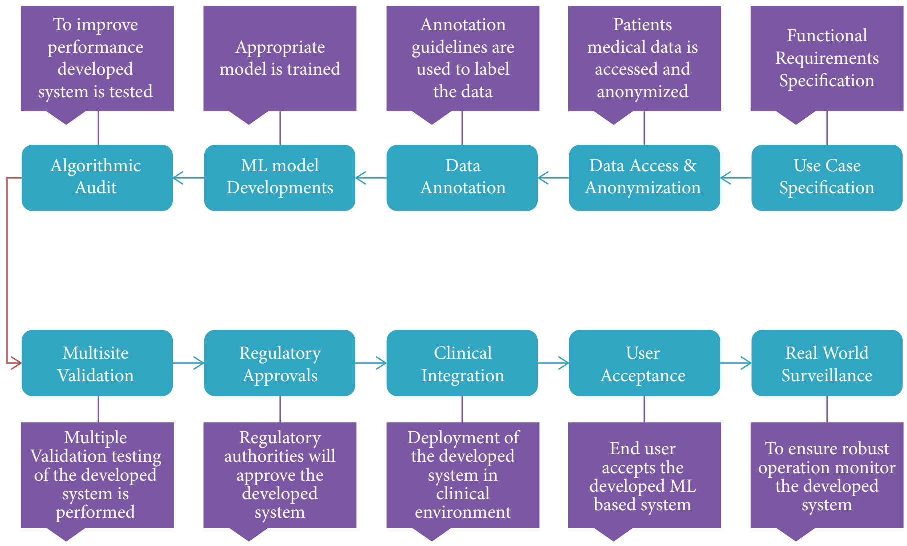


Figure 10: ML-based healthcare systems phases of development.

Table 3: Neural networks comparison.

Machine learning applications in	Domain	Areas	References
Prognosis		In clinical practice, the process of forecasting the disease development is called prognosis. Prognosis can predict various things like survival chances, corresponding disease related health issues detection, will the disease be stable, improve or become worse and finally, prognosis can forecast the symptoms of the disease	[67–69]
Diagnosis	Health records in electronic format	Patient's full history of medication is generated by healthcare services and hospitals which is stored in terms of records in electronic formats. This generated data will be consisting of unstructured and structured data. For aiding the process of diagnosis ML-based methods are applied over electronic health records for fetching the features	[70–73]
Treatment	Analysis of image in the medical fields	Image retrieval, image registration, image reconstruction, image segmentation, image classification, image detection, image enhancement	[74–93]
Clinical workflow	Interpretation of an image	Radiologists are responsible for analyzing the images and the results of the study will be written in the radiology reports	[94–97]

3.3. Machine Learning for Cyber Security. Artificial Intelligence and Machine Learning are widely accepted and utilized in various fields like Cyber Security [94–103], design and manufacturing [104], medicine [105–108], education [109], and finance [110–112]. Machine Learning techniques are used widely in the following areas of cyber security intrusion detection [113–116], dark web or deep web sites [117, 118], phishing [119–121], malware detection [122–125], fraud detection [126–129], and spam classification [130–133]. As time changes there is a need for vigorous and novel techniques to address the issues of cyber security. Machine Learning is suitable for evolutionary attacks as it learns from experiences.

In article [118], the authors analyzed and evaluated the dark web which is a hacker's social network by using the ML approach for threat prediction in the cyberspace. In article [134], the author used an ML model with social network features for predicting cyberattacks on an organization during a stipulated period. This prediction uses a dataset consisting of darkweb's 53 forum's data in it. Advancements in recent areas can be found in [135–138].

Antivirus, firewalls, unified threat management [139], intrusion prevention system [140], and SEIM solutions [141] are some of the classical cyber security systems. As in article [142], the author concluded that, in terms of post-cyber-attack response, performance, and in error rate classical cyber security systems are poor when compared with AI-based systems. As in the article [143], once there is cyber-space damage by the attack then only it's identified and this situation happens in almost 60%. Both on the cyber security side and attackers' side, there is a stronger hold by ML. On the cyber security side as specified in this article [144, 145] to safeguard everything from the damage done by the attackers and for detecting attacks at an early stage and finally for performance enhancement ML is used. ML is used on the attacker's end to locate weaknesses and system vulnerabilities as well as techniques to get beyond firewalls and other defence walls As in [146], the author concludes that to further enhance the classification performance ML approaches are combined.

3.4. Machine Learning for Renewable Energy. Forecasting Renewable Energy Generation can be done using Machine Learning, state-of-art works are presented in Table 4.

3.5. Machine Learning for Smart Manufacturing. The following table shows the ML applicability to Smart Manufacturing. State-of-art works are presented in Table 5.

3.6. Machine Learning for Smart Grids. This subsection discusses machine learning applicability to smart grids. State-of-the-art works are presented in Table 6.

3.7. Machine Learning for Computer Networks

3.7.1. Traffic Prediction. As networks are day by day becoming diverse and complex, it becomes difficult to manage and perform network operations so huge importance is given to traffic forecast in the network to properly manage and perform network operations. Time Series Forecasting is forecasting the traffic in near future.

3.7.2. Traffic Prediction. To manage and perform network operations, it's quite important to perform classification of the network traffic which includes provisioning of the resource, monitoring of the performance, differentiation of the service and quality of service, intrusion detection and security, and finally capacity planning.

3.7.3. Congestion Control. In a network, excess packets will be throttled using the concept called congestion control. It makes sure the packet loss ratio is in an acceptable range, utilization of resources is at a fair level, and stability of the network is managed.

Table 7 presents ML state of art systems in networking.

3.7.4. Machine Learning for Civil Engineering. The first uses of ML programs in Civil Engineering involved testing different existing tools on simple programs [210–213], more difficult problems are addressed in [214–216].

3.7.5. Machine Learning for Energy Systems. Hybrid ML models, ensembles, Deep Learning, Decision Trees, ANFIS, WNN, SVM, ELM, MLP, ANN are among the ten key ML models often employed in energy systems, according to the approach.

Table 8 presents ML state of art systems in the Energy Systems domain.

4. Current Challenges on Machine Learning Technology

While machine learning offers promise and is already proving beneficial to businesses around the world, it is not without its hurdles and issues. For instance, machine learning is useful for spotting patterns, but it performs poorly at generalizing knowledge. There is also the issue of "algorithm weariness" among users.

In ML, for model training, decent amount of data and resources that provide high performances are needed. This challenge is addressed by involving multiple GPU's. In Real Time Engineering Applications, an ML approach is needed which is modeled to address a particular problem robustly. As the same model designed to address one task in real-time engineering application cannot address all the tasks in a variety of domains, so there is a need to design a model for each task in the Real Time Engineering Applications.

ML approaches should have the skill to prevent issues in the early stages as this is an important challenge to address in most real-time engineering applications. In the medical domain, ML can be used in predicting diseases and ML techniques can also be used for forecasting the detection of terrorism attacks. As in [243], the catastrophic consequences cannot be avoided by having faith blindly in the ML predictions. As in article [244], author states that ML approaches are used in various domains, but in some domains as an alternative to accuracy and speed ML approaches require correctness at very high levels. To convert a model into trustworthy, there is a need to avoid a shift in dataset, which means the model is to be trained and tested on the same dataset which can be ensured by avoiding data leakages [245].

Moving object's location can be identified by using the enabling technologies like GPS and cell phones and this information to be maintained securely as tamper-proof is one of the crucial tasks for ML. As in article [246], author states that an object's location information from multiple sources is compared and tries to find the similarity, and as in article [247] author confirms that due to network delays the location change of the objects there is always ambiguousness in the location information gathered from multiple sources and the trustworthiness of such information needs to be addressed using ML techniques.

In a connected web system, to have interaction between consumers and service providers with trustworthiness an ontology of trust is proposed in the article [248]. In text classification also trustworthiness is used. As in article [249] author states that in semantic and practical terms where the meaning of the text is interpreted trustworthiness can be fused. In article [250] author validates the software's trustworthiness using a metric model. As in article [251], the author states that in companies and data centers the consumption of power can be mitigated by utilizing ML approaches for designing strategies that are power-aware. To reduce the consumption in its entirety, it's better to turn off the machines dynamically. Which machine to be turned off will be decided by the forecasting model and it's very

Table 4: ML state-of-the-art systems in renewable energy domain.

Reference	Machine learning technique	Purpose
Wind power generation		
[147]	Statistical machine learning techniques	Short and medium forecasting
[148]	Autoregressive integrated moving average and autoregressive moving average	Wind power forecasting and forecasting of wind speed
[149]	Kalman filter model is used	Wind-generated power and wind speed forecasting through online
[150]	Review on two machine learning techniques is done	Wind speed forecasting
[151]	ANN	TIME SERIES PREDICTION
[152]	ANN variant is used called recurrent multi-layer perception	Used for the prediction of long-term power generation
[153]	SVM is used	To measure the wind speed
[154]	Fuzzy models are used	For the prediction of wind power generation.
[155]	Numerical weather prediction model is used	Wind power consumption and generation forecast
[156]	Ensemble model is used	Wind power consumption and generation forecast
[157]	ANN and k-nearest neighbor approaches are used	Wind power generation forecast
[158]	Particle swarm optimization, k-NN and SVM are used	Wind power generation forecast
[159]	Techniques considered are (i) Random forest (ii) Regression trees (iii) ANN (iv) MLP (v) SVM	Review on machine learning techniques for wind power generation forecasting
Solar energy generation		
[160]	ANN is used	Solar energy generation
[161]	SVM is used	Power generation using solar
[162]	Ensemble method	Forecasting solar power generation
[163]	Statistical methods	Review on solar energy power generation
Hydro power generation		
[164–166]	RNN, SVM	Rainfall prediction
[167]	RNN, SVM	Forecast values of rainfall depth
[168]	Ensemble learning	Forecast the hydro energy consumption
[169–171]	ANN	Hydropower plant management

Table 5: ML state-of-the-art systems in the smart manufacturing domain.

Reference	Machine learning technique	Purpose
Decision support		
[172]	SVM	To estimate the products, optimize cost
[173]	Least square SVM, backpropagation NN	Cost estimation of the product life-cycle
[174]	Genetic algorithm based ML techniques are used	For decision support systems
[175]	NN and ML techniques are used	Rotating equipment's life is predicted
Plant operation and health management		
[176]	Markov decision is used	For predicting plant operation and health management
Data management		
[177]	Various ML techniques are used	Discussed about hadoop framework and cloud computing utilization for managing data
[178]	Various ML techniques are used	How cloud can be incorporated into various product life cycles can be incorporated
LifeCycle management		
[179]	Various ML techniques are used	Hardware and software changes in the manufacturing can be tracked

important to have trust in this forecasting model before setting up the machine to be switched off.

Fatigue in the alarm is generating false alarms at higher rates. This will reduce the response time of the security staff and this issue is an interesting area in cyber security [252, 253].

Some concerns associated with machine learning have substantial repercussions that are already manifesting now. One is the absence of explainability and interpretability, also known as the "black box problem." Even its creators are unable to comprehend how machine learning models generate their own judgments and behaviors. This makes it

Table 6: ML state-of-the-art systems in smart grids domain.

Reference	DL/ML technique	Application
Energy forecasting		
[180]	Weighted regularized extreme learning machine	Prediction of wind speed can be improved
[181]	Fuzzy logic, NN	Predict solar irradiance
[182]	RFR, GBR, SVR	Solar radiation can be forecasted
[183]	Gradient boost, random forest, regression tree	Solar irradiance is forecasted
[184]	RNN	Predict power and wind speed
Securing smart grids		
[185]	SVM	Stealthy attacks can be detected
[186]	ANN	Consumption of energy can be analyzed
[187]	DCNN	Theft of electricity can be detected by analyzing data
[188]	RNN	Smart grids' false attacks can be identified
[189]	RNN	Attacks on network and frauds in the networks based on blockchain can be identified
[190]	Kalman filter, chi-square detector, and cosine similarity matching	Attacks on communication system can be identified

Table 7: ML state-of-the-art systems in computer networking domain.

Reference	ML techniques	Applications
Traffic prediction		
[191, 192]	Supervised: MLP-NN	Prediction of network traffic
[193, 194]	Supervised: KBR, LSTM-RNN, MLP-NN	Prediction of traffic volume
Traffic classification		
[195–197]	Supervised: SVM	Classification of traffic based on host behavior
[198]	Unsupervised: HCA	Classification of traffic based on host behavior
[199]	Supervised: AdaBoost	Classification of traffic based on host behavior
[200–202]	Supervised k-NN, NBKE, BAGGING	Supervised flow feature based traffic classification
[203–205]	Unsupervised DBSCAN, AutoClass,k-means	UnSupervised flow feature based traffic classification
[206]	Supervised *k*-NN,Linear-SVM, Radial-SVM, DT, RF, extended tree, AdaBoost, Gradient-AdaBoost, NB, MLP	NFVand SDN-based traffic classification
[207–209]	Supervised: (i) MLP-NN (ii) MART (iii) Bagging DT (iv) Extra-trees (v) SVR (vi) BN	Congestion inference from the estimation of different network parameters

difficult to correct faults and ensure that a model's output is accurate and impartial. When it was discovered that Apple's credit card algorithm offered women much lesser credit lines than men, for instance, the corporation was unable to explain the reason or address the problem.

This pertains to the most serious problem affecting the field: data and algorithmic bias. Since the debut of the technology, machine learning models have been frequently and largely constructed using data that was obtained and labeled in a biassed manner, sometimes on intentionally. It has been discovered that algorithms are frequently biased towards women, African Americans, and individuals of other ethnicities. Google's DeepMind, one of the world's leading AI labs, issued a warning that the technology poses a threat to queer individuals.

This issue is pervasive and well-known, yet there is resistance to taking the substantial action that many experts in the field insist is necessary. Timnit Gebru and Margaret Mitchell, co-leaders of Google's ethical AI team, were sacked in retaliation for Gebru's refusal to retract research on the dangers of deploying huge language models, according to tens of thousands of Google employees. In a survey of researchers, policymakers, and activists, the majority expressed concern that the progress of AI by 2030 will continue to prioritize profit maximization and societal control over ethics. The nation as a whole is currently debating and enacting AI-related legislation, particularly with relation to immediately and blatantly damaging applications, like facial recognition for law enforcement. These discussions will probably continue. And the evolving data

Table 8: ML state-of-the-art systems in energy systems domain.

ML technique	Reference	Application
ANN	[217]	In the buildings of industry savings of the energy are verified and measured
	[218]	Forecast solar radiation and predict wind speed
	[219]	Electricity cost is forecasted
	[220]	Power generation plans are created and scheduled and fluctuations in the wind power are controlled
	[221]	Various capacities of the renewable energy generation are optimized
MLP	[222]	Plants are ranked
	[223]	Forecast solar radiation
	[224]	Predict solar power generation
	[225]	Predict load
	[226]	Solar irradiation is forecasted
SVM	[227]	Forecast price of the electricity in the market
	[228]	Estimate the power quality
	[229]	Disturbances in the power quality will be classified
WNN	[230]	Time series forecast
	[231]	Predict the speed of the wind
	[232]	In forecasting the wind power fluctuations can be mitigated
ANFIS	[233]	A protection system is presented
	[234]	Demand of power is forecasted
	[235]	Solar radiation is forecasted
Decision tree	[236]	Blackout risk is forecasted
	[237]	Cost minimization in energy systems
Deep learning	[238]	Estimation of state-of-charge of battery
	[239]	Predicting the electricity demand in the households
	[240]	Energy consumption is forecasted
Ensemble model	[241]	Building electricity demand is forecasted
	[242]	Predict buildings cooling load

privacy rules will soon influence data collecting and, by extension, machine learning.

5. Machine Learning Applications

Because of its ability to make intelligent decisions and its potential to learn from the past, machine learning techniques are more popular in industry 4.0.

Here we discuss and summarize various machine learning techniques application areas.

5.1. Intelligent Decision-Making and Predictive Analytics. By making use of data-driven predictive analytics, intelligent decisions are made by applying machine learning techniques [2, 254]. To predict the unknown outcomes by relying on the earlier events by exploiting and capturing the relationship between the predicted variables and explanatory variables is the basis for predictive analytics [7], for example, credit card fraud identification and criminal identification after a crime. In the retail industry, predictive analytics and intelligent decision-making can be used for out-of-stock situation avoidance, inventory management, behavior, and preferences of the consumer are better understood and logistics and warehouse are optimized. Support Vector Machines, Decision Trees, and ANN are the most widely used techniques in the above areas [8, 10]. Predicting the outcome accurately can help every organization like social networking, transportation, sales and marketing, healthcare, financial services, banking services, telecommunication, e-commerce, industries, etc., to improve.

5.2. Cyber-Security and Threat Intelligence. Protecting data, hardware, systems, and networks is the responsibility of cyber-security and this is an important area in Industry 4.0 [5]. In cyber-security, one of the crucial technologies is machine learning which provides protection by securing cloud data, while browsing keeps people safe, foreseen the bad people online, insider threats are identified and malware is detected in the traffic. Machine learning classification models [255], deep learning-based security models [9], and association rule learning techniques [3] are used in cyber-security and threat intelligence.

5.3. Smart Cities. In IoT, all objects are converted into things by equipping objects with transmitting capabilities for transferring the information and performing jobs with no user intervention.

Some of the applications of IoT are business, healthcare, agriculture, retail, transportation, communication, education, smart home, smart governance [2], and smart cities [256, 257]. Machine learning has become a crucial technology in the internet of things because of its ability to analyze the data and predict future events [1]. For instance, congestion can be predicted in smart cities, take decisions based on the surroundings knowledge, energy estimation for a particular period, and predicting parking availability.

5.4. Transportation and Traffic Prediction. Generally, transportation networks have been an important part of every country's economy. Yet, numerous cities across the world are witnessing an enormous amount of traffic volume, leading to severe difficulties such as a decrease in the quality of life in modern society, crises, accidents, CO_2 pollution increased, higher fuel prices, traffic congestion, and delays [258]. As a result, an ITS, that predicts traffic and is critical, and it is an essential component of the smart city. Absolute forecasting of traffic based on deep learning and machine learning models can assist to mitigate problems [259–261]. For instance, machine learning may aid transportation firms in identifying potential difficulties that may arise on certain routes and advising that their clients choose an alternative way based on their history of travel and pattern of travel by taking variety of routes. Finally, by predicting and visualizing future changes, these solutions will assist to optimize flow of the traffic, enhance the use and effectiveness of sustainable forms of transportation, and reduce real-world disturbance.

5.5. Healthcare and COVID-19 Pandemic. In a variety of medical-related application areas, like prediction of illness, extraction of medical information, data regularity identification, management of patient data, and so on, machine learning may assist address diagnostic and prognostic issues [262–264]. Here in this article [265], coronavirus is considered as an infectious disease by the WHO. Learning approaches have recently been prominent in the fight against COVID-19 [266, 267].

Learning approaches are being utilized to categorize the death rate, patients at high risk, and various abnormalities in the COVID-19 pandemic [268]. It may be utilized to fully comprehend the virus's origins, predict the outbreak of COVID-19, and diagnose and treat the disease [269, 270]. Researchers may use machine learning to predict where and when COVID-19 will spread, and then inform those locations to make the necessary preparations. For COVID 19 pandemic [271–273], to address the medical image processing problems, deep learning can provide better solutions. Altogether, deep and machine learning approaches can aid in the battle against the COVID-19 virus and pandemic, and perhaps even the development of intelligent clinical judgments in the healthcare arena.

5.6. Product Recommendations in E-commerce. One of the most prominent areas in e-commerce where machine learning techniques are used is suggesting products to the users of the e-commerce. Technology of machine learning can help e-commerce websites to analyze their customers' purchase histories and provide personalized product recommendations based on their behavior and preferences for their next purchase. By monitoring browsing tendencies and click-through rates of certain goods, e-commerce businesses, for example, may simply place product suggestions and offers. Most merchants, such as flipkart [274] and amazon, can avert out-of-stock problems, manage better inventory, optimize storage, and optimize transportation by using machine learning-based predictive models. Future of marketing and sales is to improve the personalized experience of the users while purchasing the products by collecting their data and analyzing the data and use them to improve the experience of the users. In addition, to attract new customers and also to retain the existing ones the e-commerce website will build packages to attract the customers and keep the existing ones.

5.7. Sentiment Analysis and NLP (Natural Language Processing). An act of using a computer system to read and comprehend spoken or written language [275] is called Natural Language Processing. Thus, NLP aids computers in reading texts, hearing speech, interpreting it, analyzing sentiment, and determining which elements are important, all of which may be done using machine learning techniques. Some of the examples of NLP are machine translation, language translation, document description, speech recognition, chatbot, and virtual personal assistant. Collecting data and generating views and mood of the public from news, forums, social media, reviews, and blogs is the responsibility of sentiment analysis [276] which is a subfield of NLP. In sentiment analysis, texts are analyzed by using machine learning tasks to identify the polarity like neutral, negative and positive and emotions like not interested, have interest, angry, very sad, sad, happy, and very happy.

5.8. Image, Speech and Pattern Recognition. Machine Learning is widely used in the image recognition [277] whose task is to detect the object in an image. Some of the instances of image recognition are social media suggestions tagging, face detection, character recognition and cancer label on an X-ray image. Alexa, Siri, Cortana, Google Assistant etc., are the famous linguistic and sound models in speech recognition [286282]. The automatic detection of patterns and data regularities, such as picture analysis, is characterized as pattern recognition [278]. Several machine learning approaches are employed in this field, including classification, feature selection, clustering, and sequence labeling.

5.9. Sustainable Agriculture. Agriculture is necessary for all human activities to survive [279]. Sustainable agriculture techniques increase agricultural output while decreasing negative environmental consequences [279–281]. In articles [281–284] authors convey those emerging technologies like mobile devices, mobile technologies, Internet of Things can be used to capture the huge amounts of data to encourage the adoption of practices of sustainable agriculture by encouraging knowledge transfer among farmers. By using technologies, skills, information knowledge-intensive supply chains are developed in sustainable agriculture. Various techniques of machine learning can be applied in processing phase of the agriculture, production phase and preproduction phase, distribution phases like consumer analysis, inventory management, production planning, demand estimation of livestock, soil nutrient management, weed detection, disease detection, weather prediction, irrigation requirements, soil properties, and crop yield prediction.

5.10. Context-Aware and Analytics of User Behavior. Capturing information or knowledge about the surrounding is called context-awareness and tunes the behaviors of the system accordingly [285, 286]. Hardware and software are used in context-aware computing for automating the collection and interpreting of the data.

From the historical data [1, 287] machine learning will derive knowledge by using their learning capabilities which is used for bringing tremendous changes in the mobile app development environment.

Smart apps can be developed by the programmers, using which uses can be entertained, support is provided for the user and human behavior is understood [288–290] and can build a variety of context-aware systems based on data-driven approaches like context-aware smart searching, smart interruption management, smart mobile recommendation, etc., for instance, as in [291] phone call app can be created by using association rules with context awareness. Clustering approaches [292] are used and classification methods [10, 293, 294] are used for predicting future events and for capturing users' behavior.

6. Challenges and Future Research Directions

In this review, quite a few research issues are raised by studying the applicability of variety of ML approaches in the analysis of applications and intelligent data. Here, opportunities in research and potential future directions are summarized and discussed.

Research directions are summarized as follows:

(i) While dealing with real-world data, there is a need for focusing on the in-detail study of the capturing techniques of data

(ii) There is a huge requirement for fine-tuning the preprocessing techniques or to have novel data preprocessing techniques to deal with real-world data associated with application domains

(iii) Identifying the appropriate machine learning technique for the target application is also one of the research interests

(iv) There is a huge interest in the academia in existing machine learning hybrid algorithms enhancement or modification and also in proposing novel hybrid algorithms for their applicability to the target applications domain

Machine learning techniques' performance over the data and the data's nature and characteristics will decide the efficiency and effectiveness of the machine learning solutions. Data collection in various application domains like agriculture, healthcare, cyber-security etc., is complicated because of the generation of huge amounts of data in very less time by these application domains. To proceed further in the analysis of the data in machine learning-based applications relevant data collection is the key factor. So, while dealing with real-world data, there is a need for focusing on the more deep investigation of the data collection methods.

There may be many outliers, missing values, and ambiguous values in the data that is already existing which will impact the machine learning algorithms training. Thus, there is a requirement for the cleansing of collected data from variety of sources which is a difficult task. So, there is a need for preprocessing methods to be fine-tuned and novel preprocessing techniques to be proposed that can make machine learning algorithms to be used effectively.

Choosing an appropriate machine learning algorithm best suited for the target application, for the extraction insights, and for analyzing the data is a challenging task, because the characteristics and nature of the data may have an impact on the outcome of the different machine learning techniques [10]. Inappropriate machine learning algorithm will generate unforeseen results which might reduce the accuracy and effectiveness of the model. For this purpose, the focus is on hybrid models, and these models are fine-tuned for the target application domains or novel techniques are to be proposed.

Machine learning algorithms and the nature of the data will decide the ultimate success of the applications and their corresponding machine learning-based solutions. Machine Learning models will generate less accuracy and become useless when the data is the insufficient quantity for training, irrelevant features, poor quality, and non-representative and bad data to learn. For an intelligent application to be built, there are two important factors i.e., various learning techniques handling and effective processing of data.

Our research into machine learning algorithms for intelligent data analysis and applications raises a number of new research questions in the field. As a result, we highlight the issues addressed, as well as prospective research possibilities and future initiatives, in this section.

The nature and qualities of the data, as well as the performance of the learning algorithms, determine the effectiveness and efficiency of a machine learning-based solution. To gather information in a specific domain, such as cyber security, IoT, healthcare, agriculture, and so on. As a result, data for the target machine learning-based applications is collected. When working with real-world data, a thorough analysis of data collection methodologies is required. Furthermore, historical data may contain a large number of unclear values, missing values, outliers, and data that has no meaning.

Many machine learning algorithms exist to analyze data and extract insights; however, the ultimate success of a machine learning-based solution and its accompanying applications is largely dependent on both the data and the learning algorithms. Produce reduced accuracy if the data is bad to learn, such as non-representative, poor-quality, irrelevant features, or insufficient amount for training. As a result, establishing a machine learning-based solution and eventually building intelligent applications, correctly processing the data, and handling the various learning algorithms is critical.

7. Conclusion and Future Scopre

In this study on machine learning algorithms, a comprehensive review is conducted for applications and intelligent analysis of data. Here, the real-world issues and how solutions are prepared by using a variety of learning algorithms are discussed briefly. Machine Learning techniques' performance and characteristics of the data will decide the machine learning model's success. To generate intelligent decision-making, machine learning algorithms need to be acquainted with target application knowledge and trained with data collected from various real-world situations. For highlighting the applicability of ML approaches to variety of issues in the real world and variety of application areas are discussed in this review. At last, research directions and other challenges are discussed and summarized. All the challenges in the target applications domain must be addressed by using solutions effectively. For both industry professionals and academia, this study will serve as a reference point and from the technical perspective, this study also works as a benchmark for the decision makers on a variety of application domains and various real-world situations. Machine Learning's application is not restricted to any one sector. Rather, it is spreading across a wide range of industries, including banking and finance, information technology, media and entertainment, gaming, and the automobile sector. Because the breadth of Machine Learning is so broad, there are several areas where academics are trying to revolutionize the world in the future.

Data Availability

No data were used to support this study..

Conflicts of Interest

The authors declare that they have no conflicts of interest to report regarding the present study.

Authors' Contributions

All authors substantially contributed to the conception and design of the article and interpretation of the relevant literature.

References

[1] I. H. Sarker, M. M. Hoque, M. D. K. Uddin, and A. Tawfeeq, "Mobile data science and intelligent apps: concepts, ai-based modeling and research directions," *Mobile Networks and Applications*, vol. 26, no. 1, pp. 1–19, 2020.

[2] M. S. Mahdavinejad, M. Rezvan, M. Barekatain, P. Adibi, P. Barnaghi, and A. P. Sheth, "Machine learning for internet of things data analysis: a survey," *Digital Communications and Networks*, vol. 4, no. 3, pp. 161–175, 2018.

[3] I. H. Sarker, A. S. M. Kayes, S. Badsha, H. Alqahtani, P. Watters, and A. Ng, "Cybersecurity data science: an overview from machine learning perspective," *Journal of Big Data*, vol. 7, no. 1, pp. 41–29, 2020.

[4] I. H. Sarker and R. FurhadNowrozy, "Ai-driven cybersecurity: an overview, security intelligence modeling and research directions," *SN Computer Science*, vol. 2, no. 3, p. 173, 2021.

[5] B. Slusarczyk, "Industry 4.0 - are we ready?" *Polish Journal of Management Studies*, vol. 17, no. 1, pp. 232–248, 2018.

[6] Google, "Trends In," 2019, https://trends.google.com/trends/.

[7] J. Han, J. Pei, and M. Kamber, *Data Mining: Concepts and Techniques*, Elsevier, Amsterdam, 2011.

[8] I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, Morgan Kaufmann, Burlingto,MA, USA, 2005.

[9] I. H. Sarker, "Deep cybersecurity: a comprehensive overview from neural network and deep learning perspective," *SN Computer Science*, vol. 2, no. 3, 154 pages, 2021.

[10] I. H. Sarker, P. Watters, and A. S. M. Kayes, "Effectiveness analysis of machine learning classifcation models for predicting personalized context-aware smartphone usage," *J Big Data*, vol. 6, no. 1, pp. 1–28, 2019.

[11] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement learning: a survey," *Journal of Artificial Intelligence Research*, vol. 4, pp. 237–285, 1996.

[12] R. Jhaveri, S. Ramani, G. Srivastava, T. Reddy Gadekallu, and V. Aggarwal, "Fault-Resilience for Bandwidth Management in Industrial Software-Defined Networks," *IEEE Transactions on Network Science and Engineering*, IEEE, vol. 8, no. 4, pp. 3129–3139, Aug 2021.

[13] ICT-Fact, "ICT-Fact and Figures 2017 Accessed," 2020, https://www.itu.int/en/ITU-D/Statistics/Documents/facts/ICTFactsFigures2017.pdf.

[14] ICT-Facts, "Telecommunication Union," 2017, https://www.itu.int/en/ITU-D/Statistics/Pages/facts/default.aspx.

[15] D. K. Bhattacharyya and J. K. Kalita, *Network Anomaly Detection: A Machine Learning Perspective*, Chapman & Hall, London, U.K, 2013.

[16] V. Ambalavanan and B. P. Shanthi, "Cyber threats detection and mitigation using machine learning," in *Handbook of Research on Machine and Deep Learning Applications for Cyber Security*, pp. 132–149, IGI Global, Hershey, PA, USA, 2020.

[17] T. Thomas, A. Vijayaraghavan, and S. Emmanuel, "Machine Learning and Cybersecurity," in *Machine Learning Approaches in Cyber Security Analytics*, pp. 37–47, Springer, Berlin, Germany, 2020.

[18] S. Latif, J. Qadir, S. Farooq, and M. Imran, "How 5G wireless (and concomitant technologies) will revolutionize healthcare?" *Future Internet*, vol. 9, no. 4, p. 93, 2017.

[19] Y. Zhan, Y. Zhennan, P. Zhigang et al., "Multi-instance deep learning: discover discriminative local anatomies for bodypart recognition," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1332–1343, 2016.

[20] M. Anthimopoulos, S. Christodoulidis, L. Ebner, A. Christe, and S. Mougiakakou, "Lung pattern classification for interstitial lung diseases using a deep convolutional neural network," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1207–1216, 2016.

[21] W. Shen, M. Zhou, F. Yang, C. Yang, and J. Tian, "Multi-scale convolutional neural networks for lung nodule classification," *Lecture Notes in Computer Science*, vol. 24, pp. 588–599, 2015.

[22] J. Schlemper, J. Caballero, J. V. Hajnal, A. Price, and D. Rueckert, "A deep cascade of convolutional neural networks for MR image reconstruction," *Lecture Notes in Computer Science*, pp. 647–658, 2017.

[23] J. Majumdar and A. Majumdar, "Rodeo: robust de-aliasing autoencoder for real-time medical image reconstruction," *Pattern Recognition*, vol. 63, pp. 499–510, 2017.

[24] A. Davy, M. Havaei, D. Warde-Farley et al., "Brain tumor segmentation with deep neural networks," *Medical Image Analysis*, vol. 35, pp. 18–31, 2017.

[25] H. Wang, X. Li, H. Jhaveri et al., "Sparse Bayesian learning based channel estimation in FBMC/OQAM industrial IoT networks," *Computer Communications*, Elsevier, vol. 176, pp. 40–45, 2021.

[26] L. Xing, E. A. Krupinski, and J. Cai, "Artificial intelligence will soon change the landscape of medical physics research and practice," *Medical Physics*, vol. 45, no. 5, pp. 1791–1793, 2018.

[27] S. Latif, M. Asim, M. Usman, J. Qadir, and R. Rana, "Automating motion correction in multishot MRI using generative adversarial networks," in *Proceedings of the 32nd International Conference on Neural Information Processing Syste*, Montréal Canada, December 2018.

[28] X. W. Xiaotong and X. Lin, "Big data deep learning: challenges and perspectives," *IEEE Access*, vol. 2, pp. 514–525, 2014.

[29] R. H. Jhaveri, R. Tan, A. Easwaran, and S. V. Ramani, "Managing Industrial Communication Delays with Software-Defined Networking," in *Proceedings of the 25th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA 2019)*, pp. 1–11, IEEE, Hangzhou, China, August 2019.

[30] C. Campolo, A. Molinaro, and R. Scopigno, "From today's VANETs to tomorrow's planning and the bets for the day after," *Vehicular Communications*, vol. 2, no. 3, pp. 158–171, 2015.

[31] Q. Mao, F. Hu, and Q. Hao, "Deep learning for intelligent wireless networks: a comprehensive survey," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 4, pp. 2595–2621, 2018.

[32] C. Zhang, P. Patras, and H. Haddadi, "Deep Learning in mobile and Wireless Networking: A Survey," 2018, https://arxiv.org/abs/1803.04311.

[33] N. C. Luong, D. T. Hoang, S. Gong et al., "Applications of Deep Reinforcement Learning in Communications and Networking: A Survey," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 4, pp. 3133–3174, 2018.

[34] United States Department of Energy, "Renewable Energy Data Book," 2012, http://www.nrel.gov/docs/fy14osti/60197.pdf.

[35] M. Deshmukh and S. Deshmukh, "Modeling of hybrid renewable energy systems," *Renewable and Sustainable Energy Reviews*, vol. 12, no. 1, pp. 235–249, 2008.

[36] O. I. Elgerd, *Electric Energy Systems Theory: An Introduction*, McGraw-Hill, New York, NY, USA, 1982.

[37] A. Gomez-Exposito, A. J. Conejo, and C. Canizares, *Electric Energy Systems: Analysis and Operation*, CRC Press, Boca Raton, FL, USA, 2018.

[38] V. Quaschning, *Understanding Renewable Energy Systems*, Routledge, Abingdon, UK, 2016.

[39] O. Isabella, A. Smets, K. Jäger, M. Zeman, and R. van Swaaij, *Solar Energy: The Physics and Engineering of Photovoltaic Conversion, Technologies and Systems*, UIT Cambridge Limited, Cambridge, UK, 2016.

[40] P. Dixit, R. Kohli, A. Acevedo-Duque, R. R. Gonzalez-Diaz, and R. H. Jhaveri, "Comparing and Analyzing Applications of Intelligent Techniques in Cyber-Attack Detection," *Security and Communication Networks*, Hindawi, vol. 2021, Article ID 5561816, 23 pages, 2021.

[41] M. Torabi, S. Hashemi, M. R. Saybani, S. Shamshirband, and A. Mosavi, "A Hybrid clustering and classification technique for forecasting short-term energy consumption," *Environmental Progress & Sustainable Energy*, vol. 38, no. 1, pp. 66–76, 2018.

[42] B. Najafi, S. Ardabili, A. Mosavi, S. S. Band, and T. Rabczuk, "An intelligent artificial neural network-response surface methodology method for accessing the optimum biodiesel and diesel fuel blending conditions in a diesel engine from the viewpoint of exergy and energy analysis," *Energies*, vol. 11, no. 4, p. 860, 2018.

[43] M. Hosseini-Imani, S. Zalzar, A. Mosavi, and S. S. Band, "Strategic behavior of retailers for risk reduction and profit increment via distributed generators and demand response programs," *Energies*, vol. 11, no. 6, p. 1602, 2018.

[44] A. Dinev, A. Mosavi, S. Faizollahzadeh-Ardabili et al., "Review of soft computing models in design and control of rotating electrical machines," *Energies*, vol. 12, no. 6, p. 1049, 2019.

[45] L. W. Chong, Y. W. Wong, R. K. Rajkumar, and D. Isa, "Hybrid energy storage systems and control strategies for stand-alone renewable energy power systems," *Renewable and Sustainable Energy Reviews*, vol. 66, pp. 174–189, 2016.

[46] R. Panigrahi, S. Borah, A. Kumar-Bhoi et al., "A Consolidated Decision Tree-Based Intrusion Detection System for Binary and Multiclass Imbalanced Datasets," *Mathematics*, vol. 9, p. 751, 2021.

[47] K. Amarasinghe, D. L. Marino, and M. Manic, "Deep neural networks for energy load forecasting," in *Proceedings of the 2017 IEEE 26th International Symposium on Industrial Electronics (ISIE)*, pp. 1483–1488, Edinburgh, UK, June, 2017.

[48] V. H. Quej, J. Almorox, J. A. Arnaldo, and L. Saito, "ANFIS, SVM and ANN soft-computing techniques to estimate daily global solar radiation in a warm sub-humid environment," *Journal of Atmospheric and Solar-Terrestrial Physics*, vol. 155, pp. 62–70, 2017.

[49] M. A. Mat-Daut, M. Y. Hassan, H. Abdullah, H. A. Rahman, M. P. Abdullah, and F. Hussin, "Building electrical energy consumption forecasting analysis using conventional and artificial intelligence methods: a review," *Renewable and Sustainable Energy Reviews*, vol. 70, pp. 1108–1118, 2017.

[50] B. Yildiz, J. I. Bilbao, and A. B. Sproul, "A review and analysis of regression and machine learning models on commercial building electricity load forecasting," *Renewable and Sustainable Energy Reviews*, vol. 73, pp. 1104–1122, 2017.

[51] T. R. Gadekallu, M. K. Manoj, S. Sivarama-Krishna, K. Neeraj, S. Hakak, and S. Bhattacharya, "Blockchain-based attack detection on machine learning algorithms for IoT-based e-health applications," *IEEE Internet of Things Magazine*, vol. 4, no. 3, pp. 30–33, 2021.

[52] S. Faizollahzadeh-Ardabili, B. Najafi, M. Alizamir, A. Mosavi, S. Shamshirband, and T. Rabczuk, "Using SVM-RSM and ELM-RSM approaches for optimizing the production process of methyl and ethyl esters," *Energies*, vol. 11, no. 11, p. 2889, 2018.

[53] M. H. Abidi, H. Alkhalefah, K. Moiduddin et al., "Optimal 5G network slicing using machine learning and deep learning concepts," *Computer Standards & Interfaces*, vol. 76, no. 2021, Article ID 103518, 2021.

[54] K. Amasyali and N. M. El-Gohary, “A review of data-driven building energy consumption prediction studies,” *Renewable and Sustainable Energy Reviews*, vol. 81, pp. 1192–1205, 2018.
[55] Y. Peng, A. Rysanek, Z. Nagy, and A. Schlüter, “Using machine learning techniques for occupancy-prediction-based cooling control in office buildings,” *Applied Energy*, vol. 211, pp. 1343–1358, 2018.
[56] A. Tsanas and A. Xifara, “Accurate quantitative estimation of energy performance of residential buildings using statistical machine learning tools,” *Energy and Buildings*, vol. 49, pp. 560–567, 2012.
[57] C. Robinson, B. Dilkina, J. Hubbs et al., “Machine learning approaches for estimating commercial building energy consumption,” *Applied Energy*, vol. 208, pp. 889–904, 2017.
[58] D. S. Prashanth, R. V. K. Mehta, K. Ramana, and V. Bhaskar, “Handwritten devanagari character recognition using modified lenet and alexnet convolution neural networks,” *Wireless Personal Communications*, vol. 122, no. 1, pp. 349–378, 2021.
[59] H. C. Kao, K. F. Tang, and E. Y. Chang, “Context-aware symptom checking for disease diagnosis using hierarchical reinforcement learning,” *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.
[60] D. Silver, A. Huang, C. J. Maddison et al., “Mastering the game of go with deep neural networks and tree search,” *Nature*, vol. 529, no. 7587, pp. 484–489, 2016.
[61] A. Zahin and R. Q. Hu, “Sensor-based human activity recognition for smart healthcare: a semi-supervised machine learning,” *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pp. 450–472, 2019.
[62] W. Oktay, M. Sinclair, H. Suzuki et al., “Semi-supervised learning for network-based cardiac MR image segmentation,” *Lecture Notes in Computer Science*, pp. 253–260, 2017.
[63] M. N. Sohail, J. Ren, and M. Uba-Muhammad, “A Euclidean group assessment on semi-supervised clustering for healthcare clinical implications based on real-life data,” *International Journal of Environmental Research and Public Health*, vol. 16, no. 9, p. 1581, 2019.
[64] M. Ponnavaikko and A. Subramanyam, “A global dispatcher load balancing (GLDB) approach for a web server cluster,” in *International Conference on Communications and Cyber Physical Engineering*, pp. 341–357, Springer, Singapore, 2018, Lecture Notes in Electrical Engineering.
[65] K. Polat and S. Güne, “Prediction of hepatitis disease based on principal component analysis and artificial immune recognition system,” *Applied Mathematics and Computation*, vol. 189, no. 2, pp. 1282–1291, 2007.
[66] A. K. Pandey, P. Pandey, K. Jaiswal, and A. K. Sen, “Datamining clustering techniques in the prediction of heart disease using attribute selection method,” *Heart Disease*, vol. 14, pp. 16-17, 2013.
[67] A. Collins and Y. Yao, “Machine learning approaches: data integration for disease prediction and prognosis,” in *Applied Computational Genomics*, pp. 137–141, Springer, Berlin, Germany, 2018.
[68] P. Afshar, A. Mohammadi, and K. N. Plataniotis, “Brain tumor type classification via capsule networks,” in *Proceedings of the. 25th IEEE International Conference Image Process*, pp. 3129–3133, Athens, Greece, October, 2018.
[69] W. Zhu, C. Liu, W. Fan, and X. Xie, “Deeplung: deep 3D dual path nets for automated pulmonary nodule detection and classification,” in *Proceedings of the 2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 673–681, Lake Tahoe, NV, USA, March, 2018.
[70] P. B. Jensen, L. J. Jensen, and S. Brunak, “Mining electronic health records: towards better research applications and clinical care,” *Nature Reviews Genetics*, vol. 13, no. 6, pp. 395–405, 2012.
[71] Z. Wang, A. D. Shah, A. R. Tate, S. Denaxas, J. ShaweTaylor, and H. Hemingway, “Extracting diagnoses and investigation results from unstructured text in electronic health records by semi-supervised machine learning,” *PLoS One*, vol. 7, no. 1, Article ID e30412, 2012.
[72] T. Zheng, W. Xie, L. Xu et al., “A machine learning-based framework to identify type 2 diabetes through electronic health records,” *International Journal of Medical Informatics*, vol. 97, pp. 120–127, 2017.
[73] B. Nestor, M. McDermott, W. Boag et al., “Feature Robustness in Non-stationary Health Records: Caveats to Deployable Model Performance in Common Clinical Machine Learning Tasks,” 2019, https://arxiv.org/abs/1908.00690.
[74] M. Lustig, D. L. Donoho, J. M. Santos, and J. M. Pauly, “Compressed sensing MRI,” *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 72–82, 2008.
[75] L. Gondara, “Medical image denoising using convolutional denoising autoencoders,” in *Proceedings of the IEEE International Conference on Data Mining Workshops (ICDM Workshops)*, pp. 241–246, Barcelona, Spain, December 2016.
[76] Y. Chen, Y. Xie, Z. Zhou, F. Shi, A. G. Christodoulou, and D. Li, “Brain MRI super resolution using 3D deep densely connected neural networks,” in *Proceedings of the2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pp. 739–742, Washington, DC, USA, April 2018.
[77] K. Sirinukunwattana, N. Rajpoot, S. E. A. Raza, Y. W. Tsang, D. Snead, and I. Cree, “Locality sensitive deep learning for detection and classification of nuclei in routine colon cancer histology images,” *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1196–1206, 2016.
[78] H. Wang, A. Basavanhally, H. Gilmore et al., “Mitosis detection in breast cancer pathology images by combining handcrafted and convolutional neural network features,” *Journal of Medical Imaging*, vol. 1, no. 3, Article ID 34003, 2014.
[79] Y. Yu, H. Lin, J. Meng, X. Wei, H. Guo, and Z. Zhao, “Deep transfer learning for modality classification of medical images,” *Information*, vol. 8, no. 3, p. 91, 2017.
[80] J. Antony, K. McGuinness, N. E. O’Connor, and K. Moran, “Quantifying radiographic knee osteoarthritis severity using deep convolutional neural networks,” in *Proceedings of the 2016 23rd International Conference on Pattern Recognition (ICPR)*, pp. 1195–1200, Cancun, Mexico, December, 2016.
[81] E. Kim, M. Corte-Real, and Z. Baloch, “A deep semantic mobile application for thyroid cytopathology,” in *Proceedings of the Medical Imaging 2014: PACS and Imaging Informatics: Next Generation and Innovations*, vol. 9789, San Diego, CA, USA, April 2016.
[82] M. F. Stollenga, W. Byeon, M. Liwicki, and J. Schmidhuber, “Parallel multi-dimensional LSTM, with application to fast biomedical volumetric image segmentation,” in *Proceedings of the 28th International Conference on Neural Information Processing Systems*, pp. 2998–3006, Montreal Canada, December, 2015.
[83] O. Ronneberger, P. Fischer, and T. Brox, “U-net: convolutional networks for biomedical image segmentation,” in *Proceedings of the International Conference on Medical Image*

Computing and Computer-Assisted Intervention, pp. 234–241, 2015.

[84] F. Milletari, N. Navab, and S. A. Ahmadi, “V-Net: Fully convolutional neural networks for volumetric medical image segmentation,” in *Proceedings of the 2016 Fourth International Conference on 3D Vision (3DV)*, pp. 565–571, Stanford, CA, USA, October, 2016.

[85] M. H. Hesamian, W. Jia, X. He, and P. Kennedy, “Deep learning techniques for medical image segmentation: achievements and challenges,” *Journal of Digital Imaging*, vol. 32, no. 4, pp. 582–596, 2019.

[86] H. Chen, Y. Zhang, M. K. Kalra et al., “Low-dose CT with a residual encoder-decoder convolutional neural network,” *IEEE Transactions on Medical Imaging*, vol. 36, no. 12, pp. 2524–2535, 2017.

[87] M. Usman, S. Latif, M. Asim, B. D. Lee, and J. Qadir, “Retrospective motion correction in multishot MRI using generative adversarial network,” *Scientific Reports*, vol. 10, no. 1, pp. 4786–4811, 2020.

[88] F. El-Gamal, M. Elmogy, and A. Atwan, “Current trends in medical image registration and fusion,” *Egyptian Informatics Journal*, vol. 17, no. 1, pp. 99–124, 2016.

[89] J. Ker, L. Wang, J. Rao, and T. Lim, “Deep learning applications in medical image analysis,” *IEEE Access*, vol. 6, pp. 9375–9389, 2018.

[90] X. Yang, R. Kwitt, M. Styner, and M. Niethammer, “Quicksilver: fast predictive image registration - a deep learning approach,” *NeuroImage*, vol. 158, pp. 378–396, 2017.

[91] S. Miao, Z. J. Wang, and R. Liao, “A CNN regression approach for real-time 2D/3D registration,” *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1352–1363, 2016.

[92] G. Kothai and E. Poovammal, “A new hybrid deep learning algorithm for prediction of wide traffic congestion in smart cities,” *Wireless Communications and Mobile Computing*, vol. 2021, pp. 1–13, 2021.

[93] A. Qayyum, S. M. Anwar, M. Awais, and M. Majid, “Medical image retrieval using deep convolutional neural network,” *Neurocomputing*, vol. 266, pp. 8–20, 2017.

[94] J. Zech, M. Pain, J. Titano et al., “natural language-Based machine learning models for the annotation of clinical radiology reports,” *Radiology*, vol. 287, no. 2, pp. 570–580, 2018.

[95] B. Jing, P. Xie, and E. Xing, “On the automatic generation of medical imaging reports,” *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics*, pp. 2577–2586, 2018.

[96] X. Wang, Y. Peng, L. Lu, Z. Lu, and R. M. Summers, “Tienet: textimage embedding network for common thorax disease classification and reporting in chest x-rays,” in *Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9049–9058, Salt Lake City, UT, USA, June 2018.

[97] Y. Xu, L. Rodney Long, Z. Xue, S. Antani, G. R. Thoma, and X. Huang, “Multimodal recurrent model with attention for automated radiology report generation,” in *Proceedings of the Medical Image Computing and Computer Assisted Intervention - MICCAI 2018*, pp. 457–466, Granada, Spain, September,2018.

[98] C. Virmani, T. Choudhary, A. Pillai, and M. Rani, “Applications of Machine Learning in Cyber Security,” in *Handbook of Research on Machine and Deep Learning Applications for Cyber Security*, pp. 83–103, IGI Global, Hershey, PA, USA, 2020.

[99] R. Calderon, “The benefits of artificial intelligence in cybersecurity,” Tech. Rep. Winter, La Salle University, Philadelphia, PA, USA, 2019.

[100] M. Taddeo, “Three ethical challenges of applications of artificial intelligence in cybersecurity,” *Minds and Machines*, vol. 29, no. 2, pp. 187–191, 2019.

[101] K. Shaukat, A. Rubab, I. Shehzadi, and R. Iqbal, “A sociotechnological analysis of cyber crime and cyber security in Pakistan,” *Transylvanian Rev*, vol. 1, no. 3, pp. 4187–4190, 2017.

[102] R. Sagar, R. Jhaveri, and C. Borrego, “Applications in security and evasions in machine learning: a survey,” *Electronics*, vol. 9, no. 1, p. 97, Jan. 2020.

[103] K. Shaukat, S. Luo, V. Varadharajan et al., “Performance comparison and current challenges of using machine learning techniques in cybersecurity,” *Energies*, vol. 13, no. 10, p. 2509, 2020.

[104] B. H. Li, B. C. Hou, W. T. Yu, X.-B. Lu, and C. W. Yang, “Applications of artificial intelligence in intelligent manufacturing: a review,” *Frontiers of Information Technology & Electronic Engineering*, vol. 18, no. 1, pp. 86–96, 2017.

[105] S. Topol and E. J. Topol, “Adapting to artificial intelligence,” *JAMA*, vol. 316, no. 22, pp. 2353-2354, 2016.

[106] A. I. Tekkesin, “‘Artificial intelligence in healthcare: past, present and future,” *The Anatolian Journal of Cardiology*, vol. 2, no. 4, pp. 230–243, 2019.

[107] K. Shaukat, N. Masood, A. Bin Shafaat, K. Jabbar, H. Shabbir, and S. Shabbir, “Dengue Fever in Perspective of Clustering Algorithms,” 2015, http://arxiv.org/abs/1511.07353.

[108] A. Balakrishnan, K. Ramana, G. Dhiman et al., “Multimedia concepts on object detection and recognition with F1 car simulation using convolutional layers,” *Wireless Communications and Mobile Computing*, vol. 2021, pp. 1–21, 2021.

[109] K. Shaukat, I. Nawaz, S. Aslam, S. Zaheer, and U. Shaukat, “Student’s performance in the context of data mining,” in *Proceedings of the 2016 19th International Multi-Topic Conference (INMIC)*, Islamabad, Pakistan, December, 2016.

[110] S. Gu, B. T. Kelly, and D. Xiu, “Empirical asset pricing via machine learning,” in *Proceedings of the 31st Australasian Finance and Banking Conference 2018*, Sydney, Australia, December 2018.

[111] P. Mathur, “Overview of Machine Learning in Finance,” in *in Machine Learning Applications Using Python*Apress, Berkeley, CA, USA, 2019.

[112] S. Emerson, R. Kennedy, L. O’Shea, and J. O’Brien, “‘Trends and applications of machine learning in quantitative finance,” *8th International Conference on Economics and Finance Research (ICEFR 2019)*, pp. 1–9, France Lyon, June 2019.

[113] K. A. P. da Costa, J. P. Papa, C. O. Lisboa, R. Munoz, and V. H. C. de Albuquerque, “Internet of Things: a survey on machine learning-based intrusion detection approaches,” *Computer Networks*, vol. 151, pp. 147–157, Mar. 2019.

[114] S. Dey, Q. Ye, and S. Sampalli, “A machine learning based intrusion detection scheme for data fusion in mobile clouds involving heterogeneous client networks,” *Information Fusion*, vol. 49, pp. 205–215, 2019.

[115] M. A. Ferrag, L. Maglaras, S. Moschoyiannis, and H. Janicke, “Deep learning for cyber security intrusion detection: approaches, datasets, and comparative study,” *Journal of Information Security and Applications*, vol. 50, Article ID 102419, 2020.

[116] M. Pradhan, C. K. Nayak, and S. K. Pradhan, *Intrusion Detection System (IDS) and Their Types," in Securing the Internet of Things: Concepts, Methodologies, Tools, and Applications*, pp. 481–497, IGI Global, Hershey, PA, USA, 2020.

[117] M. Almukaynizi, A. Grimm, E. Nunes, J. Shakarian, and P. Shakarian, "Predicting Cyber Threats through the Dynamics of User Connectivity in Darkweb and Deepweb Forums," in *Proceedings of the 2017 International Conference of The Computational Social Science Society of the Americas*, Santa Fe NM USA, October 2017, https://usc-isi-i2.github.io/papers/kristina02.pdf.

[118] M. Almukaynizi, A. Grimm, E. Nunes, J. Shakarian, and P. Shakarian, "Predicting cyber threats through hacker social networks in darkweb and deepweb forums," *Proc. Int. Conf. Comput. Social Sci. Soc. Americas (CSS)*, pp. 1–7, 2017.

[119] R. S. Pais and A. R. Pais, "Detection of phishing websites using an efficient feature-based machine learning framework," *Neural Computing & Applications*, vol. 31, no. 8, pp. 3851–3873, 2019.

[120] O. K. Sahingoz, E. Buber, O. Demir, and B. Diri, "Machine learning based phishing detection from URLs," *Expert Systems with Applications*, vol. 117, pp. 345–357, Mar. 2019.

[121] M. Alauthman, A. Almomani, M. Alweshah, W. Omoushd, and K. Alieyane, "'Machine learning for phishing detection and mitigation," *in Machine Learning for Computer and Cyber Security: Principle, Algorithms, and Practices*, CRC Press, New York, NY, USA, 2019.

[122] Z. Ma, H. Ge, Y. Liu, M. Zhao, and J. Ma, "A combination method for android malware detection based on control flow graphs and machine learning algorithms," *IEEE Access*, vol. 7, Article ID 21235, 2019.

[123] S. Saad, W. Briguglio, and H. Elmiligi, "The Curious Case of Machine Learning in Malware Detection," 2019, http://arxiv.org/abs/1905.075.

[124] P. Jain, "Machine Learning versus Deep Learning for Malware Detection,", Master's Projects, San Jose State University , San Jose, WA, USA, 2019.

[125] D. Sahoo, C. Liu, and S. C. H. Hoi, "Malicious URL Detection Using Machine Learning: A Survey," 2017, http://arxiv.org/abs/1701.07179.

[126] D. Prusti, S. S. Padmanabhuni, and S. K. Rath, "'Credit card fraud detection by implementing machine learning techniques," in *Technical Reports Series*Springer, Berlin, Germany, 2019, https://dspace.nitrkl.ac.in/dspace/handle/2080/3273.

[127] H. A. Shukur and S. Kurnaz, "'Credit card fraud detection using machine learning methodology," *International Journal of Computer Science and Mobile Computing*, vol. 8, no. 3, pp. 257–260, Mar. 2019.

[128] M. Lokanan, V. Tran, and N. H. Vuong, "Detecting anomalies in financial statements using machine learning algorithm," *Asian Journal of Accounting Research*, vol. 4, no. 2, pp. 181–201, Oct. 2019.

[129] A. M. Mubalaike and E. Adali, "'Deep learning approach for intelligent financial fraud detection system," in *Proceedings of the 3rd Int. Conf. Comput. Sci. Eng. (UBMK)*, pp. 598–603, September 2018.

[130] S. Selvaraj, "Applying of Machine Learning for Spam Classification," M. S. thesis, Instytut Telekomunikacji, Kraków, Poland, 2019.

[131] A. K. Cherian, E. Poovammal, N. S. Philip, K. Ramana, S. Singh, and I. H. Ra, "Deep learning based filtering algorithm for noise removal in underwater images," *Water*, vol. 13, no. 19, p. 2742, 2021.

[132] E. G. Dada, J. S. Bassi, H. Chiroma, S. M. Abdulhamid, A. O. Adetunmbi, and O. E. Ajibuwa, "Machine learning for email spam filtering: review, approaches and open research problems," *Heliyon*, vol. 5, no. 6, Article ID e01802, 2019.

[133] A. K. Jain, D. Goel, S. Agarwal, Y. Singh, and G. Bajaj, "Predicting spam messages using back propagation neural network," *Wireless Personal Communications*, vol. 110, no. 1, pp. 403–422, Jan. 2020.

[134] S. Sarkar, M. Almukaynizi, J. Shakarian, and P. Shakarian, "Predicting enterprise Cyber Incidents Using Social Network Analysis on Dark Web Hacker Forums," *Cyber Defense Review*, vol. 2019, pp. 87–102, Mar. 2019.

[135] A. Zenebe, M. Shumba, A. Carillo, and S. Cuenca, "'Cyber threat discovery from dark Web," in *Proceedings of the 28th International Conference on Software Engineering and Data Engineering*, vol. 64, San Diego, CA, USA, October 2019.

[136] E. Nunes, A. Diab, A. Gunn et al., "'Darknet and deepnet mining for proactive cybersecurity threat intelligence," in *Proceedings of the 2016 IEEE Conference on Intelligence and Security Informatics (ISI)*, pp. 7–12, Tucson, AZ, USA, September. 2016.

[137] M. Kadoguchi, S. Hayashi, M. Hashimoto, and A. Otsuka, "'Exploring the dark Web for cyber threat intelligence using machine leaning," in *Proceedings of the 2019 IEEE International Conference on Intelligence and Security Informatics (ISI)*, pp. 200–202, Shenzhen, China, July. 2019.

[138] X. Chow and K. Chow, "A framework for dark web threat intelligence analysis," *Cyber Warfare and Terrorism*, IGI Global, Hershey, PA, USA, pp. 266–276, 2020.

[139] Unified-threat-management, "Accessed," 2020, https://en.wikipedia.org/wiki/Unified_threat_management.

[140] Top-Intrusion-Detection-and-Prevention-Systems, "Guide to IDPS," 2020, https://www.esecurityplanet.com/products/top-intrusion-detection-preventionsystems.htm.

[141] Security, "information and event management (SIEM)," https://www.esecurityplanet.com/products/top-siem-products.html.

[142] B. Geluvaraj, P. Satwik, and T. A. Ashok Kumar, "The future of cybersecurity: major role of artificial intelligence, machine learning, and deep learning in cyberspace," in *Proceedings of the International Conference on Computer Networks and Communication Technologies*Singapore, Springer, 2019.

[143] A. Sharma, Z. Kalbarczyk, J. Barlow, and R. Iyer, "'Analysis of security data from a large computing organization," in *Proceedings of the 2011 IEEE/IFIP 41st International Conference on Dependable Systems & Networks (DSN)*, pp. 506–517, Hong Kong, China, June. 2011.

[144] T. T. Nguyen and V. J. Reddi, "Deep Reinforcement Learning for Cyber Security," 2019, http://arxiv.org/abs/1906.05799.

[145] K. Geis, "Machine Learning: Cybersecurity that Can Meet the Demands of Today as Well as the Demands of Tomorrow," Ph.D. dissertation, Master Sci. Cybersecurity, Utica College, Utica, NY, USA, 2019.

[146] M. Thangavel, A. S. Tgr, P. Priyadharshini, and T. Saranya, "'Review on machine and deep learning applications for cyber security," *in Handbook of Research on Machine and Deep Learning Applications for Cyber Security*, pp. 42–63, IGI Global, Hershey, PA, USA, 2020.

[147] M. Lei, L. Shiyan, J. Chuanwen, L. Hongling, and Z. Yan, "A review on the forecasting of wind speed and generated

power," *Renewable and Sustainable Energy Reviews*, vol. 13, no. 4, pp. 915–920, 2009.

[148] W. Guoyang, X. Yang, and W. Shasha, "Discussion about short-term forecast of wind speed on wind farm," *Jilin Electric Power*, vol. 181, pp. 21–24, 2005.

[149] M. Ding, L. J. Zhang, and Y. C. Wu, "Wind speed forecast model for wind farms based on time series analysis," *Electric Power Automation Equipment*, vol. 25, pp. 32–34, 2005.

[150] A. Sfetsos, "A comparison of various forecasting techniques applied to mean hourly wind speed time series," *Renewable Energy*, vol. 21, no. 1, pp. 23–35, 2000.

[151] A. Sfetsos, "A novel approach for the forecasting of mean hourly wind speed time series," *Renewable Energy*, vol. 27, no. 2, pp. 163–174, 2002.

[152] S. Li, "Wind power prediction using recurrent multilayer perceptron neural networks," *Proceedings of the 2003 IEEE Power Engineering Society General Meeting*, vol. 4, pp. 2325–2330, 2003.

[153] M. A. Mohandes, T. O. Halawani, S. Rehman, and A. A. Hussain, "Support vector machines for wind speed prediction," *Renewable Energy*, vol. 29, no. 6, pp. 939–947, 2004.

[154] I. G. Damousis, M. C. Alexiadis, J. B. Theocharis, and P. S. Dokopoulos, "A fuzzy model for wind speed prediction and power generation in wind parks using spatial correlation," *IEEE Transactions on Energy Conversion*, vol. 19, no. 2, pp. 352–361, 2004.

[155] G. Giebel, J. Badger, L. Landberg et al., "Wind Power Prediction Ensembles," Technical Report 1527, Ris National Laboratory, Denmark, 2005.

[156] S. Al-Yahyai, Y. Charabi, A. Al-Badi, and A. Gastli, "Nested ensemble NWP approach for wind energy assessment," *Renewable Energy*, vol. 37, no. 1, pp. 150–160, 2012.

[157] R. Jursa and K. Rohrig, "Short-term wind power forecasting using evolutionary algorithms for the automated specification of artificial intelligence models," *International Journal of Forecasting*, vol. 24, no. 4, pp. 694–709, 2008.

[158] R. Jursa, "Wind power prediction with different artificial intelligence models," in *Proceedings of the 2007 European Wind Energy Conference and Exhibition (EWEC)*, pp. 1–10, Milan, Italy, May 2007.

[159] A. M. Foley, P. G. Leahy, A. Marvuglia, and E. J. McKeogh, "Current methods and advances in forecasting of wind power generation," *Renewable Energy*, vol. 37, no. 1, pp. 1–8, 2012.

[160] S. A. Kalogirou, S. Panteliou, and A. Dentsoras, "Artificial neural networks used for the performance prediction of a thermosiphon solar water heater," *Renewable Energy*, vol. 18, no. 1, pp. 87–99, 1999.

[161] N. Sharma, P. Sharma, D. Irwin, and P. Shenoy, "Predicting solar generation from weather forecasts using machine learning," in *Proceedings of the 2011 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pp. 528–533, Brussels, Belgium, October 2011).

[162] M. R. Hossain, M. T. O. Amanullah, and A. B. M. Shawkat-Ali, "Hybrid prediction method of solar power using different computational intelligence algorithms," in *Proceedings of the 22nd Australasian Universities Power Engineering Conference (AUPEC)*, pp. 1–6, Bali, Indonesia, September 2012.

[163] M. Diagne, M. David, P. Lauret, J. Boland, and N. Schmutz, "Review of solar irradiance forecasting methods and a proposition for small-scale insular grids," *Renewable and Sustainable Energy Reviews*, vol. 27, pp. 65–76, 2013.

[164] G. Kechriotis, E. Zervas, and E. S. Manolakos, "Using recurrent neural networks for adaptive communication channel equalization," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 267–278, 1994.

[165] M. I. Jordan, "Attractor dynamics and parallelism in a connectionist sequential machine," *Proceedings of the 8th Annual Conference of the Cognitive Science Society (CogSci)*, pp. 531–546, 1987.

[166] R. J. Williams and D. Zipser, "A learning algorithm for continually running fully recurrent neural networks," *Neural Computation*, vol. 1, no. 2, pp. 270–280, 1989.

[167] W. C. Hong, "Rainfall forecasting by technological machine learning models," *Applied Mathematics and Computation*, vol. 200, no. 1, pp. 41–57, 2008.

[168] S. Wang, L. Tang, and L. Yu, "SD-LSSVR-based decomposition-and-ensemble methodology with application to hydropower consumption forecasting," in *Proceedings of the 4th International Joint Conference on Computational Sciences and Optimization (CSO)*, pp. 603–607, Kunming, Yunnan China, April 2011.

[169] M. Djukanovic, M. Novicevic, D. J. Dobrijevic, D. BabicSobajic, and Yoh-Han Pao, "Neural-net based coordinated stabilizing control for the exciter and governor loops of low head hydropower plants," *IEEE Transactions on Energy Conversion*, vol. 10, no. 4, pp. 760–767, 1995.

[170] M. B. Djukanovic, M. S. Calovic, B. V. Vesovic, and D. J. Sobajic, "Neuro-fuzzy controller of low head power plants using adaptive-network based fuzzy inference system," *IEEE Transactions on Energy Conversion*, vol. 12.

[171] M. B. Djukanovic, D. M. DobrijevicDobrijevic, M. S. CalovicCalovic, M. NovicevicNovicevic, and D. J. Sobajic, "Coordinated stabilizing control for the exciter and governor loops using fuzzy set theory and neural nets," *International Journal of Electrical Power & Energy Systems*, vol. 19, no. 8, pp. 489–499, 1997.

[172] S. Deng and T. H. Yeh, "Using least squares support vector machines for the airframe structures manufacturing cost estimation," *International Journal of Production Economics*, vol. 131, no. 2, pp. 701–708, 2011.

[173] T.-H. Yeh and S. Deng, "Application of machine learning methods to cost estimation of product life cycle," *International Journal of Computer Integrated Manufacturing*, vol. 25, no. 4-5, pp. 340–352, 2012.

[174] J. Woodward and N. Gindy, "A Hyper-Heuristic Multi-Criteria Decision Sup809 Port System for Eco-Efficient Product Life Cycle," in *Proceedings of the 5th International Conference on Responsive Manufacturing - Green Manufacturing (ICRM 2010)*, Ningbo, China, January2010.

[175] S.-J. Wu, N. Gebraeel, and Y. Lawley, "A neural network integrated decision support system for condition-based optimal predictive maintenance policy," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 37, no. 2, pp. 226–236, 2007.

[176] B. Y. Choo, S. C. Adams, B. A. Weiss, J. A. Marvel, and P. A. Beling, "820 Adaptive multi-scale prognostics and health management for smart man821 ufacturing systems, International Journal of Prognostics and Health 822 Management (IJPHM) Special Issue," *PHM for Smart Manufacturing 823 Systems*, vol. 7, p. 014, 2016.

[177] A. Kumar, R. Shankar, A. Choudhary, and L. S. Thakur, "A big data MapReduce framework for fault diagnosis in cloud-based manufacturing," *International Journal of Production Research*, vol. 54, no. 23, pp. 7060–7073, 2016.

[178] X. Xu, "From cloud computing to cloud manufacturing," *Robotics and Computer-Integrated Manufacturing*, vol. 28, no. 1, pp. 75–86, 2012.

[179] C. E. S. da Silva, E. G. Salgado, C. H. P. Mello, E. da Silva Oliveira, and F. Leal, "Integration of computer simulation in design for manufacturing and assembly," *International Journal of Production Research*, vol. 52, no. 10, pp. 2851–2866, 2014.

[180] J. Wang, Y. Wang, and Y. Li, "A novel hybrid strategy using three-phase feature extraction and a weighted regularized extreme learning machine for multi-step ahead wind speed prediction," *Energies*, vol. 11, no. 2, p. 321, 2018.

[181] L. Martín, L. F. Zarzalejo, J. Polo, A. Navarro, R. Marchante, and M. Cony, "Prediction of global solar irradiance based on time series analysis: application to solar thermal power plants energy production planning," *Solar Energy*, vol. 84, no. 10, pp. 1772–1781, 2010.

[182] Y. Gala, Á. Fernández, J. Díaz, and J. R. Dorronsoro, "Hybrid machine learning forecasting of solar radiation values," *Neurocomputing*, vol. 176, pp. 48–59, 2016.

[183] C. Voyant, G. Notton, S. Kalogirou et al., "Machine learning methods for solar radiation forecasting: a review," *Renewable Energy*, vol. 105, pp. 569–582, 2017.

[184] T. G. Barbounis, J. B. Theocharis, M. C. Alexiadis, and P. S. Dokopoulos, "Long-term wind speed and power forecasting using local recurrent neural network models," *IEEE Transactions on Energy Conversion*, vol. 21, no. 1, pp. 273–284, 2006.

[185] M. Esmalifalak, L. Liu, N. Nguyen, R. Zheng, and Z. Han, "Detecting stealthy false data injection using machine learning in smart grid," *IEEE Systems Journal*, vol. 11, no. 3, pp. 1644–1652, 2017.

[186] V. Ford, A. Siraj, and W. Eberle, "Smart grid energy fraud detection using artificial neural networks," in *Proceedings of the 2014 IEEE Symposium on Computational Intelligence Applications in Smart Grid (CIASG)*, pp. 1–6, IEEE, Orlando, FL, USA, December2014.

[187] Z. Zheng, Y. Yang, X. Niu, H. N. Dai, and Y. Zhou, "Wide and deep convolutional neural networks for electricity-theft detection to secure smart grids," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 4, pp. 1606–1615, 2018.

[188] A. Ayad, H. E. Farag, A. Youssef, and E. F. El-Saadany, "Detection of false data injection attacks in smart grids using recurrent neural networks," in *Proceedings of the 2018 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, pp. 1–5, IEEE, Washington, DC, USA, February2018.

[189] M. A. Ferrag and L. D. C. Maglaras, "A novel deep learning and blockchain-based energy exchange framework for smart grids," *IEEE Transactions on Engineering Management*, 2019.

[190] D. B. Rawat and C. Bajracharya, "Detection of false data injection attacks in smart grid communication systems," *IEEE Signal Processing Letters*, vol. 22, no. 10, pp. 1652–1656, 2015.

[191] S. Chabaa, A. Zeroual, and J. Antari, "Identification and prediction of internet," *Journal of Intelligent Learning Systems and Applications*, vol. 2, no. 3, pp. 147–155.

[192] Y. Zhu, G. Zhang, and J. Qiu, "Network traffic prediction based on particle swarm bp neural network," *Journal of Networks*, vol. 8, no. 11, pp. 2685–2691, 2013.

[193] Y. Li, H. Liu, W. Yang, D. Hu, and W. Xu, "Inter-data-center Network Traffic Prediction with Elephant Flows," in *Proceedings of the NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium*, pp. 206–213, IEEE, Istanbul, Turkey, April 2016.

[194] Z. Chen, J. Wen, and Y. Geng, "Predicting future traffic using hidden Markov models," in *Proceddings of 24th IEEE International Conference on Network Protocols (ICNP)*, pp. 1–6, IEEE, 2016.

[195] P. Bermolen, M. Mellia, M. Meo, D. Rossi, and S. Valenti, "Abacus: accurate behavioral classification of P2P-tv traffic," *Computer Networks*, vol. 55, no. 6, pp. 1394–1411, 2011.

[196] R. Bellman, *Dynamic Programming*, Princeton University Press, Princeton, 1st edition, 1957.

[197] D. Schatzmann, W. Mühlbauer, T. Spyropoulos, and X. Dimitropoulos, "Digging into https: flow-based classification of webmail traffic," in *Proceedings of the 10th ACM SIGCOMM Conference on Internet Measurement*, pp. 322–327, Melbourne Australia, November 2010.

[198] J. Ma, K. Levchenko, C. Kreibich, S. Savage, and G. M. Voelker, "Unexpected means of protocol inference," in *Proceedings of the 6th ACM SIGCOMM Conference on Internet Measurement*, pp. 313–326, Rio de Janeriro Brazil, October 2006.

[199] P. Haffner, S. Sen, O. Spatscheck, and D. Wang, "Acas: automated construction of application signatures," in *Proceedings of the 2005 ACM SIGCOMM Workshop on Mining Network Data*, pp. 197–202, ACM, Philadelphia PA USA, August 2005.

[200] H. Jiang, A. W. Moore, Z. Ge, S. Jin, and J. Wang, "Lightweight application classification for network management," in *Proceedings of the 2007 SIGCOMM Workshop on Internet Network Management*, pp. 299–304, ACM, Kyoto Japan, August2007.

[201] A. W. Moore and D. Zuev, "Internet traffic classification using bayesian analysis techniques," in *ACM SIGMETRICS Performance Evaluation Review*vol. 33, pp. 50–60, ACM, 2005.

[202] J. Park, H. R. Tyan, and C. C. J. Kuo, "Internet traffic classification for scalable qos provision," in *Proceedings of the 2006 IEEE International Conference on Multimedia and Expo*, pp. 1221–1224, IEEE, Toronto, ON, Canada, July 2006.

[203] Y. Liu, W. Li, and Y. C. Li, *Network Traffic Classification Using K-Means Clustering*, IEEE, in *Proceedings of the Second International Multi-Symposiums on Computer and Computational Sciences (IMSCCS 2007)*, pp. 360–365, IEEE, Iowa City, IA, USA, August 2007.

[204] J. Erman, A. Mahanti, and M. Arlitt, "Internet traffic identification using machine learning," in *Proceedings of the Global Telecommunications Conference*, pp. 1–6, IEEE, San Francisco, CA, USA, November 2006.

[205] J. Erman, M. Arlitt, and A. Mahanti, "Traffic classification using clustering algorithms," in *Proceedings of the 2006 SIGCOMM Workshop on Mining Network Data*, pp. 281–286, ACM, Pisa Italy, September 2006.

[206] L. He, C. Xu, and Y. Luo, "vtc: machine learning based traffic classification as a virtual network function," in *Proceedings of the 2016 ACM International Workshop on Security in Software Defined Networks & Network Function Virtualization*, pp. 53–56, ACM, New Orleans Louisiana USA, March 2016.

[207] I. El Khayat, P. Geurts, and G. Leduc, "Machine-learnt versus analytical models of tcp throughput," *Computer Networks*, vol. 51, no. 10, pp. 2631–2644, 2007.

[208] M. Mirza, J. Sommers, P. Barford, and X. Zhu, "A machine learning approach to tcp throughput prediction," *IEEE/ACM Transactions on Networking*, vol. 18, no. 4, pp. 1026–1039, 2010.

[209] G. Quer, H. Meenakshisundaram, B. Tamma, B. S. Manoj, R. Rao, and M. Zorzi, "Cognitive Network Inference through Bayesian Network Analysis," in *Proceedings of the 2010 IEEE Global Telecommunications Conference GLOBECOM 2010*, pp. 1–6, Miami, FL, USA, December 2010.
[210] T. Arciszewski, M. Mustafa, and W. Ziarko, "A methodology of design knowledge acquisition for use in learning expert systems," *International Journal of Man-Machine Studies*, vol. 27, no. 1, pp. 23–32, 1987.
[211] J. Ghaboussi, J. H. Garrett Jr, and X. Wu, "Knowledge-based modeling of material behavior with neural networks," *Journal of Engineering Mechanics*, vol. 117, no. 1, pp. 132–153, 1991.
[212] S. McLaughlin and J. S. Gero, "Acquiring expert knowledge from characterized designs," *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, vol. 1, no. 2, pp. 73–87, 1987.
[213] Y. Reich, *Converging to Ideal Design Knowledge by Learning*, [Carnegie Mellon University], Engineering Design Research Center, Pittsburgh, PA USA, 1989.
[214] Y. Reich and S. J. Fenves, "Inductive learning of synthesis knowledge," *International Journal of Expert Systems Research and Applications*, vol. 5, p. 275, 1992.
[215] J. R. Stone, D. I. Blockley, and B. W. Pilsworth, "Towards machine learning from case histories," *Civil Engineering Systems*, vol. 6, no. 3, pp. 129–135, 1989.
[216] J. Zarka and J. M. Hablot, "Learning expert systems in numerical analysis of structures," in *Expert Systems in Structural Safety Assessment*, pp. 305–314, Springer, Berlin, Heidelberg, 1989.
[217] C. V. Gallagher, K. Bruton, K. Leahy, and D. T. O'Sullivan, "The suitability of machine learning to minimise uncertainty in the measurement and verification of energy savings," *Energy and Buildings*, vol. 158, pp. 647–655, 2018.
[218] P. Chatziagorakis, C. Ziogou, C. Elmasides et al., "Enhancement of hybrid renewable energy systems control with neural networks applied to weather forecasting: the case of Olvio," *Neural Computing & Applications*, vol. 27, no. 5, pp. 1093–1118, 2016.
[219] T. Boukelia, O. Arslan, and M. Mecibah, "Potential assessment of a parabolic trough solar thermal power plant considering hourly analysis: ANN-based approach," *Renewable Energy*, vol. 105, pp. 324–333, 2017.
[220] M. B. Anwar, M. S. El Moursi, and W. Xiao, "Novel power smoothing and generation scheduling strategies for a hybrid wind and marine current turbine system," *IEEE Transactions on Power Systems*, vol. 32, p. 1, 2016.
[221] F. Abbas, S. Habib, D. Feng, and Z. Yan, "Optimizing generation capacities incorporating renewable energy with storage systems using genetic algorithms," *Electronics*, vol. 7, no. 7, p. 100, 2018.
[222] B. A. Shimray, K. M. Singh, T. Khelchandra, and R. K. Mehta, "Ranking of sites for installation of hydropower plant using MLP neural network trained with ga: a madm approach," *Computational Intelligence and Neuroscience*, vol. 2017, pp. 1–8, 2017.
[223] H. Loutfi, A. Bernatchou, and R. Tadili, "Generation of horizontal hourly global solar radiation from exogenous variables using an artificial neural network in fes (Morocco)," *International Journal of Renewable Energy Resources*, vol. 7, pp. 1097–1107, 2017.
[224] H. A. Kazem and J. H. Yousif, "Comparison of prediction methods of photovoltaic power system production using a measured dataset," *Energy Conversion and Management*, vol. 148, pp. 1070–1081, 2017.
[225] F. Chahkoutahi and M. Khashei, "A seasonal direct optimal hybrid model of computational intelligence and soft computing techniques for electricity load forecasting," *Energy*, vol. 140, pp. 988–1004, 2017.
[226] A. Ahmad, T. Anderson, and T. Lie, "Hourly global solar irradiation forecasting for New Zealand," *Solar Energy*, vol. 122, pp. 1398–1408, 2015.
[227] T. Pinto, T. M. Sousa, I. Praça, Z. Vale, and H. Morais, "Support Vector Machines for decision support in electricity markets′ strategic bidding," *Neurocomputing*, vol. 172, pp. 438–445, 2016.
[228] S. Özdemir, M. Demirtaş, and S. Aydin, "Harmonic estimation based support vector machine for typical power systems," *Neural Network World*, vol. 26, no. 3, pp. 233–252, 2016.
[229] C. Arikan and M. Ozdemir, "Classification of power quality disturbances at power system frequency and out of power system frequency using support vector machines," *Przeglad Elektrotechniczny*, vol. 89, pp. 284–291, 2013.
[230] B. Doucoure, K. Agbossou, and A. Cardenas, "Time series prediction using artificial wavelet neural network and multiresolution analysis: application to wind speed data," *Renewable Energy*, vol. 92, pp. 202–211, 2016.
[231] Q. He, J. Wang, and H. Lu, "A hybrid system for short-term wind speed forecasting," *Applied Energy*, vol. 226, pp. 756–771, 2018.
[232] J. Sarshar, S. S. Moosapour, and M. Joorabian, "Multi-objective energy management of a micro-grid considering uncertainty in wind power forecasting," *Energy*, vol. 139, pp. 680–693, 2017.
[233] A. Hadi Abdulwahid and S. Wang, "A novel method of protection to prevent reverse power flow based on neuro-fuzzy networks for smart grid," *Sustainability*, vol. 10, no. 4, p. 1059, 2018.
[234] K. Kampouropoulos, F. Andrade, E. Sala, A. G. Espinosa, and L. Romeral, "Multiobjective optimization of multi-carrier energy system using a combination of ANFIS and genetic algorithms," *IEEE Transactions on Smart Grid*, vol. 9, no. 3, pp. 2276–2283, 2018.
[235] K. Mohammadi, S. Shamshirband, A. Kamsin, P. Lai, and Z. Mansor, "Identifying the most significant input parameters for predicting global solar radiation using an ANFIS selection procedure," *Renewable and Sustainable Energy Reviews*, vol. 63, pp. 423–434, 2016.
[236] S. Kamali and T. Amraee, "Blackout prediction in interconnected electric energy systems considering generation redispatch and energy curtailment," *Applied Energy*, vol. 187, pp. 50–61, 2017.
[237] S. O. Ottesen, A. Tomasgard, and S. E. Fleten, "Prosumer bidding and scheduling in electricity markets," *Energy*, vol. 94, pp. 828–843, 2016.
[238] E. Chemali, P. J. Kollmeyer, M. Preindl, and A. Emadi, "State-of-charge estimation of Li-ion batteries using deep neural networks: a machine learning approach," *Journal of Power Sources*, vol. 400, pp. 242–255, 2018.
[239] I. M. Coelho, V. N. Coelho, E. J. D. S. Luz, L. S. Ochi, F. G. Guimarães, and E. Rios, "A GPU deep learning metaheuristic based model for time series forecasting," *Applied Energy*, vol. 201, pp. 412–418, 2017.
[240] E. Mocanu, P. H. Nguyen, M. Gibescu, and W. L. Kling, "Deep learning for estimating building energy

consumption," *Sustainable Energy, Grids and Networks*, vol. 6, pp. 91–99, 2016.
[241] E. M. Burger and S. J. Moura, "Gated ensemble learning method for demand-side electricity load forecasting," *Energy and Buildings*, vol. 109, pp. 23–34, 2015.
[242] G. Fu, "Deep belief network based ensemble approach for cooling load forecasting of air-conditioning system," *Energy*, vol. 148, pp. 269–282, 2018.
[243] M. T. Ribeiro, S. Singh, and C. Guestrin, "'"Why should I trust you?' Explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1135–1144, San Francisco California USA, August. 2016.
[244] S. Ghosh, P. Lincoln, A. Tiwari, and X. Zhu, "'Trusted machine learning: model repair and data repair for probabilistic models," *Proceeding of the Workshops 31st AAAI Conf. Artif. Intell.*, 20.
[245] J. Quionero-Candela, M. Sugiyama, A. Schwaighofer, and N. D. Lawrence, *Dataset Shift in Machine Learning*, MIT Press, Cambridge, MA, USA, 2009.
[246] C. Dai, H.-S. Lim, E. Bertino, and Y.-S. Moon, "'Assessing the trustworthiness of location data based on provenance," in *Proceedings of the 17th SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pp. 276–285, Seattle Washington, November 2009.
[247] G. Trajcevski, O. Wolfson, K. Hinrichs, and S. Chamberlain, "Managing uncertainty in moving objects databases," *ACM Transactions on Database Systems*, vol. 29, no. 3, pp. 463–507, 2004.
[248] M. Zhu and Z. Jin, "'A trust measurement mechanism for service agents," " *in Proc. IEEE/WIC/ACM Int. Joint Conf. Web Intell. Intell. Agent Technol., Sep.*, pp. 375–382, 2009.
[249] Q. Su, C. R. Huang, and H. K. Y. Chen, "Evidentiality for text trustworthiness detection," in *Proceedings of the 2010 Workshop on NLP and Linguistics: Finding the Common Ground*, pp. 10–17, Uppsala Sweden, July 2010.
[250] H. Tao and Y. Chen, "'A metric model for trustworthiness of softwares," in *Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology*, pp. 69–72, Milan, Italy, September 2009.
[251] J. L. Berral, Ì. Goiri, R. Nou et al., "'Towards energy-aware scheduling in data centers using machine learning," in *Proceedings of the 1st International Conference on Energy-Efficient Computing and Networking*, pp. 215–224, E-Energy), Passau Germany, April 2010.
[252] X. Wang, Y. Gao, J. Lin, H. Rangwala, and R. Mittu, "'A machine learning approach to false alarm detection for critical arrhythmia alarms," in *Proceedings of the 2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA)*, pp. 202–207, Miami, FL, USA, December. 2015.
[253] L. M. Eerikainen, J. Vanschoren, M. J. Rooijakkers, R. Vullings, and R. M. Aarts, "'Decreasing the false alarm rate of arrhythmias in intensive care using a machine learning approach," in *Proceedings of the 2015 Computing in Cardiology Conference (CinC)*, pp. 293–296, Nice, France, September2015.
[254] L. Cao, "Data science," *ACM Computing Surveys*, vol. 50, no. 3, pp. 1–42, 2018.
[255] I. H. Sarker, Y. B. Abushark, F. Alsolami, and A. I. Khan, "Intrudtree: a machine learning based cyber security intrusion detection model," *Symmetry*, vol. 12, no. 5, p. 754, 2020.
[256] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of things for smart cities," *IEEE Internet of Things Journal*, vol. 1, no. 1, pp. 22–32, 2014.
[257] Y. Zheng, S. Rajasegarar, and C. Leckie, "Parking availability prediction for sensor-enabled car parks in smart cities," in *Proceedings of the 2015 IEEE Tenth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*, pp. 1–6, IEEE, Singapore, April 2015.
[258] J. Guerrero-Ibáñez, S. Zeadally, and J. Contreras-Castillo, "Sensor technologies for intelligent transportation systems," *Sensors*, vol. 18, no. 4, p. 1212, 2018.
[259] A. Boukerche and J. Wang, "Machine learning-based trafc prediction models for intelligent transportation systems," *Computer Networks*, p. 181, 2020.
[260] A. Essien, I. Petrounias, P. Sampaio, and S. Sampaio, "Improving urban trafc speed prediction using data source fusion and deep learning," in *Proceedings of the2019 IEEE International Conference on Big Data and Smart Computing (BigComp)*, vol. 1, February 2019.
[261] A. Essien, I. Petrounias, P. Sampaio, and S. Sampaio, "A deep-learning model for urban trafc fow prediction with trafc events mined from twitter," *World Wide Web*, vol. 24, 2020.
[262] M. Fatima and M. Pasha, "Survey of machine learning algorithms for disease diagnostic," *Journal of Intelligent Learning Systems and Applications*, vol. 9, no. 1, pp. 1–16, 2017.
[263] M. Nilashi, O. B. Ibrahim, H. Ahmadi, and L. Shahmoradi, "An analytical method for diseases prediction using machine learning techniques," *Computers & Chemical Engineering*, vol. 106, pp. 212–223, 2017.
[264] G. Silahtaroğlu and N. Yılmaztürk, "Data analysis in health and big data: a machine learning medical diagnosis model based on patients' complaints," *Communications in Statistics - Theory and Methods*, vol. 50, no. 7, pp. 1547–1556, 2019.
[265] W.H.O, "WHO," http://www.who.int/.
[266] S. Kushwaha, S. Bahl, A. K. Bagha et al., "Signifcant applications of machine learning for covid-19 pandemic," *J Ind Integr Manag*, vol. 5, no. 4, 2020.
[267] S. Lalmuanawma, J. Hussain, and L. Chhakchhuak, "Applications of machine learning and artifcial intelligence for covid-19 (sarscov-2) pandemic: a review," *Chaos, Solitons & Fractals*, p. 1, 2020.
[268] S. F. Ardabili, A. Mosavi, P. Ghamisi et al., "Covid-19 outbreak prediction with machine learning," *Algorithms*, vol. 13, no. 10, p. 2, 2020.
[269] M. Jamshidi, A. Lalbakhsh, J. Talla et al., "Artificial intelligence and COVID-19: deep learning approaches for diagnosis and treatment," *IEEE Access*, vol. 8, Article ID 109581, 2020.
[270] T. B. Alakus and I. Turkoglu, "Comparison of deep learning approaches to predict covid-19 infection," *Chaos, Solitons & Fractals*, p. 140, 2020.
[271] Y. Oh, S. Park, and J. C. Ye, "Deep learning covid-19 features on cxr using limited training data sets," *IEEE Transactions on Medical Imaging*, vol. 39, no. 8, pp. 2688–2700, 2020.
[272] C. Shorten, T. M. Khoshgoftaar, and B. Furht, "Deep learning applications for covid-19," *Journal of Big Data*, vol. 8, no. 1, pp. 18–25, 2021.
[273] A. Marchand and P. Marx, "Automated product recommendations with preference-based explanations," *Journal of Retailing*, vol. 96, no. 3, pp. 328–343, 2020.
[274] D. W. Otter, J. R. Medina, and J. K. Kalita, "A survey of the usages of deep learning for natural language processing,"

IEEE Transactions on Neural Networks and Learning Systems, vol. 32, no. 2, pp. 604–624, 2021.

[275] K. Ravi and V. Ravi, "A survey on opinion mining and sentiment analysis: tasks, approaches and applications," *Knowledge-Based Systems*, vol. 89, pp. 14–46, 2015.

[276] H. Fujiyoshi, T. Hirakawa, and T. Yamashita, "Deep learning-based image recognition for autonomous driving," *IATSS Research*, vol. 43, no. 4, pp. 244–252, 2019.

[277] C.-C. Chiu, T. N. Sainath, Y. Wu et al., "State-of-the-art speech recognition with sequence-to-sequence models," in *Proceedings of the 18 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4774–4778, IEEE, Calgary, AB, Canada, April 2018.

[278] R. Sharma, S. S. Kamble, A. Gunasekaran, V. Kumar, and A. Kumar, "A systematic literature review on machine learning applications for sustainable agriculture supply chain performance," *Computers & Operations Research*, vol. 119, p. 104926, 2020.

[279] N. Adnan, M. Nordin Shahrina, I. Rahman, and A. Noor, "The efects of knowledge transfer on farmers decision making toward sustainable agriculture practices," *World J Sci Technol Sustain Dev*, 2018.

[280] H. I. Cobuloglu and I. E. Büyüktahtakın, "A stochastic multi-criteria decision analysis for sustainable biomass crop selection," *Expert Systems with Applications*, vol. 42, no. 15-16, pp. 6065–6074, 2015.

[281] S. S. Kamble, A. Gunasekaran, and S. A. Gawankar, "Sustainable industry 4.0 framework: a systematic literature review identifying the current trends and future perspectives," *Process Safety and Environmental Protection*, vol. 117, pp. 408–425, 2018.

[282] S. S. Kamble, A. Gunasekaran, and S. A. Gawankar, "Achieving sustainable performance in a data-driven agriculture supply chain: a review for research and applications," *International Journal of Production Economics*, vol. 219, pp. 179–194, 2020.

[283] A. K. Dey, "Understanding and using context," *Personal and Ubiquitous Computing*, vol. 5, no. 1, pp. 4–7, 2001.

[284] I. H. Sarker, "Context-aware rule learning from smartphone data: survey, challenges and future directions," *Journal of Big Data*, vol. 6, no. 1, pp. 95–25, 2019.

[285] H. Zhu, H. Cao, E. Chen, H. Xiong, and J. Tian, "Exploiting enriched contextual information for mobile app classifcation," in *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, pp. 1617–1621, ACM, Maui Hawaii USA, October 2012.

[286] I. H. Sarker and K. Salah, "Appspred: predicting context-aware smartphone apps using random forest learning," *Internet Things*, vol. 8, 2019.

[287] H. Zhu, E. Chen, H. Xiong, K. Yu, H. Cao, and J. Tian, "Mining mobile user preferences for personalized context-aware recommendation," *ACM Transactions on Intelligent Systems and Technology*, vol. 5, no. 4, pp. 1–27, 2015.

[288] S. Zulkernain, P. Madiraju, S. Ahamed, and K. Stamm, "A mobile intelligent interruption management system," *J UCS*, vol. 16, no. 15, pp. 2060–2080, 2010.

[289] I. H. Sarker and A. S. M. Kayes, "ABC-RuleMiner: user behavioral rule-based machine learning method for context-aware intelligent services," *Journal of Network and Computer Applications*, vol. 168, Article ID 102762, 2020.

[290] I. H. Sarker, A. Colman, M. A. Kabir, and J. Han, "Individualized time-series segmentation for mining mobile phone user behavior," *The Computer Journal*, vol. 61, no. 3, pp. 349–368, 2018.

[291] S. Zulkernain, P. Madiraju, and S. I. Ahamed, "A context aware interruption management system for mobile devices," in *Mobile Wireless Middleware, Operating Systems, and Applications*, pp. 221–234, Springer, Berlin, Germany, 2010.

[292] K. S. Perera, Z. Aung, and W. L. Woon, "Machine learning techniques for supporting renewable energy generation and integration: a survey," in *Proceedings of the International Workshop on Data Analytics for Renewable Energy Integration*, Springer, Berlin, Germany, pp. 81–96, September 2014.

[293] T. S. Bomfim, "Evolution of machine learning in smart grids," in *Proceedings of the 2020 IEEE 8th International Conference on Smart Energy Grid Engineering (SEGE)*, pp. 82–87, IEEE, Oshawa, ON, Canada, August 2020.

[294] P. K. R. Maddikunta, Q. V. Pham, B. B et al., "Industry 5.0: a survey on enabling technologies and potential applications," *Journal of Industrial Information Integration*, vol. 26, Article ID 100257, 2022.

sensors

Article

BENS−B5G: Blockchain-Enabled Network Slicing in 5G and Beyond-5G (B5G) Networks

Saurabh Singh [1], C. Rajesh Babu [2], Kadiyala Ramana [3], In-Ho Ra [4] and Byungun Yoon [1,*]

[1] Department of Industrial and System Engineering, Dongguk University, Seoul 04620, Korea
[2] Department of Networking and Communications, SRM Institute of Science and Technology, Chennai 603203, India
[3] Department of IT, Chaitanya Bharathi Institute Technology, Hyderabad 500075, India
[4] School of Computer, Information and Communication Engineering, Kunsan National University, Gunsan 54150, Korea
* Correspondence: postman3@dongguk.edu

Abstract: Fifth-generation (5G) technology is anticipated to allow a slew of novel applications across a variety of industries. The wireless communication of the 5G and Beyond-5G (B5G) networks will accommodate a wide variety of services and user expectations, including intense end-user connectivity, sub-1 ms delay, and a transmission rate of 100 Gbps. Network slicing is envisioned as an appropriate technique that can meet these disparate requirements. The intrinsic qualities of a blockchain, which has lately acquired prominence, mean that it is critical for the 5G network and B5G networks. In particular, the incorporation of blockchain technology into B5G enables the network to effectively monitor and control resource utilization and sharing. Using blockchain technology, a network-slicing architecture referred to as the Blockchain Consensus Framework is introduced that allows resource providers to dynamically contract resources, especially the radio access network (RAN) schedule, to guarantee that their end-to-end services are effortlessly executed. The core of our methodology is comprehensive service procurement, which offers the fine-grained adaptive allocation of resources through a blockchain-based consensus mechanism. Our objective is to have Primary User—Secondary User (PU—SU) interactions with a variety of services, while minimizing the operation and maintenance costs of the 5G service providers. A Blockchain-Enabled Network Slicing Model (BENS), which is a learning-based algorithm, is incorporated to handle the spectrum resource allocation in a sophisticate manner. The performance and inferences of the proposed work are analyzed in detail.

Keywords: blockchain; network slicing; 5G communications; radio access network; 5G-CORE network functions; beyond-5G

check for updates

Citation: Singh, S.; Babu, C.R.; Ramana, K.; Ra, I.-H.; Yoon, B. BENS−B5G: Blockchain-Enabled Network Slicing in 5G and Beyond-5G (B5G) Networks. *Sensors* **2022**, 22, 6068. https://doi.org/10.3390/s22166068

Academic Editor: Nikos Fotiou

Received: 11 July 2022
Accepted: 11 August 2022
Published: 14 August 2022

1. Introduction

As a result of the Internet of Things era, new time- and mission-critical applications that incorporate 5G or B5G have been created for every sector of human activity. These end-to-end applications are organized using a series of network services [1]. As a result, infrastructure operators must bring computational capabilities closer to end-users to meet the delay requirements of cloud computing. Wireless data traffic will surge in the coming years as the number of mobile users and the wide range of bandwidth-hungry apps they use increase dramatically. Next-generation (5G) and 5G wireless communication networks will support a broader communication ecosystem, including the Internet of Things (IoT) and Internet of Vehicles (IoV) [2]. The 5G and beyond 5G wireless communications are expected to constitute the foundation for several new applications to enable this progress. Customer's needs for applications have a wide range of complexity and customers will want to be met with 5G and beyond wireless communications. Some vertical sectors, such as industrial automation control systems and the Internet of Vehicles, need an exceptionally high reliability and low latency communications to meet rigorous QoS requirements [3]. To meet the diverse and personalized QoS needs of 5G and beyond networks, it is necessary to re-examine networking technology and network design. The International Telecommunication Union (ITU) has categorized three sorts of users in terms of service classification: enhanced mobile broadband (eMBB), ultra-reliable low-latency communications (URLLC), and massive machine-type communications (mMTC), which are all examples of new technologies that are being developed to meet the needs of today's mobile devices [3]. Enterprises are searching for creative solutions to satisfy their demands and address new prospects as a result of the arrival of new technologies brought about by 5G, as well as the new business chances that have been created across all sectors. When it comes to enterprise customers, they demand automated business and operational processes from the time they buy the service through activation, delivery, and decommissioning. They anticipate that services will be provided more quickly while maintaining a high level of safety. Through the use of network slicing, communication service providers can fulfil all of the requirements posed by their corporate clients.

The 5G network design, based on network slicing is expected to play a significant role in the future generation of networks. The virtual network is described as a slice in network slicing, which allows numerous independent and separated virtual networks to coexist in the same physical network infrastructure. Using Software Defined Networks (SDN) and Network Function Virtualization (NFV), networks may be sliced to accommodate new services with a broad range of needs, while still using the same physical network (PN). Slices are created by abstracting control logic and resources from the SDN controller and making them available to the SDN nodes. SDN network slicing also makes it possible for several tenants to share the same PN resources. On the other hand, NFV was created to address a lack of particular communication equipment in the market. Virtual network functions are at the heart of NFV, and they may be performed on standard servers without the need for specialized hardware. Due to these attractive benefits, network slicing is often used [4]. Multi-tenancy is supported via network slicing, which allows the same physical infrastructure to be used by many virtual network operators. Network slicing may be used to provide differentiated service and meet service level agreements. Network slicing facilitates the capacity to build and alter network slices on demand, which increases the adaptability and flexibility of network administration. Figure 1 shows a network situation using network slicing. To simplify management and coordination, network slices are used to represent the underlying physical infrastructure. Core networks (CNs), radio access networks (RANs), and other physical resources are separated into many logical components, resulting in various network slices that may be tailored to meet the needs of different users [5]. Network slicing has numerous slices for diverse services and sits above the underlying layer of the network stack. It is widely regarded as one of the most promising technologies for 5G and beyond networks because of its ability to provide a wide range of QoS requirements for diverse services.

By splitting the same PN into many isolated logical networks, RAN slicing may deliver tailored services for isolated logical networks. This cost-effective and high-efficiency network management approach is built on the concept of PN sharing. Ref. [6] has found that RAN sharing might save the global economy about 60 billion dollars by 2022 in terms of both capital and operating expenses. The 3GPP has conducted an extensive study of 5G network slicing in practice.

To meet diverse QoS requirements, it is necessary to find a method for allocating network resources that is both flexible and efficient, and this is where RAN slicing comes into play. RAN slicing allows network operators to effectively and flexibly distribute resources based on the performance needs of individual users. In addition to wireless resources, these resources include those for computing and cache storage. The resources may be used more efficiently and at a higher rate with the help of resource-allocation technology, which can also merge diverse slices. Network slicing may be categorized as either a static or dynamic allocation, depending on the circumstance. After deciding on a resource allocation and mapping approach for network slicing, the allocation will remain static, no matter what changes occur in the environment. A key characteristic of dynamic resource management is the capacity to adapt resource-allocation tactics in response to changes in the environment, ultimately resulting in improved communication service quality.

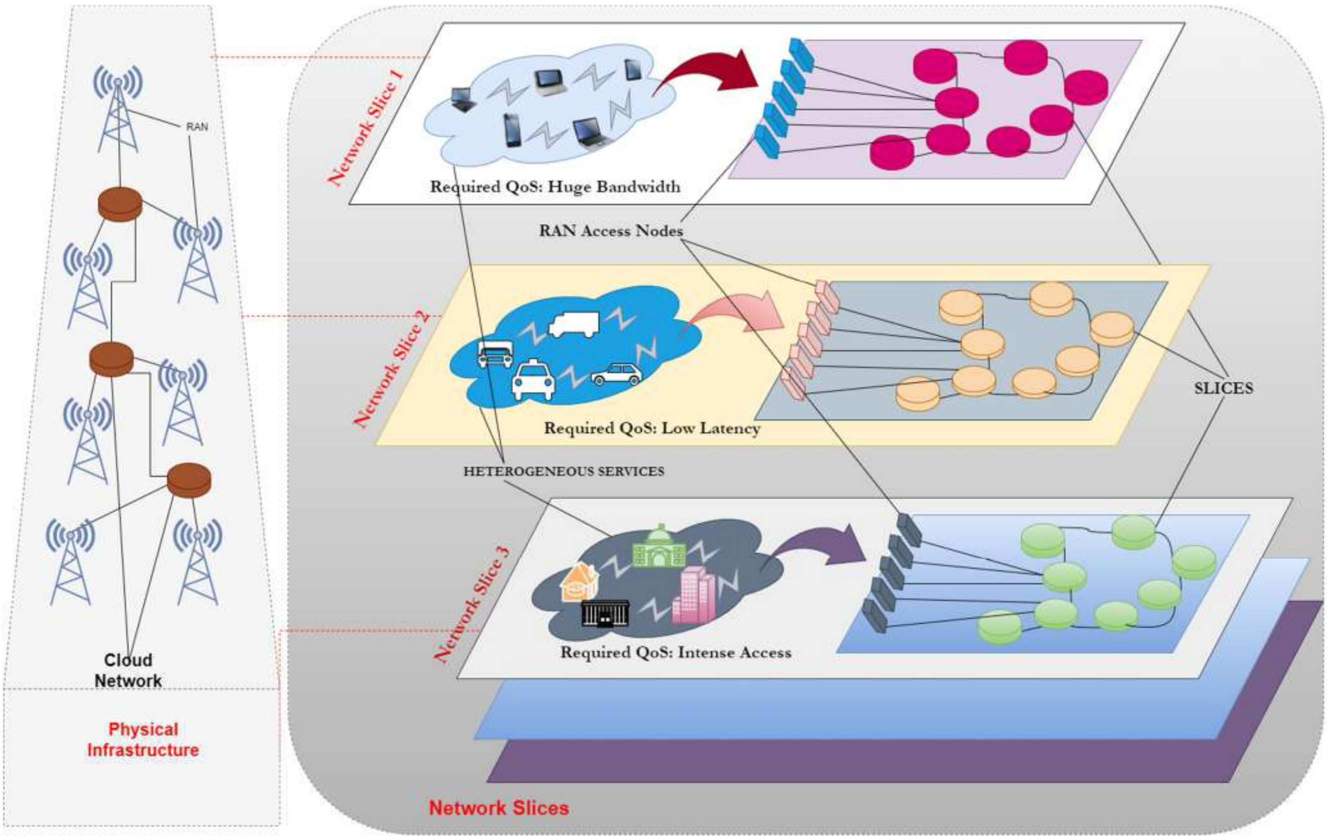


Figure 1. Network slicing scenario.

It is critical to construct cross-domain RAN slicing that incorporates different operators and infrastructure suppliers. For example, an autonomous driving module requires a RAN slice that can cover a large geographical region by using services from a variety of local operators located across the city [7]. On the other hand, traditional cross-domain orchestrators are built on a master–slave design, which has a number of problems. A self-interested master virtual mobile network operator (VMNO) can capture super profits over other players, preventing them from entering the system, since they are responsible for collecting occupants' slice requests, resource allocations, and incentive distributions. Second, the master VMNO must negotiate the cost of access to all the tenants' resources on a frequently time-intensive and wasteful basis. To alleviate incumbents' concerns about the 'Master', both research and commerce have recently laid great emphasis on the blockchain, a distributed immutable data recorder that is capable of establishing trust between untrusted peers. Additionally, smart contracts, which are contracts that have been encoded and managed by computers, are optionally utilized by the blockchain without the approval of the authority. With smart contracts, blockchain technology enables the management systems to easily handle the complicated operations.

In this paper, we present a *tabula rasa* model referred to as BeNS−B5G, a RAN slicing framework based on blockchain, which improves fairness and orchestration efficiency, and we use reinforcement learning methods to further improve its performance. In contrast to conventional designs, the suggested one is distributed, with several cloud service providers (CSPs), virtual machine network operators (VMNOs), and infrastructure providers (InPs) co-managing a blockchain system. Rather than merely listening to a committed master, all incumbents fight for leadership. Without the need for human discussions, smart contracts running on the blockchain automatically record resource statuses, process RAN slice requests, and pay incentives to resource providers. When the tenants release the slice requests, an intelligent resource optimizer (RO) is included to assist VMNOs, in determining the optimal resource combination that meets the requested QoS criteria of RAN slice.

The rest of this article is organized in the following manner. We begin with a comprehensive analysis of RAN slicing and the blockchain method, followed by a discussion of the obstacles associated with implementing the blockchain-enabled network slicing framework. We propose a RAN slicing architecture that leverages the distributed and automated nature of blockchain technology and incorporates a novel consensus mechanism to assure system performance, while aligning with incumbents' economic objectives. Following that, we further reduce slice requesters' costs by using Federated Learning (FL) methods to determine the ideal resource combination. The learning algorithm's benefits are assessed via simulations, and associated future approaches are considered as well as outstanding concerns.

2. Related Work

The exhaustive related work on cognitive radio networks has been studied based on the following categories:

- Cognitive Radio over 5G and Beyond Networks;
- Network Management in 5G and Beyond Networks.

2.1. Cognitive Radio over 5G and beyond Networks

Abubakar Makarf et al. (2020) [8] explored combining the Radio Information System (RIS) and Monte Carlo concept within a network to maximize the potential benefits. Two different RIS-based network models were investigated, and many performance measures connected with the CR secondary user were implemented. The obtained equations were validated using Monte Carlo simulations. In the presence of an RIS-enhanced main network, the results showed the influence of key system parameters and a clear improvement in the CR network. According to Zhaoyuan Shi et al., the massive MIMO which is, underlying the cognitive radio user selection schema, is aware of the QoS requirements of the channel requested (2019) [9]. There are two major ways in which a CR may be implemented: The Channel State Indicator (CSI) of any cross-network is inaccessible at the secondary base station (SBS), but the SBS has access to the CSI of the cross-channel channel state. Low-complexity algorithms for increasing users while using the least amount of power (IUMP) and methods for decreasing users while using the most amount of power (DUMP) were developed to solve user selection via power allocation. The intractable challenge was addressed using a deep reinforcement-learning-based method, enabling the SBS to accomplish effective and intelligent user selection. In simulations, these algorithms dramatically outperform the current user selection approaches. Our neural network was able to rapidly learn the best user selection strategy in an unknown dynamic environment with a high success rate and fast convergence, as the findings also revealed. Kok-Lim Alvin Yau et al. [10] focused on how CR and the Cognition Cycle have been integrated into 5G to deliver spectrum efficiency, energy efficiency, enhanced quality of service and experience, and cost-efficiency (2018). Open research opportunities and platform implementation were made accessible to inspire new research in this area. Gianfranco Nencioni et al. (2018) [11] discussed the fifth-generation (5G) of cellular networks, which is expected to represent a significant advancement in wireless technology. A variety of new wireless technologies will be implemented to better serve 5G's wide set of requirements, including upgrades to the radio access network. It was demonstrated that the convergence of many communication technologies has been facilitated by embedding softwarizations such as Software-Defined Networking (SDN) and Network Functions Virtualization (NFV). Through network slicing, 5G networks may be constructed at low cost. By using an SDN/NFV architecture, 5G radio access and core networks will be able to deliver network services more efficiently, flexibly, and in a more scalable manner. The authors also discussed software-defined 5G radio access and core networks, as well as a wide range of future research topics in orchestration and control. Johana Hernández et al. proposed cognitive radio management (2018) [12].

2.2. Network Management in 5G and beyond Networks

The most available route for the opportunistic transmission of secondary user data is chosen throughout the decision-making process as a result of the main user characterization-model's efficiency. It was claimed that an approach based on deep learning and long short-term memory might lessen the forecasting error now present in future significant user estimates in the Global System for Mobile Communication (GSM) and WiFi frequency bands. When compared to alternative approaches, such as multi-layer perceptron neural networks, Bayesian networks, and adaptive neuro-fuzzy inference systems, the results indicate that a lengthy short-term memory may significantly enhance the estimates of channel utilization (ANFIS−Grid). The neural structure has input, forget, and output gates, which complicates its implementation in cognitive radio networks (CRNs) based on core network topologies, despite the fact that a long short-term memory fared better at time series forecasting. Aaron Yi Ding et al. developed the criteria for assessing the restrictions of 5G-driven applications (2018) [13]. The usual hurdles and needs for various application domains were studied using 5G networks as a basis. The major objective was to create a network architecture that could adapt to changing traffic patterns, while also supporting diverse technologies, such as edge computing, blockchain-based distributed ledgers, software-defined networking, and virtualization. We underlined the need to perform 5G application pilots to better understand how 5G networks are deployed and utilized in different vertical industries. Xingjian Li et al. investigated the issue of spectrum sharing in a cognitive radio system with a main and secondary user (2018) [14]. Secondary users are at odds with prime users. In particular, it was assumed that the primary user would alter its transmitted power in accordance with a pre-defined power management strategy. The secondary user has no idea what the main user's transmission power and power control method are. For the secondary user, a power control system based on learning was developed to share the same spectrum with the main. To assist the secondary user, a network of sensor nodes was strategically placed across the wireless network to collect data on the received signal strength. The secondary user's transmission power may be automatically adjusted using a deep reinforcement learning algorithm. This may be performed after a few rounds of interaction with the principal user. The results showed that secondary users might effectively connect with main users to achieve the desired state from any beginning situation in a few steps. When Maria Massaro et al. (2017) [15] examined the Licensed Shared Access (LSA) and Shared Access Spectrum (SAS) regimes in the EU, they discovered significant disparities between the LSA and SAS regimes. To acquire information on the technical and regulatory components of current and forthcoming spectrum-sharing regimes, policy documents, research publications, position papers, and analytical studies were studied. The LSA regime is notable for providing mobile operators with the regulatory certainty they need to invest in 5G, while also granting them access to an additional spectrum below 6 GHz. Other spectrum-sharing regimes will not safeguard cell operators from hazardous interference or ensure trustworthy Quality of Service (QoS) while utilizing sub-6 GHz airwaves. Due to its lower level of technical complexity, the LSA regime can be deployed more rapidly and with less effort when the two techniques are compared. However, as technology progresses, the LSA regime is projected to be surpassed in the long run by the SAS regime. More persons may share the same frequency channels under the SAS setup. A cognitive WLAN overlay over an OFDMA TDD main network was tested for saturation throughput by Parisa Rahimzadeh et al. (2017) [16], e.g., using LTE or WiMAX. The successful node delivers its data packet in the main network's downlink and uplink subframes that have empty resource blocks (RBs). Unlike the OFDMA structure and time-scheduled resources in the primary network, the opportunity length in the secondary network does not follow a straightforward exponential on–off pattern. There is a mathematical model for the dynamic behavior of secondary node opportunities and contentions that incorporates a discrete-time Markov chain and two connected open multi-class queueing networks (QNs). As a random number is created when data are downloaded and uploaded, our research includes the random packet transmission time on WLANs, the dependency on the number of empty RBs in the subsequent frames, and aspects of the 802.11 MAC protocol. Multiple resource allotments were inserted into the main network for the purpose of conducting the analysis. We were able to demonstrate the correctness of our method via simulations in a variety of different situations. Cheng Wu et al. (2016) [17] developed a multi-agent reinforcement learning-based spectrum management approach. For efficient spectrum and transmit power allocation, the approach employed value functions to evaluate the advantages of different transmission features, maximizing long-term return. Using a variety of learning factors, students were exposed to a variety of real-world circumstances, and their communication skills were evaluated. A Kanerva-based function approximation was utilized to enhance the management of large CRNs, and to examine the impact on communication performance in these networks. The proposed reinforcement learning-based spectrum management in a cognitive radio ad hoc network is shown to considerably minimize interference to licensed users, while preserving a high probability of successful transmissions in this network. The secondary users' average sum rate was increased by Yang Yang et al. (2017) [18] by determining the best way to access and regulate their power in multiple bands (ASR). We represented the random distributions of PUs and SUs using Poisson point processes (PPPs) based on stochastic geometry, from which we were able to compute the closed-form outage probability and estimate the ASR of SUs. On a number of bands, the ASR maximization problem included an outage probability. The optimal density of SUs with a given power was determined using closed-form convex optimization, and the optimal SU power was calculated and ASR convexity was checked. These findings prompted the creation of a spectrum access and power management strategy aimed at optimizing the ASR of SUs over several bands. The results of the simulations reveal that PUs and network interference limit SUs' density and power, and that the proposed approach can achieve the maximum ASR for the SUs.

With the exhaustive survey on network management in 5G and beyond networks, it is becoming ever clearer that resource allocation becomes a tedious process with the dynamics of 5g and Beyond 5G networks. Hence, to focus on an efficient and reliable resource allocation technique, we incorporate a state-of-the-art blockchain-enabled network-slicing method with 5G and beyond networks.

3. Blockchain-Enabled Network Slicing in 5G

In this section, we develop a blockchain-enabled RAN slicing architecture that encompasses all stakeholders in the 5G and beyond network slicing scenario, ensuring invisible automation, transparency, and optimal interactions between them. The suggested design is consistent with the smart contract as a management platform paradigm, in which all services are supplied by contracts and are invoked by users through transactions. InPs, VMNOs, MNOs, and CSPs are all incumbent members of the RAN slicing architecture. All of these players are agents in the FL and blockchain-enabled RAN slicing, and all users except occupants are users [19]. The following criteria are used to choose agents. On the one hand, agents should have robust capabilities for storing and validating blockchain data, as well as the ability to remain online at all times. As a result, occupants would be less qualified for the task.

3.1. Spectrum Sharing and Slicing in 5G

We are proposing a dynamic spectrum sharing and slicing system using broad sensing to overcome this limitation of sharing most of the unused spectrum bands to gain additional spectrum resources for 5G.

Terminals and fixed sensors track the spectrum of use of a primary user in this system. Another system's transmission spectrum or power must be dynamically identified from enormous tracking data when spectrum sharing is necessary. During spectrum sharing, the secondary user must determine whether or not they are interfering with the PU. Interference between the PU and SU is avoided by changing the transmission spectrum, power, or linked network. Regardless of whether the principal user is migrating or not, this technique may be used in a variety of systems. A paradigm for dynamic spectrum sharing in the 5G−CORE network is also proposed in this paper. We concentrate on the scenario in which the base station is responsible for implementing the dynamic spectrum-sharing features. As an addition to the DSA, we offer an implementation strategy for dynamic spectrum sharing.

Figure 2 shows the dynamic spectrum-sharing mechanism. Dynamic spectral monitoring, the sharing of existing spectral bands, and the coordination of the utilization of unused spectra comprise the system. Terminals and fixed sensors monitor the primary user's spectrum utilization via dynamic spectrum monitoring. A spectrum sharing condition, such as the availability of a spectrum band or transmission power, is dynamically defined in the current spectral bands for sharing. During spectrum sharing, interference between primary and secondary users might alter the sharing conditions.

3.2. Bands That Share a Spectrum Range

The received powers of primary users with several terminals and fixed sensors are calculated using these functions. With terminals and mobile-dispersed monitoring, a watch can be kept on a large region. On the other hand, fixed sensors in shared spectrum bands can keep an eye on a wider spectral range. The base station receives a summary of the monitoring data. However, when the monitoring data is delivered, the system's overhead would be enormous. Data collection through street-pass communication between terminals may be explored to reduce the quantity of transferred data. To determine how well the sharable spectrum is being utilized, and to avoid interference, data from the base station are collected and analyzed.

3.3. Dynamic Environmental Monitoring

This function determines whether large numbers of data should be shared. Fast and dynamic data analysis using edge computing and distributed databases are used to determine the current state of affairs. The interpolation and inference of frequent utilization situations are used to determine the sharing condition [20]. The secondary base station and secondary terminal interact according to the set conditions.

3.4. Utilization of Shared Spectrum Bands

PU—SU interference may arise while they switch locations throughout normal usage. To avoid interference, the secondary user's transmission spectrum or power should be modified without affecting the PU. The PU—SU communication regions are estimated using the data gathered through monitoring. Autonomous radio coordination or autonomous network coordination is carried out based on the estimated regions [21]. The power or spectrum of the transmission in the same band may be changed automatically through radio coordination. If the secondary user is still interfering with the primary user after radio coordination, the network connection for the secondary user is replaced. Figure 2 shows the dynamic environmental changes of the PU and SU.

3.5. Dynamic Spectrum-Sharing Specification for the 5G−CORE Network

Figure 3 presents a framework for dynamic spectrum sharing in the 5G−CORE network, where three application situations in terms of the placement of the component functions are explored. Configuration the base station, on top of an AMF, and outside the 5G−CORE network are the first two possibilities.

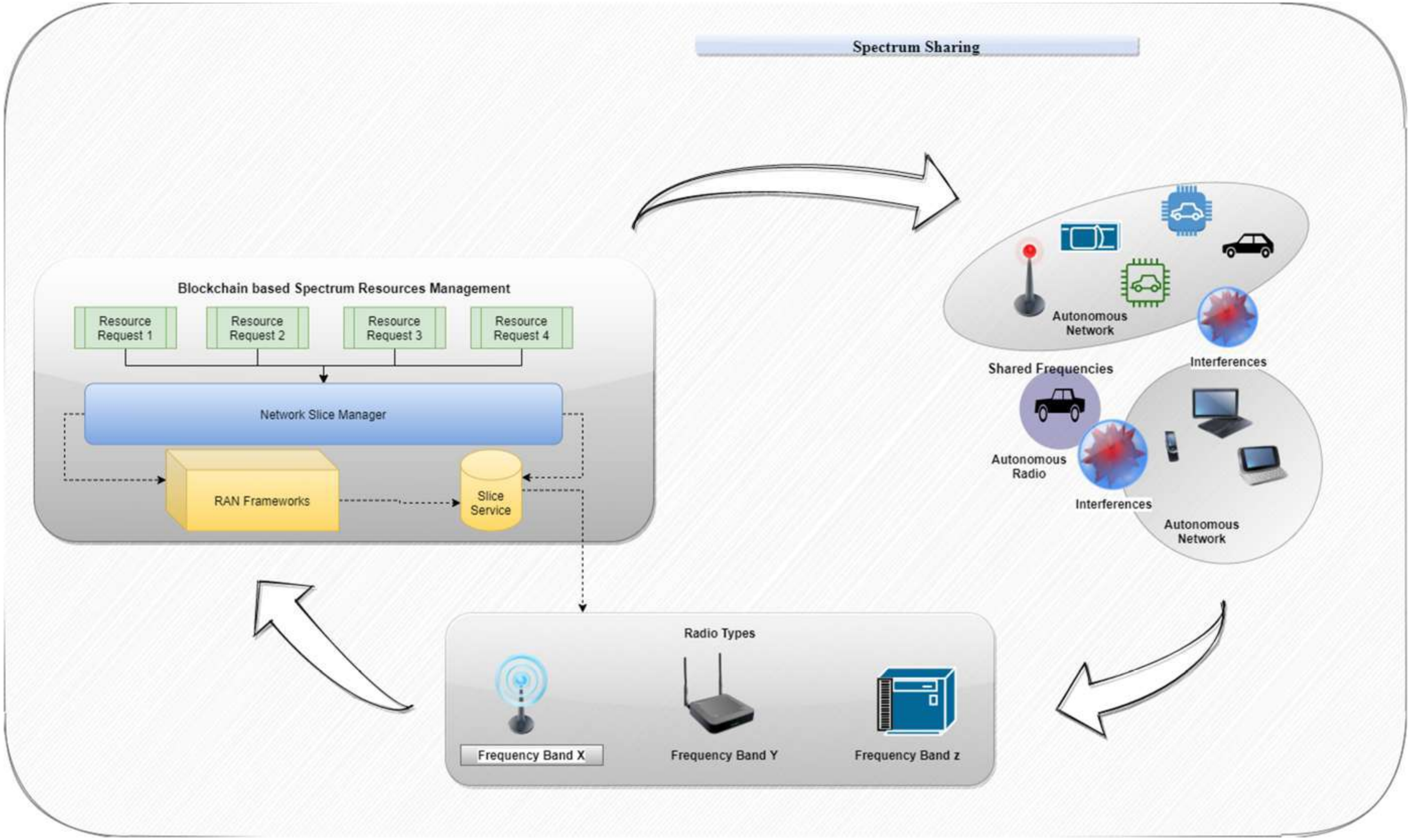


Figure 2. Dynamic spectrum-sharing environment.

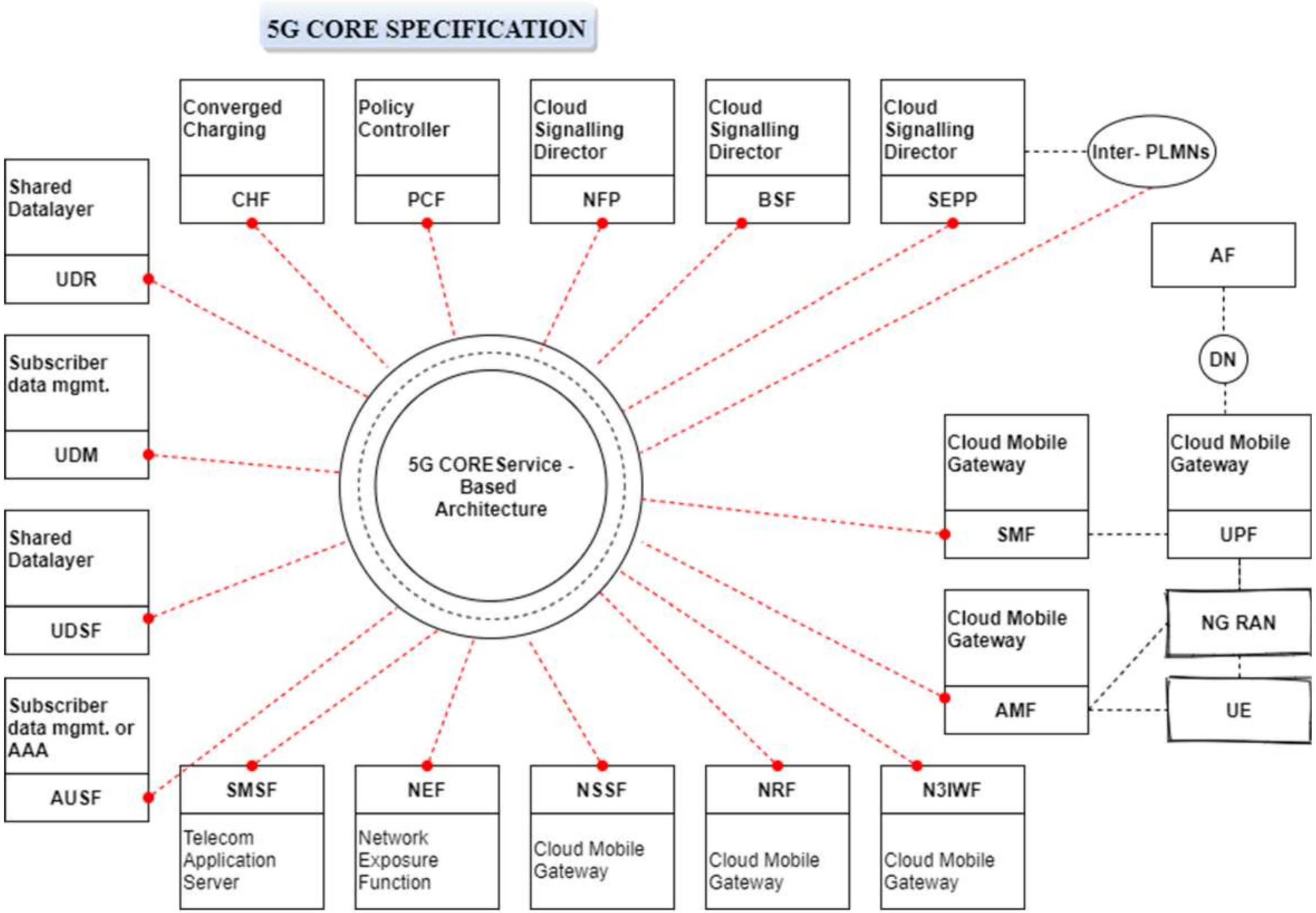


Figure 3. 5G−CORE specification for dynamic spectrum sharing.

3.5.1. Scenario 1—Configuration in Radio Access Network

The Radio Access Network regulates the user equipment linked to the network. The RAN may use the sharable spectrum, and coordinate when interference occurs without a burden on the main network. The RAN cannot identify the interference until the primary user reaches the edge of the secondary user's territory, since the RAN has information about its own area. This indicates the PU—SU interference. Therefore, the design becomes successful in communication systems where the primary user does not relocate.

3.5.2. Scenario 2—Configuration of Access and Mobility Management Function

Since the core network may gather data from several RANs, it is possible to cover a large region by using multiple RANs. The core network is far more sensitive to the presence of the primary user than the RAN. As a result, in systems where the principal user travels, this use case is effective. There is a delay in detecting and coordinating interference, because the core network must gather data from many RANs.

The following section discusses the configuration of the RAN and a learning-based channel allocation mechanism for the specified model.

3.6. Configuration of Radio Access Network

The 5G cellular networks are divided into four sub-segments using a Multi-Layer Stratified Networking paradigm that incorporates Machine Learning and Cognitive Radio technology. We use a three-level MAS paradigm for the core network, secondary users, and primary users to create a complex Multi-Agent System that incorporates AI and Cognitive Radio technologies. In the age of AI, robots are able to see and learn from their surroundings in the same way as humans. Furthermore, in an Artificial Intelligence (AI) system, people may achieve great abilities via clustering algorithms. Meanwhile, a plethora of AI solutions have opened the way for optimizing network connectivity and allocating resources. Furthermore, 5G cellular networks may benefit from the merging of AI and CRNs, providing the network with the same intelligence and autonomy as a human. The AI and machine learning methods are presented in such a way that how far it is important to incorporate a learning-based system for the CRNs [22]. Reinforcement Learning can be collaborated with the CRNs for channel sensing and allocation and dynamic routing. With the help of RL, CRNs' primary and secondary users may more efficiently use their local operating environments' spectrum resources by using online learning behavior and adopting the best actions possible.

Our architecture integrates AI technologies with channel resource management and Base Stations' resources management in 5G communications to ensure the QoS needs of CRN users, maximize spectrum utilization, and optimize the BSs resources control approach. While the reinforcement learning algorithm performs well in CRNs, learning strategies also rely on the network's system features, which include solitary, multi-agent, hierarchical, and distributed networks. We have utilized multi-agent reinforcement learning for cooperative power distribution in CRNs. It is also claimed that CRNs use a reinforcement learning system to estimate the throughput and discover accessible idle channels. In addition, the distributed optimization technology of a heterogeneous small cell network is given. Cognitive Radio consumers may expect improved resource efficiency and QoS assurances with AI-based hierarchical and distributed network technologies.

4. Learning Based Slicing (Channel Resource Allocation) for the RAN in 5G-CORE

We examine the channel resource allocation technique for a Multi-Agent System with several primary users and secondary users. An agent is a CR user, and the environment considers all the PU—SUs over the 5G network. Each agent acquires information and makes decisions in response to an environmental input and spectrum resources are allocated in a dynamic manner to maximize each agent's advantage. The Multi-Agent Reinforcement Learning method is seen in this light as addressing a decision problem in the Multi-Agent System. Additionally, we must investigate agent behavior, and provide an appropriate technique for Multi-Agent Reinforcement Learning [23]. From this vantage point, interventions need three components: an actor, a context, and rules. Each of these components is discussed in depth in the following sections. In the suggested paradigm, agents are basic, consisting of the PU and SU. Each agent is a self-contained entity capable of sensing, observation, learning, and decision making. These agents behave according to well-defined rules, while interacting with the environment of a CRN. We abstract a model of a Multi-Agent System that incorporates an intelligent Base Station control mechanism and dynamic spectrum allocation. Every agent decides on an action by observing, learning, and deriving a perceptive state from the environment

4.1. Environment

A partially observable Markov decision-making process can be used to define the environment. In the Multi-Agent System model, while an agent makes a decision, the rest of the agents become idle, and a Markov decision process is used to define the environment. While a PU-agent and environment interaction takes place, CRNs consider all other agents, including the PU and SU, to be the environmental components.

4.2. Rules

Rules are critical in Multi-Agent Reinforcement Learning. This article discusses two distinct kinds of agent rules: isomorphic or heterogeneous. Primary User-to-Primary User and Secondary User-to-Secondary User policies are isomorphic, while Primary User-to-Secondary User policies are heterogeneous agent rules. PUs–SUs interactions provide incentives and rules.

In the Multi-Agent System, Primary User-to-Primary User competition occurs. The Primary Users begin by acquiring channel resources given by BSs. The Primary Users then use a part of their resources and distribute the remnants to the Secondary Users. The Secondary Users must compete for unused channel resources. As a consequence, a system for resource allocation among Primary Users should be developed to ensure that their advantages are maximized.

4.3. Secondary User to Secondary User

The SU agent is the first to receive information about the channel's resource consumption status, which includes the active channel count and the PU—SU count. Between the CR users, there are two sorts of relationships: competitive and neutral. Mathematically, we may assume that the numbers of channel resources, Primary Users, and Secondary Users are A, B, and C. Secondary Users are in a state of neutrality when A–B.C. As a consequence of bargaining among the Secondary Users, the different idle channel resources may be occupied. When A–B.C occurs, a competitive link between the Secondary Users exists as well as A–B subchannels. These Secondary Users compete directly for spectrum resources in the A–B subchannels. As a consequence, each Secondary User must develop an appropriate policy.

4.4. Primary Users to Secondary Users

PUs−SUs communicate and learn inside CRNs. PUs make the most efficient use of the spectrum resources made available by the Base Stations during the encounter. Following that, the Secondary Users may be allocated idle resources. As previously mentioned, N Secondary Users compete for spectrum resources with S = (s1, s2, . . . , sn) data needs. Primary and Secondary Users both strive to maximize resource usage in a dynamic multi-agent environment. To perform this, Primary Users must be able to distinguish Secondary Users and their data needs. Each Primary User has an allocation, a request, and a wait strategy. When a Primary User's idle resources are adequate for provision to the Secondary Users, the Primary User chooses to wait; however, when the Primary User's idle resources are insufficient for allocation to the Secondary Users, the Primary User decides to seek resources from other Primary Users. Thirdly, when another Primary User submits a request, the current Primary User determines whether to approve it.

4.5. Evaluation Methods

Machine learning is concerned with deriving a function from a noisy set of data, referred to as the training set, which was created by an unknown true function. Supervised learning and reinforcement learning are two machine learning techniques that are useful in this context. Supervised learning derives a function from a series of data pairs supplied by a supervisor, each of which has an input and a desired outcome. Artificial Neural Networks are a family of function approximators that may be customized for a particular job by adjusting their weights appropriately [24]. Training an ANN entails progressively adjusting its weights to minimize the error function between the ANN-represented function and the real function's actual noisy data samples. The term "backpropagation" refers to the practice of applying gradients to ANNs. On the other hand, reinforcement learning is concerned with how a software agent learns to act in a particular environment to accomplish a certain goal, such as the maximization of a particular kind of reward. As a result, it is well-suited to resolving control difficulties, such as those seen in Radio Resource Manager. Following that, we look at a model-free case in which the issue is fully described with three components: the state, action, and reward. The state s is a tuple of variables referred to as features that uniquely describe the agent's surroundings in terms of the task at hand. The action a denotes the agent's modification of the surrounding environment. The reward function r is a multi-objective scalar function that quantitatively represents the agent's goal. The agent's interaction with the environment throughout time is characterized by a sequence of tuples (s_t, a_t, r_{t+1}, s_{t+1}), each of which represents a state transition caused by the agent executing actions on the environment and receiving rewards. The goal of reinforcement learning is to construct a policy from a collection of transitions that, given a state, generates the action to perform to maximize the cumulative long-term reward. Thus, a reinforcement learning system establishes a connection between rewards and distantly related behaviors—the technical word for this is credit assignment. A reinforcement learning approach should quickly transition an agent from a blank slate state, in which it has no idea of how to behave, to an ideal state. Making the fewest potential mistakes on the way to quasi-optimal behavior is referred to as regret minimization, a notion that is strongly connected to the problem of balancing environmental exploration and knowledge extraction. This progression from exploration to exploitation may take a number of forms.

4.6. Blockchain-Enabled Network Slicing Agents (BENS-Agents)—An Intelligent Multi Agent System

For the sake of the design, a private blockchain has been used as shown in Figure 4. Employing the user front end, consent is written or recorded on the network of blockchains that are based on the Hyperledger Fabric after signing up for the service and saving the appropriate information on the database. The Core Network Slice ID, the Access Network Slice ID, and the Network Access Slice Information are all included in the details of the consent. A user has the option of mentioning whether they are ready to grant entire or partial access to the resource. When a user records their consent data on blockchain, an administrator has the ability to verify those details and then provide access to research organizations. Using the front end of the organization, the organization on the network is able to view the consent information of the user, and access may be sought from the healthcare administrator if the user has provided full access. The organization's front-end interface may also be used to request more access if necessary. When a research organization asks for further consent from a user, the user's account will be updated with a notice. The information will be logged in the network as a transaction regardless of the user's decision on whether or not to comply with the request. In the event that the request has been granted, the administrator will have permission to exchange the data with the organization on an as-requested basis. It is possible for the network to keep a record of the specifics of an individual user's request to have their data removed. The transaction will be updated with the new information. After verifying the information provided by the user, the administrator promptly removes the information from the database.

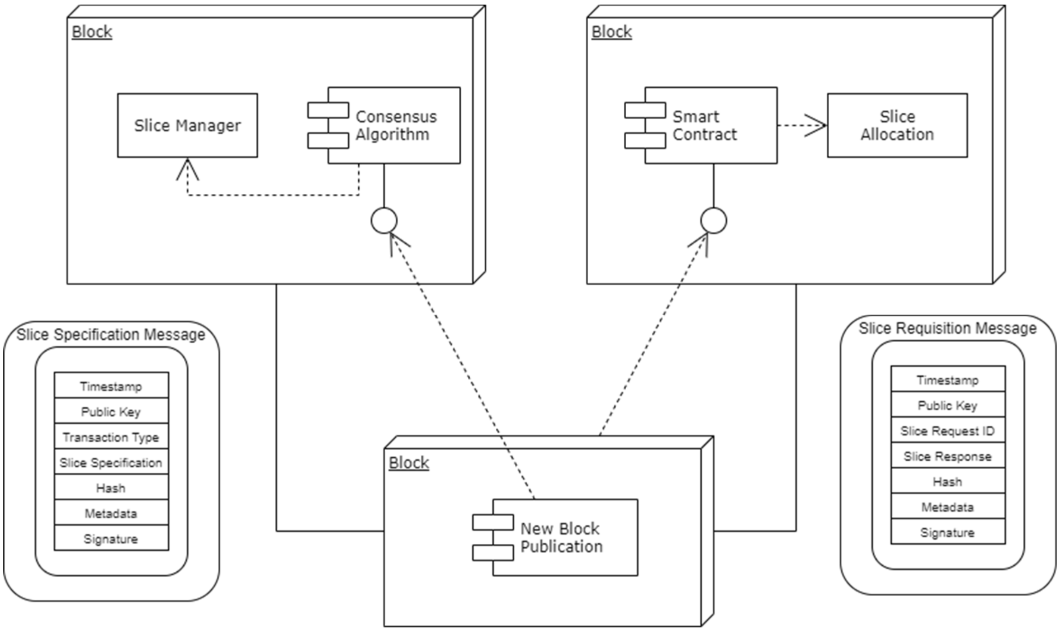


Figure 4. BENS-B5G model.

An Intelligent Agent (IA) adapts the AI properties, such as:

- Actions—the responding capability of an agent towards environmental changes and events;
- Perceptions—environment-provided data, which on further process, the data will be accumulated as information;
- Goals—system objectives;
- Events—update beliefs and perform actions;
- Beliefs—the processing of environment-provided accumulated information;
- Messages—the interaction of agents;
- Plans—achieving goals and handling events;
- Protocols—interaction rule sets.

4.6.1. BENS Agent Blocks

Figure 4 represents the blockchain consortium which depicts the slice requirements and requisition message formats. In addition to the vast amount of user equipment competing to access a large number of spectra, various applications will have their own QoS requirements. The QoS requirements are to be considered for channel assignment. The QoS-aware Application-specific agent behavioral model is formulated in such a way that each device enabled with Intelligent Agents will intelligently make spectrum access decisions with the help of their observations. QoS requirements are incorporated into the learning process, and the learning process is enhanced with learning transfer and cooperative learning mechanisms to adapt to the distributed heterogeneous environment. When new user equipment joins the environment, or when it applies for a new service, it can directly search for the expert agent among the neighbors, and by utilizing the learning transfer model, it can infer the knowledge about the data it seeks from the expert agent. The actions and events are in the cooperative learning model, and the data shared can be utilized for decision making.

BENS incorporates Neural Networks to switch from the observed state to actions through various layers, instead of relying on Q-values and its memory storage. Any high-scale complex model can be realized using Neural Networks with the help of multi-dimensional data. Additionally, by using the experience replay and generalization capabilities of Neural Networks, BENS may enhance network performance. A large number of communication entities can attempt to access a highly constrained spectrum resource in 5G and beyond 5G networks. This can be framed as a learning problem of the Multi Agent System. Every agent expects the network environment to offer spectrum resources and allied services by being in a state in which they can observe the environment. Our proposed approach to design a multi-agent system comprises of two stages, as follows:

4.6.2. Training Stage

The training stage for a BENS multi-agent system is proposed to have the following list of behavioral characteristics

- The agents are all the communication linkages, and the wireless network serves as the environment.
- By integrating with the environment, each agent intelligently monitors its present condition.
- Then, based on the learned policy, it makes a decision, and selects an action.
- Following that, the environment provides each agent with a new state and an instant reward.
- All agents intelligently learn new policies in the future time step based on the input.
- An established replaying method is used to increase the rate of learning, the efficiency of learning, and the stability of learning toward the ideal policy for expansive access control.
- The storage memory is used to store the training data.

4.6.3. BENS Training Stage Algorithm (BENS-T)

The BENS-T Algorithm [Algorithm 1] is a reinforcement learning algorithm that operates inside a 5G environment. The algorithm is based on a Markov model that has numerous states and actions. The algorithm decides the action to take in each state by accumulating knowledge about the values associated with each possible action. In addition to that, it combines many exploration methodologies within its overall architecture. It has been shown that the algorithm will eventually arrive at the best course of action for each state.

Algorithm 1. BENS—Training Algorithm (BENS-T).

1. **Input**: BENS framework, scenario emulator, and all applications' QoS requirements.
2. **For** every session i = (1, 2, ... , to M)$^{\text{instancs}}$, perform the following:
3. **Initialize**: every agent Q-network where, function $Q(p, b)$, rule-based approach $\varphi(p, b)$, load β, refactor W;
4. **Perform** process *Instance* = (0, 1, 2, to F);
5. Every agent keeps track of its condition T_s;
6. Randomly **select** action A_c with probability λ;
7. Alternatively, randomly select action A_c with probability = max arg a∈A $Q_t(P_s, b_s, \mu_s)$;
8. Carry out an **action** at, then collect a reward W_{ins};
9. **Record** a fresh state T_{s+1};
10. **Store** refactor $W_{ins} = (p_s, b_s, W(p_s, b_s), T_{s+1})$ in memory **O**;
11. **Repeat for** every agent, proceed;
12. Randomly sample a micro-data k_s in **O**;
13. **Add** μ_s—Update;
14. Incline to update μ_{s+1};
15. **Update** φ, the policy with Q-max_value;
16. **Perform** action on φ;
17. **Conclude** the loop for;
18. **Conclude** the **loop for;**
19. Return: Return trained BENS models.

4.6.4. BENS-Agent Learning Algorithm (BENS-AL)

Algorithm 2 aggregates and groups agent data by observing and identifying the neighborhood agents. The agents are performed with actions and rewards with the decisions based on aggregated data.

Algorithm 2. BENS-Agent Learning Algorithm (BENS-AL).

1. **Input:** BENS structure, simulation of environment parameters, and QoS parameters of end applications.
2. **Initialize:** Every agent in observation stage S;
3. Learning transfer: If found new agent/service;
4. **Identify** the neighbors of agent and transfer information;
5. **Analyze If** the information exist
6. Identify Expert Agent EA;
7. **Extract** EA information;
8. Check and **do** database update D_t;
9. Set transmission rate μ;
10. **Select** an appropriate action A_t;
11. Cooperative learning: Every group CG share actions and observed data;
12. Aggregate the group data CG_D;
13. Set join policy $\pi_g(CG_g)$ and aggregated value $A_g(CG_g, b_g)$, and Set aggregated action AG_g;
14. Execute action AG_g, receive reward RW_g;
15. **end if.**
16. **End BENS-AL**

5. BENS-5G: Procedure for RAN Slicing in 5G-CORE

We devised a mathematical model of the CRN-enabled 5G in different scenarios:

1. No channels are reserved;
2. Certain idle channels are allocated for the PUs. To begin, a wireless channel is mathematically designed for NONRES and RES.

Whenever a wireless channel is devised, it is always as a two-state Markov process. An Occupied State indicates that a PU is now occupying the current channel. An Unoccupied State is a channel that is not in use by a PU, and is thereby accessible to an SU. Allow for M wireless channels in the system. As a result, the transition probability $P(a,b)$ represented in Equation (1) indicates that there are a number of vacant channels in the present frame, and b in the following frame.

$$P_{a,b} = \sum_{y'=\max(a-b,o)}^{\min(a,a-b)} \binom{a}{y'} P_{00}^{a-y'} P_{01}^{y'} \binom{A-a}{x'} P_{11}^{a-y'} P_{10}^{y'} \quad (1)$$

where, in this case, $x = b - a + y'$. The y' and x' represent the number of channels that have changed their state from Vacant to Occupied and Occupied to Vacant, respectively. Spectrum sensing with inaccuracies introduces malfunction and misses detection issues. As a result, the number of observed vacant channels is a measure of the total range of actual vacant channels, indicated by Equation (2): $a'(a)$, which is given by:

$$a'(a) = a - a\left(p_{fa}\right) + (A-a)(1-p_{md}) \quad (2)$$

where p_{fa} is the false alarm probability;
p_{md} is the detection probability;
$1 - p_{md}$ is the miss-detection probability.

5.1. Assumptions

There are N channels, each of which is shared by primary and secondary users, with the main user taking precedence over the secondary user(s). Calls from primary users occur at a rate of λ1, while calls from secondary users occur at a rate of λ2. The µ1 and µ2 are the respective service rates. The following assumptions are made:

(1) When a PU joins a channel already occupied by SUs, the SU is constantly aware of the PU's presence. Notably, this causes a temporary clash with the principal user. A search for a new channel is then initiated by the SU. The SU will check the remaining channels at random until it finds a free one or determines that all the bands are occupied. The likelihood of a free channel being occupied is dictated by the probability of a false alarm occurring, and the probability of an occupied channel being free is given by the probability of a miss detection occurring.

(2) All state changes are immediate, implying that the time required to find a free channel is insignificant.

(3) An SU is aware of the channels that are being used by other SUs and will avoid them. For example, the relevant information may be conveyed through a signaling channel.

(4) A PU is aware of the channels that other PUs are using, ensuring that there are no clashes between primary users.

(5) In the event of a channel collision (between the PU and SU), both conflicting users continue the transmission in the channel. Notably, the collision between the PU and SU is believed to be brief, and does not result in the PU exiting the channel.

(6) The sequence in which new free channels are identified is random. When a free channel is found, or all channels are confirmed to be occupied, the search comes to an end. Many channels may be searched before halting the entire procedure.

5.2. Procedure Model

A two-dimensional Markov chain is used to represent the system. Tuples P and Q are used to represent the number of channels used by primary and secondary users, respectively, in the system states. There are two primary user and one SU in the (2,1) example. Assume that K is the number of channels in the system that are accessible. N is the maximum number of users, primary and secondary, on a given channel. As a result, the limitations being enforced are $0 \leq P \leq K$, $0 \leq Q \leq K$ and $0 \leq P + Q \leq K$. Let $z(P, Q)$ be the duration spent in state by the system state (P, Q), which signifies the probability of steady-state.

5.3. State Transitions

The following chart depicts the state transition diagram where three channels are used. Iterating through all possible channel and detection event sequences yielded the state-dependent transition rates. Consider the progression from state (1,1) to state (2,1). (1,0). The detection events and the channel search order changes are shown in this section. One of two occurrences might cause this transition to occur. During the first scenario, the current secondary user call is terminated at a rate of two, which reduces the number of secondary users by one. When a new primary user joins an existing secondary user's channel, it forces the secondary user to hunt for new channels. A collision between an existing secondary user and an already existing PU results in both calls being dropped, leaving just the new primary (1,0). To go a little deeper, start with the current state of affairs (1,1). It is possible for the new PU to utilize either an existing channel, or one that is currently being used by a secondary user, since the main user is aware of any existing PU. Since the new primary user is not worried about secondary users, they will approach the primary user's channel with a probability of 0.5. This results in a collision with probability P_M when the secondary user connects straight to the current user's channel with a probability of 0.5. Secondary users may connect to a free channel (with a chance of 0.5) first, and then continue to the primary user channel, and obtain a misdetection if they are unsuccessful. The process of movement is from a state (2,1) to another (3,0). The secondary user will always be forced by the new primary user to leave the channel, and search for other accessible channels. It is possible for the secondary user to accurately identify that the primary user is consuming the two extra channels with probability P. Figure 5 represents the state transitions which happen in 5G CORE functions during slice mapping.

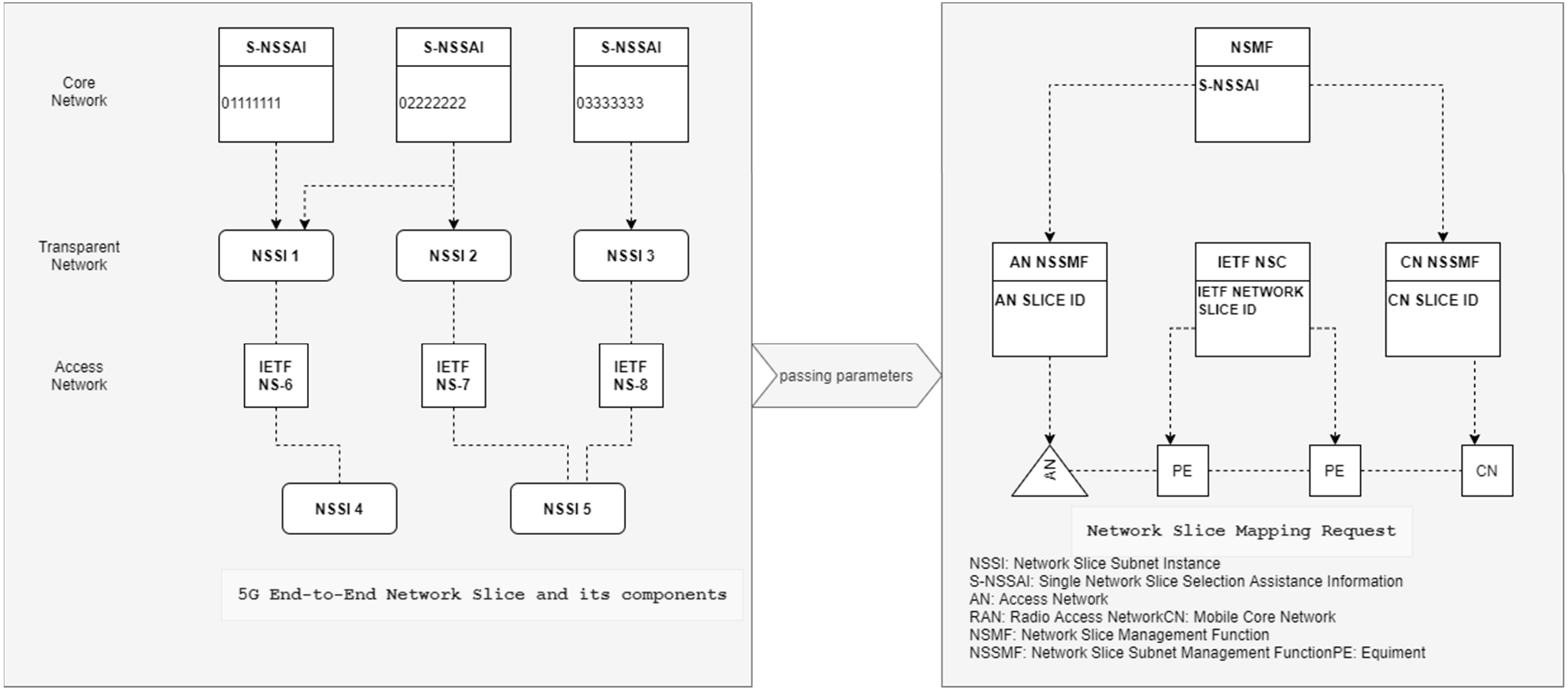


Figure 5. Network slice mapping structure for 5G.

5.4. BENS Algorithm for Channel Reservation (BENS-R)

The BENS-R algorithm [Algorithm 3] optimises channel usage by implementing an adaptive channel reservation method based on PU activity. It also offers priority-based access to various kinds of SU traffic for which the sample given in Table 1, which leads to an increase in network system capacity and heterogeneity.

Table 1. Simulation Parameters.

Parameters	Values
Cell Radius	500 m
Carrier Spectrum/Bandwidth	2 GHz, 100 MHz
Inter-device Distance (max.,)	75 m
Number of Primary devices	100, (200–600)
Number of channels	100
Number of SU devices	500, (1000−3000)
Latency benchmark value	(1, 2, 4, 6, 8)
SINR Threshold	5 dB
Max Transmit power of each device	500 mW
Device power consumption	50 mW
Device power consumption	50 mW
Packet size	1024 bytes

Algorithm 3. BENS Algorithm For Reservation (BENS-R).

Input: N number of available channels in the CRN,
N_{np} Number of PU-occupied channels in Unreserved CRN
N_{ns} Number of SU-occupied channels in Unreserved CRN
N_{rp} Number of PU-occupied channels in Reserved CRN
N_{rs} Number of SU-occupied channels in Reserved CRN
f_a number of failed channels in the CRN
max_{res} number of maximum reserved channels
values K_i such that $K_{a+1} < K_a < K_{a-1} < K_{a-2} \ldots < K_i < \ldots < K_0$;
i = 1 to a, where K_a and $K_{a+1} = 0$
Output: M number of reserved channels in the CRN,
Step 1: Calculate $\beta = (p_n + q_n + p_r + q_r)/(Q - e)$
Step 2: Find $M_{Available} = Q - (p_n + q_n + p_r + q_r + e)$
Step 3: Find $R_N = p_r + q_r$
Step 4: while(k > 0) do
if $(k_{i+1} \leq \beta < k_i)$
thentraffic_threshold_level = j;
k = k − 1; repeat
Step 5: do break;
Step 6: end
Step 7: Switch to operating mode:

- Operating mode = 0 indicates that the objective is to increase the retainability of current monitoring services.
- Operating mode = 1 denotes the increase in the availability of channels for fresh users.

Step 8: if selected Operating mode = 0 then
Step 9: $M' = max_{res} -$ traffic_threshold_level
Step10: end
Step 11: if selected Operating mode = 1 then
Step 12: $M' = max_{res} -$ (a-traffic_threshold_level)
Step 13: end

6. Performance Analysis

We analyzed the benefits of the reservation of channels strategy in the CRN for individual SUs and the entire system. To begin, the analytical models developed in previous sections are utilized to analyze the traffic for each SU. Additionally, a simulation model for SU traffic-based CRNs is developed in OMNET++ to investigate and confirm the performance increase in the Reservation-enabled CRNs, compared to CRNs with unreserved channels.

From Figure 6, as the channel reservation reduces the Secondary User's spectrum use, it is evident from the figure that the reserved CRN has a lower Secondary User throughput than the unreserved CRN. As the number of reserved channels for the Pus decreases, the number of Sus allowed into the CRN increases, resulting in an increase in the overall system throughput, as depicted. However, this increases the amount of interference and the number of incorrect packets received by the Secondary Users.

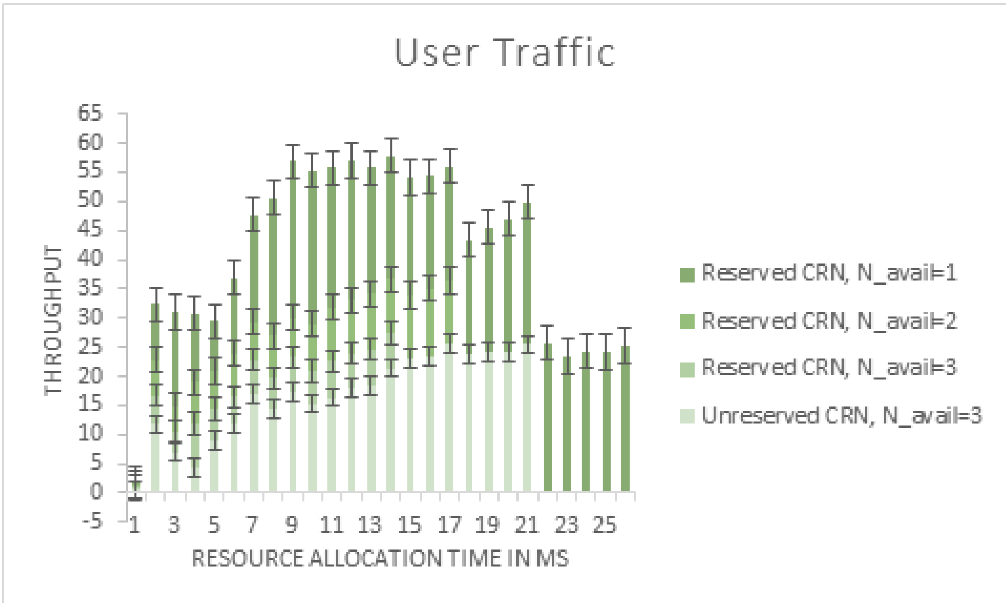


Figure 6. Variation in user traffic with respect to channel reservation time (slice request).

Figure 7 depicts a temporal average plot of the proportion of time during which incorrect packets are received by receivers at the SUs' end. It is observed that in an unreserved CRN, incorrect packets account for about 4/5th of the entire transmission time, but in a reserved CRN, this proportion is drastically decreased. The overall number of SUs that drop likewise reduces significantly in the reserved CRN. The analytical model for SU traffic demonstrates the user-centric requirement of a channel reservation system for ensuring QoS in real-time applications. On the other hand, the simulation model demonstrates the reserved CRN's benefits as a system, and further verifies the reserved channels for the PUs. Thus, the contribution of this study to a comprehensive performance assessment has been effectively demonstrated.

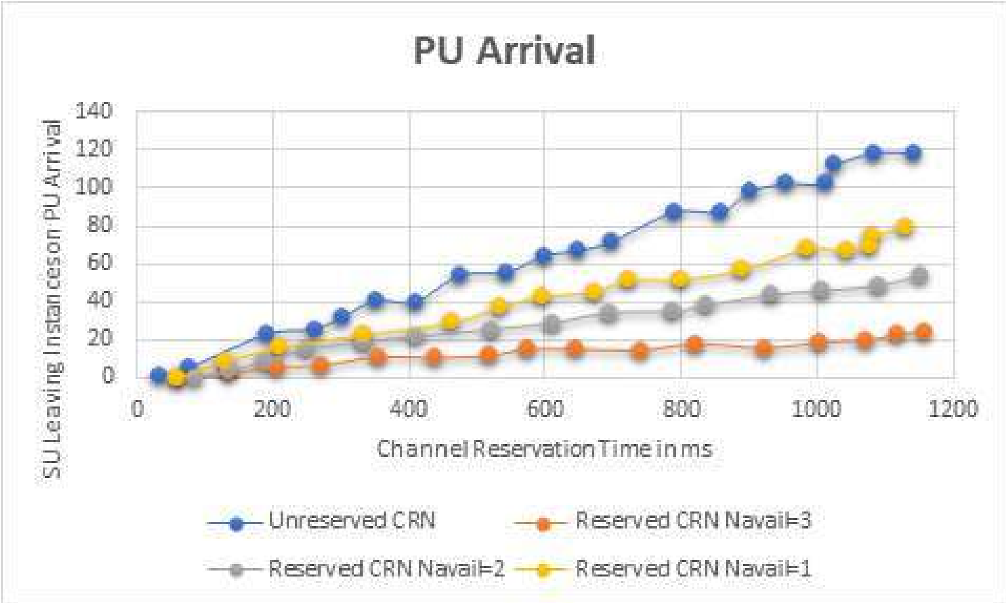


Figure 7. PU Channel Switching.

6.1. Simulation Results and Inferences for BENS-R

Table 1 depicts the simulation parameters being configured in Omnet++ to analyze the performance of network convergence of our proposed BENS-5G model multi-agent system with distributed and centralised multi-agent system

We have compared the proposed Blockchain-Enabled Network Slicing (dubbed BENS, which incorporates both transfer and cooperative learning mechanisms) to the fully distributed multi-agent RL-based massive access approach, in which each communication link determines its sub-channel assignment and transmission power strategy independently of other communication links. As seen in the Figure 8, the suggested learning technique greatly outperforms the fully distributed learning and centralized MA approaches in terms of EE performance and transmission success probability. The suggested technique achieves a quicker convergence rate via the use of transfer learning and cooperative learning mechanisms to boost the learning efficiency and speed. Since it does not need device cooperation, the completely dispersed strategy is straightforward; however, it results in a poor global performance, which results in a low EE value, and a low likelihood of transmission success. One of the most important contributions made by the algorithms developed for this work is the efficient management of network resources, with a particular emphasis on the computational and networking resources of core networks. In 5G networks, however, computing resources have been placed closer to the end users via edge-computing technologies in order to reduce excessive end-to-end latency. As a result, the problem of resource allocation will become much more difficult, particularly for online conditions. This is due to the fact that these conditions will require more computational time in order to more accurately categorize the best resources and paths when it comes to allocating network services that are sensitive to delay. Additionally, in migration situations, a shorter migration time is an essential key performance indicator regarding the effectiveness of performing resource allocations for moving services in preparation for a real-time scenario.

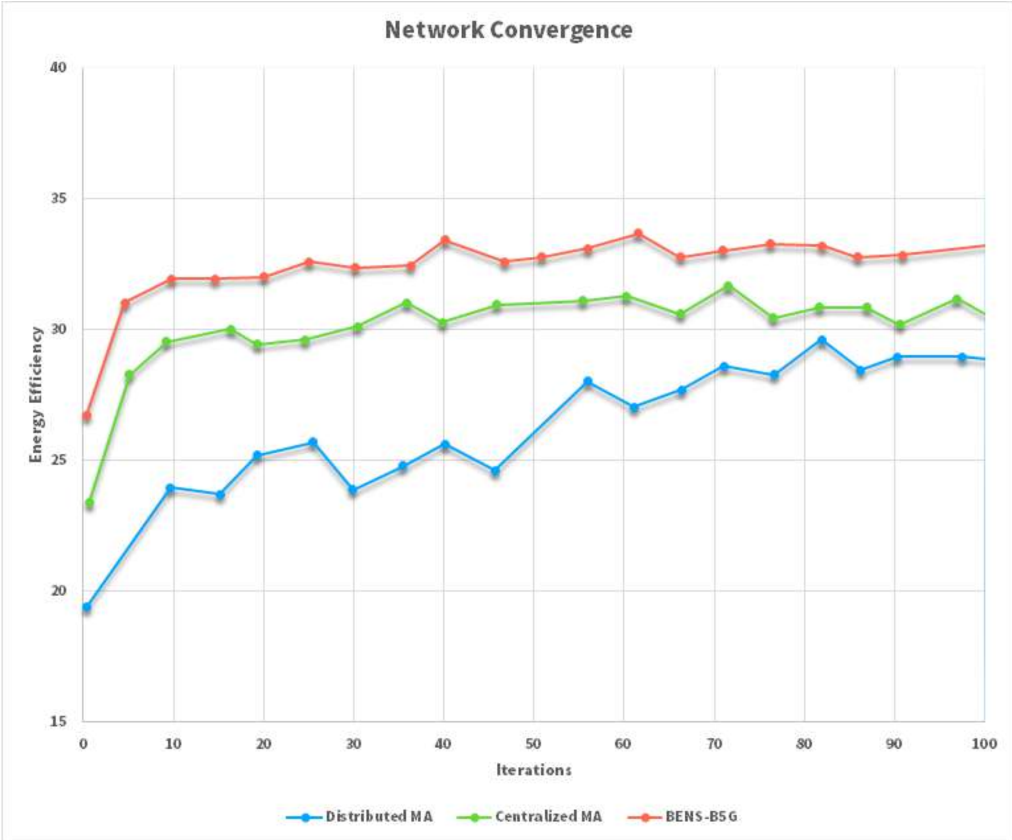


Figure 8. Convergence comparison.

6.2. Inference

As mentioned earlier, most research on dynamic spectrum allocation in CRNs is focused on spectrum sensing and allocation. However, minimal research on dynamic spectrum access with channel reservation and the mobility of the CRNs is based on channel reservation. Very few studies have examined the use of PU and SU dynamics in CRNs to enhance hand-off and mobility management in typical cellular networks. We allow for a change of the PU—SU characteristics to satisfy the needs of new services. These modifications are accomplished via the use of consensus and BENS algorithms that regulate PUs and SUs through negotiation and allocation procedures. As a result of the discussion above, we can affirm that the Blockchain Consensus Spectrum Management algorithm and BENS Model used to control RAN in 5G is capable of providing very efficient solutions to a variety of cognitive radio difficulties. The use of the collaborated framework in 5G−CORE is intriguing, and represents unexplored research.

7. Conclusions

In a 5G−CORE network, we presented a system for dynamic spectrum sharing, and concentrated on the setup of RAN Slicing in 5G−CORE using a consensus algorithm. We also presented a learning strategy for the extension of spectrum sharing in 5G−CORE to overcome the dynamic spectrum access issue in 5G. Our proposed BENS approach achieves higher energy-efficient performance and greater probability-of-success rate for transmission than other distributed and centralized learning approaches. Our proposed approach proves to have a good and faster convergence speed by assuming learning handover and combining collaborative techniques for learning. Furthermore, we discussed the issues of interference and power allocation and proposed the methods for solving the issues. The proposed methods are feasible for implementation in any vendor-specific 5G−CORE architecture. The Dynamic Spectrum Allocation Framework for the 5G environment did not consider the applications that operate on 5G radio environmental dynamics. We suggest that future research directions must be to identify the heterogenous applications and their traffic over 5G and beyond networks, to enhance the proposed work.

Author Contributions: Conceptualization: S.S., C.R.B., B.Y. and K.R.; methodology: C.R.B., B.Y., K.R., S.S. and I.-H.R.; software: C.R.B., B.Y. and K.R.; resources: B.Y., K.R., S.S. and I.-H.R.; writing—original draft preparation: S.S., C.R.B. and K.R.; writing—review and editing: B.Y. and I.-H.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by a grant (21163MFDS502) from ministry of food and drug safety in 2022. And this work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2021R1A2C2014333).

Institutional Review Board Statement: Not Applicable.

Informed Consent Statement: Not Applicable.

Data Availability Statement: The data used to support the findings of this study are available from the author (postman3@dongguk.edu) upon request.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. An, N.; Kim, Y.; Park, J.; Kwon, D.H.; Lim, H. Slice management for quality of service differentiation in wireless network slicing. *Sensors* **2019**, *19*, 2745. [CrossRef] [PubMed]
2. Theodorou, V.; Lekidis, A.; Bozios, T.; Meth, K.; Fernandez-Fernandez, A.; Tavlor, J.; Diogo, P.; Martins, P.; Behravesh, R. Blockchain-based zero touch service assurance in cross-domain network slicing. In Proceedings of the 2021 Joint European Conference on Networks and Communications and 6G Summit, EuCNC/6G Summit, Porto, Portugal, 8–11 June 2021; pp. 395–400. [CrossRef]
3. Zanzi, L.; Albanese, A.; Sciancalepore, V.; Costa-Perez, X. NSBchain: A Secure Blockchain Framework for Network Slicing Brokerage. In Proceedings of the IEEE International Conference on Communications, Dublin, Ireland, 7–11 June 2020. [CrossRef]
4. Sattar, D.; Matrawy, A. Towards Secure Slicing: Using Slice Isolation to Mitigate DDoS Attacks on 5G Core Network Slices. In Proceedings of the 2019 IEEE Conference on Communications and Network Security, CNS, Washington, WA, USA, 10–12 June 2019; pp. 82–90. [CrossRef]
5. Gonçalves, J.P.d.B.; de Resende, H.C.; Villaca, R.d.S.; Municio, E.; Both, C.B.; Marquez-Barja, J.M. Distributed Network Slicing Management Using Blockchains in E-Health Environments. *Mob. Netw. Appl.* **2021**, *26*, 2111–2122. [CrossRef]
6. Li, X.; Samaka, M.; Chan, H.A.; Bhamare, D.; Gupta, L.; Guo, C.; Jain, R. Network Slicing for 5G: Challenges and Opportunities. *IEEE Internet Comput.* **2018**, *21*, 20–27. [CrossRef]
7. Makarfi, A.U.; Kharel, R.; Rabie, K.M.; Kaiwartya, O.; Li, X.; Do, D.-T. Reconfigurable Intelligent Surfaces Based Cognitive Radio Networks. 2021. Available online: http://arxiv.org/abs/2008.10946 (accessed on 20 February 2022).
8. Shi, Z.; Xie, X.; Lu, H. Deep Reinforcement Learning Based Intelligent User Selection in Massive MIMO Underlay Cognitive Radios. *IEEE Access* **2019**, *7*, 110884–110894. [CrossRef]
9. Yau, K.L.A.; Komisarczuk, P.; Paul, D.T. Enhancing network performance in Distributed Cognitive Radio Networks using single-agent and multi-agent Reinforcement Learning. In Proceedings of the Local Computer Networks, LCN, Denver, CO, USA, 10–14 October 2010; pp. 152–159. [CrossRef]
10. Nencioni, G.; Garroppo, R.G.; Gonzalez, A.J.; Helvik, B.E.; Procissi, G. Orchestration and Control in Software-Defined 5G Networks: Research Challenges. *Wirel. Commun. Mob. Comput.* **2018**, *2018*, 6923867. [CrossRef]
11. Hernández, J.; López, D.; Vera, N. Primary user characterization for cognitive radio wireless networks using long short-term memory. *Int. J. Distrib. Sens. Netw.* **2018**, *14*, 1550147718811828. [CrossRef]
12. Ding, A.Y.; Janssen, M. 5G Applications: Requirements, Challenges, and Outlook. In Proceedings of the Seventh International Conference on Telecommunications and Remote Sensing, Barcelona, Spain, 8–9 October 2018. [CrossRef]
13. Li, X.; Fang, J.; Cheng, W.; Duan, H.; Chen, Z.; Li, H. Intelligent Power Control for Spectrum Sharing in Cognitive Radios: A Deep Reinforcement Learning Approach. *IEEE Access* **2018**, *6*, 25463–25473. [CrossRef]
14. Massaro, M. Next generation of radio spectrum management: Licensed shared access for 5G. *Telecommun. Policy* **2017**, *41*, 422–433. [CrossRef]
15. Rahimzadeh, P.; Ashtiani, F. Analytical Evaluation of Saturation Throughput of a Cognitive WLAN Overlaid on a Time-Scheduled OFDMA Network. *IEEE Trans. Mob. Comput.* **2017**, *16*, 634–647. [CrossRef]
16. Wu, Q.; Wang, Y.; Yin, Z.; Deng, H.; Wu, C. A Novel Approach of Cognitive Base Station with Dynamic Spectrum Management for High-Speed Rail. 2016. Available online: https://openreview.net/forum?id=B1bmYVz_br (accessed on 20 February 2022).
17. Yang, Y.; Dai, L.; Li, J.; Mumtaz, S.; Rodriguez, J. Optimal spectrum access and power control of secondary users in cognitive radio networks. *Eurasip J. Wirel. Commun. Netw.* **2017**, *2017*, 98. [CrossRef]

18. Rajesh Babu, C.; Amutha, B. Blockchain and extreme learning machine based spectrum management in cognitive radio networks. *Trans. Emerg. Telecommun. Technol.* **2020**, e4174. [CrossRef]
19. Babu, C.R.; Amutha, B. A novel energy-efficient data aggregation protocol for cognitive radio based wireless multimedia networks. *Peer Peer Netw. Appl.* **2021**, *14*, 2452–2461. [CrossRef]
20. Nasser, A.; Hassan, H.A.H.; Chaaya, J.A.; Mansour, A.; Yao, K.C. Spectrum sensing for cognitive radio: Recent advances and future challenge. *Sensors* **2021**, *21*, 2408. [CrossRef]
21. Mockel, B.P.; Makala, B. Artificial Intelligence and 5G Mobile Technology Can Drive Investment Opportunities in Emerging Markets. Available online: https://openknowledge.worldbank.org/bitstream/handle/10986/33388/Artificial-Intelligence-and-5G-Mobile-Technology-Can-Drive-Investment-Opportunities-in-Emerging-Markets.pdf?sequence=1 (accessed on 1 July 2022).
22. Arasteh, S.; Aghili, S.F.; Mala, H. A new lightweight authentication and key agreement protocol for Internet of Things. In Proceedings of the 13th International ISC Conference on Information Security and Cryptology, ISCISC, Tehran, Iran, 7–8 September 2016; pp. 52–59. [CrossRef]
23. Zheng, S.; Han, T.; Jiang, Y.; Ge, X. Smart Contract-Based Spectrum Sharing Transactions for Multi-Operators Wireless Communication Networks. *IEEE Access* **2020**, *8*, 88547–88557. [CrossRef]
24. Kotobi, K.; Bilen, S.G. Blockchain-enabled spectrum access in cognitive radio networks. In Proceedings of the Wireless Telecommunications Symposium, Chicago, IL, USA, 26–28 April 2017. [CrossRef]

Hindawi
Complexity
Volume 2022, Article ID 1388941, 10 pages
https://doi.org/10.1155/2022/1388941

WILEY | Hindawi

Research Article

COCO: Coherent Consensus Schema For Dynamic Spectrum Allocation For 5G

C. Rajesh Babu,[1] **Kadiyala Ramana**,[2] **R. Jeya**,[3] **and Asadi Srinivasulu**[4]

[1]*Department of Network and Communications, SRM Institute of Science and Technology, Chennai, India*
[2]*Department of Information Technology, Chaitanya Bharathi Institute of Technology, Hyderabad, India*
[3]*Department of Computing Technologies, SRM Institute of Science and Technology, Chennai, India*
[4]*Data Science Research Lab, BlueCrest University, Monrovia, Liberia*

Correspondence should be addressed to Asadi Srinivasulu; head.research@bluecrestcollege.com

Received 14 April 2022; Revised 12 May 2022; Accepted 31 May 2022; Published 6 July 2022

Academic Editor: Muhammad Ahmad

Numerous wireless technologies have been integrated to provide 5th generation (5G) communication networks capable of delivering mission-critical applications and services. Despite considerable developments in a variety of supporting technologies, next-generation cellular deployments may still face severe bandwidth constraints as a result of inefficient radio spectrum use. To this end, a variety of appropriate frameworks have recently emerged that all aid mobile network operators (MNOs) in making effective use of the abundant frequency bands that other incumbents reserve for their own use. The proposed COCO model for Dynamic Spectrum Allocation (DSA) has 2 functionalities such as 1. Coherent PU-SU packet acceptance algorithm for Secondary User (SU) in DSA. 2. Consensus Algorithm for PU-SU Channel Reservation in DSA. To enable a 5G service with one-millisecond latency, interconnection ports between operators are expected to be required at every base station, which would have a significant influence on the topological structure of the core network. Additionally, just one radio network infrastructure would need to be created, which all operators would then be able to use. We allow change of PU SU characteristics to satisfy the needs of new services. These modifications are accomplished via the use of Coherent and Consensus Algorithms that regulate PU and SU through negotiation and allocation procedures. Our primary objective was to decrease interference, handoff latency, and the chance of blocking. In this paper, we describe our idea for employing COCO Model to address the issues of spectrum mobility, sharing, and handoff for Cognitive Radio Networks in 5G.

1. Introduction

In order to meet the technical requirements for 5G, the sub 1 ms latency rate must be achieved. Content must be supplied from a location near to the user's device if a delay time of less than 1 millisecond is required. In order to provide a service with such low latency, content must be placed extremely near to the client, potentially at the base of every cell, including the numerous tiny cells that are expected to be important in achieving densification needs [1].To enable a 5G service with one millisecond latency, interconnection ports between operators are expected to be required at every base station, which would have a significant influence on the topological structure of the core network [2]. Additionally, just one radio network infrastructure would need to be created, which all operators would then be able to use.

Figure 1 illustrates the "overlay distribution" strategy by radio regulating bodies, which allows wide access to the majority of the frequency band, even if the frequency band is authorized for a certain application. Uncoordinated use of spectrum in both the time and frequency domains can be achieved by using the overlay information exchange. Techniques are used to disperse the generated signal across a wide range of frequencies so that existing licensed radio equipment does not detect an unacceptable level of power [3]. Examples of these approaches include frequency hopping, Multiplexing, and Ultra-Wide Band. This kind of

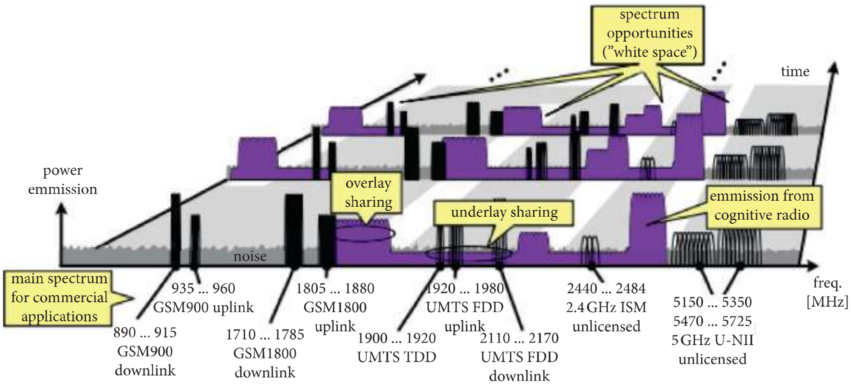


Figure 1: Cognitive radio and spectrum management in 5G.

interference is minimized by placing rigorous limits on the transmission power in the underlying dynamic spectrum.

1.1. Motivation for the Work. The Primary and Secondary users' activation and addition are dynamic with the fluctuating modalities of their interest and their requirements. The resource availability changes according to the usability, and the spectrum wastage occurs due to unused spectrum utilization. A dynamic spectrum sensing and sharing framework based on user needs is the need of the hour topic in Cognitive Radio [4]. There are disruptions possible while providing effective and efficient services meeting the QoS aspects. There is a precondition required to understand the users' requirements with a satisfactory resource provisioning mechanism along with validation with artificial intelligence in place, which remain a community hole to be filled through research.

2. Related Work

Liu et al. (2020) [5] presented a blockchain-based safe FL architecture for creating smart contracts and preventing hostile or untrustworthy parties from participating in FL. To fight against poisoning assaults, the central aggregator identified harmful and untrustworthy individuals by automatically executing smart contracts. Furthermore, membership inference attacks are prevented using local differential privacy approaches. The proposed approach, according to numerical findings, may effectively prevent poisoning and membership inference attacks, thereby boosting the security of FL in 5G networks. According to Rubayet Shafi et al. (2020) [6], Resilience, performance, and complexity were major technological hurdles to overcome while using AI in 5G and beyond 5G. Beyond-5G and sixth generation (6G) networks with AI-enabled cell networks were presented as a possible roadmap for future research to identify top challenges such as training issues, lack of bounding performance, and uncertainty in generalization, as well as a possible roadmap to realize the vision. In order to safeguard the user's identity and location, Hui Li et al. (2019) [7] implemented UGG, IPP, and LPP algorithms in the SBMs upload of a blockchain-based VANET. Two factors were used to assess the availability of k-anonymity unity: connection and average distance. Extensive simulations have shown the effectiveness of a blockchain-based VANET. The simulation took into account a number of elements, such as system time, average distance, connection quality, and privacy breaches. In terms of processing time, the suggested design outperforms the current designs, according to the simulation results. They also demonstrate that their proposed architecture provides a higher degree of privacy for their users' identities and locations. Huijuan Jiang et al. (2019) [8] proposed a distributed user association strategy based on multi-agent reinforcement learning to offer load balancing for cognitive radio networks with several independent APs. APs used reinforcement learning to find the optimum user association rules in the technique they proposed. It is portrayed as a dynamic match game between the APs and SUs. Every iteration, the APs picked which SUs they wanted to be associated with, and the SUs then choose which AP they wanted to be affiliated with based on all the APs' offers. Comparing the suggested multi-agent reinforcement learning technique to the classic max-SINR method, simulation results demonstrate that system performance is greatly improved while excellent resilience is maintained by the latter. According to Semba Yawada et al., (2019) [9] the spectrum mobility in cognitive radio networks has many key aspects. Despite the difficulty of upkeep and upgrading, the novel methods to mobility and connection management attempt to decrease latency and information loss during spectrum handoff. The reasons for spectrum handoff and the methods that lead to it were examined. Protocols have been developed to show how the handoff process works. The different spectrum handoff methods were compared. The suggested technique outperformed the pure reactive handoff method in simulations. There are still many unanswered questions about software-defined networks in 5G and 6G networks that have yet to be addressed by Long et al. (2019) [10]. (SDN). SDN technologies are utilized to introduce 5G and 6G mobile network system designs. It was thus necessary to draw attention to the main issues and common SDN-5G/6G application scenarios. There are also comparisons and descriptions of three kinds of software-defined 5G/6G mobility management frameworks. We took a look at how wireless cellular networks handle interference right now. An overview of interference control techniques was provided in SDN-5G/6G. For software-defined 5G and 6G networks, the mm-Wave spectrum, the absence of standard channel models, huge MIMO, low latency and Quality of Experience (QoE), energy efficiency, and scalability are all investigated. According to Daniel Minoli et al. (2019) [11] in IEEE Spectrum, IoT applications in Smart City environments face a number of challenges, including the requirement for small cells and millimeter-wave transmission issues, building penetration issues, the requirement for Distributed Antenna Systems, and the near-term introduction of pre-5G IoT technologies such as NB-IoT and LTE-M, which could serve as proxies for commercial deployment and acceptance of 5G. An improved target channel selection method was developed by Atif Shakeel et al., (2019) [12] in order to facilitate spectrum handoffs among the SUs in a CRAHN. It was recommended that SUs be organized during channel access according to the shortest job first idea using an improved frame structure that facilitates in cooperation among them in an ad hoc setting. By enabling SUs affected by inaccurate channel state predictions to compete for channel access within a single transmission cycle, the proposed system improves throughput over previous prediction-based spectrum handoff techniques. Since there are less collisions in the proposed method, it outperformed conventional spectrum handoff techniques in terms of throughput and data delivery time. Priority was given to Base Stations, PUs, and SUs in addressing the issue of spectrum resource distribution, according to Wang Bin Song et al. In the research, a multi-layer network model and a multi-agent system model were introduced. The RF Plan's BS structure and the number of Base Stations were both reduced thanks to the use of MAS for resource distribution. Designing a stratified multi-layer Multi Agent System that works in a dispersed environment and provides higher network performance while using less power is the research gap highlighted. A network environment for algorithm simulation is set up by Xiaomo Yu et al (2022) [13], after which they analyse the overall performance of the improved genetic algorithm and investigate the influence of genetic algorithm-related parameters and network environment-related parameters on the overall performance of the algorithm. The findings demonstrate the effectiveness of the enhanced genetic algorithm. It is possible to enhance network efficiency by approximately 2% while simultaneously reducing the frequency of spectrum switching by approximately 69%.

The summary of related work is shown in Table 1.

2.1. Research Gap. The basic purpose of cognitive radio is to detect interference on the channel that will be shared, as well as to protect the mobile user (PU) from interference. Due to the passive nature of cognitive radio, there is currently no practical means for identifying its impact on a given channel. According to popular belief, recognising PU signals is the same as finding spectrum possibilities, which is not the case [14].

3. Dynamic Spectrum Management Framework

As shown in figure 2, the DSMF incorporates the important components of a Licensed Spectrum Access domain. The Spectrum Manager is composed of 2 sub system blocks namely, Request Manager, which does the management of prioritizing based on DSA domain spectrum application rules, and Radio Spectrum Resource component, computes accessible resources for allocating the PU, on the basis of spectrum application rules and details saved in a Repository [15]. Regarding the PU block, a multi-PU channel is applied where a licensed system has User Equipment or UE associated to PU's Base Station (BS) provides optimal power which is received. Primary users, the DSA Repository, the DSA Spectrum Manager, and a number of Secondary Users are all implemented in the DSMF. Spectrum sharing policies may be implemented in both macro and small-cell scenarios using the framework's techniques for centralized and distributed allocation of resources. There are two separate modules in the Spectrum Manager, one for managing priority based on DSA spectrum use rules, and one for calculating available resources for Primary Users based on spectrum consumption rules and data stored in the Repository. It is now possible to have each UE (User Equipment) in the licensed network associate with the Base Station (BS) that offers the strongest received power for the PU frame, which is currently the case. The Secondary User's behavior is recorded in the DSA Repository as an array of "pixels." Each SU relates to a specific DSA channel since there is only one array for each DSA channel. It is necessary for DSA licensees to comply to an Interference-to-Noise Ratio (INR) of -6dB inside the SU's zone of protection. By sharing resources that might otherwise go unused, DSA aims to increase the framework's utility [16]. It is possible that greedy operators may pursue their own interests at the expense of other Research on node misbehavior was conducted utilizing the cognitive radio analogy. An analysis of a fictional cognitive radio system revealed that hostile or selfish nodes may engage in acts to disrupt or enhance their own value in the community of cognitive radio nodes. For enforcing shared access agreements both before and after the fact, the authors provide a variety of options. It is hoped that the preventative measures and punitive measures would inhibit tampering with the devices' software and hardware layers, while the latter is designed to identify and punish disobedient users. Instead, we advocate for the creation of a grading system to keep tabs on the DSA's behavior.

4. Network Model

The model analyzes a single channel Cognitive Radio Network. As in practice, with the consensus policy with mobile network operators, the arrival of PU in a channel is arbitrary and has the right to access the channel. SU packets come in two forms: SU1 packets and SU2 packets, and they're both utilized by the system. Priority is given to SU1 packets over SU2 packets. When it comes to taking up a single channel, the PU packets of the system take precedence. Additionally, this research makes the assumption that the spectrum sensing for the SUs is optimal. This suggests that the system model does not take into account the interactions between distinct SUs.

There is, in fact, to limit the contention of SU flooding over the channel, a Secondary User buffer prepared for packets from Secondary User2. If the channel is already full at the time of an incoming Secondary User2 packet, the newly arriving Messages from Secondary User2 will be held

Table 1: Related work Summary.

Reference	Proposed Technique	Allocation/ Sharing/ Sensing	Centralized/ Distributed	Simulation/Frequency Band	Efficiency	Parameters Improved
[5]	Blockchain-based safe FL framework has been presented by the authors in order to build smart contracts and prohibit malevolent or unreliable FL players.	Sharing	Centralized	MATLAB	Improved	PU Arrival Service time Number of Hand-offs
[6]	AI-enabled cellular networks for Beyond-5G and 6th generation (6G) networks are on the horizon.	Sharing	Centralized	Google Tensorflow on MNIST dataset and CIFAR-10 dataset, respectively.	Improved	Data Accuracy
[7]	SBMs are uploaded to the blockchain-based VANET in a revolutionary decentralized architecture employing blockchain technology that integrates UGG, IPP, and LPP algorithms with the method of dynamic threshold encryption and k-anonymity unity.	Sharing and Allocation	Centralized	-	-	AI-enabled fault identification Self-Recovery Mechanism
[8]	User association rules are learned via a reinforcement learning process by access points on their own in order to pick secondary user protocols for handoff and mobility management.	Sharing/ Privacy Protection	Distributed	MATLAB	Improved	system time, average distance, connectivity,
[9]	Protocols for Handoff and Mobility Management	Sharing and Allocation	Distributed	Simulated OFDM based CR	Improved	System Throughput SU Transmission Rate
[10]	Examined new 5G and 6G software defined networks (SDN) technology, which encompasses system design, resource management, mobility management, and interference control. SDN is a cutting-edge technology.	Sharing	Distributed	MATLAB	Improved	Bandwidth Collision Probability Delay
[11]	The millimeter wave spectrum must be addressed in order to accommodate high data rates in 5G cellular technology.	Sharing	Distributed	MAC Protocol from IEEE 802.11a	Improved	Collision with SU Extended Data Delivery Throughput
[12]	The spectrum handoff between SUs in a CRAHN is improved using an upgraded frame structure that promotes coordination among SUs based on an imperfect channel state prediction.	Sharing and Allocation	Distributed	Deployed a Hierarchical Multi Agent System Model for 5G	Improved	Channel Resource Allocation
[13]	A new paradigm for 5G cellular communication networks has been proposed, which balances resource distribution between main users, secondary users, and base stations.	sharing	Distributed	MATLAB	Improved	Switching probability

in the Secondary User2 buffer until they are ready to be sent out. To configure the buffers for Primary User and Secondary User1 packets, there are no alternatives Because Primary User packets have the greatest priority, if a Primary User packet arrives while a Secondary User packet is being sent, the transmission of the Secondary User packet will be immediately halted. For the sake of efficiency, the transmission of a freshly received Secondary User1 packet may only be temporarily halted by another Secondary User1. In both Secondary User1 and Secondary User2 packets, the interrupted packet's heightened need for transmission continuation causes both to become impatient. If a Secondary User packet is stopped during transmission, it will not be saved in the system and will not be able to be sent

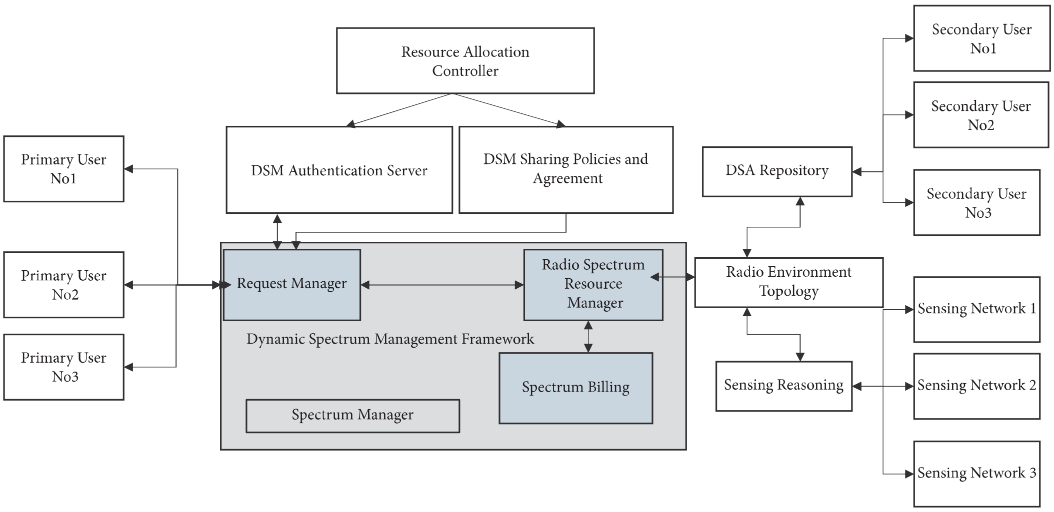


Figure 2: Dynamic Spectrum Management Framework.

again. It is possible for Secondary User packets to exit the system and broadcast on another open channel if they are interrupted. In light of the digital nature of current networks, the discrete-time Markov chain model shown below may be constructed by using the spectrum allocation approach described above. R = 1, 2,... is a convenient way to represent the division of time into equal-sized slots on an axis. Estimated arrival rates of Primary User, Secondary User 1, and Secondary User 2 packet transmission rates are predicted to follow geometric distributions at intervals determined by their respective rates of service μ_1, μ_{21}, and μ_{22}. If PU arrives back in channel, it is the responsibility of the algorithm to release the channel from SU to PU.

A system with a large number of Secondary User2 packets, a large number of Secondary User1 packets, and an even larger number of Primary User packets may be described by the following equation: A three-dimensional process comprised of the number $S(2)$ of Secondary User packets, the number $S(1)$ of Secondary User1 packets, and the number Pn of Primary User packets may be used to abstract the changes in the number of distinct packet types in the system. A discrete-time three-dimensional Markov chain is thus formed by $S(2)$, $S(1)$, and Pn. The state-space of $S(2)$, $S(1)$, and Pn may be expressed using the model assumption mentioned before.

$$\theta = \{(K,0,1) \cup (K,0,0) \cup (K,1,0)\colon\ 0 \le K \le \infty\}. \qquad (1)$$

5. EVALUATION OF MODEL

Assume that P is the probability matrix of state transitions in the Markov chain S(2), S(1), and Pn. The system model implies that the buffer capacity of Secondary User2 packets is limitless in order to accommodate additional Secondary User2 packets in the system. A consequence of this is that P is unlimited in its size by the number of packets experiencing state change for Secondary User2. Here's an example of how you can represent P and it is shown in (1).

$$p = \begin{pmatrix} a & b & & & \\ & d & a & b & \\ & & d & a & b \\ & & & \ddots & \ddots \end{pmatrix}. \qquad (2)$$

Secondary User1 packet transmissions are unaffected by Secondary User2 packets in the analyzed cognitive radio network with categorized Secondary Users and impatient packets. The primary and secondary user1 packet transmission operations may be seen as a single-server pure losing priority queueing architecture.

The transmissions of the Secondary User2 packets, on the other hand, are impacted not only by the transmissions of the Primary User packets, but also by the transmissions of the Secondary User1 packets. As a consequence, the performance of Secondary User2 packets will be the exclusive emphasis of this section. The γ rate of interruptions, throughput, and average latency of Secondary User2 packets are among the key performance measures that we create algorithms for. The number of Secondary User2 packets that are interrupted and leave the system per slot is what is meant by the interruption rate for Secondary User2 packets. The rate at which Secondary User2 packets are being interrupted may be described as in (3):

$$\gamma = \sum_{a=0}^{\infty} \pi_{a,0,0}\mu_{22}\left(1 - \lambda_{21}\lambda_{22}\right). \qquad (3)$$

The throughput ϕ of total number of the Secondary User2 packets successfully sent per slot is defined as Secondary User2 packets. A Secondary User2 packet may be successfully broadcast if and only if it is not interrupted during transmission. Equation (4) may be used to describe the throughput of packets transmitted by Secondary User2:

$$\phi = \lambda_{22} - \gamma. \qquad (4)$$

The average delay β in arrival of a Secondary User2 packets and their exit from system is defined as the Secondary User2 packets. Little's calculation for the mean delay of Secondary User2 traffic is expressed in (5)

$$\beta = \frac{E[SU2]}{\lambda_{22}}, \quad (5)$$

where the system's mean SU2 traffic in steady-state is denoted in (6)

$$E[SU2] = \sum_{a=0}^{\infty} \pi_{a,0,0} + \pi_{a,0,1} + \pi_{a,1,0}. \quad (6)$$

The Secondary User2 packet cannot ascertain the total traffic in the CRN prior to making a decision. If a Secondary User2 successfully sent a packet, it may receive the reward indicated by R, but if it decides to join the system, it incurs the penalty indicated by C per slot. If a single Secondary User2 packet elects to join the system, the Secondary User2 packet's individual net benefit function $We(\lambda_{22})$ may be expressed in (7)

$$We(\lambda_{22}) = \left(1 - \frac{\gamma}{\lambda_{22}}\right)R - \beta C. \quad (7)$$

6. Coherent Pu-Su Packet Acceptance Algorithm for DSA

In contrast to standard cognitive radios, adaptive transceivers transmit data using "dynamic" resources, which are always changing. In order to make use of this special quality of cognitive radio systems, the proposed adaptive algorithm distributes bandwidth for individual traffic regarding the knowledge of congestion queues' Quality of Service standards and the related data of the currently available spectrum. As an added bonus, the suggested method, which is applicable to multimedia applications that generate both genuine and quasi traffic, dynamically adapts the distribution probabilities between actual and potential real-time traffic based on the fluctuation of accessible spectrums.

To minimize the latency of SU packets, they are allocated a limited buffer with a capacity of N (N > 0). On the other hand, no buffer is supplied for the PUs, ensuring that the PUs' latency requirements are satisfied to the maximum degree feasible [16].

(1) In an adaptive admission control strategy, the central controller counts the number of packets in the system on a regular basis. A fresh SU packet will be accepted or rejected by the system's central controller based on its likelihood with acceptance probability as β= 1/(λ + 1) or reject it with probability β to the packet count of the system multiplied by the Coherent Factor. The system access probability is inversely proportional to the system packet count.

(2) When an SU packet is permitted into the system, it is queued in the buffer if the channel is currently in use by another packet. If the buffer is full, this SU packet will be terminated.

(3) To prevent conflict, the newly coming PU packet will be stopped if the channel is already in use by another packet of the same type [17]. If a packet of the same type is already in use by another PU, the newly arriving PU packet will interrupt this SU packet's transmission and take over the channel.

A packet from an SU that was in transit is returned to the SUs' buffer and placed at the front of the queue if the transmission was interrupted. If the buffers of the SUs become full, the system will force the last SU packet queued to exit. Since there is only one gap in the buffer when a fresh admission of an SU packet happens concurrently with an interruption of an SU packet, the newly admitted SU packet will be rejected by the system. Since the interrupted packet has a greater priority than the newly accepted one, it takes precedence. The greater the number of packets in a system, the greater the likelihood that it will be accessed. When the system is overloaded with packets, it's less probable that one of the freshly arrived SU packets will be accepted. Coherent PU-SU Packet Acceptance Procedure is represented inj figure 3.

7. Algorithm Analysis

7.1. Spectrum Analyzer Setup. Each cellular band was measured throughout the day where table 1 detail the spectrum analyzer and antenna specifications, respectively. The spectrum analyzer's frequency range is set at 100KHz to 3GHz. The spectrum occupancy measurement is recorded and plotted using the Rohde & Schwarz FSH Remote and MATLAB software. Spectrum occupancy was determined using a spectrum analyzer operating at frequencies ranging from 100 kHz to 3GHz.

7.2. Parameters and Values. Typical Wi-Fi network parameters, such as RTS = 44 bytes, CTS = 38 bytes, payload = 250 bytes, SIFS and DIFS = respectively 15, 34 microseconds and a slot size of 1 ms are used in the numerical results. We also employ a transmission rate of 10 Mbps while operating in the 2.4 GHz frequency range. The packet transmission rate is determined as $\mu 1$, $\mu 2$ = 0.5.

Additionally, the Malleable or Coherent Factor β data set is set to = {0.0 through 0.1 to, 1.0} with $\beta = 0$ being the standard access to system method without using admission control. This demonstrates the Coherent Factor's effect on the system's performance and the suggested adaptive admission control scheme's efficiency.

Simultaneously, we adjust the arrival rates of the PU and SU packets to λ 1 = 0.2 and 0.3 and λ 2 = 0.4 and 0.6 and investigate the impacts of the arrivals of the Primary User and Secondary User packets on measuring performance in a novel approach. For the sake of illustration, we assume that the SU's buffer capacity is set at N = 10. We derive that increasing the Secondary User buffer capacity increases the throughput and mean delay of the Secondary User packets.

From Figure 4, we infer that as the Factor of Coherent Function grows, SU packets' throughput decreases in the same way with the λ 1, the arrival rate of the packets from PU

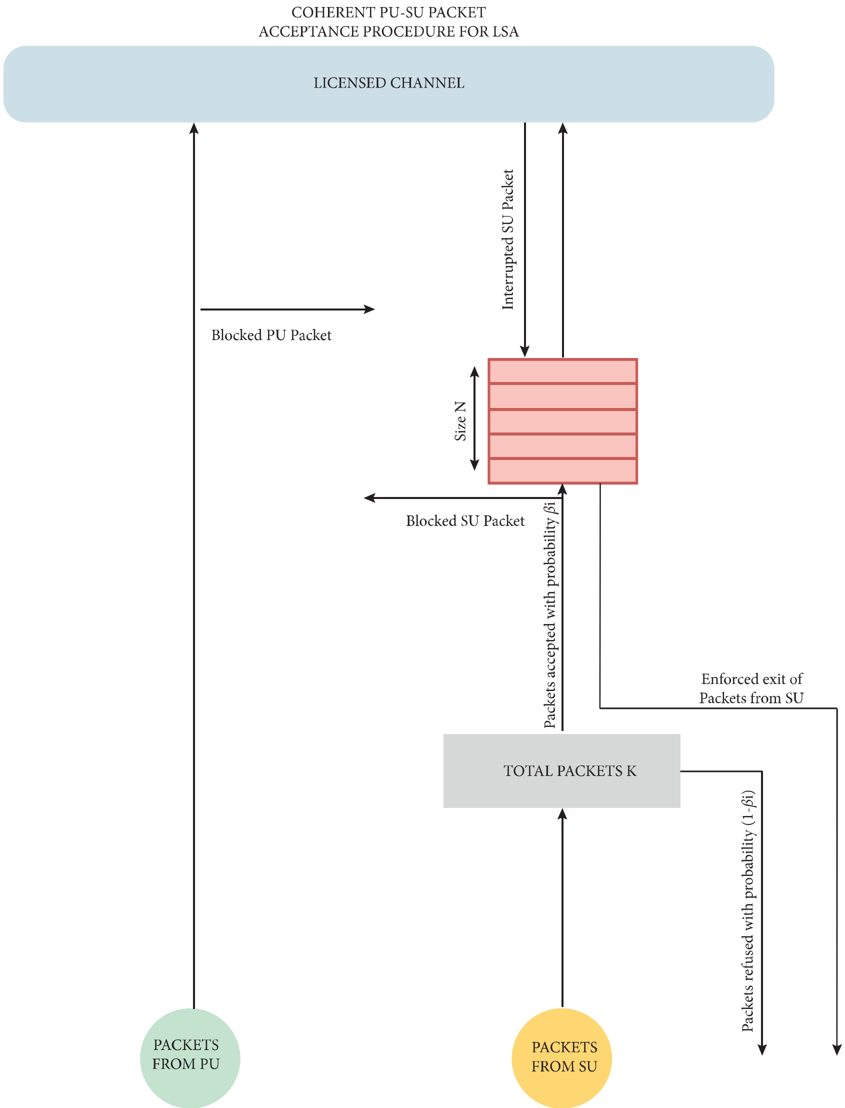


Figure 3: Coherent PU-SU Packet Acceptance Procedure.

and λ 2 of the SU packets. The reason for this is that when the Coherent Factor β increases, the likelihood of a fresh incoming Secondary User packet being acknowledged for the system decreases, resulting in a throughput decrease of the Secondary User packets.

7.3. Performance Optimization Assumptions. Certain assumptions are made that will be used in further optimizations.

(1) In this scenario, it is assumed that an arriving SU packet does not know the current number of packets in the system and is unsure whether the system would allow it. This is a distinct assumption. A SU packet will either enter the system permanently or not at all, depending on its state when received.

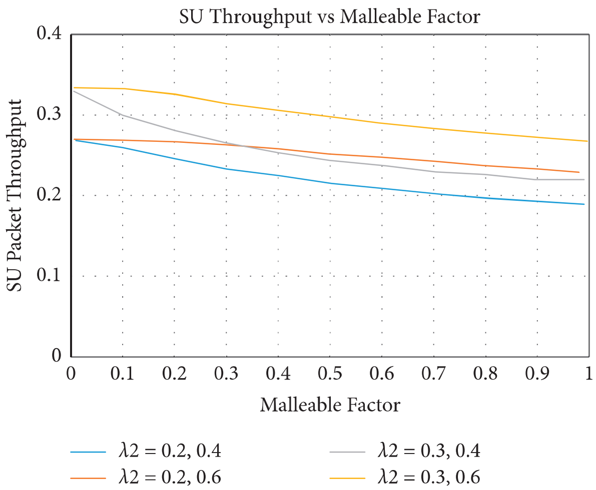


Figure 4: SU throughput versus Malleable factor.

Table 2: SU packet arrival scenarios for analysis.

Scenario 1	Scenario 2	Scenario 3
$\beta(0^+)$	$\beta(T)$	$\beta(0^+) > 0$ and $\beta(0^+) < 0$
Even if no additional SU packets enter the system, the SU packet that attempts to join will get no advantage. Thus, the trial strategy with probability $\eta_e = 0$ is the ideal approach, and there is no alternative optimal strategy [18].	Even if all prospective incoming SU packets attempt to enter the system, they will all get non-negative advantages in this situation. Thus, the trial strategy with probability $\eta_e = 1$ is the ideal approach, and there is no alternative optimal strategy.	If all SU packets enter the system with probability q=1, the Secondary User traffic packet that attempts to join will get a negative net benefit. As a result, q=1 is not the best option. On the other hand, if all SU packets enter the process with probability q=0, the packet from SU that attempts to combine will get a net positive benefit. Thus, q=0 is also not an ideal option. As a result, the best trial probability is ηe = e, where e is determined by solving the equation $\beta(\Gamma) = 0$

(2) Assume R is the reward for successfully transmitting an SU packet. Because the admission of SU packets into the system is not assured, the introduction of a new SU packet is known as a trial. Each trial is charged of fee T. (T R). That is, whenever a SU packet is received in the system, regardless of whether it is effectively delivered, it incurs a cost T for attempting to enter the system.

(3) Γ represents the expected pace of arrival of the SU packages

We would get the following equation for the likelihood of sending an SU packet successfully:

$$\omega(\lambda^2) = \frac{S}{\lambda_2}. \tag{8}$$

S denotes the throughput of Secondary User traffic, and λ_2 denotes the arrival rate of Secondary user traffic. The individualized net positive $\beta(\lambda_2)$ for an SU packet seeking to enter the service is calculated as in eqaution (9):

$$\beta(\lambda_2) = \omega(\lambda_2)(R - T) - (1 - \omega(\lambda^2))T = \omega(\lambda^2)R - T. \tag{9}$$

The unique net gain of an SU packet decreases asymptotically as the rate of SU packet arrival rises. We analyze three scenarios to determine the ideal trial technique for a single SU packet as given in the Table 2:

The term "Collective Net Gain" refers to the following in equation (10):

$$\beta c(\lambda_2) = (\lambda^2)\omega(\lambda^2)R - T. \tag{10}$$

Additionally, we investigate the consistent characteristic of BS(2) in ith numerical findings. We demonstrate through the following figure 5, how the Collective Net Gain varies with the arrival rate 2 of SU packets for various Coherent Factors.

8. Inference From Numberical Results

From Table 3, as PU packets' arrival rate of λ^1 or SU1 packets' arrival rate of λ^{21} rises, both the optimum indivual and collective access rate of effectiveness drop. Since when the arrival rate of Primary User packets or Secondary User1 packets rises, the probability of successfully transmitting

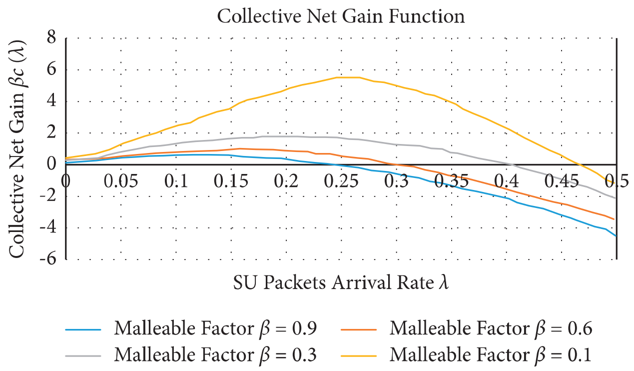


Figure 5: Collective net gain variation.

Secondary User2 packets always decreases. As a consequence, a greater number of Secondary User2 packets will decline in reaching the system. Additionally, when incentive R grows, both the individually optimum and collectively ideal access rates increase. The explanation for this trend is self-evident: if the incentive is bigger, the additional Secondary User2 packets will prefer to contact the system since their interests are higher. The numerical findings for establishing the appropriate additional price are shown in the table. However if the rate of arrival of Primary User or Secondary User1 packets improves, decline in the ideal extra price f is observed. Secondary User2 packets are more likely to drop out of the system if Primary User or Secondary User1 packet rates rise.

8.1. Discussions. We concentrated on the limits imposed by channel dynamics and the varying attitudes of PU and SU during times when research must be enforced. We proposed a new algorithm called Coherent PU-SU Packet Acceptance Algorithm For DSA that deals with Channel Reservation using white space terminology, Spectrum Selection and Allocation to the CR-SU by taking into account the functional outputs of the PU-SU dynamics, which involves the contention of SUs for a defined frequency and the constraints imposed by the PUs on the same frequency. The proposed algorithm offers additional services for 5G CORE such as User behaviour based spectrum resource allocation and Policy based dynamic spectrum management framework.

Table 3: Numerical results of individual and Collective Net Gain of cognitive radio users.

Reward R	PU Arrival Rate λ^1	SU Arrival Rate λ^{21}	Individual Gain λ^g		Collective Net Gain $\beta(\Gamma)$		λ^*	β^*	Additional Price A^P
			MIN	MAX	MIN	MAX			
8	0.09	0.14	0.13	0.14	0.6	0.59	0.07	0.3	1.3
8	0.09	0.19	0.09	0.10	0.4	0.34	0.05	0.2	0.8
11	0.09	0.19	0.12	0.13	0.5	0.45	0.07	0.3	1.4
11	0.14	0.19	0.09	0.10	0.4	0.34	0.05	0.2	1.12

9. Conclusion and Future Work

As a result of the discussion above, we can affirm that the Dynamic Spectrum Management framework used to control spectrum in Cognitive Radio networks is capable of providing very efficient solutions to a variety of cognitive radio difficulties. The use of DSMF in cognitive radio networks is intriguing and represents an unexplored research topic. The Dynamic Spectrum Allocation Framework for the 5G environment did not consider the applications which operate on 5G radio environmental dynamics. To address the Application Specific QoS parameters in a 5G scenario, we need to focus on collaborating our proposed Dynamic Spectrum Management framework with the 5G CORE [19]. A fundamental promise of 5G is to offer stakeholders with significantly faster speeds. 5G networks are projected to operate at higher frequencies, including 3-6GHz, which intersects with satellite C-band, and 26-30GHz, which intersects with satellite Ka-band. To allow CR-enabled spectrum management at these frequencies, CORE functionality must be separated from wireless networks [20,21]. Our proposed algorithm in conjunction with 5G CORE is intended to give the ideal solutions for such a high-capacity 5G network situation. As a future direction customized Access Modification Function (AMF) with our proposed algorithm in 5G-CORE is suggested for better performance.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request (head.research@bluecrestcollege.com).

Conflicts of Interest

“The authors declare that they have no conflicts of interest to report regarding the present study.”

Authors’ Contributions

C.Rajesh Babu: Conceptualization, Data curation, Formal Analysis, Methodology, Software, Writing - original draft; Kadiyala Ramana: Supervision, Writing – review & editing, Project administration, Visualization; R.Jeya: Visualization, Investigation, Formal Analysis, Software; Asadi Srinivasulu: Data Curation, Investigation, Resources, Software, Writing - original draft, Methodology

References

[1] Y. Zhang and Z. Fang, “Dynamic double threshold spectrum sensing algorithm based on block chain,” in *Proceedings of the IEEE 3rd International Conference on Electronic Information Technology and Computer Engineering*, pp. 1090–1095, Xiamen, China, October 2019.

[2] A. Agarwal and R. Gangopadhyay, “Predictive spectrum occupancy probability-based spatio-temporal dynamic channel allocation map for future cognitive wireless networks,” *Transactions on Emerging Telecommunications Technologies*, vol. 29, no. 8, Article ID e3442, 2018.

[3] V. Frascolla, M. M. Butt, N. Marchetti et al., “Dynamic licensed shared access - a new architecture and spectrum allocation techniques,” in *Proceedings of the IEEE Vehicular Technology Conference*, pp. 1–5, Montreal, Canada, September 2016.

[4] A. A. Cheema and S. Salous, “Spectrum occupancy measurements and analysis in 2.4 GHZ WLAN,” *Electronics*, vol. 8, no. 9, p. 1011, 2019.

[5] Y. Liu, J. Peng, J. Kang, A. M. Iliyasu, D. Niyato, and A. A. A. El-Latif, “A secure federated learning framework for 5G networks,” *IEEE Wireless Communications*, vol. 27, no. 4, pp. 24–31, 2020.

[6] R. Shafin, L. Liu, V. Chandrasekhar, H. Chen, J. Reed, and J. C. Zhang, “Artificial intelligence-enabled cellular networks: a critical path to beyond-5G and 6G,” *IEEE Wireless Communications*, vol. 27, no. 2, pp. 212–217, 2020.

[7] H. Li, L. Pei, D. Liao, G. Sun, and D. Xu, “Blockchain meets VANET: an architecture for identity and location privacy protection in VANET,” *Peer-to-Peer Networking and Applications*, vol. 12, no. 5, pp. 1178–1193, 2019.

[8] H. Jiang, T. Wang, and S. Wang, “Multi-agent reinforcement learning for dynamic spectrum access,” in *Proceedings of the IEEE International Conference on Communications*, pp. 1–6, Shanghai, China, May 2019.

[9] P. S. Yawada and M. T. Dong, “Intelligent process of spectrum handoff/mobility in cognitive radio networks,” *Journal of Electrical and Computer Engineering*, vol. 2019, Article ID 7692630, 12 pages, 2019.

[10] Q. Long, Y. Chen, H. Zhang, and X. Lei, “Software defined 5G and 6G networks: a survey,” *Mobile Networks and Applications*, 2019.

[11] D. Minoli and B. Occhiogrosso, “Practical aspects for the integration of 5G networks and IoT applications in smart cities environments,” *Wireless Communications and Mobile Computing*, vol. 2019, Article ID 5710834, 30 pages, 2019.

[12] A. Shakeel, R. Hussain, A. Iqbal, I. L. Khan, Q. U. Hasan, and S. A. Malik, “Spectrum handoff based on imperfect channel state prediction probabilities with collision reduction in

cognitive radio ad hoc networks," *Sensors*, vol. 19, no. 21, p. 4741, 2019.

[13] X. Yu, Y. Cai, W. Li, X. Zhou, and L. Tang, "Research on dynamic spectrum allocation algorithm based on cyclic neural network," *Wireless Communications and Mobile Computing*, vol. 2022, Article ID 7928300, 14 pages, 2022.

[14] F. Beltrán, S. K. Ray, and J. Gutiérrez, "Dynamic Spectrum management in 5G: lessons from technological breakthroughs in unlicensed bands use," *Social-Informatics and Telecommunications Engineering, LNICST*, vol. 245, pp. 250–259, 2018.

[15] M. B. H. Weiss, K. Werbach, D. C. Sicker, and C. E. C. Bastidas, "On the application of blockchains to spectrum management," *IEEE Transactions on Cognitive Communications and Networking*, vol. 5, no. 2, pp. 193–205, 2019.

[16] P. Ren, Y. Wang, and Q. Du, "CAD-MAC: a channel-aggregation diversity based MAC protocol for spectrum and energy efficient cognitive Ad Hoc networks," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 2, pp. 237–250, 2014.

[17] C. R. Babu and B. Amutha, "A novel energy-efficient data aggregation protocol for cognitive radio based wireless multimedia networks," *Peer-to-Peer Networking and Applications*, vol. 14, pp. 2452–2461, 2020.

[18] C. Rajesh Babu and B. Amutha, "Blockchain and extreme learning machine based spectrum management in cognitive radio networks," *Transactions on Emerging Telecommunications Technologies*, pp. 1–13, 2020.

[19] Y. Xu, J. Yu, and R. M. Buehrer, "The application of deep reinforcement learning to distributed spectrum access in dynamic heterogeneous environments with partial observations," *IEEE Transactions on Wireless Communications*, vol. 19, no. 7, pp. 4494–4506, 2020.

[20] E. Z. Tragos, S. Zeadally, A. G. Fragkiadakis, and V. A. Siris, "Spectrum assignment in cognitive radio networks: a comprehensive survey," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1108–1135, 2013.

[21] C. Jothikumar, V. Kadiyala Ramana, D. Chakravarthy, S. Singh, S. Singh, and I. H. Ra, "An efficient routing approach to maximize the lifetime of IoT-based wireless sensor networks in 5G and beyond," *Mobile Information Systems*, vol. 2021, Article ID 9160516, 11 pages, 2021.

sensors

Article

Elite-CAM: An Elite Channel Allocation and Mapping for Policy Engine over Cognitive Radio Technology in 5G

C. Rajesh Babu [1], Amutha Balakrishnan [1], Kadiyala Ramana [2], Saurabh Singh [3] and In-Ho Ra [4,*]

1 Department of CSE, SRM Institute of Science and Technology, Chennai 603203, India; rajeshbc@srmist.edu.in (C.R.B.); amuthab@srmist.edu.in (A.B.)
2 Department of IT, Chaitanya Bharathi Institute Technology, Hyderabad 500075, India; ramana.it01@gmail.com
3 Department of Industrial and System Engineering, Dongguk University, Seoul 04620, Korea; saurabh89@dongguk.edu
4 School of Computer, Information and Communication Engineering, Kunsan National University, Gunsan 54150, Korea
* Correspondence: ihra@kunsan.ac.kr

Abstract: The spectrum allocation in any auctioned wireless service primarily depends upon the necessity and the usage of licensed primary users (PUs) of a certain band of frequencies. These frequencies are utilized by the PUs as per their needs and requirements. When the allocated spectrum is not being utilized in the full efficient manner, the unused spectrum is treated by the PUs as white space without believing much in the concept of spectrum scarcity. There are techniques invented and incorporated by many researchers, such as cognitive radio technology, which involves software-defined radio with reconfigurable antennas tuned to particular frequencies at different times. Cognitive radio (CR) technology realizes the logic of the utility factor of the PUs and the requirements of the secondary users (SU) who are in queue to utilize the unused spectrum, which is the white space. The CR technology is enriched with different frequency allocation engines and with different strategies in different parts of the world, complying with the regulatory standards of the FCC and ITU. Based on the frequency allocation made globally, the existing CR technology understands the nuances of static and dynamic spectrum allocation and also embraces the intelligence in time allocation by scheduling the SUs whenever the PUs are not using the spectrum, and when the PUs pitch in the SUs have to leave the band without time. This paper identifies a few of the research gaps existing in the earlier literature. The behavioral aspects of the PUs and SUs have been analyzed for a period of 90 days with some specific spectrum ranges of usage in India. The communal habits of utilizing the spectrum, not utilizing the spectrum as white space, different time zones, the requisites of the SUs, the necessity of the applications, and the improvement of the utility factor of the entire spectrum have been considered along with static and dynamic spectrum usage, the development of the spectrum policy engine aligned with cooperative and opportunistic spectrum sensing, and access techniques indulging in artificial intelligence (AI). This will lead to fine-tuning the PU and SU channel mapping without being hindered by predefined policies. We identify the cognitive radio transmitter and receiver parameters, and resort to the same in a proposed channel adaption algorithm. We also analyze the white spaces offered by spectrum ranges of VHF, GSM-900, and GSM-1800 by a real-time survey with a spectrum analyzer. The identified parameters and white spaces are mapped with the help of a swotting algorithm. A sample policy has been stated for ISM band 2.4 GHz where such policies can be excited in a policy server. The policy engine is suggested to be configured over the 5G CORE spectrum management function.

Keywords: cognitive radio; wireless communications; 5G CORE; spectrum allocation; 5G communications

Citation: Babu, C.R.; Balakrishnan, A.; Ramana, K.; Singh, S.; Ra, I.-H. Elite-CAM: An Elite Channel Allocation and Mapping for Policy Engine over Cognitive Radio Technology in 5G. *Sensors* **2022**, *22*, 5011. https://doi.org/10.3390/s22135011

Academic Editor: Qammer Hussain Abbasi

Received: 16 May 2022
Accepted: 27 June 2022
Published: 2 July 2022

1. Introduction

The 5G wireless structure desires to support varied service models, different delays, and reliable demands. The 5G broadband wireless communication results in several issues relevant to real-time radio resource management. For example, ultra-reliable transmission demands real-time applications of wireless infrastructure for delay and reliability. The model of an autonomous wireless network with simultaneous service delivered in actual timing can be attained by enhanced changes to the recent deterministic control and optimization of radio resource management models.

Cognitive radio resource management necessitates a close linkage between spectrum management capability and software-defined radio properties, i.e., physical layer-supported modes of operation. The concept behind distributed artificial intelligence (DAI) is to shift from individual to collective behavior in order to overcome the limitations of traditional AI when solving complex problems that require the distribution of intelligence across multiple entities. DAI conducts research in three key areas: distributed problem solving, parallel artificial intelligence, and multi-agent systems. AI and 5G mobile technology have enormous potential to greatly improve profitability, productivity, and efficiency in many areas of business and society, allowing the creation of previously imagined products and services. The combination of AI with 5G will have a significant impact on growing industries like agriculture, healthcare, and education, despite the fact that mainstream applications have not yet materialized. Interest in 5G networks is growing quickly although many mobile operators are still recouping their investments in older network standards. 5G networks are able to transport more data every second and per hertz of the spectrum than previous generations of cellular networks because of improved spectral efficiency. Due to the limited supply and expensive cost of spectrum, this is critical. More concurrent users may be served at a lower cost in order to improve spectral efficiency. Automation and machine-enabled decision-making will transform almost every area of everyday life as a result of breakthroughs in emerging technologies, the rise of artificial intelligence, and higher data speeds. AI and 5G are undeniably the driving forces behind this next technology revolution. However, the vast majority of AI-based 5G use cases are not really 5G use cases since they do not need a technological transition. Apps with a response time of less than one millisecond will stand out because of their decreased latency. In emerging economies, the AI and 5G journey will very certainly require strengthening existing use cases and developing new ones that are not served by present technology. Combining artificial intelligence (AI) with 5G's higher data speeds opens up a slew of new possibilities. The following are among them: Enhancements to mobile communications networks and services. Mobile communication service providers face increasing challenges. In order to deal with the complexity of 5G networks and the billions of Internet of Things (IoT) devices they can support, the deployment of 5G networks is significantly more difficult than in previous generations due to the upgrades required for radio, edge, transport, core, and cloud infrastructure. All of these upgrades should be managed optimally by AI. 5G deployment is being hampered by a lack of available spectrum. The frequency at which 5G operates determines its power and speed. Utilizing frequencies with smaller bandwidths that have advantageous characteristics is one way to get around spectrum limitations. When it comes to 5G, a frequency of 600 MHz may not deteriorate quickly and may still be capable of reaching devices even if they are encased in columns [1].

Here, dynamic spectrum access used by CR tools demands abandoning present pre-programmed strategies. A policy-based model for CR platforms and a new policy reasoning structure for the same have been applied for envisaging policies for the CR system. The essence of CR [2], reconfigures the use of policies from radio devices, and as a result, the adaptive as well as flexible nature of radio systems yields optimized use of spectrum resources. Hence, the strategies are exploited and dynamically modified by shareholders.

The development of a policy-based spectrum allocation system for 5G networks is crucial, thus we built:

1. An elite channel selection system that determines the parameters that must be selected for successful spectrum allocation from an exhaustive collection of parameters.
2. A new parameter mapping technique for translating the needed channel specification to available spectrum holes.

3. A POMDP-enabled policy management system that provides secondary users with service based on pre-mapped channels.

The remaining sections of the paper are organized as follows. Work on 5G networks, channel selection and allocation systems, and policy-based systems are all discussed in Section 2. The policy engine framework is presented in Section 3. We demonstrate the parameter mapping and POMDP-enabled reward methods in Sections 4 and 5, and the findings and testing performance in Section 6. In Section 7, we will also share our findings.

2. Related Work

An exhaustive survey on cognitive radio sensing and allocation has been carried out and the summary of the survey is given in Table 1.

Table 1. Comparison Chart for Related Work.

Reference	Proposed Technique	Focus	Network	Simulation/Frequency Band	Parameters Improved
[1]	A detailed survey on spectrum sensing technique in 5G.	Sensing	Distributed	-	Spectrum trading and leasing Multi-user interference
[2]	A CR-enabled NOMA network capable of wireless data and power transmission at the same time.	Sensing	Distributed	10^6 iterations Monte Carlo simulations	Power outage probability
[3]	One of SenPUI's key issues, detecting and Primary User Interference, is addressed by a unique cognitive radio algorithm.	Sensing	Centralized	Real-time implementation with IEEE 802.15.4	Throughput Primary User Interference
[4]	Models for analyzing and evaluating sharing procedures under a wide variety of situations were provided.	Sharing	Distributed	MATLAB	PU activity SU Arrival
[5]	In cognitive radio networks, the hidden Markov model (HMM) was used for opportunistic spectrum access (OSA) through cooperative spectrum sharing.	Sharing	Distributed	HMM in OSA	Detection Probability
[6]	The CR networks may now be made aware of the needs of unlicensed users through a new method that reduces sensing latency.	Sensing	Distributed	Q-learning for different radio access techniques	Sensing Latency
[7]	Two machine learning (ML) approaches that have been developed to increase spectrum sensing performance are k-nearest neighbors and random forest.	Policy	Distributed	Energy detection using k-NN and RF Algorithms	Energy Detection
[8]	Massive MIMO cognitive radio underlay user selection was proposed using a QoS-aware technique.	Policy	Distributed	Deployed a DNN with MIMO CR	Loss Function Success Rate Average transition time

2.1. Survey on Spectrum Sensing

Ahmad, W.S et al. [3] studied the most current 5G enablers and SS (recent spectrum sharing) technologies in detail. In order to better understand the importance of 5G networks, SS methods were categorized and SS surveys and related research on SS techniques were analyzed. One of the main SS techniques is based on network design, spectrum allocation behavior, and the manner of spectrum access. Cognitive radio technology with SS was also thoroughly investigated in relation to 5G rollout. There are discussions on the existing deployment of SS and CR and the steps to allow efficient 5G progress for a complete examination. Dinh-Thuan Do et al. [4] showed off a unique system design that combined the benefits of both a NOMA and CR network with the Simultaneous Wireless Information and Power Transfer technology. Downlink spectrum sensing and wireless energy harvesting are the primary functions of a radio spectrum unit (RSU). Remote vehicle outage probabilities may be accurately expressed using CR-enabled NOMA systems. Overlay CR-enabled NOMA network performance is severely limited by the transmit power and the number of RSU units designed. Simultaneous gains from energy collection and RSU selection were enabled by the CR, according to the findings. KedirMamo Besher et al. [5] described the Sensing Primary User Interface (SenPUI), a brand new cognitive radio algorithm. During the dynamic inactive phase of communication, an energy scan was conducted. It is therefore suggested that an application packet-based main user ID be used as the foundation for Primary User Interface (PUI) prevention. As a result of this paper's recommendations, the target issues have been decreased by an average of 10–30%. The memory, battery life, and size limits of wireless sensor networks (WSNs) were taken into account while developing and implementing new solutions. Vicent Pla et al. [6] developed a set of Markovian models that may be used to analyze and evaluate spectrum sensing tactics in various scenarios. Markov phase renewal is used to describe the alternating idle and busy times of a channel's occupancy by main users. Secondary users' activity is shown by the length of transmissions, sensing periods, and idle intervals between subsequent sensing periods. These durations are modeled using phase-type distributions, allowing the model to be very flexible. Models for both finite and infinite queues are developed using the Markovian arrival process, which depicts the arrival of secondary users. Additionally, provided was a basic explanation of how an SU transmission restarts after being stopped by a PU action in the suggested models. Many essential performance measures in cognitive radio networks are established by a thorough study of the suggested models. Despite their generality and flexibility, the proposed models may be numerically evaluated.

2.2. Survey on Spectrum Allocation and Algorithms

Yung-Fa Huang et al [7] used the hidden Markov model (HMM) to examine cooperative spectrum allocation in cognitive radio (CR) networks for opportunistic spectrum access (OSA). Slot-by-slot spectrum sensing was performed, assuming that the main channel operates in TDMA mode. Although typical Bayesian updating relies on a single observation, our method concatenates all of the data from secondary users (SUs). In the suggested technique, the channel activity was determined using a predetermined threshold based on belief. In contrast to the simple majority voting method, which did not have a threshold, the suggested approach was more flexible in system functioning. To test the proposed concatenated update approach, a simulation was run using the number of SUs and the majority vote strategy. The simulation indicated that a busy state and an idle state can be reliably detected at around one per SU. METIN OZTURK et al. [8] allowed CRNs to be made aware of unlicensed user needs while also reducing sensing latency. As a consequence, a unique QoS-based optimization phase and two distinct decision methods were presented. Artificial neural networks (ANNs) were used to predict future radio access technology (RAT) traffic loads in various frequency bands (ANNs). Assumptions based on this information have led to two possible solutions. The original approach was only concerned with reducing latency. As a result, it was conceivable to create a virtual wideband (WB) sensing solution by using traffic loads in WB to enable narrowband (NB) sensing. The second strategy, which was based on Q-learning, tried to reduce sensing latency while still meeting other user needs. Random selection's sensing latency was lowered by 59% with the first technique whereas Q-learning helped boost satisfaction by 95.7% with the second method.

2.3. Survey on Learning-Based Allocation

Magorzata Wasilewska et al. [9] recommended the use of k-nearest neighbors and random forest as two machine learning (ML) approaches for improving spectrum sensing performance. By using methods such as energy detection (ED) and data based on energy vectors (EVs), a new 5G radio system was able to make use of the resource blocks made accessible by the previous 4G LTE network. Time, frequency, and location relationships were exploited by the algorithms. If the input training data set is correctly selected, ML algorithms may considerably increase spectrum sensing performance. The benefits and drawbacks of using input data sets with ED assessments and energy values in the actual world were investigated. According to ZHAOYUAN SHI et al. [10], the massive multiple-input and multiple-output that underlays cognitive radio user selection are always QoS-aware. There are two major ways in which a CR may be implemented: the CSI of any cross-network is inaccessible at the secondary base station, but the secondary base station (SBS) has access to the CSI of the cross-channel channel state. Low-complexity algorithms for increasing users while using the least amount of power (IUMP) and methods for decreasing users while using the most amount of power (DUMP) were developed to solve user selection with power allocation. The intractable challenge was addressed using a deep reinforcement learning-based method, enabling the SBS to accomplish effective and intelligent user selection. These algorithms outperform current user selection approaches dramatically in simulations. The neural network was able to rapidly learn the best user selection strategy in an unknown dynamic environment with a high success rate and fast convergence, as the findings also revealed. Arunthavanathan et al. [11] demonstrated that a Markov decision process (MDP) might be used to make decisions based on the current transmissions in the channel. Decisions were made and the effects of interference and waste were evaluated for a range of occupancy rates. Comparative studies of the partially observable Markov decision process (POMDP) and the Markov decision process (MDP) were also conducted. Reinforcement learning and policy-based multi-agent systems were suggested by Jaishanthi et al. [12]. Because the system is powered by AI, the nodes are able to make autonomous choices about which channel to use and how to move between multiple channels because they have all of the necessary information stored in the repository. Maximizing the simulation's use of the spectrum yielded better outcomes.

3. Policy Management for Cognitive Radio

In this section, we explore and understand the fundamentals of policies to be defined for channel allocation and channel mapping. Policies are declarative statements that describe the administrative norms that different organizational bodies follow. There is a business basis for automatic policy management of resources in network and information system administration. Policy management allows you to define organizational goals that can be read and enforced by the network as part of the policy management process [13]. In order to meet administrative objectives and constraints on privacy, allocation of resources, app prioritization, and service quality via the use of automated management rules, it is possible to alter network device settings at runtime. Due to the existing policy's centralized, static character, wireless communication is confronted with two important issues: spectrum shortages and implementation delays. The following capabilities are required to achieve opportunistic spectrum access:

1. Sensing across a broad frequency spectrum and primary channel connection establishment
2. Defining potential opportunities
3. Coordinating the usage of recognized opportunities via communication across devices
4. Defining and enforcing interference-restricting regulations

Enforcing compliance with relevant regulations while capitalizing on highlighted possibilities. Devices may be made to load and alter their policies at any time by using declarative language that is based on formal logic. Instead of relying on hardware, firmware, or device-specific software to implement rules, our solution allows devices to be flexible and respond to policy changes. Decoupling policy formulation from implementation and optimization decisions specific to individual devices is the basis for this policy-based approach to radio operation. For starters, there's less need for certification work. In order to be accredited, a device must have the ability to dynamically load the rules that control its behavior. It is possible to independently validate the policy reasoner and every rule and then test the configuration of devices to ensure that they appropriately understand the outputs of the policy reasoner. In order for a policy reasoner to be effective, it must have a policy language that is sufficiently expressive and methodologies for analyzing and automating policy enforcement.

3.1. Policy Engine

To govern network resource behavior, a computer or process must be capable of consuming and applying machine-readable rules. The computer/process is referred to as the policy engine, as in Figure 1. For events requiring changes to the configuration of a resource, the policy engine is responsible for providing a response. Configuration directives or authorizations tailored to specific network devices are often the results of the policy engine. The policy engine as shown in Figure 1 connects domain-specific goals with the capabilities of the devices they are associated with [14]. In spite of the fact that policy engines must interact with particular vendor devices, they have been seen as general-purpose instruments susceptible to logical thinking through rules. Strategic reasoner (SR) and rule-based reasoner (RBR), the policy engines for cognitive radio, work together to determine if spectrum access options are available that are compatible with the currently active set of regulatory requirements. The SR sends a channel request to the RBR with a transmission plan it has developed based on the behavior data it has collected from the channel's primary users and its present state [15,16]. For the purpose of making sure the allocation plan is in compliance with the rules, the RBR compares channel requests to those policies. The RBR provides a denial-of-service response to the SR if it detects that a portion of configuration states in the channel request is erroneous. Constraints for opportunistic access are the adjusted parameters derived by the RBR. The basic reasoning challenge in policy-enabled cognitive radio is to determine whether a channel request should be processed or refused based on a given set of rules and allocation techniques, and, if denied, to calculate the limitations for opportunistic access to the channel.

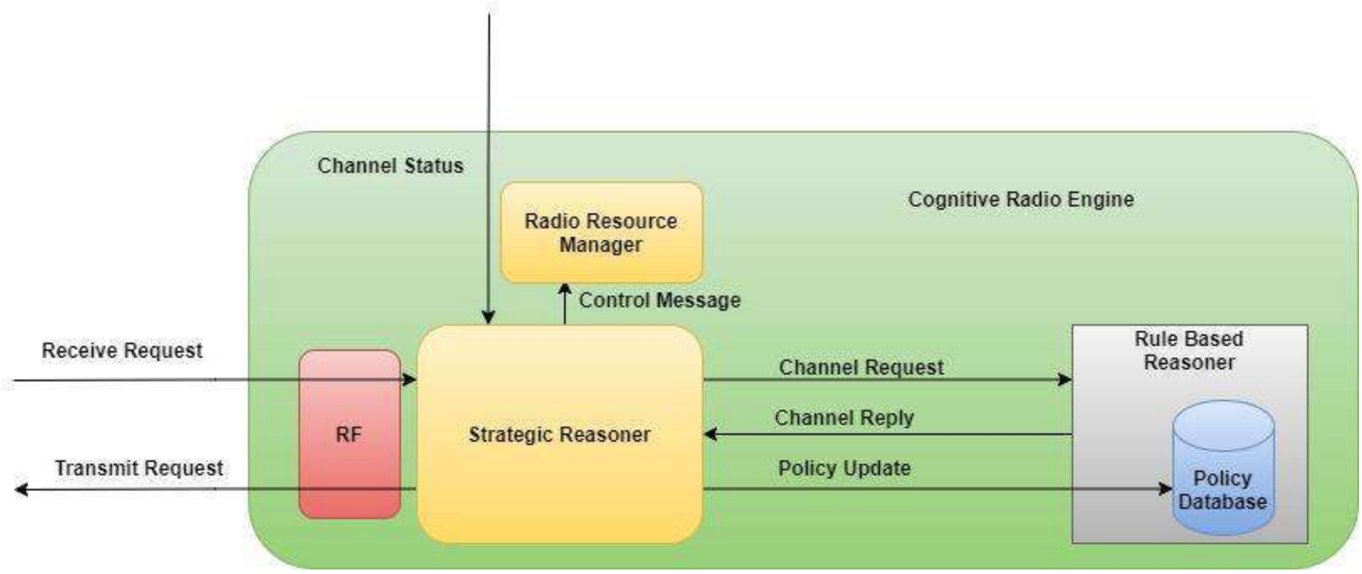


Figure 1. Policy Engine.

3.2. Assumptions for Elite Channel Allocation Algorithm

- It is impossible for hardcode regulations allowing secondary users to use available spectrum holes across several radio bands that fluctuate in frequency and location.
- The spectrum occupancy denoted by Γ by primary users, is given by [17,18]:

$$\Gamma = (1/\mathrm{KN}) \sum_{\mathrm{K}=1}^{\mathrm{K}} \sum_{\mathrm{N}=1}^{\mathrm{N}} \Gamma \mathrm{f}(\mathrm{n}), \mathrm{t}(\mathrm{k}) \quad (1)$$

- where total number of operating frequencies in a band is denoted by N and K denotes the number of total samples associated with each point of frequency fb [(99 kHz–3 GHz), (100 MHz–1000 MHz) (50 MHz–4400 MHz)]
- The anticipated spectrum characteristic needs for secondary users (bandwidth Ba, power dB req, time access period, SNR) for various types of spectral intensity, geographic location, and regulatory authority [11]

$$R(x) = \sum_{n=0}^{n-1}\sum_{m=0}^{m-1}\left(ch_{mn}B_{nm}\log_2\frac{Pow_{mn}}{\lambda}\cdot\frac{PoG_{mn}}{\mu}\right) \tag{2}$$

- where R(x) denotes SU utility function,
- ch_{mn} channel m utilized by node n
- B_{mn} bandwidth of channel m utilized by node n
- Pow_{mn} power required in dB for channel m utilized by node n
- PoG_{mn} power gain in dB for all fading and path loss over the channel
- λ and μ denotes SNR and minimal threshold power required to transmit in the channel.
- The bandwidth available is divided into a predetermined number of channel bands.
- Each channel is optimal.
- At any one time, each channel must serve a single user (primary user or secondary user).
- Secondary user time sensing and synchronization are intended to occur at guard band frequency.
- At guard band frequency, time sensing and hand off to the primary user from the secondary user are anticipated.
- Each user's entrance is a distinct event that occurs independently of subsequent users. Distribution of Poisson.
- The number of available channels is more than the number of PUs.
- Primary users are authorized users who take precedence over secondary users.

Figure 2 and Table 2 summarize the set of exhaustive cognitive radio parameters for transmitters, receivers, and channels. The parameters [19,20] are considered by the swotting algorithm which is mentioned in Algorithm 1, for doing parameter mapping before channels are mapped.

Table 2. Cognitive Radio Parameters.

PARAMETERS		
Channel	**Transmitter**	**Receiver**
Doppler Frequency f^d (0.0001)	ASCII message maximum size Maxmsglen (100)	Equalizer feedback tap number N1
Multipath-Link Information md_p (10, 0.2, 0.001, 0.010)	Message + control message Payload (128 bits)	Equalizer feedforward tap number N2
Variance of Quadrature noise varnoise (0.01)	Hamming code index M (3)	N1 + N2 = md_p length
Distortion Flag distort flag (0 for no distortion and 1 for noise)	Samples/Channel symbol Fracspace (2)	Fracspace (2)
Coherence Time Cohtime (100,000)	PacketID ACKID Power in dBm maxpower	Cohtime Memory

Algorithm 1. Elite-CAM—Algorithm for Cognitive Radio Channel Selection and Allocation

for All Channels,
    SetIdle
        {
If channel Ch is IDLE at time T_a
    If authorized
        Check Primary User back-off time B_{ta}
        If available IDLE for Time $T_{b\text{-}a}$, Mark Channel specification Γ, and release for Secondary User
        Channel Specification as in Equation (1)

$$\Gamma = (1/KN)\sum_{K=1}^{K}\sum_{N=1}^{N}\Gamma f(n), t(k)$$

            N—Number of frequency points in the band
            K—Number of time samples for each frequency point
} Repeat SetIdle for all channels;

For all Idle Channels
{
AllocateSU
Check SU Queue
    If unauthorized
        Transfer Channel specifications
        $\Gamma_1, \Gamma_2 \ldots \Gamma_N$ and
        Idle Time Slots—$T_{b-a}(\Gamma_1)$, $T_{b-a}(\Gamma_2)$, … $T_{b-a}(\Gamma_N)$,
        to SU
    Identify the QoS requests from Secondary User
For

$$R(SU1) = \sum_{n=0}^{n-1}\sum_{m=0}^{m-1}\left(ch_{mn}B_{nm}\log_2\frac{Pow_{mn}}{\lambda}\cdot\frac{PoG_{mn}}{\mu}\right)$$

If Γ_x satisfies the requirements R(SUy)
Allocate
Ch_x to SUy with B_x Bandwidth of channel m utilized by node y, $Pow_{x,}$ the Power required in dB for channel m utilized by node y, PoG_x Power Gain in dB for all fading and path loss over the channel

} Repeat AllocateSU for all Idle Channels
When PU arrives,
    If the channel is occupied by SUz,
    Pre-empt SUz from Channel Chz and release for PU
For SUz, call AllocateSU and identify spectrum hole for re-allocation
end

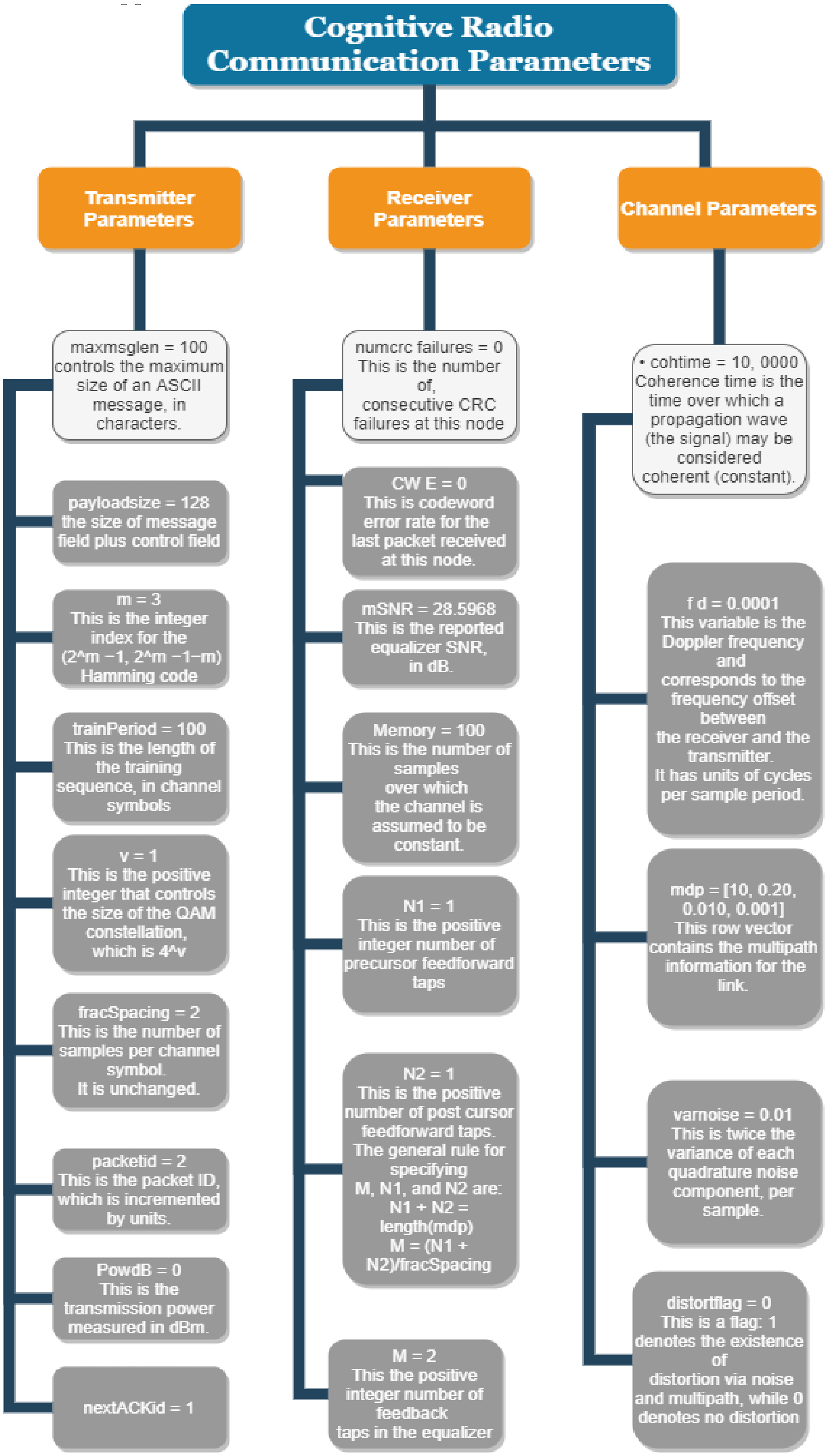


Figure 2. Cognitive Radio Parameters.

4. Elite Mapping—Swotting Algorithm for Parameter Mapping Repository

The swotting algorithm given in Algorithm 2, is a reinforcement Q-learning algorithm that determines decision-making policies without comprehensive modeling of the radio environment. This suggests that Q-learning describes and improves the performance measures of interest, rather than addressing network performance-related aspects such as wireless channel state and mobility, for instance. It has four parameters: states, action a, probabilistic transition function Qr, r, and reward function Sb, b. The state may represent internal events occurring inside the agent, such as the size of the instantaneous queue, or external phenomena occurring outside the agent, such as the agent's use of the wireless medium. The reward function shows the system's response to the quality of its actions, and as a result, the system acquires experience. At time t, the agent observes the environment's state st. The action cat is chosen in accordance with the state cst. According to cat and Qr, r, the environment changes the state and reward function rt = R(cst, cat) obtained as a consequence of the change is recorded and supplied back to the agent. The optimum Q-value is a measure defined for each state-action pair in the process of determining the best policy s, and it is assessed in the following manner [21,22].

$$\mathrm{SWQ}(a,s) = \mathrm{E}\{\mathrm{P}(a,s)\} + \gamma \sum_{a \in s} \mathrm{P}_{a,a'} \max_{c \in A} \mathrm{SWQ}(a',c) \tag{3}$$

where a, s—available state sets,

γ—Factor for discount

The ideal policy can be identified using the following criteria [12]:

$$\pi s = \arg\max a \in A\ \mathrm{SWQ}(a,s) \tag{4}$$

The elite swotting algorithm finds SWQ (a,s) in a recursive manner using the following rule [19]

$$\mathrm{SWQ}(a,s) = (1-\alpha)\mathrm{SWQ}(a,s) + \alpha(\gamma\,(a,s) + \gamma \max_{b \in A} \mathrm{SWQ}(a',b)) \tag{5}$$

Appropriate actions are rewarded with a rise in their Q-value. In comparison, an incorrect action results in a penalty and a fall in the Q-value. The Q-value is stored in a size-dependent two-dimensional lookup Q-table. It was shown that when each state-action pair is visited infinitely often, this updated algorithm converges to the optimum Q-value. Thus, learning is composed of two processes that govern the selection of actions: exploitation and exploration [23]. Exploitation is the process of choosing the optimum action based on previously discovered optimal policies, whereas exploration is the process of randomly picking nonoptimal actions and finding new ones. The exploration rate E determines the balance between exploration and exploitation.

Algorithm 2. Elite Mapping Swotting Algorithm for Parameter Mapping

Prerequisites: present state cs(t), previous state ps(x), and F(t)
Ascertain that the Swotting algorithm selected produces the maximum possible output. OP
Training: given the state of the network cs(t);
Probabilistic exploration γ;
Choose one action at random;
New Update $\mathrm{UP}(t) = \{b | W(cs(t), b) = 1\}$ for cs(t);
Probabilistic Exploitation is $1 - \gamma$;
Choose α records UP″(CS′(ps(x),F(t)) from actions F in accordance with CS′(ps(x),F(t))
Resolve Re(cs(t), r) in accordance with next r(x) and fill UP^{R}(cs(t));
if $(y * \text{ exist} = \max_y(y|cs(y) \in$ UP″(CS′(cs(t)),F(t)) ∩ UP′(cs(t)) then
Choose the action r(x);
Else if
Choose action from UP^{R}(cs(t));
end if
do update UP(cs(t), r)
do
repeat SWQ;
while (t = t + 1 and s = s.t + 1)

Parameter Mapping Repository (PMR)

Assisting the cognitive network, the PMR accepts sensory input from the communication layer, selects suitable channels [24,25], and fine-tunes transmission parameters to enhance the cognitive network's performance for each generic interface; the PMR is responsible for storing and processing sensory data that has been sent to it as in Figure 3. In addition, it prepares and sends various transmission parameter setting options through the corresponding interfaces along the stack levels. The parameter mapper is responsible for making the final choice of temporary data storage repository for parameter mapper modules' data flow. Following are the modules that make up the parameter mapper framework.

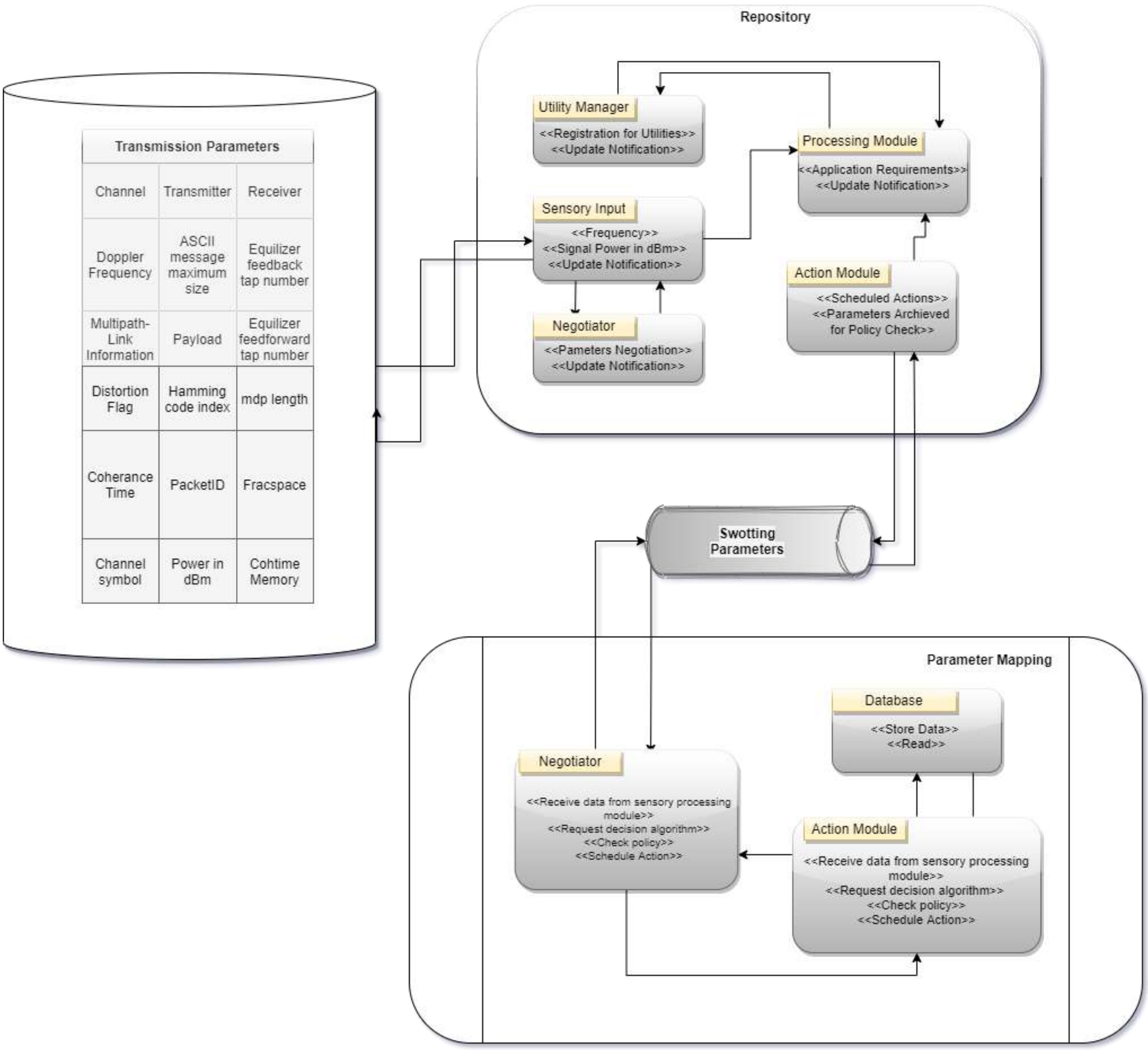


Figure 3. Swotting for cognitive radio parameters.

Negotiator Module: It serves as a coordinator for the decision-making layer's duties. The decision module receives sensory data from the repository and uses it to make a choice. Negotiators may also use policy queries to check whether the transmission parameters they've selected are in compliance with any constraints as in Table 3, placed by the policy layer. Following receipt of the policy-layer answer, the negotiator communicates the choice to the repository's action module.

Table 3. Cognitive Channel Allocation Policies.

Cognitive Channel Allocation Policies				
Scenario	Pu is occupying the spectrum. SUs are in queue.	The spectrum is free, PU is idle and not utilizing the spectrum. SU is in need.	PU has left the spectrum free, No. of the competing SUs is more for the same spectrum.	PU has been left with a fading channel. Spectral density is low. The secondary user arrival rate is Poisson.
Policy	Wait	Allocate	Assign by Rank	Random wait
Reason	As the history of the PU activity and the requirement of the SU is known, mapping is already done, the mapping table is verified, and then policy 1 is triggered.	The spectrum is freed by PU. SU is in need of the spectrum. Spectral density is good. SU's requirement lies within the availability of the spectrum. SU's Qos is also satisfied by the network and spectrum parameters.	SUs are prioritized for their effective utilization of the spectrum and their active participation without wasting the spectrum. Their spectral density and utilization factor are the criteria for the decision.	The spectrum analyzer has to detect the quality of the channel by sending a few random packets at different time intervals.

Resolver Module: It accepts requests from the negotiator and performs the following activities. Determine whether there is a greater priority in the case that many channel requests are pending. After the spectrum sensing data from the spectrum sniffer is gathered, the process of allocating channels starts. As a final step, a decision module runs the optimization process and passes the policy-validated transmission parameters to the action broker.

5. PMDP Model for Channel State and Reward

In a Markov decision process (MDP), [26] decisions are made based on the most current state data in a discrete-time stochastic management process as in Figure 4. Current state is S(t) ∈ S, where S implies the complete state space. In real-time application, the recent states are adaptable to conditions, and partially observable MDP (POMDP) has been applied for computing the decision policy according to the partially accessible data as well as observations from the physical atmosphere. Generally, optimization models are applied for gaining the solution for POMDP-related issues.

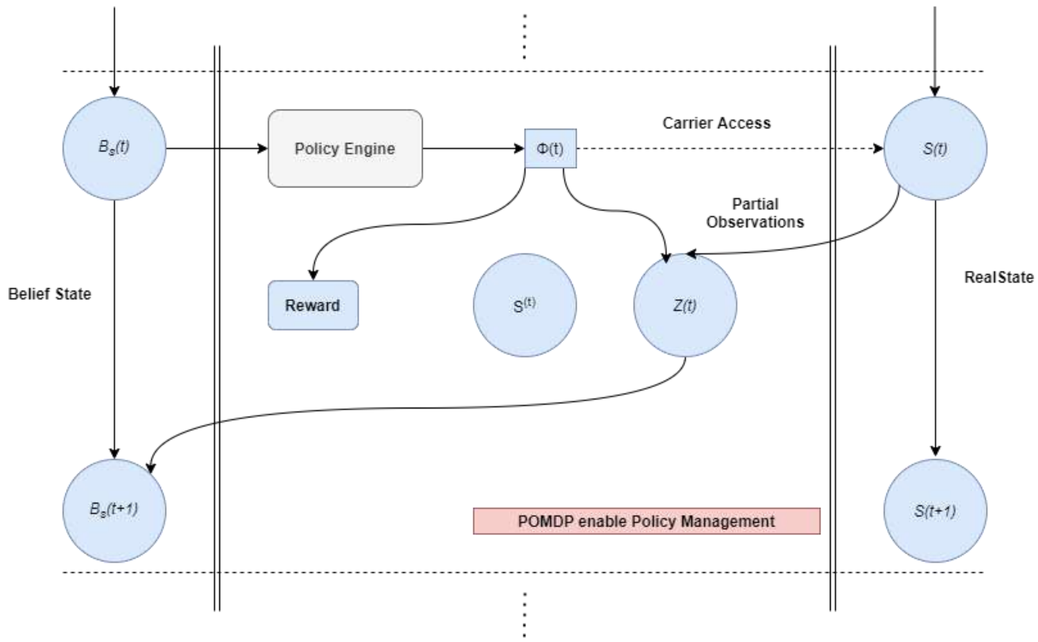


Figure 4. POMDP-enabled policy management.

Consider that $c_r^1(i)(t)$ implies the condition of rth channel of ith carrier in time slot t, the status of complete system in tth time slot can be expressed as [13]

$$S(t) = \{C(1)(t), \ldots, C(i)(t), \ldots, C(N)(t)\}, \forall S(t) \in S \text{ and} \tag{6}$$

$$C(i)(t) = \{c(i)1(t), \ldots, c(i)r(t), \ldots, c(i)R(i)(t)\}, \forall c(i)r(t) \in \{0, 1\} \tag{7}$$

where $C(i)(t)$ defines the channel set of ith carrier from time slot t. $c(i)r(t) = 0$ refers the constant state and $c(i)r(t) = 1$ indicates the busy state, while [13]

$$S = \{S1, \ldots, SM\} \text{ with } M = 2\sum N\ i = 1\ R(i) \tag{8}$$

No states can be observed directly inside the POMDP infrastructure as in Figure 4; the collection of observations $Z(t) \in Z$ is required to render the representation of the physical state. Then, observations are assumed with probabilistic behavior in which the observation function O is depicted as a probability distribution across the feasible observations $Z(t)$, in action $\Phi(t)$, and final states $S(t)$, which is represented by [27],

$$O(S(t), \Phi(t), Z(t)) = Pr(Z(t)|\Phi(t), S(t)), \forall Z(t) \in Z, \Phi(t) \in \mathrm{A}, \tag{9}$$

where $\Phi(t)$ denotes the action set at t-th time slot. The attribute $\Phi(t) = \{\Phi(1)(t), \Phi(2)(t), \ldots, \Phi(N)(t)\}$ implies the action selected by POMDP development, and $S(t)$ refers to the final state after implementing $\Phi(t)$. It is notable that $\Phi(i)(t) \in \{0, 1\}$ where 0 and 1 signify that carrier i is not applicable and accessed at time slot t. As the state transition and observation function are possible, it requires immediate reward $W(S(t), \Phi(t))$ which can be accomplished by,

$$W(S(t), \Phi(t)) = \sum_{S(t) \in S} \Gamma(S(t), S(t)) \sum Z(Z\ \ O(S(t), \Phi(t), Z(t))\ )C(\Phi(t), S(t)) \tag{10}$$

POMDP-Enabled Policy-Based Spectrum Allocation (PPBSA)

As a result of the implementation of the suggested POMDP in LTE-A small cells, the channel status indicator on the dispersed spectrum is evaluated considerably by sensing the frequency carriers under consideration [14,21]. To accomplish high system throughput, a contention approach has been allocated in the decision-making process mentioned in Algorithm 3, to access alternate systems-distributed channels. The two influential factors in selecting target-shared channels are channel state indicator (CSI), and number of contenders (NOCs). Here, the NOCs define the overall count of determined SUs and access to a similar spectrum. As a result, taking NOCs into account lowers SU collisions on the shared spectrum. The power and channel assignments are made to enable the small cell-enhanced node B (SeNB) to function as numerous UEs [28,29] without interference from small cells as in Figure 5.

Algorithm 3. Channel Selection and Optimal Power Allocation Policy

1: **Input**: Channel State Identifier, Channel Occupancy Predictor, Belief State and Number of SUs
2: **Output**: Channel Selected, Updated CSI
3: Set Wmax = 0
4: **for i = 1 to N do**
5: Set Optimal carrier $\Phi(i)(t) = 1$ and $\Phi(j)(t) = 0$, $\forall j \in \{1, \ldots, N\}$, $j \neq i$, where $\Phi(t) = \{\Phi(1)(t), \Phi(2)(t), \ldots, \Phi(N)(t)\}$
6: Derive Optimal Power allocation $P(i) \times (t)$ and $A(i) \times (t)$
7: Compute Expected Idle time of channel and probability that small cell reserves channel r)
8: Compute reward and set Wmax = max{Wmax, W(i) ($\Phi(i)(t)$, $P(i) \times (t)$, $A(i) \times (t)$)}
9: **if** Wmax = W(i) ($\Phi(i)(t)$, $P(i) \times (t)$, $A(i) \times (t)$) then
10: Set Optimal carrier $\Phi \times (t) = \Phi(t)$
11: **end if**
12: **end for**
13: Set update for Channel Selection $A \times (t) = \{A(1) \times (t), A(2) \times (t), \ldots, A(N) \times (t)\}$

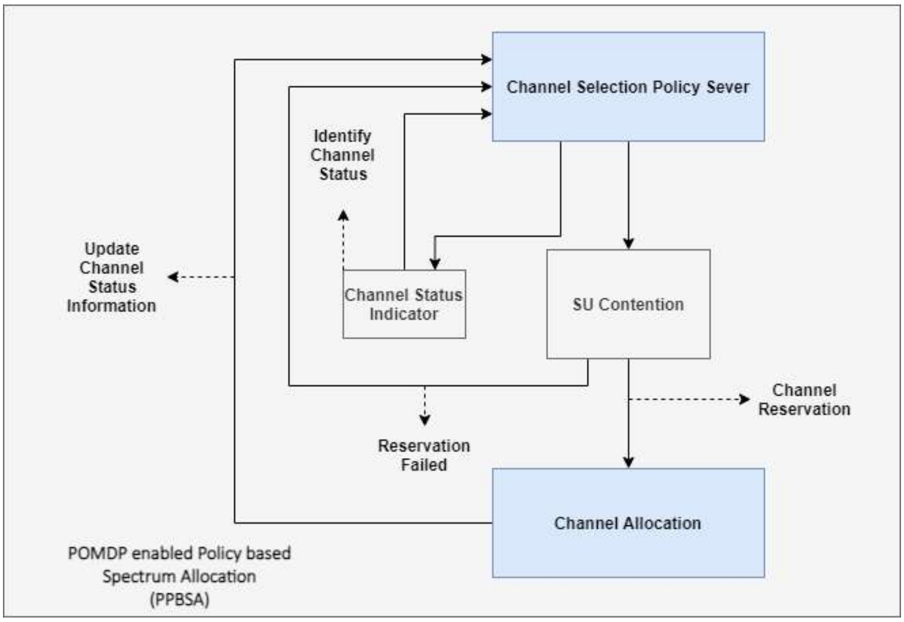


Figure 5. POMDP-enabled policy-based spectrum allocation (PPBSA).

6. Results

6.1. Spectrum Analyzer Specification

The spectrum analyzer has been configured with the parameter setting as specified in Table 4 for the use cases.

Table 4. Spectrum analyzer specification.

Parameters	Range Setup 1	Range Setup 2	Range Setup 3
Spectrum Frequency	99 kHz–3 GHz	100 MHz–1000 MHz (UHF) GSM 900 MHz 2.4–2.5 GHz (ISM band)	50 MHz–4400 MHz [GSM 900, GSM 1800]
Duration	12 h [6 a.m.–6 p.m.]	24 h	24 h
Instance	60	90	120
Sweep Time [Frequency Span = 0 Hz]	1 millisecond–100 s	Auto	Auto
Sweep Time [Frequency Span > 0 Hz]	20 milliseconds–1000 s	Auto	Auto
Bandwidth for Video	10 Hz–1 MHz	100 kHz	10 MHz
Interface	RS232	RS232	RS232
Bandwidth Resolution	10 Hz–1 MHz	100 kHz	10 MHz

6.2. USE CASE>>GSM [941.2 MHz–949.6 MHz]

The data are analyzed in Figures 6 and 7 using several sample windows, each with 1024 discrete points. To reduce overlap between subsequent observation vectors, a block of 512 points was eliminated between each pair. The resolution bandwidth for each window is of value 10 MHz and is balanced on 942 MHZ, resulting in 9.7 kHz guard space between each pair of points captured every 10 µs. Parallel to our allocation method, we did an analytical investigation utilizing the open-source project "bit-gsm", which allowed us to get an a priori understanding of the occupied PU considered territory.

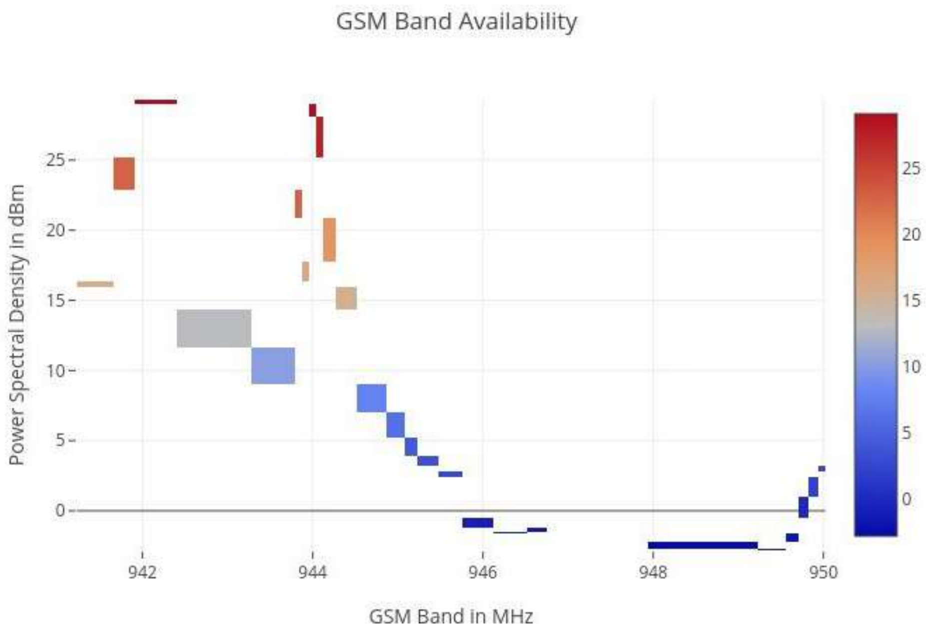


Figure 6. GSM 900 Band Availability.

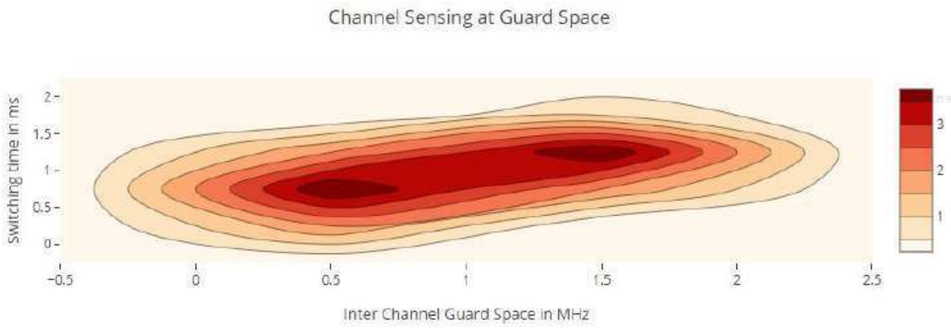


Figure 7. Channel switching time over guard space.

With the Hanning window weighted by 4096 non-overlapping windows of 1024 received samples, the Welch technique was used to estimate the power spectral density at each acquisition point shown in the graph below. The GSM channel's core frequency is shown by the vertical red lines. The PSD distribution implies tiny idle bands between 940.2 and 941.2 MHz, as well as a 4 kHz idle band for guard channels between 945.6 and 949.6 MHz.

When the number of SUs rises, the system's total utility drops [30,31]; similarly, when the number of functioning PUs increases, the system's average SU arrival declines. To minimize searching delays, it is ideal to do a limited number of channel searches before moving on to the next available channel. When the inter-sensing interval is minimal, the switching rate of the channels stays high [29,30]. This is because sensing occurs in a relatively small period of time, resulting in significant overhead and increased channel switching. As a result, detecting a signal with certainty is difficult, necessitating the addition of additional channels. Knowing which channels are most likely to be idle reduces channel switching and search times. Our channel selection algorithm picks channels for sensing intelligently. This significantly minimizes the number of channel shifts required, since only those channels with the greatest probability of being free are chosen for sensing. Channel switching diminishes as the inter-sensing interval increases, as seen in Figure 8. When sensing is carried out over an extended period of time, the results are more certain and likely to be accurate. What is critical in this case is to identify idle channels. If only idle channels are chosen, there is no need to move between several channels, resulting in fewer instances of channel switching [32,33]. The length of the sensing operation has a direct effect on network performance. A channel that has been sensed for a prolonged period of time slows data transmission, since data transmission may occur only once sensing is complete.

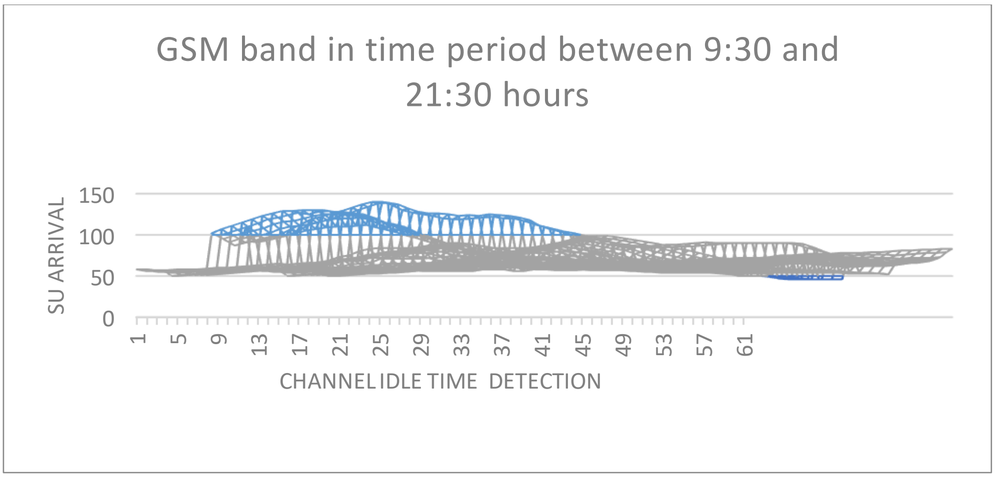


Figure 8. SU arrival over GSM 900.

6.3. *USE CASE>>VHF [180 MHz–270 MHz]:*

Using a resolution of 30 kHz, the spectrum analyzer was able to identify fixed voltages as in Figure 9. Three 90 MHz segments with 3001 continuous channels were used to measure the whole spectrum from DC to 300 MHz. Typical VHF low-end broadcasts often have a bandwidth of 30 kHz, which is wider than 30 kHz. The result for the 180–270 MHz frequency range is shown in the graph below. There is a clear depiction of broadcast television channel 10 in this image (193.25 MHz, 196.83 MHz, and 197.75 MHz). There is a strong correlation between the mean and maximum PSDs, which indicates that the channels that are most often utilized have a high frequency. Following that, we make an effort to quantify spectral occupancy. Notably, huge sections of the spectrum are virtually always occupied by extremely powerful, very broadband transmissions—particularly, broadcast television and FM stations [34]. Clearly, incorporating such sections of the spectrum in computations of spectral occupancy is of little use.

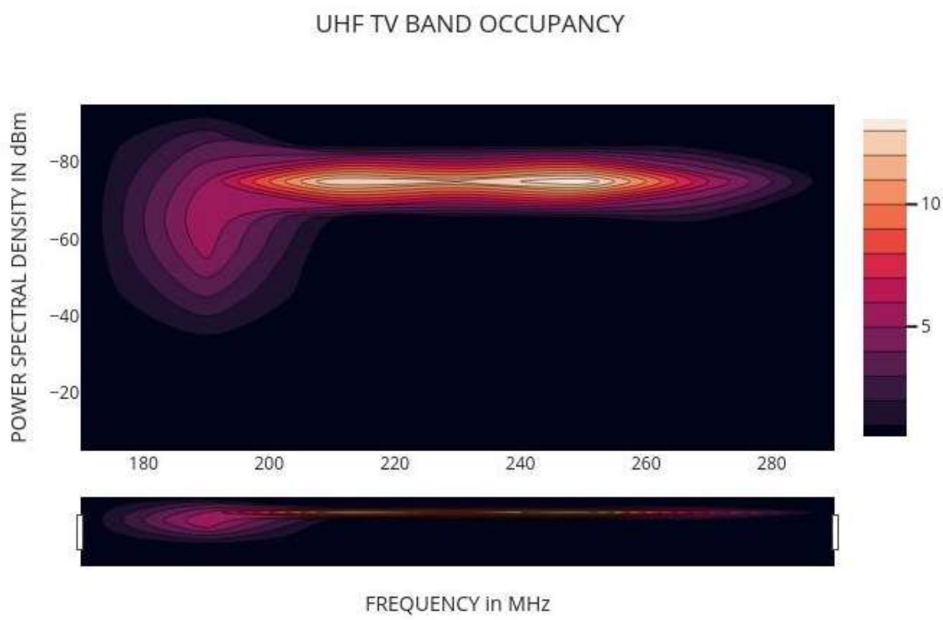


Figure 9. UHF band availability.

It should be emphasized that the spectrum occupancy in the observations presented here is actually sparse. There is virtually little indication of sustained activity greater than 87 dBm per 30 kHz. For a rural environment, the frequency sample size taken is 25–90 MHz. PSD value of 100 dBm is a good sample. Frequency-agile operation in the metropolitan area may be accomplished by using 31 to 60 MHz and 141 to 180 MHz as feasible possibilities. These bands had 40 and 67 openings with bandwidths ranging from 29 kHz to more than 2.9 MHz. This band's statistics may be affected by improvements in measurement sensitivity, which is expected. Furthermore, it should be remembered that broadcast television stations operating on many channels may be dominant in particular locations. Nonetheless, these are generally favorable results for the future of cognitive radio with frequency agility. The measurements provided are imperfect, and further effort will be required to provide a credible verdict. Next generation measurements should focus on long-term (weeks) observation at several sites with a time-frequency resolution of 1kHz in 1 msec. An evaluation of the feasibility and expected capacity of cognitive radio networks operating in the unallocated spectrum without interfering with major users must meet these severe conditions [35].

The following are the important stages in the measuring process:

- As though they were broadcasting alone, each linked SU transmits at the highest allowable power associated with the appropriate channel.
- If the television reception is compromised, reduce the transmit power of one or more secondary users until no more visual distortion is seen.
- Calculate the weighted aggregate power by recording the transmit power of each SU.

As anticipated, the weighted aggregate power is always equal to or less than the intended television signal level; hence, the suggested model is validated. The following graphic illustrates how, as the number of concurrently accessible channels rises, the maximum acceptable received interference in each adjoining channel decreases. One consequence of the accumulative impact of multichannel interference is a reduction in the amount of available TV white space for many secondary users [36]. The collective impact of neighboring channel interference is readily apparent, where a safe area is defined as the set of interference values that the SUs may create without interfering with TV reception. When the aggregate impact of adjoining channel interference is considered, we can plainly observe how the safe zone diminishes as in Figure 10.

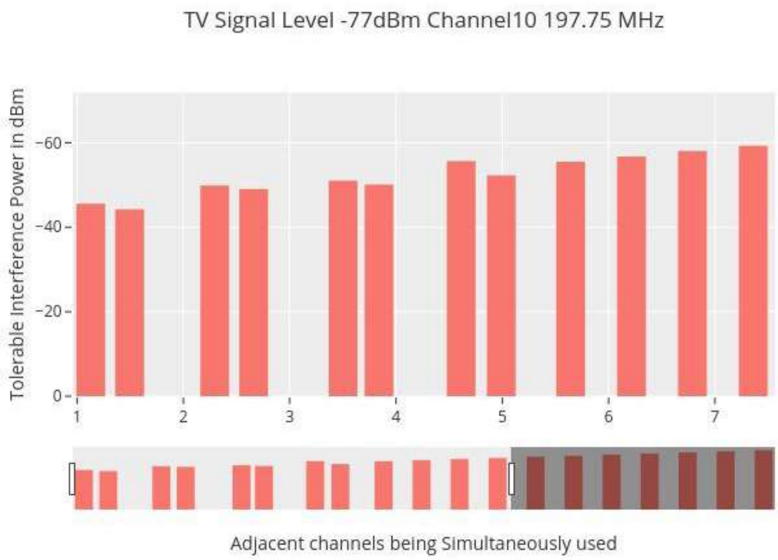


Figure 10. Adjacent channel interference.

Contrary to industrialized nations, a significant percentage of the TV band spectrum remains underutilized. Even with conservative criteria, the findings indicate that all 15 channels are free in at least 56.27% of the region. The average amount of accessible television white space was determined using two methods: first, from a protection and pollution standpoint, and second, from the perspective of FCC restrictions [36]. Both approaches indicated that the average accessible TV white space in the UHF television spectrum is more than 100 MHz as in Figure 11. A method for reassigning TV transmitter frequencies was devised in order to free up unneeded spectra. Eight television channels (about 64 MHz) were determined to be adequate to cover the current UHF/VHF television bands. In the future, we want to investigate appropriate laws for the TV white space to allow inexpensive broadband access.

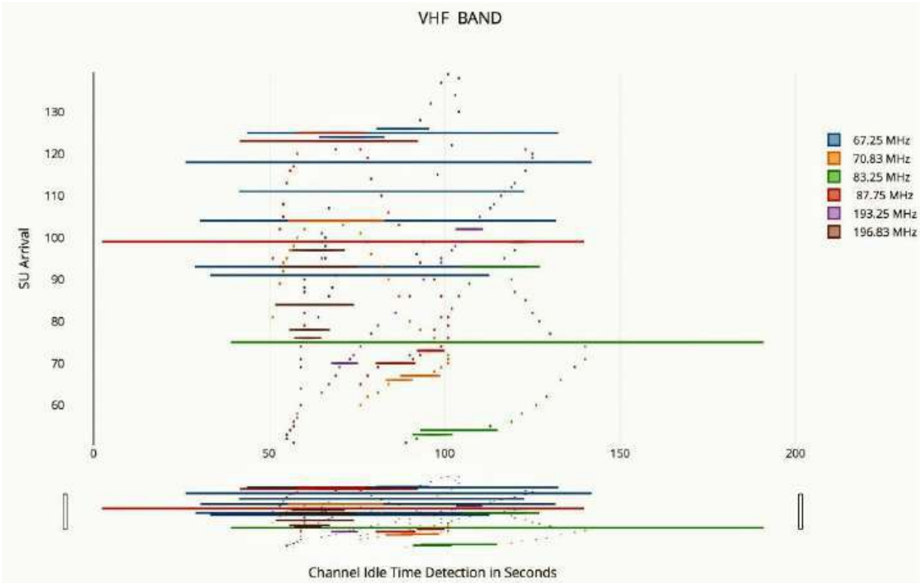


Figure 11. SU arrival over VHF band.

It is pointed out that the overhead data rate is reduced by enhancing the sensing time. The reason for data rate limitation is due to the limited transmission time which is occupied by prolonged sensing duration. As a result, it demonstrates that the sensing time is described as 1 ms. In addition, it is observed, as, in Figure 12, that our projected PPBSA approach performs better than alternate models under sparse as well as dense scenarios.

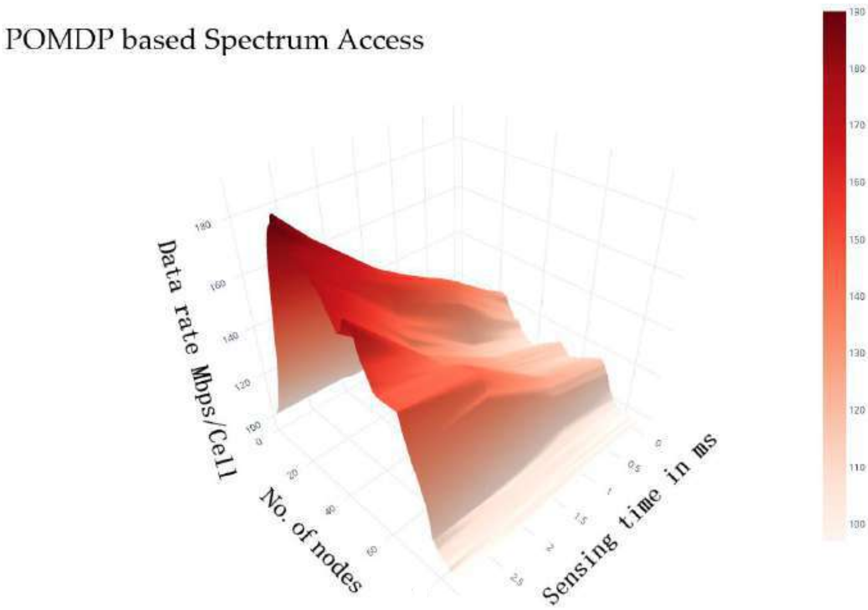


Figure 12. POMDP-based spectrum access by SU.

6.4. Inferences

Two case studies have been analyzed for frequencies GSM 900 and the TV VHF band. Both the case studies have taught that the pattern of usage of the primary user, the guard space analysis, the pattern of the secondary user, and the occupancy criteria are considered to be the same. A POMDP model has been developed to understand the usability of spectral white space [34] by the secondary user and they are prioritized in a proposed swotting model database. The prioritized secondary users are assigned ranking through the rewards generated by the models. An optimal mechanism of spectrum transferred from the primary to the secondary user and the handoff from secondary to primary user need arises and is experimented with the statistical analysis as in Figure 13.

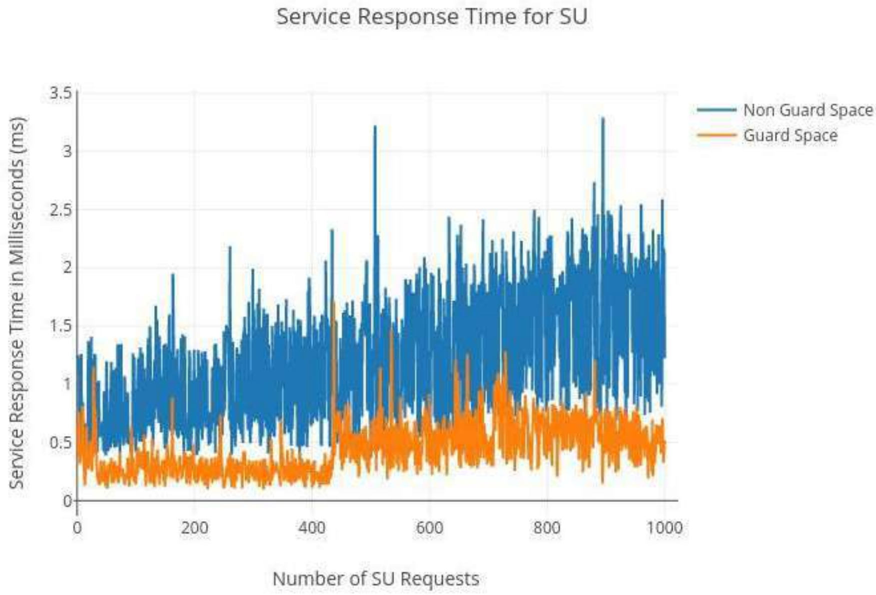


Figure 13. Service response time.

7. Conclusions

Various real-time bands were analyzed and the proposed elite channel allocation and mapping algorithms were tested to offer service to the secondary users requesting a single service. The elite-CAM is enhanced as policy and the policies are imposed on a policy engine. The policy-based spectrum allocation has been developed and implemented for ranking and rewarding the static environment. The policy engine is suggested for configuration with the spectrum management function of 5G CORE specification. The continuously changing behavior of the primary users and the users who do not fit in the ranking mechanism has not been experimented with. Therefore, it is not preferable that cognitive radio users once tested with a particular model prolong for a constant period of time. For a fast-changing dynamic environment, the spectrum allocation strategy should be fine-tuned. In addition, the handoff will be another major issue when the PU arrives back to the channel, and as future work, addressing dynamic spectrum allocation in 5G and beyond 5G networks is suggested.

Author Contributions: Conceptualization: C.R.B., A.B. and K.R.; methodology: C.R.B., A.B., K.R., S.S. and I.-H.R.; software: C.R.B., A.B. and K.R.; resources: A.B., K.R., S.S. and I.-H.R.; writing—original draft preparation: C.R.B. and A.B.; writing —review and editing: K.R., S.S. and I.-H.R. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2021R1A2C2014333).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data used to support the findings of this study are available from the author (ihra@kunsan.ac.kr) upon request.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Wang, S.; Long, Y.; Ruby, R.; Fu, X. Clustering and power optimization in mmWave massive MIMO–NOMA systems. *Phys. Commun.* **2021**, *49*, 101469. [CrossRef]
2. Wang, J.; Gao, Y.; Yin, X.; Li, F.; Kim, H.J. An enhanced PEGASIS algorithm with mobile sink support for wireless sensor networks. *Wirel. Commun. Mob. Comput.* **2018**, *2018*, 9472075. [CrossRef]
3. Ahmad, W.S.; Radzi, N.A.; Samidi, F.S.; Ismail, A.; Abdullah, F.; Jamaludin, M.Z.; Zakaria, M.N. 5G Technology: Towards Dynamic Spectrum Sharing Using Cognitive Radio Networks. *IEEE Access* **2020**, *8*, 14460–14488. [CrossRef]
4. Do, D.T.; Le, A.T.; Hoang, T.A.; Lee, B.M. Cognitive radio-assisted NOMA broadcasting for 5G cellular V2X communications: Model of roadside unit selection and SWIPT. *Sensors* **2020**, *20*, 1786. [CrossRef] [PubMed]
5. Besher, K.M.; Nieto-Hipolito, J.I.; Briseno, M.V.; Mariscal, R.B. SenPUI: Solutions for sensing and primary user interference in cognitive radio implementation of a wireless sensor network. *Wirel. Commun. Mob. Comput.* **2019**, *2019*, 2405141. [CrossRef]
6. Pla, V.; Alfa, A.S.; Martinez-Bauset, J.; Casares-Giner, V. Discrete-Time Analysis of Cognitive Radio Networks with Nonsaturated Source of Secondary Users. *Wirel. Commun. Mob. Comput.* **2019**, *2019*, 7367028. [CrossRef]
7. Huang, Y.F.; Wang, J.W. Cooperative spectrum sensing in cognitive radio using Bayesian updating with multiple observations. *J. Electron. Sci. Technol.* **2019**, *17*, 252–259.
8. Ozturk, M.; Akram, M.; Hussain, S.; Imran, M.A. Novel QoS-Aware Proactive Spectrum Access Techniques for Cognitive Radio Using Machine Learning. *IEEE Access* **2019**, *7*, 70811–70827. [CrossRef]
9. Wasilewska, M.; Bogucka, H. Machine learning for LTE energy detection performance improvement. *Sensors* **2019**, *19*, 4348. [CrossRef]
10. Shi, Z.; Xie, X.; Lu, H. Deep Reinforcement Learning Based Intelligent User Selection in Massive MIMO Underlay Cognitive Radios. *IEEE Access* **2019**, *7*, 110884–110894. [CrossRef]
11. Arunthavanathan, S.; Kandeepan, S.; Evans, R.J. A Markov Decision Process-Based Opportunistic Spectral Access. *IEEE Wirel. Commun. Lett.* **2016**, *5*, 544–547. [CrossRef]
12. Jaishanthi, B.; Ganesh, E.N.; Sheela, D. Priority-based reserved spectrum allocation by multi-agent through reinforcement learning in cognitive radio network. *Automatika* **2019**, *60*, 564–569. [CrossRef]
13. Jamoos, A.; Abdou, A. Spectrum Measurements and Analysis for Cognitive Radio Applications in Palestine. In Proceedings of the 2019 6th International Conference on Electrical and Electronics Engineering (ICEEE), Istanbul, Turkey, 16–17 April 2019; pp. 180–185. [CrossRef]
14. Liu, Y.; Peng, J.; Kang, J.; Iliyasu, A.M.; Niyato, D.; El-Latif, A.A. A Secure Federated Learning Framework for 5G Networks. *IEEE Wirel. Commun.* **2020**, *27*, 24–31. [CrossRef]
15. Wang, W.; Lv, T.; Wang, T.; Yu, X. Primary user activity based channel allocation in cognitive radio networks. In Proceedings of the IEEE Vehicular Technology Conference, Ottawa, ON, Canada, 6–9 September 2010. [CrossRef]
16. Shafin, R.; Liu, L.; Chandrasekhar, V.; Chen, H.; Reed, J.; Zhang, J.C. Artificial Intelligence-Enabled Cellular Networks: A Critical Path to Beyond-5G and 6G. *IEEE Wirel. Commun.* **2020**, *27*, 212–217. [CrossRef]
17. Zhou, H.; Elsayed, M.; Erol-Kantarci, M. RAN Resource Slicing in 5G Using Multi-Agent Correlated Q-Learning. In Proceedings of the IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC, Helsinki, Finland, 13–16 September 2021; pp. 1179–1184. [CrossRef]
18. Jang, S.J.; Han, C.H.; Lee, K.E.; Yoo, S.J. Reinforcement learning-based dynamic band and channel selection in cognitive radio ad-hoc networks. *J. Wirel. Commun. Netw.* **2019**, *2019*, 131. [CrossRef]
19. Mitola, J.; Maguire, G. Cognitive radio: Making software radios more personal. *IEEE Pers. Commun.* **1999**, *6*, 13–18. [CrossRef]
20. Haykin, S. Cognitive radio: Brain-empowered wireless communications. *IEEE J. Sel. Areas Commun.* **2005**, *23*, 201–220. [CrossRef]
21. Sengottuvelan, S.; Ansari, J.; Mahonen, P.; Venkatesh, T.G.; Petrova, M. Channel Selection Algorithm for Cognitive Radio Networks with Heavy-Tailed Idle Times. *IEEE Trans. Mob. Comput.* **2017**, *16*, 1258–1271. [CrossRef]
22. Li, H.; Pei, L.; Liao, D.; Sun, G.; Xu, D. Blockchain Meets VANET: An Architecture for Identity and Location Privacy Protection in VANET. *Peer-Peer Netw. Appl.* **2019**, *12*, 1178–1193. [CrossRef]
23. Jiang, H.; Wang, T.; Wang, S. Multi-Agent Reinforcement Learning for Dynamic Spectrum Access. In Proceedings of the 2019 IEEE International Conference on Communications (ICC), Shanghai, China, 20–24 May 2019; pp. 1–6.
24. Rajesh Babu, C.; Amutha, B. Blockchain and extreme learning machine based spectrum management in cognitive radio networks. *Trans. Emerg. Telecommun. Technol.* **2020**, e4174. [CrossRef]
25. Babu, C.R.; Amutha, B. A novel energy-efficient data aggregation protocol for cognitive radio based wireless multimedia networks. *Peer-Peer Netw. Appl.* **2021**, *14*, 2452–2461. [CrossRef]
26. Jeya, R.; Amutha, B.; Singh, M.; Arora, C. Performance Analysis and Comparison of Different Modulation Schemes with Channel Estimation Methods for MIMO-OFDM System. *Int. J. Innov. Technol. Explor. Eng.* **2019**, *8*, 2278–2307.

27. Yawada, P.S.; Dong, M.T. Intelligent Process of Spectrum Handoff/Mobility in Cognitive Radio Networks. *J. Electr. Comput. Eng.* **2019**, *2019*, 7692630. [CrossRef]
28. Long, Q.; Chen, Y.; Zhang, H.; Lei, X. Software Defined 5G and 6G Networks: A Survey. *Mob. Netw. Appl.* **2019**, 1–21. [CrossRef]
29. Minoli, D.; Occhiogrosso, B. Practical Aspects for the Integration of 5G Networks and IoT Applications in Smart Cities Environments. *Wirel. Commun. Mob. Comput.* **2019**, *2019*, 5710834. [CrossRef]
30. Shakeel, A.; Hussain, R.; Iqbal, A.; Khan, I.L.; Hasan, Q.U.; Malik, S.A. Spectrum handoff based on imperfect channel state prediction probabilities with collision reduction in cognitive radio ad hoc networks. *Sensors* **2019**, *19*, 4741. [CrossRef]
31. Wang, S.; He, M.; Zhang, Y.; Ruby, R. Equivalent channel-based joint hybrid precoding/combining for large-scale MIMO systems. *Phys. Commun.* **2021**, *47*, 101287. [CrossRef]
32. Akhtar, M.W.; Hassan, S.A.; Ghaffar, R.; Jung, H.; Garg, S.; Hossain, M.S. The shift to 6G communications: Vision and requirements. *Hum. Cent. Comput. Inf. Sci.* **2020**, *10*, 53. [CrossRef]
33. Park, J.H.; Rathore, S.; Singh, S.K.; Salim, M.M.; Azzaoui, A.E.; Kim, T.W.; Pan, Y.; Park, J.H. A comprehensive survey on core technologies and services for 5G security: Taxonomies, issues, and solutions. *Hum. Cent. Comput. Inf. Sci.* **2021**, *11*, 3.
34. Mei, P.; Ding, G.; Jin, Q.; Zhang, F. Research on Emotion Simulation Method of Large-Scale Crowd Evacuation under Particle Model. *Hum. Cent. Comput. Inf. Sci.* **2021**, *11*, 1.
35. He, S.; Xie, K.; Xie, K.; Xu, C.; Wang, J. Interference-aware multisource transmission in multiradio and multichannel wireless network. *IEEE Syst. J.* **2019**, *13*, 2507–2518. [CrossRef]
36. Luo, Y.; Yang, K.; Tang, Q.; Zhang, J.; Li, P.; Qiu, S. An optimal data service providing framework in cloud radio access network. *EURASIP J. Wirel. Commun. Netw.* **2016**, *2016*, 23. [CrossRef]

International Journal of Engineering Trends and Technology
ISSN: 2231 – 5381 / https://doi.org/10.14445/22315381/IJETT-V70I7P210

Volume 70 Issue 7, 88-93, July 2022

Original Article

A Framework for Intelligent Traffic Control System

S. Rakesh[1], Nagaratna P Hegde[2]

[1]*Chaitanya Bharathi Institute of Technology, Research Scholar, OU, Hyderabad, India.*
[2]*Vasavi College of Engineering, Hyderabad, India.*

[2]nagaratnaph@staff.vce.ac.in

Received: 29 May 2022 Revised: 07 July 2022 Accepted: 11 July 2022 Published: 18 July 2022

Abstract *- Recently, traffic congestion has been among the significant problems encountered by many large cities worldwide. The reasons for the traffic congestion are the hasty increase of motor vehicles and inadequate roadways to accommodate a large number of vehicles. Many researchers find the traffic density by applying edge detection (ED), moving object detection (MOD), and frame differencing techniques separately. However, the edge detection method detects the edges for static images and the MOD method finds the traffic density when vehicles are moving. Actually, in real-time, when the red signal is on a traffic junction, the vehicles are in an idle state; this situation is better to apply the ED method. When the green signal is on, vehicles immediately start moving; this situation is best suitable for applying the MOD method to find the real-time traffic density. This paper illustrates a novel technique named Edge Detection and Moving Object Detection (EDMOD) algorithm, which uses ED and MOD approaches to find the real-time area-wide density of the traffic at the traffic light junction by dividing the Region of Interest (ROI) into two regions. It uses ED in region1 and MOD in region2.*

Keywords - *Edge Detection, Image processing, Moving Object Detection, Traffic density.*

1. Introduction

One of the considerable and vital problems in so many large cities is traffic management. Accidents due to improper traffic management at times may be causes significant injuries and sometimes deaths. Vehicles accidents demises in most of the cities, majorly out-turn on account of congestion of traffic accidents. As many vehicles are moving into the jammed traffic roads, there is a great need to introduce new techniques to overcome the drawbacks of existing traffic management situations. Since building or constructing new flyovers, new elevated expressways, new roads, etc., become very expensive, it also may take a lot of time to construct. So, the main objective is to evolve a new traffic controlling method using existing or available infrastructures and the latest technologies.

Nowadays, the world has a great deal of digital and visual data, and a lot of information is produced by processing the data using various technologies. All large and metropolitan cities of the world installed closed-circuit television(CCTV) cameras in many places, including traffic light junctions. These CCTV cameras record and store a lot of data. This data is very helpful in analyzing and developing a solution for traffic problems faced by many big cities in most countries. So many image data analysis methods are tested to understand and analyze this immense collection of data, especially the image data, i.e., recorded and stored by CCTV cameras installed at traffic junctions. ED and MOD techniques are vital among the several image data analysis methods, which have many real-world applications.

2. Related Works

Traffic jams also generate many other overcritical issues and severe problems which influence many human lives, regular activities, and sometimes reasoning for the deaths [1-2]. For instance, assume there is some emergency vehicle, i.e., an ambulance, on the way to the hospital along with an emergency treatment needed. Suppose the emergency vehicle is struck in the congestion caused by substantial traffic. The probability of reaching the patient at the hospital in the expected or needed time will be greatly reduced. And this is going to be a serious issue. Therefore, there is a need to outline and evolve an intelligent and smart traffic light controlling process, which controls and manages the signal at the junction to avoid many accidents, traffic jams, and collisions [1-2].
Earlier, there were many technologies proposed, for instance, RFID and IoT [3-4], image processing [5-7], fuzzy logic [8-9], neural networks and big data [10-13], traffic prediction techniques [14], accident detection and ambulance control [15], real video analysis technique [16], traffic congestion estimation methods [17] and Machine vision technology [18]. The study of those and present technologies [19] allow realizing a few drawbacks in existing techniques. The best solution to overcome many existing drawbacks is to use a density-based traffic control system [20].

In the recommended method, an estimate location technique is used to identify the density of the website traffic, which uses picture processing functions. The computed traffic thickness values are used for the clever traffic signal administration system. This proposed approach can strongly assess the density of web traffic with the help of the area engrossed by the vehicle edges. Traffic density estimation using an area-based method will be added functional to manage smart traffic controlling lights than earlier conventional approaches. Prewitt and Canny edge detection methods are more popular for finding the edges in image processing, out of which canny edge detection techniques have high accuracy [21-22]. The Canny edge detection technique is used to find vehicle density in [23], and the calculated density controls the smart traffic management system.

Several techniques are used to detect the objects which are in motion. Some techniques used to find the objects in motion are recursive, and a few are non-recursive approaches. Recursive MOD approaches are Approximation media filter, Mixture of Gaussians (MoG), and Kalman filter. Non-recursive approaches are Median filter, frame differencing, and Linear predictive filter. Chandrasekhar et al. explained all these techniques' merits and demerits in detail in [24]. The importance algorithm, MoG, used to detect foreground, was explained by T. Bouwmans et al. in [25] and A. Yilmaz et al. in [26].

Optical flow estimation, frame differencing, and subtracting background are some of the moving object recognition techniques. Among these three techniques, the subtracting background paradigm is the most suitable approach for smart traffic applications. The optical flow method needs more complicated computation, and the frame differencing technique is more sensitive with its illumination switches, creating many traffic difficulties. Hence these two techniques are not suitable to use in smart traffic systems. MOD uses subtracting background paradigm. ED approaches are best suitable for static images, and Canny is the best ED technique. This paper proposed a novel technique named EDMOD algorithm, which uses Canny and MOD methods internally to produce the best solution for traffic problems.

3. Methodology

EDAMOD method for smart traffic management involves the following steps.

- Divide the ROI into two regions.
- Apply the Edge detection technique to find the presence of vehicles in the region1.
- Apply the Moving Object Detection technique to find the vehicle density in region2.
- Decision Making.

3.1. Divide the ROI into two regions

The main reason for dividing the ROI into two regions is that the edge detection method detects the edges for static images. The MOD method finds the traffic density when vehicles are moving. Generally, vehicles will wait until the green light is on at a traffic light junction, so divide the ROI into two regions in this project. In the first region, apply the edge detection technique to find the vehicle's presence. And once the green light is on, vehicles will start moving, so in the second region, applying the MOD method to find the traffic density of moving vehicles.

Fig.1 shows the sample division of ROI into two regions. Region1 is selected near the traffic light because all vehicles wait for the green light, and Region 2 is the remaining portion of ROI because once the green light is on, then focusing on the traffic coming in that direction.

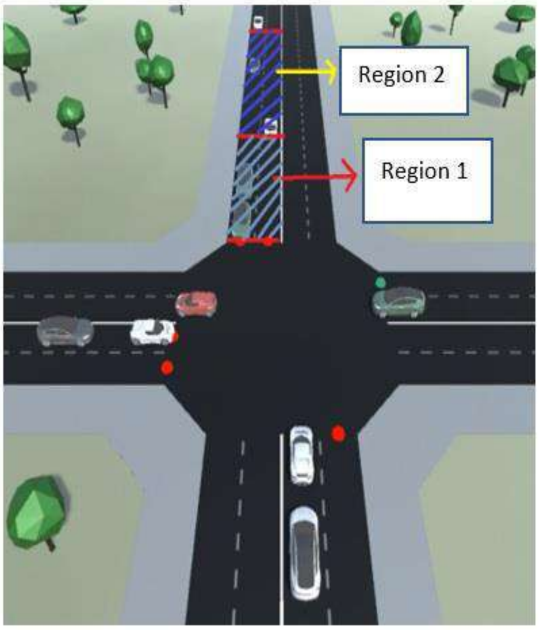


Fig. 1 Division of ROI into region1 and region2

3.2. Apply the Edge detection technique to find the presence of vehicles in region1

Canny is the best edge detection technique out of the available edge detection techniques; hence, this project uses the canny edge detection technique to find whether vehicles are present near traffic junctions. The idea is that if vehicles are waiting for the signal at the junction, then allot green signal in that direction and immediately find the traffic density in region 2. Based on the traffic density values in the region2, further decisions will be taken, i.e., explained in the decision-making section.

Step1: Noise Reduction
Step2: Gradient Calculation
Step3: Non-maximum suppression
Step4: Double threshold
Step5: Edge Tracking by Hysteresis

Fig. 2 Canny Algorithm

The key steps of the Canny algorithm are shown in fig.2, and the sample results of Canny with and without vehicles are shown in fig.3, fig.4. Here, Canny not only detects the vehicle's edges but also detects the edges of other objects or lines present in the frame. Hence first need to apply Canny for the empty road to decide the threshold value used to identify the presence of vehicles.

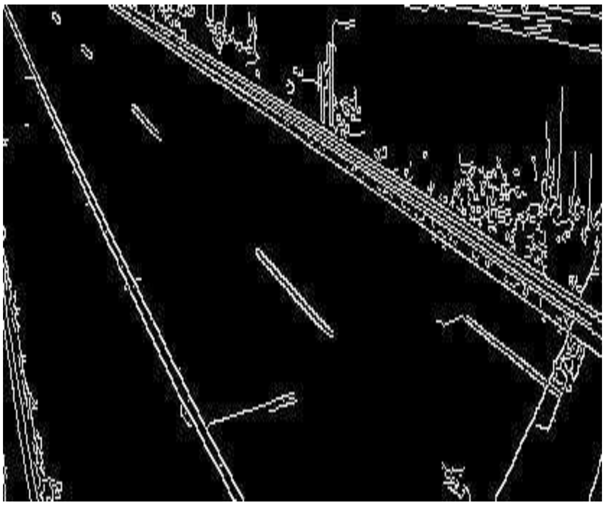


Fig. 3 Canny ED result for an empty road

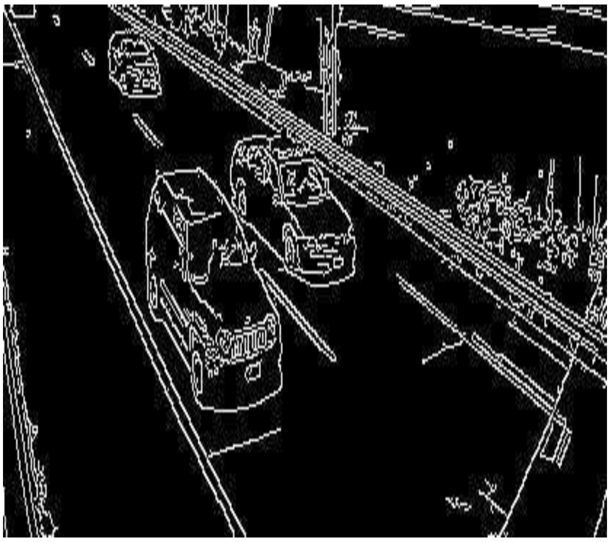
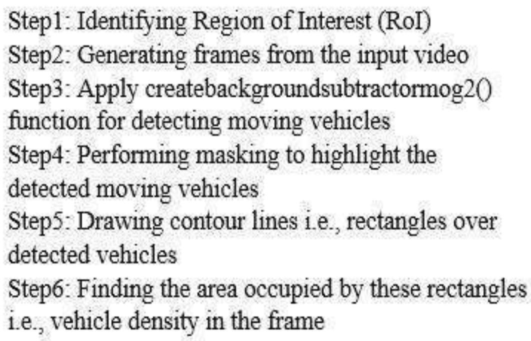


Fig. 4 Canny ED result for the road with Vehicles

3.3. Apply the Moving Object Detection technique to find the vehicle density in region2

Once the green light is on at a traffic light junction, vehicles will start moving. At that time, or after 3 seconds, I started applying the MOD method to find the density of the moving vehicle in the region2. The idea is that if a greater number of vehicles are moving in the region2, then the traffic in that direction is more, so we need to continue the green signal until it reaches its maximum time (60 seconds or 90 seconds). Suppose a moving vehicle's density in region 2 is less than some threshold value. In that case, the traffic in that direction is low, so immediately need to change the green light to red light by giving 5 seconds grace time to avoid accidents at the traffic light junction.



Step1: Identifying Region of Interest (RoI)
Step2: Generating frames from the input video
Step3: Apply createbackgroundsubtractormog2() function for detecting moving vehicles
Step4: Performing masking to highlight the detected moving vehicles
Step5: Drawing contour lines i.e., rectangles over detected vehicles
Step6: Finding the area occupied by these rectangles i.e., vehicle density in the frame

Fig. 5 MOD Algorithm

The step-by-step procedure of the MOD algorithm is presented in fig.5. The Intermediary results of applying the MOD algorithm are shown in fig.6 and fig.7. Fig.6 shows the vehicle detected image after applying it to mask, i.e., step 4 result.Fig.7 shows the contour lines drawn over the detected vehicles, i.e., step 5 result.

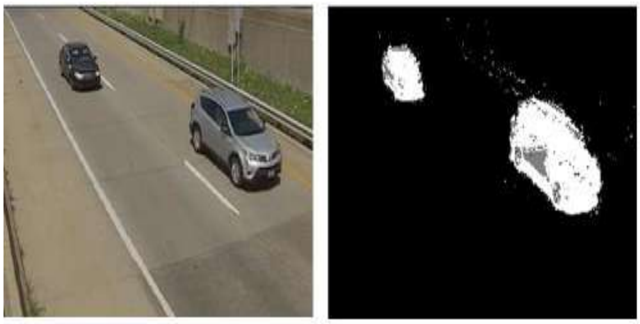


Fig. 6 Sample frame and Vehicle Detection after Masking

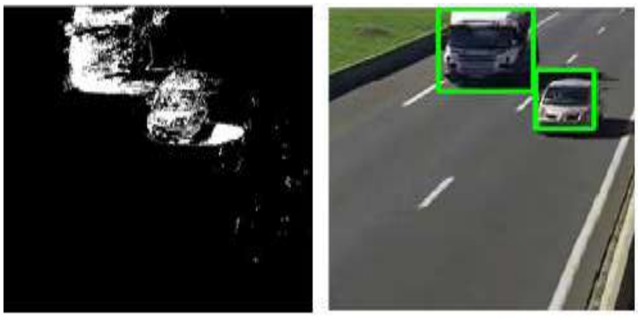


Fig. 7 Frame consisting of Contour lines over Detected Vehicles

3.4. Decision Making

A novel technique is developed and presented in this paper. The very crucial step in this novel technique in decision making. The proposed novel approach uses edge detection and moving object detection. The proposed method divides the ROI at traffic light junctions into two regions, as shown in the figure.

Initially, in the first region, the Canny edge detection technique was applied to find the area of the vehicle edges. If this value is greater than some threshold value, then the green light signal will be given in that direction; otherwise, fewer vehicles are present so that the red light will be continued for some more time, and to the next side region1 again canny edge detection applied.

If the canny result is more than some threshold value, then the green light signal will be given in that direction, and after 3 seconds, start calculating the density of the moving vehicles in the region2. Again, a decision is taken based on the density of the moving vehicle; that is, if it is greater than some threshold value, then the green light will be continued till it is exhausted. If moving vehicles' density is less than the threshold value, traffic is much less in that direction, so traffic lights need to change from green to red by giving 5 seconds grace time. This process continued in a loop at a traffic light junction. The junction may consist of 3-direction roads, 4-direction roads, or 5-direction roads. For all types of junctions, our proposed technique will work efficiently. The detailed controlled flow is presented in the flowchart shown in fig.8.

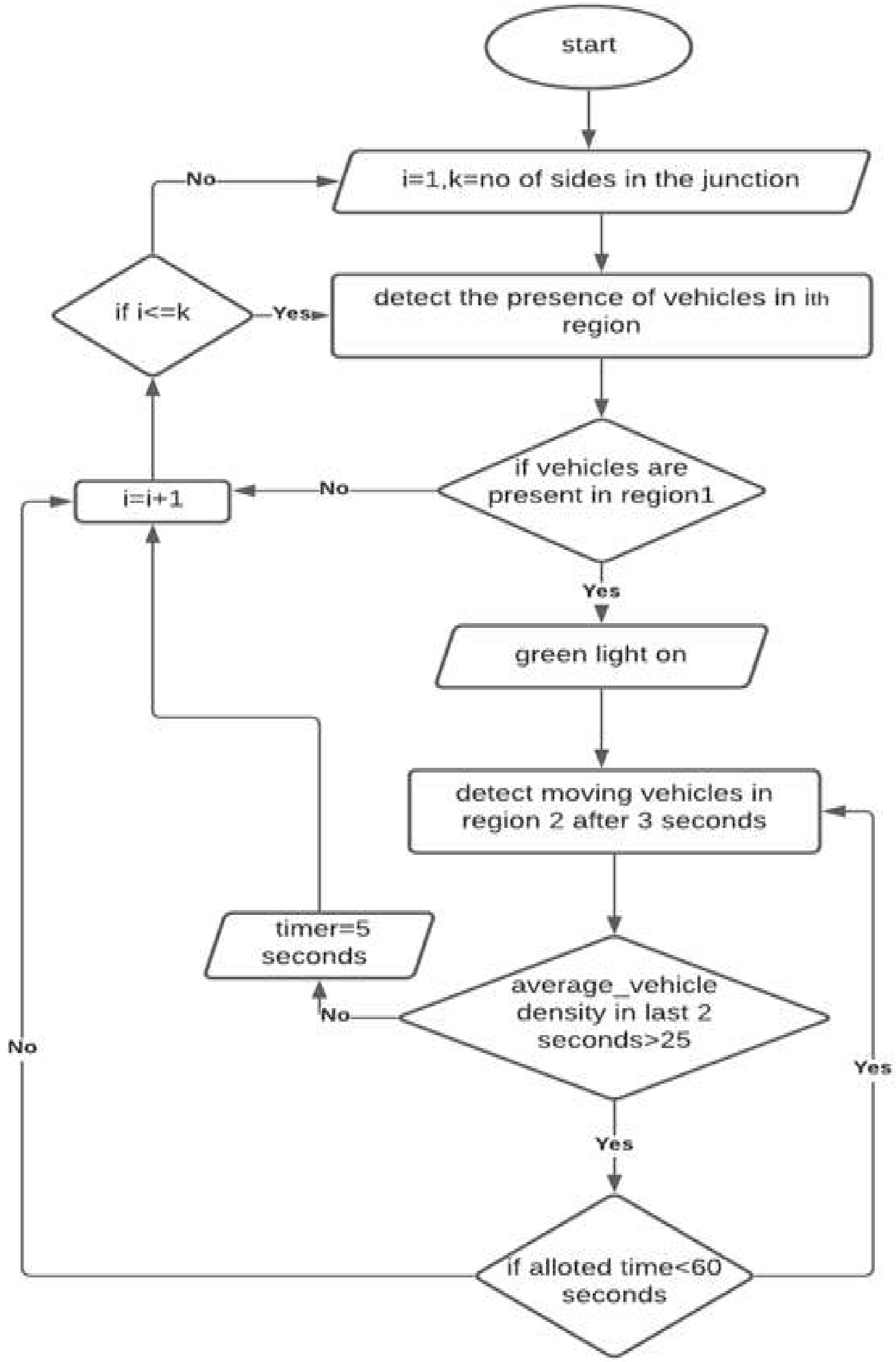


Fig. 8 Flowchart of EDMOD Algorithm

4. Experimental Results

The proposed EDMOD method was evaluated by processing three real traffic video datasets through the proposed EDMOD model. The frames are color images, and the frame rate is 30FPS.Fig.9, fig.10, and fig.11 show the sample frames, which display the traffic density values.Fig.9 consists of 31 traffic density, which means 31% of the road is occupied by the vehicles in the frame. Similarly, Fig.10 consists of 29 traffic density, which means 29% of the road is occupied by the vehicles present in the respective frame, and Fig.11 consists of 0 traffic density, which means the frame does not consist of vehicles.Fig.11 can be considered one of the proofs for extreme test cases because when the frame doesn't consist of any vehicles, it shows the result traffic density value as 0.

Fig. 9 Sample Result Frame with 31 traffic density value

Fig. 10 Sample Result Frame with 29 traffic density value

Fig. 11 Sample Result Frame with 0 traffic density value

Table 1. Traffic density value range vs. green light allocated time

Traffic Density Value using EDMOD	Greenlight estimated allocation time
Less than or equal 20	0 Sec
20 – 30	30 Sec
30 – 40	45 Sec
Greater than 40	60 Sec

The estimated green signal allocation time based on the EDMOD method, especially ED traffic density values, is shown in table 2. If the traffic density value is less than or equal to a fixed threshold value, i.e., 20, then the traffic at that instant is assumed to be very low; hence, the green signal is not allocated. Otherwise, green signal time needs to be allocated using table 1 values. The allotted green signal time may be completely used or not depending on the EDMOD algorithm that is already discussed in the methodology section, i.e., if MOD traffic density values are less than the threshold value, then the green signal is changed to a red signal by giving a grace time of 3 to 5 seconds otherwise green signal continued till the allotted time.

5. Conclusion and Future Scope

The proposed EDMOD model in this paper works efficiently and gives good results compared to existing techniques, i.e., the EDMOD approach is a novel method that works efficiently with static and dynamic frames. Using this proposed novel method, the traffic density values of live traffic frames are calculated by passing the live traffic videos as input to the proposed model. The traffic video datasets used in this paper can generate 30 FPS and the time required to process and calculate the traffic density for each frame is around 1 sec. So, there is a need to find the average value of a few latest frame's traffic densities so that it is used to monitor the traffic light, i.e., if the average density value is lesser than some threshold value, then needs to change the traffic light from green to red because this value suggests the traffic is less otherwise the green light must be continued till reaches maximum allocated time 60 seconds. In this way, the proposed system will solve the issues related to traffic congestion problems.

References

[1] Rajeshwary Sundar, Santosh Hebbar, and Varaprasad Golla, "Implementing Intelligent Traffic Control System for Congestion Control, Ambulance Clearance, and Stolen Vehicle Detection," *IEEE National conference*, 2014.

[2] Mohamed Fazil Mohamed Firdhous, B. H. Sudantha and Naseer Ali Hussien, "A Framework for Iot-Enabled Environment Aware Traffic Management," *International Journal of Electrical and Computer Engineering (IJECE),* vol. 11, no. 1, pp. 518-527, 2021.

[3] R Vijaya Kumar reddy, K Prudvi Raju, M Jogendra Kumar, L Ravi Kumar, P Ravi Prakash, and S Sai Kumar, “Comparative Analysis of Common Edge Detection Algorithms using Pre-processing Technique," *International Journal of Electrical and Computer Engineering (IJECE),* vol. 7, no. 5, pp. 2574-2580, 2017.

[4] Cuneyt Akinlar, Edward Chome, "CannySR: Using Smart Routing of Edge Drawing to Convert Canny Binary Edge Maps to Edge Segments," *IEEE National Conference,* 2015.

[5] Rita Cucchiara, Massimo Piccardi, and Paola Mello, "Image Analysis and Rule-Based Reasoning for a Traffic Monitoring System," *IEEE Transaction on Intelligent Transportation System*, vol. 1, no. 2, 2000.

[6] Madhavi Arora, V. K. Banga, "Real Time Traffic Light Control System," *2nd International Conference on Electrical, Electronics and Civil Engineering (ICEECE'2012),* pp. 172-176, 2012.

[7] Van Li and Xiaoping FAN, "Design of Signal Controllers for Urban Intersections Based on Fuzzy Logic and Weightings," in *6th IEEE Conference on Intelligent Transportation Systems*, vol. 1, pp. 867- 871, 2003.

[8] Tantaoui Mouad, Laanaoui My Driss and Kabil Mustapha "Big Data Traffic Management in a Vehicular Ad-Hoc Network," *International Journal of Electrical and Computer Engineering (IJECE),* vol. 1, no. 4, pp. 3483-3491, 2021.

[9] Celil Ozkurt and Fatih Camci, "Automatic Traffic Density Estimation and Vehicle Classification For Traffic Surveillance Systems Using Neural Networks," *Journal of Mathematical and Computational Applications,* vol. 14, no. 3, pp. 187-196, 2009.

[10] Dipti Srinivasan, Min Chee Choy, and Ruey Long Cheu,"Neural networks for real-time traffic signal control," *IEEE Transactions on Intelligent Transportation Systems,* vol. 7, no. 3, pp. 261-272, 2006.

[11] Julian Nubert, Nicholas Giai Truong, Abel Lim, Herbert Ilhan Tanujaya, Leah Lim, Mai Anh Vu, "Traffic Density Estimation using a Convolutional Neural Network Machine Learning," *National University of Singapore*, 2018.

[12] Jiri Ruzicka, Jan Silar, Zuzana Belinova, Martin Langr, "Methods of Traffic Surveys in Cities for Comparison of Traffic Control Systems – A Case Study," *IEEE International Conference*, 2018.

[13] Taqi Tahmid, Eklas Hossain, "Density Based Smart Traffic Control System Using Canny Edge Detection Algorithm for Congregating Traffic Information," *IEEE National Conference*, 2017.

[14] Papageorgiou M, Diakaki C, Dinopoulou V, Kotsialos A, "Review of Road Traffic Control Strategies," *Proceedings of IEEE,* vol. 91, no. 12, pp. 2043-2067, 2004.

[15] Georgios Vigos, Markos Papageorgioua, Yibing Wangb, "Real-time Estimation of Vehicle-Count Within Signalized Links," *Journal of Transportation Research Part C: Emerging Technologies*, vol. 16, no. 1, pp.18–35, 2008.

[16] Nisha, Rajesh Mehra, Lalita Sharma "Comparative Analysis of Canny and Prewitt Edge Detection Techniques used in Image Processing," *International Journal of Engineering Trends and Technology (IJETT),* vol. 28, no. 1, 2015.

[17] Susmitha.A Ishani Mishra, Divya Sharma, Parul Wadhwa, Lipsa Dash "Implementation of Canny's Edge Detection Technique for Real World Images," *International Journal of Engineering Trends and Technology (IJETT),* vol. 48, no. 4, 2017.

[18] Mohammad Shahab Uddin, Ayon Kumar Das, Md. Abu Taleb, "Real-time Area Based Traffic Density Estimation by Image Processing for Traffic Signal Control System:Bangladesh Perspective" in 2nd *International Conference on Electrical Engineering and Information & Communication Technology (ICEEICT)*, 2015.

[19] Er. Navreet Kaur, Er. Meenakshi Sharma "Comparative Analysis of Techniques used for Traffic Prediction," *International Journal of Engineering Trends and Technology (IJETT),* vol. 50, no. 4, 2017.

[20] Mrs Manasi patil, Aanchal Rawat, Prateek Singh, Srishtie dixit "Accident Detection and Ambulance Control using Intelligent Traffic Control System," *International Journal of Engineering Trends and Technology (IJETT),* vol. 34, no. 8, 2016.

[21] Yonghong Yue, "A Traffic-Flow Parameters Evaluation Approach Based on Urban Road Video," *International Journal of Intelligent Engineering Systems*, vol. 2, no. 1, 2009.

[22] Li Li1, Jian Huang, Xiaofei Huang, Long Chen, "A Real-time Traffic Congestion Estimation Approach from Video Imagery," *International Journal of Intelligent Engineering Systems*, 2008.

[23] Divya Jegatheesan Chandrasekar Arumugam "Intelligent Traffic Management Support System Unfolding the Machine Vision Technology Deployed using YOLO D-NET," *International Journal of Intelligent Engineering Systems,* vol. 14, no. 5, 2021.

[24] U. Chandrasekhar and T. Das, "A Survey of Techniques for Background Subtraction and Traffic Analysis on Surveillance Video," 2011.

[25] T. Bouwmans, F. El Baf, and B. Vachon, "Background Modeling Using Mixture of Gaussians for Foreground Detection-A Survey," *Recent Patents on Computer Science*, vol. 1, no. 3, pp. 219–237, 2008.

[26] A. Yilmaz, O. Javed, and M. Shah, "Object Tracking: A Survey," *ACM computing surveys (CSUR),* vol. 38, no. 4, pp. 13, 2006.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 2060875, 14 pages
https://doi.org/10.1155/2022/2060875

Research Article

Multipath Transmission Control Protocol for Live Virtual Machine Migration in the Cloud Environment

Kadiyala Ramana,[1] **Rajanikanth Aluvalu**,[1] **Vinit Kumar Gunjan**,[2,3] **Ninni Singh**,[2] and **M. Nageswara Prasadhu**[4]

[1]*Department of Information Technology, Chaitanya Bharathi Institute of Technology, Hyderabad 500075, India*
[2]*Department of Computer Science and Engineering, CMR Institute of Technology, Hyderabad, India*
[3]*University of Science and Technology Chattogram, Bangladesh*
[4]*Department of Computer Science and Engineering, Annamacharya Institute of Technology and Sciences, Rajampet, Andhra Pradesh, India*

Correspondence should be addressed to Vinit Kumar Gunjan; vinit.gunjan@ustc.ac.bd

Received 11 March 2022; Accepted 2 April 2022; Published 22 April 2022

Academic Editor: Kuruva Lakshmanna

For mobile cloud computing (MCC), a local virtual machine- (VM-) based cloudlet is proposed to improve the performance of real-time resource-intensive mobile applications. When a mobile device (MD) discovers a cloudlet nearby, it takes some time to build up a virtual machine (VM) inside the cloudlet before data offloading from the MD to the VM can begin. Live virtual machine migration refers to the process of transferring a running Virtual Machine (VM) from one host to another without interrupting its state. Theoretically, live migration process must not render the instance being migrated unavailable during its execution. However, in practice, there is always a service downtime associated with the process. This paper focuses on addressing the need to reduce the service downtime in case of live VM migration in cloud and providing a solution by implementing and optimizing the multipath transmission control protocol (MPTCP) ability within an Infrastructure as a service (IaaS) cloud to increase the efficiency of live migration. We have also introduced an algorithm, the α-best fit algorithm, to optimize the usage of bandwidth and to effectively streamline the MPTCP performance.

1. Introduction

A virtual machine (VM) refers to an entity that does not exist physically, but is hosted on a hypervisor, and yet has all the capabilities of an equivalent physical machine [1]. This virtualization gives an added advantage in the form of the ability to migrate a VM from one host machine to another. Virtual machine migration may be done in the cloud for several reasons including load balancing and energy usage optimization. When a VM is used to handle data traffic within a cloud, a relatively freer host (destination host) can be chosen to migrate it to, as an attempt to relieve the load on the source host. Similarly, energy optimization within the cloud can be done by shutting off the nodes which do not host any occupied VMs [2].

In datacenter virtualization, live migration is a crucial technology and feature. With live migration, VMs can be transferred from one physical host to another with little to no impact on the availability of running applications. This ensures that running applications are not harmed by any physical server failures, significantly improving service availability. The TCP/IP protocol is used to send live migration communications via the Ethernet network that connects the cluster servers. The content that needs be migrated is mostly the CPU cache, memory, and buffers; however, the memory content takes up the majority of the time. The content of the CPU cache and buffers is almost insignificant in comparison to the RAM content, which is what most authors assume in live migration modeling.

In the early stages of its development, VM migration was static in nature. This is to say that the general practice was to suspend the VM state before the actual migration took place. This caused the instance to be unavailable throughout the migration process. The functioning of the VM was resumed after the entire VM data, that is, the memory pages and the processor state, was copied to the destination host. It is easily imaginable how much inconvenience this sort of a process can cause when implemented in cloud environment [3]. The users pay for the services they receive, and they expect those services to be available all the time. In such scenario, where a VM handling some or all of those services is rendered unavailable till the completion of the migration process, the user experience surely tends to degrade [4].

The solution was live VM migration. In this case, the VM is not unavailable for the entire duration of the migration but only a smaller fraction of it, which is termed as the service downtime. Theoretically, the machine continues to function as if nothing had occurred at the back end during the migration process. However, as shown by Sv¨ard et al. [5], running processes on the VM are interrupted even during its live migration. This is attributed to the service downtime. Therefore, even if the service downtime is small, to ensure good end user experience, there will always be a need to reduce it as much as possible.

When we talk about the functioning of an Infrastructure as a Service (IaaS) cloud, we automatically step into the realm of software-defined networking (SDN). SDN refers to the use of programmable software components that can be installed in a networking framework and still provide the same functionality as their physical hardware counterparts [6]. Modern clouds depend heavily on SDN to function. Though physical resources are required to host the cloud instances, the networking in the cloud combines elements which are both physical and software defined in nature. Without SDN, it would be nearly impossible to cope with the growing pressure on data networks as the number of mobile networking devices increase [7]. IaaS cloud platforms such as "OpenStack" use SDN to connect the instances to the physical networking resources. With SDN, networking becomes more manageable and less cumbersome. Moreover, we can add or remove elements from a software defined network according to our needs, which would otherwise require the installation and removal of physical networking devices.

In this paper, we have implemented multipath TCP, which is a recent project by the Internet Engineering Task Force (IETF) to facilitate faster data transfer within the cloud environment [8]. The VM data can be transferred with a higher speed between the host nodes in case of data centers if the effective bandwidth of the connection between them can be increased. This ability is provided by MPTCP by allowing us to use multiple paths for TCP data transfer at the same time. By the creation of TCP subflows, MPTCP makes use of multiple interfaces on the connected hosts for the transfer of data. Hence, theoretically, the aggregate bandwidth of each of the individual paths is available for data transfer. However, in reality, this is not the case. MPTCP requires some optimization techniques to be implemented effectively in the cloud. We introduce an algorithm, the α-best fit algorithm, for the optimization of MPTCP usage, and compare it to the subflow optimization offered by Joshi and Kataoka [9]. This algorithm provides a much better optimization scheme than the one already proposed [9] in terms of bandwidth usage. The generation of optimum number of subflows in [9] leads to wastage of bandwidth by generating suboptimal results, while the α-best fit algorithm minimizes this wastage, which is crucial in the cloud scenario where bandwidth resources to be dealt with are massive in scale.

There are three forms of live VM migration: precopy, post-copy, and hybrid-copy. The most resilient migration type is precopy live migration, which we focus on in this study. The source host memory content copy begins and continues to transfer until it reaches a halting circumstance in precopy migration. The VM will then stop on the source host and start on the target host. This is the most popular migration method used by commercial and open source hypervisors since it is the most dependable.

The remaining part of this paper is organized as follows: Section 2 explains the conventional algorithm that is in use for conducting live migrations, Section 3 explains MPTCP, and its effects on live migration of cloud instances, Section 4 explains the need for optimization of MPTCP implementation in cloud environments and puts forward our proposed approach to do so, Section 5 contains the details of implementation of MPTCP in the cloud environment, Section 6 embodies the results obtained, and Section 7 includes the discussion on the results and the conclusion.

2. Related Work

Multipath TCP is a TCP enhancement that allows for simultaneous transmission of data through many pathways from one end to the other. For example, multipath TCP allows many applications to transport and receive data over multiple interfaces, such as cellular and Wi-Fi, by creating a single TCP sub flow for each interface [10]. This subflow is used by the MPTCP scheduler to transmit and receive data. MPTCP's design has various advantages, including resource utilization, throughput, and a mild response to faults, as well as good route performance.

If numerous sub flows are available, a scheduler [11] selects the subflow with the least round-trip time (RTT) to transfer the data. During the transmission of the segment, the scheduler chooses the way with the shortest round-trip time among all subflows whose congestion window is not yet full. If there are multiple such paths, the scheduler creates a bias towards one of them. It also continues to deliver information on the particular subflow until the congestion window for that sub flow is full. In prior research, MPTCP was examined in a mobile situation. The authors investigated the impact of mobility on MPTCP in [12], and in [13], they offered various MPTCP options for use by cell phones for Wi-Fi handover. However, neither of the works investigates the heterogenetic nature of the path in lossy subflows. The authors of [14] gave a comparison of single TCP vs. multipath TCP. The authors of [15] used various scheduler algorithms to calculate the impact of scheduler architecture on performance. In [16, 17], different congestion mechanism techniques are compared.

The authors proposed scheduler algorithms in [18], which select subflows based on an estimate of the amount of traffic they can handle before becoming congested. By considering large network transfers and a restricted buffering quantity, the methodology demonstrates. Another unique scheduling policy offered by the authors in [19] was to keep a strategic distance from out-of-request parts. The authors do not explain why a segment is removed from the TCP buffer after it is transmitted by another subflow. The authors presented a delay-aware packet scheduler technique on the ns-2 simulator platform in [20]. In both delay and stable settings, the proposed technique analyses path heterogeneity. The performance of MPTCP over a mobile network was examined by the authors in [21]. The analysis took into account a variety of subflows as well as precise statistics such as round-trip time (RTT) and out-of-order delivery, but the authors did not account for lossy subflows. Lim et al. proposed a new scheduler method that sends buffers by monitoring available bandwidth on each subflow, but it does not take use of the data loss rate on each sub flow.

3. The Precopy Algorithm

This is the conventional method for carrying out live VM migrations. In this method, the memory pages of the VM are sent before its processor state to the destination node. The transfer of the memory pages occurs in several iterations. These iterations continue till a set number is reached, or a small enough writable working set (WWS) is obtained [22]. In each iteration, there are certain memory pages which are modified during the transmission, and these are called dirty pages. During each iteration, there is a possibility of a memory page being dirtied, and these dirty pages are transferred to the target node in the next iteration.

The duration of service downtime starts when either a WWS is obtained or the cap on the number of iterations is reached [23]. If either of the above conditions are satisfied, the VM at the source node is suspended, and all the remaining memory pages along with the dirty pages from the previous iteration are moved to the destination node. After this transfer, the VM is resumed at the destination node, and its source node copy is destroyed. Hence, it is obvious that the service downtime depends upon the number of memory pages to be sent, which also includes the dirty pages from the previous iteration [24].

The time taken to transfer the entire RAM is given by [22]

$$t_0 = \frac{V}{B}, \tag{1}$$

where V is the total memory size of the VM, and B is the available bandwidth.

We must also consider the page dirtying rate to get a measure of the time required for their transfer in each iteration during precopy. This time required for the transfer of the remaining memory pages, which also includes the pages dirtied in each iteration, is given by [22]

$$t_1 = \frac{R \times t_0}{B}, t_2 = \frac{R \times t_1}{B}, t_k = \frac{R \times t_{k-1}}{B}, \tag{2}$$

where R is the memory dirtying rate, and k is the total number of iterations.

The precopy process gets suspended when the k -$^{\text{th}}$ iteration occurs, and the stop conditions are satisfied. Following this, the VM at the source is suspended, and the remaining dirty pages, along with the processor state of the VM, are transferred to the destination node [22]. This takes about $t_k = R \times t_{k-1}/B$ amount of time.

Hence, the total time for the precopy and stop and copy is calculated as

$$T_0 = t_0 + t_1 + t_2 + \cdots + t_k = \frac{V}{B} \times \frac{1 - (R/B)^{k+1}}{1 - (R/B)}. \tag{3}$$

Therefore, the total migration time is given by

$$T_{\text{mig}} = \text{Pr}\ e - \text{migration overhead} + T_o + \text{Post-migration overhead}, \tag{4}$$

where the premigration overhead refers to the time required for the resource reservation, and the postmigration overhead consists of the commitment and activation stage of the precopy [22]. While these overheads are largely unavoidable, the value of T_o can be varied.

4. Multipath TCP (MPTCP)

MPTCP is a recent project by the Internet Engineering Task Force (IETF). This project enables data transfer via multiple TCP subflows that are sent over different TCP paths [8]. This capability is of great value if we consider its implementation in data centers. Multiple paths to send data imply more bandwidth for data transfer. In principle, this bandwidth should be equal to the aggregate of the bandwidth capacities of the individual paths [25].

As we can see in Figure 1, data is divided into chunks, and these chunks are transferred through different subflows over different paths. This means that the client and the server in a simple MPTCP connection have multiple IP addresses, which may or may not be in the same network [26]. In case of data centers, MPTCP can therefore be utilized to obtain the ability of fast data transfer; since, at any given time, more than one paths can be used simultaneously as opposed to TCP, which can be seen in Figure 2.

Now, consider the scenario in which the implementation of the precopy algorithm described in Section 2 in combination with the MPTCP capability is being carried out.

If we consider k number of iterations to occur for the transfer of dirty pages in case of a single active path for data transfer, as was the case in equations (2) and (3), then we can see that for an iteration i, the pages dirtied will be $R \times i$, and these will be transmitted in the $(i+1)$-$^{\text{th}}$ iteration.

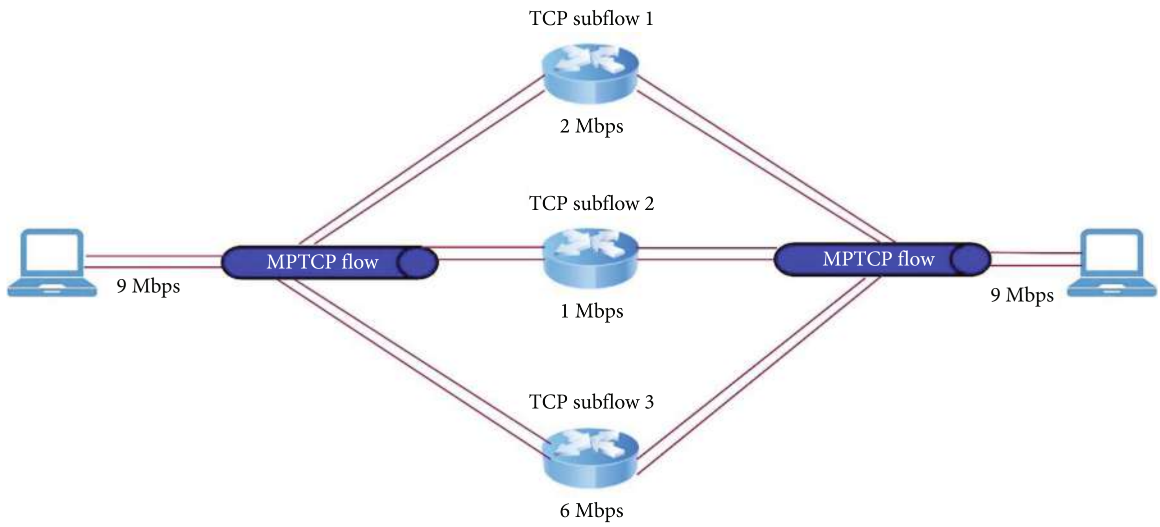


Figure 1: Division of MPTCP flow into TCP subflows.

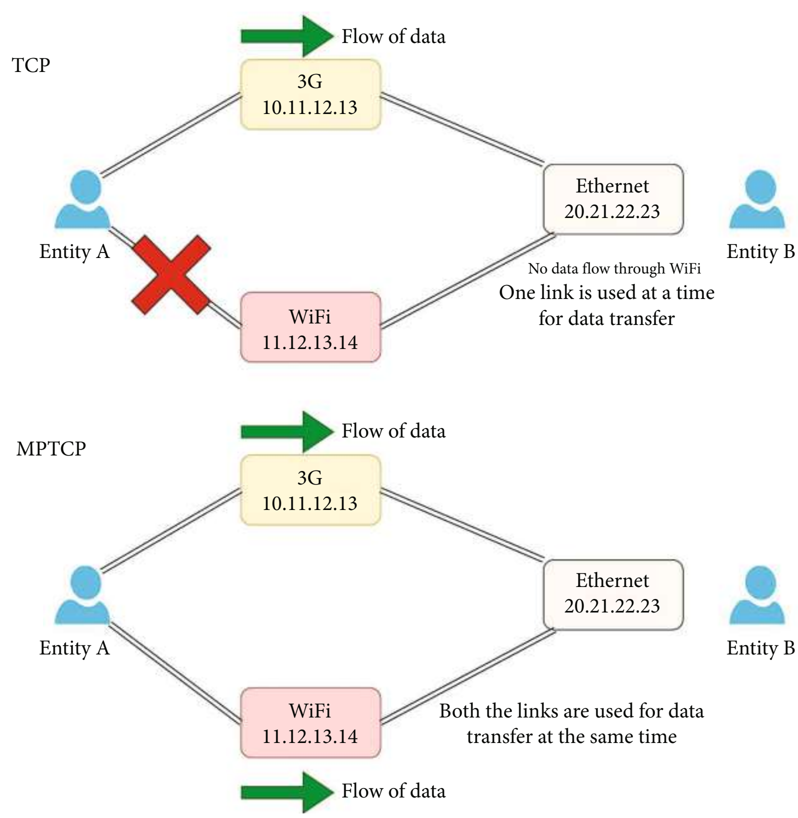


Figure 2: Comparison of MPTCP session with normal TCP session.

As stated by Nathan et al. [27], the page dirtying rate for the i-th iteration is given by the expression:

$$R = \frac{\text{number of pages dirtied in iteration } (i-1) \times \text{page size}}{\text{time taken by iteration } (i-1)}. \quad (5)$$

Now, the time taken for an iteration is dependent upon the bandwidth available, which is evident from equation (2). Hence, an increased bandwidth will reduce the page dirtying rate [28]. This implies that less pages will be dirtied in each iteration with the increase in effective bandwidth, and hence the time required for these iterations will decrease. Considering the page dirtying rate to be constant for a migration, then the advantage of added bandwidth is expected to reduce the time $t_{(i+1)} = R \times t_i/B'$, where B' is the new bandwidth.

This implies that though the iterations will require lesser time than in the case of a reduced bandwidth capacity, the actual number of iterations, k will remain the same.

This implies that the advantage MPTCP will provide due to increased aggregate bandwidth will be in the form of reduced transmission time rather than reduced number of iterations before the WWS is achieved. Also, as stated in [22], the time required after k iterations to transfer the WWS and the processor state is given by

$$t_k = \frac{R \times t_{k-1}}{B} \tag{6}$$

This time period required for the stop-and-copy phase will be reduced by an increase in bandwidth. Therefore, a decrease in service downtime is expected.

Assuming that we have an MPTCP connectivity between two host machines with x number of active paths for data transfer, let γ be the segment-wise bandwidth utilization factor, which is explained in the next section.

Hence, the aggregate effective bandwidth of the MPTCP connection will be $B' = x \times \gamma \times B$, where B is the bandwidth of each individual path in the MPTCP connection.

Using this new bandwidth, equations in Section 2 can be written for the MPTCP scenario as

$$t'_0 = \frac{V}{B'}, \tag{7}$$

$$t'_1 = \frac{R \times t'_0}{B'}, t_2 = \frac{R \times t'_1}{B'}, t'_k = \frac{R \times t'_{k-1}}{B'}, \tag{8}$$

$$T'_0 = t'_0 + t'_1 + t'_2 + \cdots + t'_k = \frac{V}{B'} \times \frac{1 - (R/B')^{k+1}}{1 - (R/B')}. \tag{9}$$

Using $B' = x \times \gamma \times B$ in equations 7and 8, it can be stated that for an iteration i,

$$t'_i = \frac{t_i}{(x\gamma)^{i+1}}. \tag{10}$$

From equation (10), it can be inferred that with the increasing number of iterations, the time required for transmission of data will decrease progressively in case of MPTCP connectivity between the hosts. When the value of "i" in equation (10) becomes equal to k, then the time period t'_k represents the time spent in the stop-and-copy phase [22], which is synonymous to the service downtime. The value x will always be positive and greater than one for an MPTCP connection, while γ will be a positive fraction representing the degree of utilization of the bandwidth resources.

On the other hand, equation (10) also provides the information that in case of TCP connection between the hosts, when $x = 1$, γ is still less than 1. This means that with each successive iteration, with increasing powers of γ, the performance of precopy for transmission of dirty pages will degrade.

Therefore, while TCP connectivity causes the performance to worsen with every iteration, MPTCP connectivity actually increases it with the increase in number of iterations till a set number of iterations, which will depend upon the product $(x\gamma)$. As long as the quantity $(x\gamma)$ remains greater than one, the performance will increase with increasing number iterations.

Rearranging equation 6 is as follows:

$$T'_0 = \sum_{i=0}^{k} \frac{t_i}{(x\gamma)^{i+1}}. \tag{11}$$

Therefore,

$$(x\gamma)^{k+1} T'_0 = \sum_{i=0}^{k} t_i (x\gamma)^{k-i}. \tag{12}$$

Dividing equation (9) by equation (3), we get

$$\frac{T'_0}{T_0} = \frac{1}{(x\gamma)^{k+1}} \times \sum_{i=0}^{k} (x\gamma)^{k-i} \tag{13}$$

Hence, it can be seen that the ratio T'_0/T_0 will always be less than one if the product $x\gamma > 1$. Therefore, the total migration time will also decrease with increasing number of active paths for data transfer.

This deduction implies that by increasing the number of active paths in an MPTCP connection, there will be a reduction in the service downtime to a negligible nonzero amount. However, this is not the case, and thus the MPTCP connection requires optimization for the reason that with increasing number of paths in an MPTCP connection, there is an increased possibility of network congestion at the points where the subflows reunite. This congestion lowers the value of the factor $(x\gamma)$, as the bandwidth utilization (γ) is drastically reduced in case of congestion.

Focusing on the value of γ, its relation to x and T_0 can be predicted to some extent using equation (11). Partially differentiating equation (11) with respect to γ,

$$\frac{\partial T'_0}{\partial \gamma} = \sum_{i=0}^{k} \frac{\partial}{\partial \gamma}\left(\frac{t_i}{(x\gamma)^{i+1}}\right) = \frac{-t_0}{x\gamma^2} - \frac{2t_1}{x^2\gamma^3} - \cdots - \frac{(k+1)t_k}{x^{k+1}\gamma^{k+2}}. \tag{14}$$

In terms of t_0, we have

$$\frac{\partial T'_0}{\partial \gamma} = \frac{t_0}{x\gamma^2}\left[-1 - \frac{2R}{Bx\gamma} - \cdots - \frac{(k+1)R^k}{B^k x^k \gamma^k}\right]. \tag{15}$$

This series can be converged, and the resulting equation is as follows:

$$\frac{\partial T_0'}{\partial \gamma} = \frac{-t_0}{x\gamma^2} \times \left[\frac{B^{-k}\gamma^{-k}R^{k+1}x^{-k}(k+1)R - B\gamma x(k+2) - B^2x^2\gamma^2}{(B\gamma x - R)^2}\right] \quad (16)$$

Figuring out the value of γ at which the value of T_o will be maximum or minimum can be achieved by finding the critical points by equating the partial differential in equation (16) to zero.

Substituting $\partial T_0'/\partial\gamma = 0$, equation (16) can be rewritten as

$$\left[\frac{B^{-k}\gamma^{-k}R^{k+1}x^{-k}(k+1)R - B\gamma x(k+2) - B^2x^2\gamma^2}{(B\gamma x - R)^2}\right] = 0. \quad (17)$$

$$\text{As}\frac{-t_0}{x\gamma^2} \neq 0. \quad (18)$$

Now, Let

$$\text{C1} = B^{-k}x^{-k}R^{k+2}(k+1), \quad (19)$$

$$\text{C2} = B^{1-k}x^{1-k}R^{k+1}, \quad (20)$$

$$\begin{aligned}\text{C3} &= B^2x^2. \\ C4 &= C2\,(k-2).\end{aligned} \quad (21)$$

Substituting the above values in equation (17),

$$\frac{C_1}{\gamma^k} - \frac{C_4}{\gamma^{k-1}} - C_3 = 0 \Rightarrow C_1 - C_4\gamma - C_3\gamma^k = 0. \quad (22)$$

Equation (22) is a polynomial of order k. Here, the value of γ can be represented by the series:

$$\gamma = \sum_{m=0}^{\infty}(-1)^m \frac{C_1^{mk-m+1}C_3^m}{C_4^{mk+1}m!}\prod_{j=0}^{m-2}(mk-j) = \frac{C_1}{C_4} - \frac{C_1^kC_3}{C_4^{k+1}} + \frac{2kC_1^{2k-1}C_3^2}{2!C_4^{2k+1}} - \frac{3k(3k-1)C_1^{3k-2}C_3^3}{3!C_4^{3k+1}} + \frac{4k(4k-1)(4k-2)C_1^{4k-3}C_3^4}{4!C_4^{4k+1}} - \cdots \quad (23)$$

This series will converge if the following condition is satisfied:

$$\left|\frac{C_1^{k-1}C_3}{C_4^k}\right| < \frac{(k-1)^{k-1}}{k^k}, \quad (24)$$

which translates to

$$\left|\left(\frac{Bx}{R}\right)^2 \times \frac{(k+1)^{k-1}}{(k-2)}\right| < \frac{(k-1)^{k-1}}{k^k}. \quad (25)$$

For this inequality, it can be clearly seen that $k \notin [0\,;2]$, because for $k \in [0\,;2]$, the inequality becomes meaningless. Therefore, $k \geq 3$ which is justified as practically the number of iterations is always higher than two. Hence, the series in equation (23) cannot be converged.

Now, the constraint on the value of γ is that $\gamma \in (0;\,1)$ If C_1, C_3, $C_4 > 0$ and $C_1 < C_3 + C_4$, there can be a unique value of γ, which will be the critical point for the equation (11). This can be accomplished by taking a sufficient number of terms so that we get a good approximation of the sum of the series in equation (23).

The page dirtying rate, as defined by Nathan et al. [27], is lesser than the value of (Bx), which is quite reasonable. Also, since k and x are positive integers, and $\gamma \in (0;\,1)$.

Therefore, using the values of C_1, C_2, C_3, and C_4 in equation (25), it can be seen that

$$\gamma = \frac{R}{Bx} \times \frac{k+1}{k-2}. \quad (26)$$

The higher terms in the series given by equation (23) are not taken into consideration, because of the fact that for $k \in (3,\infty)$, the value of higher terms will become negligible. At greater values of k, the power of the terms in the denominators of the terms in equation (23) will increase steeply. The first term yields a number that satisfies the condition of $\gamma \in (0,\,1)$, while inclusion of the higher terms does not affect its value significantly. This concludes that the value of k, given our constraints is as in equation ((26)).

Now,

$$\frac{\partial^2 T_0'}{\partial\gamma^2} = \frac{\partial^2}{\partial\gamma^2}\sum_{i=0}^{k}\left[\frac{t_i}{(x\gamma)^{i+1}}\right] = \sum_{i=0}^{k}\frac{\partial}{\partial\gamma}\left[\frac{-t_i(i+1)}{x^{i+1}\gamma^{i+2}}\right] = \sum_{i=0}^{k}\frac{t_i(i+1)(i+2)}{x^{i+1}\gamma^{i+3}}. \quad (27)$$

As all the variables in equation (27) are positive, and the critical point given by equation (26) is also a positive value, it can be inferred that the critical point is the point of maxima for the function T_0' with respect to γ with x and k constant.

Thus, it is explicit that for the value of γ given in equation (26), the migration time T_0' will be maximum. This proves that for MPTCP to be advantageous at a given number of active data paths between the hosts, the value of γ must be greater than the expression given by equation (26).

5. Optimizing MPTCP Capability

The MPTCP capability offers higher bandwidth for data transfer in clouds and data centers. In theory, by increasing the number of paths in an MPTCP session between two hosts, there should be significant improvement in the data transfer speed as an effect of increased effective bandwidth. One must therefore expect that the number of paths can be increased without any limit to obtain a nearly infinite bandwidth capacity for data transfer. However, it is not the case. While it is true that the increase in the number of paths leads to increased number of TCP subflows in an MPTCP session, their optimum usage is something that depends upon other factors.

As explained by Joshi and Kataoka [9], overutilization as well as underutilization of the MPTCP capability by creating fewer or a greater number of subflows than required leads to poor performance in terms of data transfer. [9] already provides an optimization algorithm which has been tested and implemented to provide an improved result than using MPTCP without any subflow optimization in terms of throughput.

The SFO given in [9] is modified to obtain the α-best fit algorithm, which is more suited for the cloud environment. Moreover, as the cloud environment deals with bandwidth resources that are in order of Gbps, the optimization of the SFO [9] for improved performance in terms of bandwidth utilization is needed.

The MPTCP subflow optimization as elaborated in [9] consists of three functions, the first function responsible for finding the feasible paths for the MPTCP session between the hosts, and the second function is responsible for calculating the optimum number of sub flows, while the third function deals with the assignment of paths to the MPTCP subflows.

SFO [9] uses the modified k-shortest path algorithm in order to calculate the set of shortest paths for data transfer between the hosts. While this approach suffices in the experimental topology considered in [9], it might lead to suboptimal bandwidth utilization in case of other available topologies for data centers and cloud environment. Hence, the first optimization we propose is to modify the path finder method to function not on the shortest k-path algorithm, but on maximum bandwidth algorithm, which gives priority to the path with the highest bandwidth capacity during selection.

The function responsible for the calculation of optimum number of sub flows has been modified by us to include the aspects of protection against path failure and reliability in performance along with optimization of the selection process.

5.1. The α–Best Fit Algorithm. Before discussing the α-best fit algorithm for the optimization of MPTCP subflows, it is essential to consider a factor that ought to be useful while we deal with substantial amount of bandwidth capacities, as is in the case of data centers. This factor is the segment-wise bandwidth utilization ratio (γ), including this factor will lead to a better reliability of the system, as it gives us a more realistic and practical view of the bandwidth resources. The inclusion of γ increases the accuracy of the algorithm while functioning in real time conditions.

This factor is calculated by the standard method, which involves using SNMP objects to know what fraction of the total bandwidth of a link is being considered. The bandwidth of a path is limited by the capacity of the minimum bandwidth link in its constituting links. This means that for a path consisting of n links, $\lambda_P = [\min(\lambda_i, \forall_i \in P)]_{i=1}^{n}$. Therefore, γ is calculated segment-wise.

The value of γ can be calculated as [29]

$$\gamma = \frac{\max(\Delta \text{IfInOctets}, \Delta \text{IfOutOctets}) \times 8}{\Delta \text{time period between the polls} \times \text{IfSpeed}}, \quad (28)$$

where IfInOctets and IfOutOctets refer to the ingress and egress SNMP objects on a TCP link which are sent repeatedly after a fixed time interval. IfSpeed refers to the interface speed as stated by the specifications of the link.

Let S be the total size of the VM data to be transferred during the MPTCP session, τ be the time limit in which the live VM migration should be complete, α^d be the demand of bandwidth resources for the transfer of the VM, ONS be the optimum number of subflows, OP be the set of all the optimized paths available to put the sub flows on, FP be the set of all the feasible paths, D be the set of discarded paths, and λ be the bandwidth capacity.

The subflow optimization [9] provides a method to divide the MPTCP traffic into a suitable number of sub flows; however, they do not clarify the ramifications of the case when a particular path fails in the MPTCP connection. The comparison between α-best fit algorithm and subflow optimization algorithm [9] is represented in Table 1.

In order to address the scenario, in which a path moves to a "down" state, an additional clause has been devised in the α-best fit algorithm. This part of the algorithm makes use of a function named failure protection, which uses the set of all feasible paths (K) and the bandwidth capacity of the failed path to decide which path to redirect the subflow, which was initially on the now failed path, or to divide the subflow further into smaller subflows that are redirected onto other available paths in the set K. It is ensured that the algorithm functions in a way that causes minimum wastage of bandwidth owing to the importance and the order of magnitude of the bandwidth resources in the cloud environment.

6. Implementation

The implementation MPTCP on an OpenStack Mitaka IaaS cloud consisting of four nodes–one controller node, two compute nodes, and a block storage node, was done. The networking inside the cloud makes use of the OpenStack Neutron server residing on the controller node, and ML2 is used as the core plugin with Linux Bridge agent as the mechanism driver. The controller node has 16 GB RAM, 250 GB SDD, two network interfaces, while the compute nodes have 32 GB of RAM, 150 GB SDD, and four network interfaces each. The block storage node has 32 GB of RAM, 150 GB SDD on which the Operating System is hosted, and a 500 GB SDD to provide block storage resources to the OpenStack instances. The Operating System installed on all the four nodes is Ubuntu 14.04 LTS. Figure 3 shows the host networking implemented for deploying Openstack with MPTCP capability.

In order to measure the service downtime, instances are created, and they are pinged continuously from the controller node while the migration process from one compute host to another occurs. The time for which the instance remains unavailable is observed as the service downtime. A packet sniffer (Wireshark) is used for this purpose.

The service downtime is measured without using MPTCP (normal TCP connection) and then using MPTCP using two and then three paths between the hosts, respectively. Comparisons are made using the obtained results.

In order to test the proposed algorithm for optimization of MPTCP connection, we compare the results of the algorithm given in [9] to α-best fit algorithm's performance on randomly generated bandwidth values for a Multi Path TCP connection.

Table 1: Comparison between α-best fit algorithm and subflow optimization algorithm [9].

The α–best fit algorithm	Subflow optimization [9]
The maximum bandwidth path is given priority while selection of feasible paths.	The shortest path is chosen while deciding the set of feasible paths.
Discarded paths are stored as they can be used later to make maximum use of the available bandwidth.	Paths once discarded are never used again.
Either the path with bandwidth closest to the demand is used, or the subflow is divided into smaller subflows.	If the ONS is one, it might lead to the selection of a path with far greater bandwidth than the demand.
Protection against failure of a path is taken into account.	Protection against failure is not considered.
It is more reliable to function correctly in real time environment as bandwidth utilization factor is taken into account.	Bandwidth utilization factor is not considered, hence, its reliability in real time environment is limited.

1. Compute $\alpha^{d}=s/\tau$
2. Initialise D=ϕ; ONS=0; OP=ϕ; FP←K; Sort K in ascending order of bandwidths
3. Compute $\lambda_P \forall P \in K$
$\lambda_P = \gamma[\lambda_P]_{(i=1)}^{|FP|}$
4. Compute the total capacity of all feasible paths as
$\propto = \sum_{P=1}^{(|FP|)} \lambda_P$
5. if $\alpha^d < \alpha$ **then**
6. FP←FP-P_{min}, Where P_{min} is the path ∊ FP with the minimum bandwidth
7. $D \longleftarrow D \cup P_{\min}$
8. **Compute the new bandwidth capacity**
$\alpha^s = \sum_{P=1}^{(|FP|)} \lambda_P$
9. **if** $\alpha^d < \alpha^s$ **then**
10. $\alpha \longleftarrow \alpha^s$
11. **Repeat** 5
12. while i=1 to |D **do**
13. $FP \longleftarrow FP \cup D_i$
14. $\alpha^{s'} = \sum_{P=1}^{(|FP|)} \lambda_P$
15. if $\alpha^d \geq \alpha^{s'}$ **then**
16. FP←FP-D_i
17. i++18. OP←FP, ON=|OP|
19. if ONS=1 & $\lambda_P \neq \alpha^d$ **then**
20. K←ϕ, K←D, D←ϕ
21. **goto 3**
22. return OP & ONS
23. if $[OP_i]_i^{|OP|}$ is down **then**, Where OP is the set of Optimal Paths returned from 22
24. Invoke failure_protection(K,λ(OP_i)), OP←OP-OP_i+OP_{index}, where index is a set returned by the function failure_protection
25. The function failure_protection can be elaborated as:
26. Function FAILURE_PROTECTION(K,λ(OP_i))
27. Initpos=0, K'←ϕ, index←ϕ, C=0
28. for j=1 to |K| **do**
29. if K_j=λ(OP_i) **then**
30. pos←j
31. $K' \longleftarrow K' \cup [P_k]_{k=1}^{j} : P_k \in K$
32. C←possible number of combinations in K' such that $\sum_{k=1}^{K'} = \lambda(OP_i)$
33. if C=0 **then**
34. index ←pos+1
35. else
36. Choose the combination with minimum number of paths in K'
37. index←positions of constituting in K'
38. return index

Algorithm 1: The α-best fit algorithm.

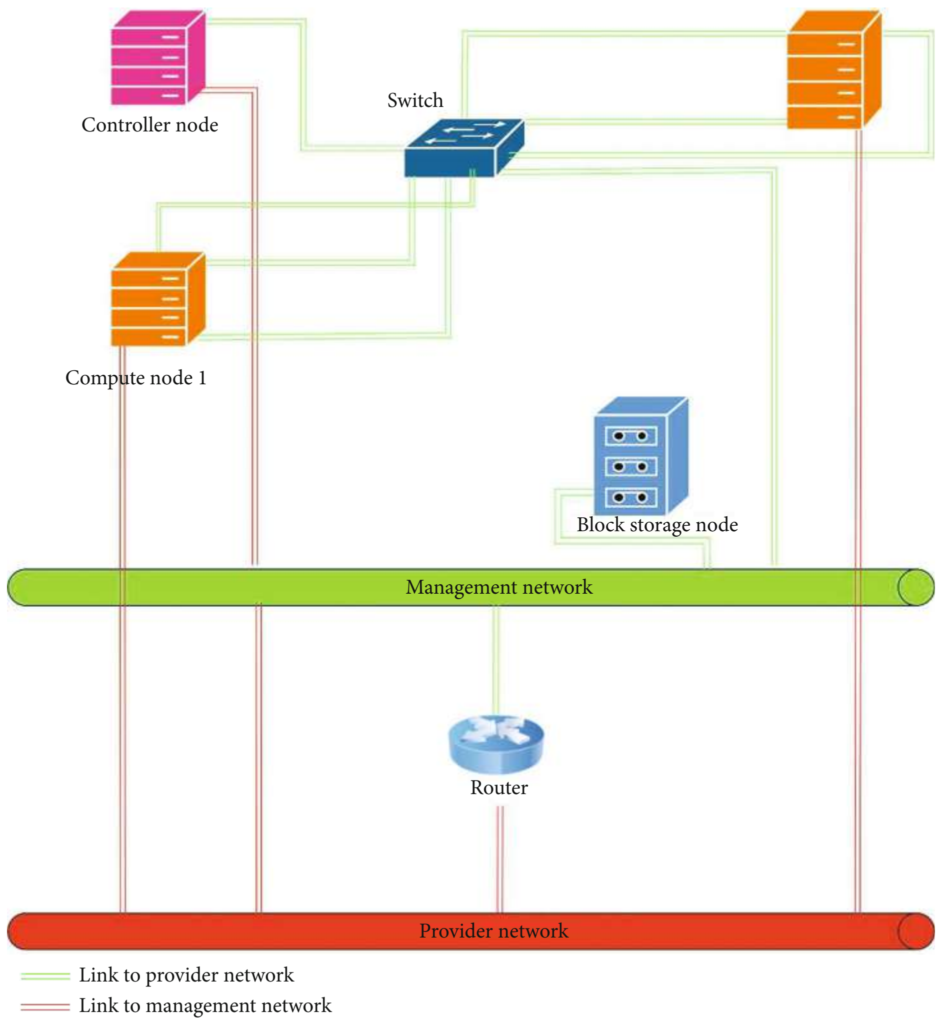


Figure 3: Host networking layout for implementation.

7. Results and Discussion

During the migration of an instance, the service downtime is detected using a packet sniffer (Wireshark). The instances are pinged every millisecond, and the time of their unavailability is measured in terms of the packets not received by them during the duration of the migration process.

It can be inferred from Figure 4 that when the migration of a CentOS 7 instance with 16 GB RAM is conducted, we have a drop for a significant amount of time in the number of ICMP packets transmitted to the OpenStack instance. This dip is shown graphically in the figure and pertains to the service downtime.

We have also measured the service downtime for 50 migrations of “Cirros” instances with different RAM specifications, and a graph displaying avg downtime vs. RAM of the instance being migrated is shown in Figure 5.

The results in Figure 5 show that with increasing RAM size of the instance, the service downtime is increased. This can be justified by considering that with the increase in RAM, the number of memory pages to be transferred in each iteration of the precopy algorithm increases as per equation (1). This leads to higher number of pages dirtied in every iteration. When the WWS is obtained, or the cap on the number of iterations is reached, with increasing RAM, the amount of data to be transferred increases as seen from equations (2) and (3). Hence, there is an increase in the service downtime.

7.1. Results of MPTCP Implementation without Optimization. After the measurement of service downtime in a single path TCP scenario, MPTCP capability is enabled, and the same procedure is carried out using two paths for the MPTCP communication between the compute hosts. The concept for performing this experiment is that with increased number of paths, the VM data will be divided into smaller streams/flows of data and transferred from both the paths at the same time, speeding up the data transfer. The results obtained provide the proof that the assumption made earlier is correct. While the average service downtime with a single path connectivity for data transfer between the compute hosts is 1.12 seconds, by increasing just one path, it is reduced to 0.58 seconds. Hence, it is observed that the service downtime has been reduced by 48% of the original value.

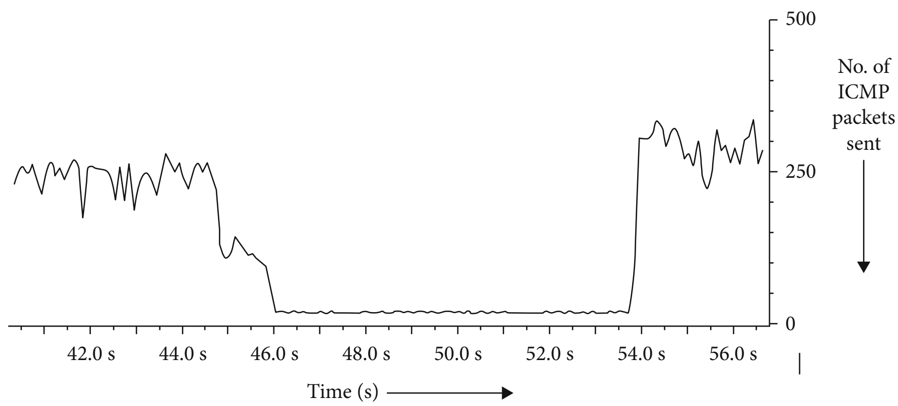


FIGURE 4: Depiction of service downtime in terms of ICMP packets sent during the migration process of an instance.

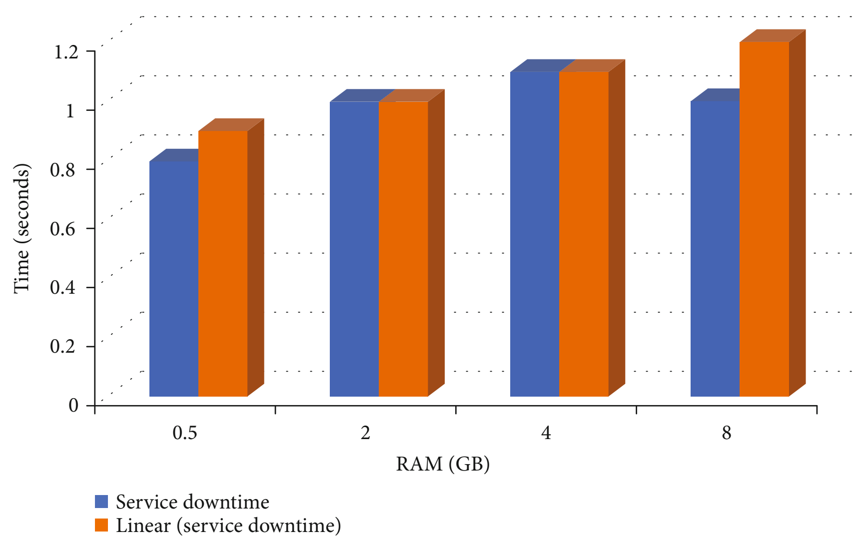


FIGURE 5: The variation of average service downtime with RAM for a “Cirros” instance.

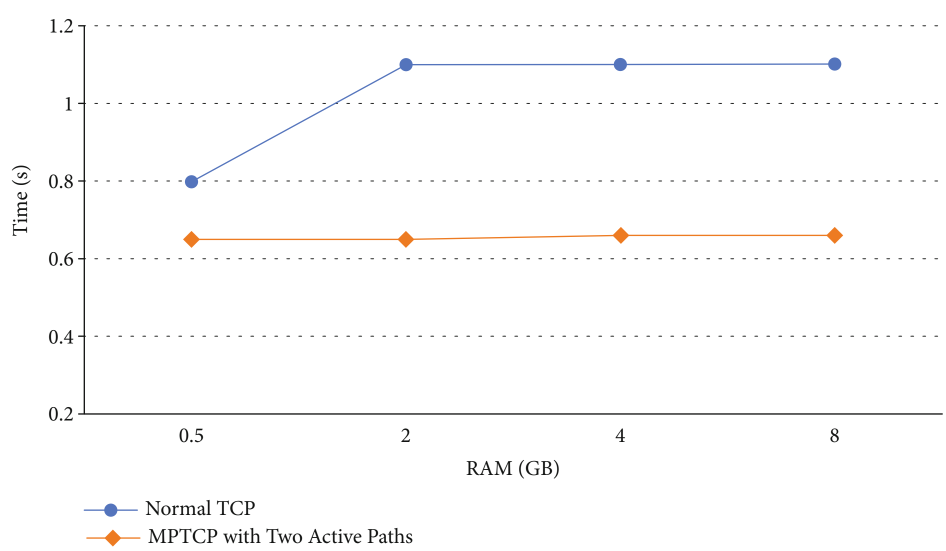
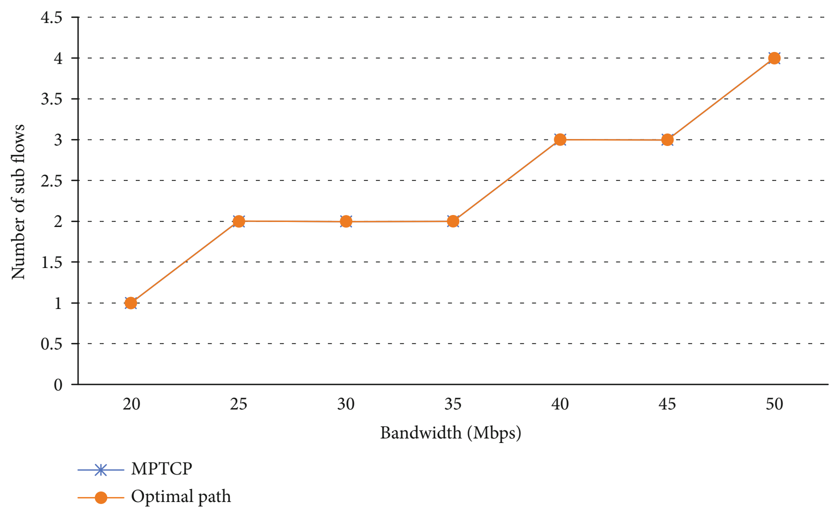
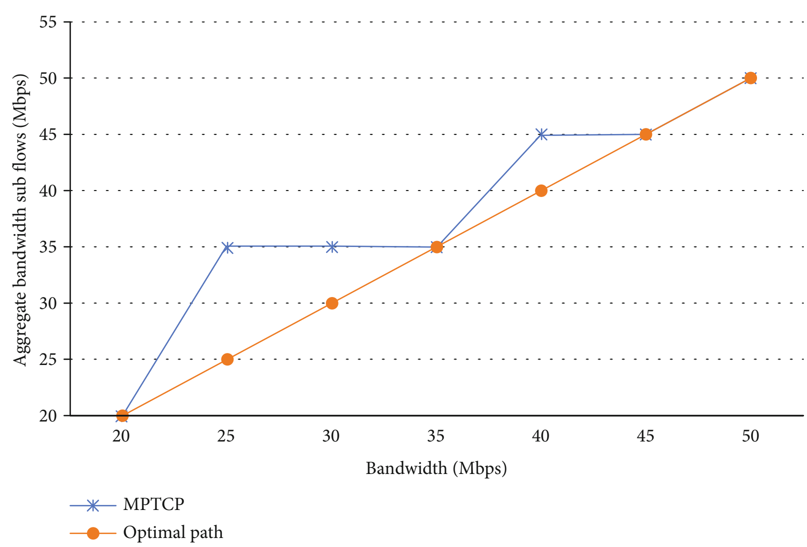


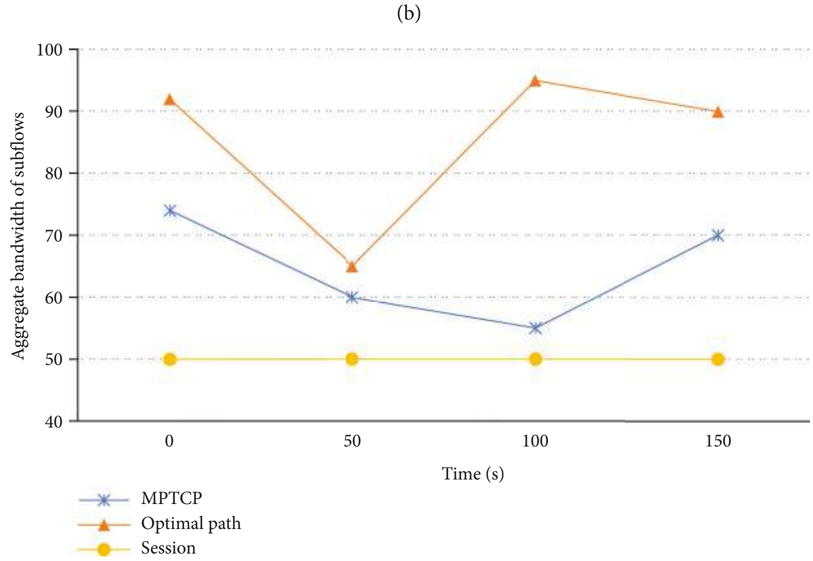
FIGURE 6: The comparison of variation of average service downtime with RAM for a “Cirros” instance using Normal TCP and MPTCP with two paths.



(a)

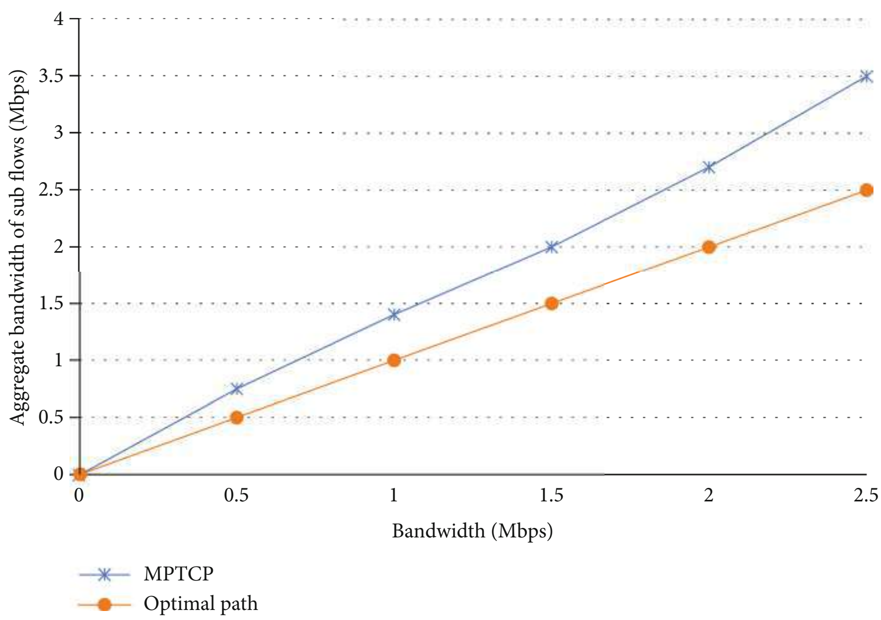


(b)

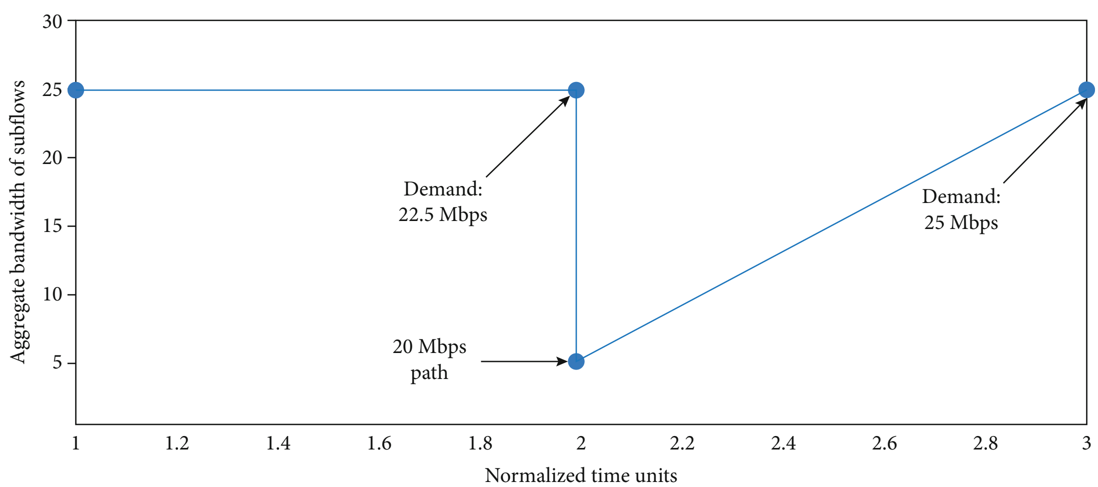


(c)

Figure 7: Continued.



(d)



(e)

Figure 7: Performance comparison of the α-best fit algorithm with SFO [9] in terms of bandwidth utilization. (a) Variation in the number of subflows with the increasing bandwidth demand (α). (b) Variation in the aggregate bandwidth ($\propto^s$) for a fixed number of subflows. (c) The variation of aggregate bandwidth of the sub flows with time. (d) Variation of the aggregate bandwidth of the subflows with the bandwidth demand. (e) The performance of the α-best fit algorithm in case of path failure.

Encouraged by the results in Figure 6, the number of MPTCP paths was increased to three. The service downtime of random migrations using single path TCP, MPTCP with two paths, and then with three paths has been plotted.

While there is a noted improvement with the increase in the number of paths by one, we observed that by adding the third path, the decrease in the service downtime is not satisfactory. The service downtime for random migrations using two path MPTCP connectivity and three path MPTCP connectivity has also been plotted.

The justification for this has been provided in Sections 3 and 4, which elaborate the functioning of the precopy mechanism with MPTCP and the need for optimization of MPTCP. The increase in the number of active paths to three for data transfer caused an increase in network congestion in the OpenStack management network, which reduced the bandwidth utilization. As the value of γ dropped, the factor $(x\gamma)$ dropped as well, thus accounting for the reduced performance as per equation (10).

The α-best fit algorithm proposed in Section 4 is tested for random bandwidth values of the available paths. We have compared the results obtained from the run of the proposed algorithm 4.1 to those obtained after the run of the algorithm given by Joshi and Kataoka for ONS computation [9].

7.2. Results from the α-Best Fit Algorithm. In order to demonstrate the comparison of the performance of the algorithm for optimization of MPTCP subflows proposed in this paper, and of the algorithm given in [9], tests were conducted in different aspects. The first aspect to be considered is the optimal number of subflows for different bandwidth demands. Evident from Figure 7(a), this is pretty much the same for both the algorithms. This means that for different values of $\propto^d$, the same number of subflows will be created using either of the algorithms.

Figure 7(b) shows the variation of the aggregate bandwidth of the subflows ($\propto^s$) with the demand of bandwidth ($\propto^d$) for a given value of ONS, thereby displaying the comparison between the α-best fit algorithm and SFO [9]. It can be inferred from the graph that our proposed algorithm delivers precisely the same amount of bandwidth as required during an MPTCP session.

Figure 7(c) shows the variation in the aggregate bandwidth of the subflows as a function of time. The MPTCP session bandwidth demand is assumed to be fixed at 50 Mbps. It can be seen that the range of variation of the aggregate bandwidth of the optimal paths is narrower, and the extremities are closer to the demand value in case of the α-best fit algorithm.

Figure 7(d) shows the relationship between $\propto^s$ and = $\propto^d$. As $\propto^d$ increases, $\propto^s$ increases in both the cases. However, in the algorithm given in [9], the rise is steeper in comparison with the α-best fit proposed algorithm. This means that the algorithm proposed in this paper adheres to the demand closer than the initial algorithm. Hence, the bandwidth usage is optimized.

In general, the utilization of bandwidth for the same demand in the α-best fit algorithm is more optimized than the initial algorithm. SFO [9] leads to wastage of bandwidth, as more bandwidth than required is taken up for the same data transfer. In case of cloud environments, this makes a huge difference, as the bandwidth is in the order of Gbps.

Finally, the action taken by the α-best fit algorithm in case of path failure is illustrated in Figure 7(e). Consider four paths with bandwidths 5, 10, 15, and 20 Mbps. If the demand is fixed to 25 Mbps, and the number of ONS is two, then initially, the 20 Mbps and 5 Mbps paths are chosen to put the subflows on. However, in case of the failure of the 20 Mbps path, the algorithm invokes the path_failure function and chooses the 15 Mbps and 10 Mbps paths to put the subflows on. Hence, the α-best fit algorithm offers a greater degree of optimization along with failure protection when compared to SFO [9].

8. Conclusion and Future Enhancements

As can be shown, MPTCP can be quite useful in a cloud setting for live VM migration. This capacity reduced service downtime, allowing for enhanced service quality. The results in Section 6 show that adding a single active data channel reduced service downtime by nearly half. In addition to several paths for data transfer between the host machines, the service downtime was decreased from 1.12 seconds to 0.58 seconds. Reduced service downtime enhances end-user experience and overall cloud performance in terms of responsiveness and reliability. We addressed the necessity to optimize MPTCP connections by providing an algorithm 4.1. A more elaborate approach was presented that takes into account not only the optimization of subflows but also the improvement of reliability and performance in case of path breakdown.

The proposed α-best fit approach can be used in conjunction with the precopy in the cloud to improve the efficiency of live VM migration. The α-best fit algorithm saves a lot of bandwidth by optimizing its consumption. The algorithm's failure prevention clause boosts its reliability. Also, the aggregate bandwidth of the α-best fit algorithm's subflows is less than the SFO's subflows [9]; yet, it meets the bandwidth requirements. Our results also show that the aggregate bandwidth of subflows from the α-best fit method changes within a narrow region near the bandwidth requirement.

The service downtime for live migration of instances can be further reduced by using a separate migration method in conjunction with ours. For example, adaptive memory compression [24], CR/TR motion [30], delta compression [5], and CPU scheduling for synchronization have all been added to precopy. These changes will reduce service downtime. Others like the postcopy algorithm [31] or a hybrid strategy that incorporates the benefits of both precopy and postcopy algorithms can also be employed in the cloud using MPTCP in future.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request (vinit.gunjan@ustc.ac.bd).

Conflicts of Interest

The authors declare that they have no conflicts of interest to report regarding the present study.

Authors' Contributions

Kadiyala Ramana contributed to the conceptualization, data curation, formal analysis, methodology, software, and writing-original draft. Rajanikanth Aluvala contributed to the supervision, writing—review and editing, project administration, and visualization. Vinit Kumar Gunjan contributed to the software, validation, writing-original draft, and methodology. Ninni Singh contributed to the supervision, writing—review and editing, funding acquisition, and visualization. M. Nageswara Prasadhu contributed to the data curation, investigation, resources, and software.

References

[1] M. Nelson, "Michael Nelson. Virtual machine migration," US Patent 7, 484, 2009.

[2] B. Sowmiya, E. Poovammal, K. Ramana, S. Singh, and B. Yoon, "Linear elliptical curve digital signature (LECDS) with Blockchain approach for enhanced security on cloud server," *Access*, vol. 9, pp. 138245–138253, 2021.

[3] M. Vashishtha, P. Chouksey, D. S. Rajput et al., "Security and detection mechanism in IoT-based cloud computing using hybrid approach," *Transactions*, vol. 11, no. 5–6, pp. 436–451, 2021.

[4] V. Rajanikanth Aluvalu, U. Maheswari, K. K. Chennam, and S. Shitharth, “Data security in cloud computing using Abe-based access control,” in *Architectural wireless networks solutions and security issues*, vol. 196, no. 1pp. 47–62, Springer, Singapore, 2021.

[5] P. Svärd, B. Hudzia, J. Tordsson, and E. Elmroth, “Evaluation of delta compression techniques for efficient live migration of large virtual machines,” *ACM SIGPLAN notices*, vol. 46, no. 7, pp. 111–120, 2011.

[6] K. K. Chennam, R. Aluvalu, and S. Shitharth, “An authentication model with high security for cloud database,” in *Architectural wireless networks solutions and security issues*, vol. 196, no. 1pp. 13–25, Springer, Singapore, 2021.

[7] S. P. Rm, S. Bhattacharya, P. K. R. Maddikunta, S. R. K. Somayaji, K. Lakshmanna, R. Kaluri et al., “Load balancing of energy cloud using wind driven and firefly algorithms in internet of everything,” *Journal of parallel and distributed computing*, vol. 142, pp. 16–26, 2020.

[8] A. Ford, C. Raiciu, M. Handley, and O. Bonaventure, “Tcp extensions for multipath operation with multiple addresses, draft-Ietf-Mptcp-Multiaddressed-09,” in *Internetdraft*, p. 2012, Internet Engineering Task Force, 2012.

[9] K. D. Joshi and K. Kataoka, “Sfo: Subflow optimizer for mptcp in sdn,” *2016 26th International Telecommunication Networks and Applications Conference*, 2016, pp. 173–178, Dunedin, New Zealand, Dec. 2016.

[10] C. Raiciu, D. Niculescu, M. Bagnulo, and M. J. Handley, “Opportunistic mobility with multipath TCP,” in *Proceedings of the sixth international workshop on MobiArch*, pp. 7–12, Bethesda Maryland USA, 2011.

[11] C. Paasch, G. Detal, F. Duchene, C. Raiciu, and O. Bonaventure, “Exploring Mobile/Wi-fi handover with multipath TCP,” in *Proceedings of the 2012 ACM SIGCOMM workshop on Cellular networks: operations, challenges, and future design*, pp. 31–36, Helsinki Finland, 2012.

[12] S. Deng, R. Netravali, A. Sivaraman, and H. Balakrishnan, “WiFi, LTE, or both? Measuring multi-homed wireless internet performance,” in *Proceedings of the 2014 Conference on Internet Measurement Conference*, pp. 181–194, Vancouver BC Canada, 2014.

[13] B. Arzani, A. Gurney, S. Cheng, R. Guerin, and B. T. Loo, “Impact of path characteristics and scheduling policies on MPTCP performance,” in *28th International Conference on Advanced Information Networking and Applications Workshops*, pp. 743–748, Victoria, BC, Canada, May 2014.

[14] R. Khalili, N. Gast, M. Popovic, U. Upadhyay, and J.-Y. Le Boudec, “MPTCP is not Pareto-optimal: Performance issues and a possible solution,” in *Proceedings of the 8th International Conference on Emerging Networking Experiments and Technologies. ACM*, pp. 1–12, Nice France, 2012.

[15] D. Wischik, C. Raiciu, A. Greenhalgh, and M. Handley, “Design, implementation and evaluation of congestion control for multipath TCP,” *8th USENIX Symposium on Networked Systems Design and Implementation*, vol. 11, p. 8, 2011.

[16] C. Raiciu, D. Wischik, and M. Handley, *Practical congestion control for multipath transport protocols*, University College London, London/United Kingdom, Technical Report/Representative, 2009.

[17] K. Noda and Y. Ito, “Proposal of novel MPTCP congestion control to suppress QoS fluctuation for WebQoE improvement,” in *2018 IEEE 8th international conference on consumer electronics-Berlin*, pp. 1–3, Berlin, Germany, Sept. 2018.

[18] F. Yang, P. Amer, and N. Ekiz, “A scheduler for multipath TCP,” in *2013 22nd International Conference on Computer Communication and Networks*, pp. 1–7, Nassau, Bahamas, Aug. 2013.

[19] F. Yang, Q. Wang, and P. D. Amer, “Out-of-order transmission for InOrder arrival scheduling for multipath TCP,” *28th International Conference on Advanced Information Networking and Applications Workshops*, 2014, pp. 749–752, Victoria, BC, Canada, May 2014.

[20] N. Kuhn, E. Lochin, A. Mifdaoui, G. Sarwar, O. Mehani, and R. Boreli, “DAPS: Intelligent delay-aware packet scheduling for multipath transport,” in *International Conference on Communications*, pp. 1222–1227, Sydney, NSW, Australia, June 2014.

[21] Y.-C. Chen, Y.-s. Lim, R. J. Gibbens, E. M. Nahum, R. Khalili, and D. Towsley, “A measurement-based study of multipath TCP performance over wireless networks,” in *Proceedings of the 2013 Conference on Internet Measurement Conference. ACM*, pp. 455–468, Barcelona Spain, 2013.

[22] H. Chen, H. Kang, G. Jiang, and Y. Zhang, *Coordinating virtual machine migrations in Enterprise data centers and clouds*, Institute of Electrical and Electronics Engineers, New York, 2013.

[23] K. Kumar, K. R. Singamaneni, G. Dhiman, S. Singh, and B. Yoon, “A novel Blockchain and bi-linear polynomial-based QCP-ABE framework for privacy and security over the complex cloud data,” *Sensors*, vol. 21, no. 21, p. 7300, 2021.

[24] H. Jin, L. Deng, S. Wu, X. Shi, and X. Pan, “Live virtual machine migration with adaptive, memory compression,” in *2009 IEEE International Conference on Cluster Computing and Workshops*, New Orleans, LA, USA, Sept. 2009.

[25] K. Ramana, T. Krishna, C. Narayana, and M. P. Kumar, “Comparative analysis on cloud computing and service oriented architecture,” *International Journal of Advanced Research in Technology*, vol. 1, no. 1, pp. 22–28, 2011.

[26] K. Ramana, “A survey on cloud computing and service oriented architecture,” *VSRD International Journal*, vol. 1, no. 9, 2011.

[27] S. Nathan, P. Kulkarni, and U. Bellur, “Resource availability based performance benchmarking of virtual machine migrations,” in *Proceedings of the 4th ACM/SPEC International Conference on Performance Engineering*, pp. 387–398, Prague Czech Republic, April 2013.

[28] T. Wang, Y. Quan, X. S. Shen, T. R. Gadekallu, W. Wang, and K. Dev, “A privacy-enhanced retrieval technology for the cloud-assisted internet of things,” *IEEE transactions on industrial informatics*, p. 1, 2021.

[29] “How to calculate bandwidth utilization using snmp,” pp. 10–26, 2005, http://www.cisco.com/c/en/us/support/docs/ip/simple-network-management-protocol-snmp/8141-calculate-bandwidth-snmp.

[30] H. Liu, H. Jin, X. Liao, H. Liting, and Y. Chen, “Live migration of virtual machine based on full system trace and replay,” in *Proceedings of the 18th ACM international symposium on High performance distributed computing*, p. 101, Garching Germany, June 2009.

[31] M. R. Hines and K. Gopalan, “Post-copy based live virtual machine migration using adaptive pre-paging and dynamic self-ballooning,” in *Proceedings of the 2009 ACM SIGPLAN/SIGOPS international conference on Virtual execution environments*, pp. 51–60, Washington DC USA, March 2009.

Hindawi
Mobile Information Systems
Volume 2022, Article ID 7632892, 36 pages
https://doi.org/10.1155/2022/7632892

Review Article

Autonomous Vehicles and Intelligent Automation: Applications, Challenges, and Opportunities

Gourav Bathla,[1] **Kishor Bhadane**,[2] **Rahul Kumar Singh**,[1] **Rajneesh Kumar**,[3] **Rajanikanth Aluvalu**,[4] **Rajalakshmi Krishnamurthi**,[5] **Adarsh Kumar**,[1] **R. N Thakur**,[6] **and Shakila Basheer**[7]

[1]*School of Computer Science, University of Petroleum and Energy Studies, Dehradun, India*
[2]*Electrical Engineering, Amrutvahini College of Engineering, Sangamner, Ahmednagar, India*
[3]*Software Architecture Department, Airtel X Labs, Gurugram, India*
[4]*Department of Information Technology, Chaitanya Bharathi Institute of Technology, Hyderabad, India*
[5]*Department of Computer Science and Engineering, Jaypee Institute of Information Technology, Noida, India*
[6]*LBEF Campus, Kathmandu, Nepal*
[7]*Department of Information Systems, College of Computer and Information Science, Princess Nourah Bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia*

Correspondence should be addressed to Gourav Bathla; gbathla@ddn.upes.ac.in, Adarsh Kumar; adarsh.kumar@ddn.upes.ac.in, and R. N Thakur; rn.thakur@lbef.edu.np

Received 5 April 2022; Accepted 5 May 2022; Published 6 June 2022

Academic Editor: M. Praveen Kumar Reddy

Intelligent Automation (IA) in automobiles combines robotic process automation and artificial intelligence, allowing digital transformation in autonomous vehicles. IA can completely replace humans with automation with better safety and intelligent movement of vehicles. This work surveys those recent methodologies and their comparative analysis, which use artificial intelligence, machine learning, and IoT in autonomous vehicles. With the shift from manual to automation, there is a need to understand risk mitigation technologies. Thus, this work surveys the safety standards and challenges associated with autonomous vehicles in context of object detection, cybersecurity, and V2X privacy. Additionally, the conceptual autonomous technology risks and benefits are listed to study the consideration of artificial intelligence as an essential factor in handling futuristic vehicles. Researchers and organizations are innovating efficient tools and frameworks for autonomous vehicles. In this survey, in-depth analysis of design techniques of intelligent tools and frameworks for AI and IoT-based autonomous vehicles was conducted. Furthermore, autonomous electric vehicle functionality is also covered with its applications. The real-life applications of autonomous truck, bus, car, shuttle, helicopter, rover, and underground vehicles in various countries and organizations are elaborated. Furthermore, the applications of autonomous vehicles in the supply chain management and manufacturing industry are included in this survey. The advancements in autonomous vehicles technology using machine learning, deep learning, reinforcement learning, statistical techniques, and IoT are presented with comparative analysis. The important future directions are offered in order to indicate areas of potential study that may be carried out in order to enhance autonomous cars in the future.

1. Introduction

Autonomous vehicles (AVs) and associated technologies have rapidly gained the attention of the research community. AV utilizes sensorial technologies such as computer vision, odometry, GPS, laser lights, sensors, and a mapping system to navigate. These technologies can be used to determine environments and locations and recognize the suitable routes amid obstacles and signage [1, 2]. AVs are supposed to minimize vehicle accidents, enhance the flow of traffic and movability, reduce the utilization of fuel, be free from driving, and facilitate business operation and transportation [3–6]. Despite the massive potential advantages, there are many unsolved safety, security, legal and regulatory, social, ethical, and technology issues [7–10]. In the AV system, it is expected to solve all the problems to avoid failure. In this survey, design, hardware, AI-based, and safety issues and current solutions of autonomous vehicles are discussed. Furthermore, scope of improvement in these solutions is provided as directions for AV research community.

Intelligent software and tools are required for efficient design and development of AVs. These tools are used during path planning, object detection, perception, act, operational testing, and risk assessment phases. In this survey, comprehensive analysis of tools is provided. Various tools and frameworks such as SysWeaver, SysAnalyzer, AutoSim, Flow, OpenCV, JESS, FuzzyJ, AuRa, and PaddleCV are analyzed based on functionality and applications. The latest releases and versions such as AutoSim 200, OpenCV 4.5.5, and FuzzyJ 1.2.2 are discussed so that researchers can contribute in various open-source tools and frameworks.

Since the middle of the 1980s, several car companies, research institutes, universities, and industries worldwide have studied and developed AV. To promote AV technology, there are well-known competitions. For example, 2007 Urban Challenge and 2005 DARPA Grand Challenge are organized by Defense Advanced Research Projects Agency (DARPA). In the USA, the first competition of DARPA Grand Challenge was organized in which AV was required to navigate 142 miles long desert track within 10 hours. In the first few miles, all the AV failed to navigate. The second competition DARPA Grand Challenge was organized in 2005 in which AV was required to navigate 132 miles long track that contains mountain passes, approximately 100 right and left turns, three narrow tunnels, and flat and dry lake beds [11]. In this competition, 4 AVs among 23 finalists completed the track in time. Stanley of Stanford University secured first place the AV, and second and third place was secured by AV of Carnegie Mellon University Sandstorm and Highlander, respectively. The third competition DARPA Urban Challenge was organized in 2007 in California, USA. AV was required to navigate 60 miles' long track containing human-driven cars and virtual urban atmosphere, and 6-hour time limits [12]. In this competition, 6 AVs among 11 finalists completed the track in time. In this competition, first place was secured by Boss AV of Carnegie Mellon University, Junior AV of Stanford University claimed second place, and Odin AV of Virginia Tech finished in third place. However, these competitions did not include tough challenges as presented in everyday traffic life. After the DARPA competition, there are several trials and competitions performed by different organizations. Some examples of these competitions are as follows: ELROB from 2006 to till now [13], (SparkFun), the AV Competition from 2009 to 2017, and Intelligent Vehicle Future Challenge from 2009 to 2013 [14]. In recent time, both industry and academic community accelerate the research work in the field of AV. Some notable companies which are performing cutting-edge research in AV are Google, Argo AI, Nvidia, Mercedes Benz, Ford, Volvo, Lyft, and Aptiv. Some universities such as Virginia Tech, MIT, Carnegie Mellon University, Stanford University, and University of Ulm have also conducted research in AV.

According to SAE J3016 standard, there are six levels in vehicle automation from 0 to 5 [15, 16]. Each level has its own functionalities such as (i) Level 0: the individual operator is in-charge of all operating activities (No Automation), (ii) Level 1: the vehicle is controlled by a human driver, but the automation system assists in operating (Assistance to Drivers such as Tesla AutoPilot) [17], (iii) Level 2: the vehicle used automated features but the control and environment of the driving process require human intervention (Partially Automated Driving such as Tesla AutoPilot) [17], (iv) Level 3: the human driver should be ready to take control of the vehicle at any moment (Automated Conditional Driving), (v) Level 4: under some conditions, the automation system can drive the car automatically, but the human operator will still be able to control it (high-level automation of driving such as Waymo driverless cars [18]), and (vi) Level 5: under all the conditions, the automation system can drive the car automatically, but the human operator will be able to control it (fully automation Waymo driverless cars [18]). The driving choices of the vehicle consist of three different levels: tactical level (comprising lane-keeping and lane-changing), operational level (consisting of break and pedal control), and strategic level (containing routing) [19]. The tactical and operating controls are further divided into lateral and longitudinal control categories [19]. Several researchers and orgranizations are trying to achieve Level 5 automation. These challenges are covered in this work comprehensively.

AI is a critical technology for efficient autonomous vehicles functionality. AV utilizes AI and sensory technologies and minimizes the risk. In the field of object detection, computer vision, and semantic segmentation, deep learning has been very effective. On several common object detection datasets, deep learning techniques have raised the standard [20, 21] and have been commonly used in AV especially detection of people [22, 23], vehicle [24, 25], road signal [26, 27], and traffic lights [28, 29]. AI techniques play an important role (perception, decision-making, localization, and mapping) in a given area to improve the performance of AV [30]. Perception is described as an AV's repeatedly scanning and monitoring the environment with sensors, like human vision [31]. Several deep learning approaches have been utilized for perception and are considered one of AV's challenging areas [32]. AI also plays an important role in AV decision-making, such as automatic parking [33] and path planning [34]. The computational problem of creating or updating a map of an uncertain area is known as simultaneous localization and mapping (SLAM) [35].

The significant contributions of this work are as follows:

(i) A comprehensive survey of AI and IoT-based autonomous vehicles research works is carried out.

(ii) Safety standards and challenges for autonomous vehicles are discussed with currently available solutions.

(iii) Research and development challenges for AI and IoT-enabled autonomous vehicles are presented.

(iv) Tools and frameworks for autonomous vehicles used by researchers and organizations are highlighted.

(v) Recent advancements in autonomous vehicles using cloud computing, machine learning, and deep learning are discussed as future directions for researchers and organizations.

This work is organized as follows: the research methodology, data collection, and analysis methods are discussed in Section 2. Section 3 presents the theoretical background and recent artificial intelligence trends for autonomous vehicles. Section 4 presents the recent studies and developments over autonomous driving decision systems. Section 5 offers the current observations in safety standards and ethical challenges in autonomous vehicles. Section 6 presents the importance of artificial intelligence in IoT-enabled autonomous vehicles in recent studies. Section 7 shows the research challenges in integrating artificial intelligence-enabled autonomous vehicles. The intelligent system software and tools used for autonomous vehicles are elaborated in Section 8. Section 9 presents the artificial intelligence-enabled testing techniques for autonomous vehicles. Section 10 presents the importance of artificial intelligence in autonomous electric vehicles and associated applications. Section 11 shows the role of artificial intelligence in power train energy management and electric vehicles. Section 12 presents the autonomous driving subsystems in electric vehicles. Section 13 presents the advanced technologies and their roles in autonomous vehicles. Here, importance is drawn towards integrating artificial intelligence and other advanced technologies with autonomous vehicles. Finally, Section 14 concludes the paper with future directions.

2. Survey Materials and Methods

This section explains the survey method and data collection and analysis followed during the survey. Details are presented as follows.

2.1. Survey Research Method. The following survey methodology is followed in this work to survey artificial intelligence and its importance to autonomous vehicles.

This work has focused on those contributions that integrate artificial intelligence with autonomous vehicles. Furthermore, those systems and proposed approaches that apply artificial intelligence or its variant to improve the autonomous vehicle's experiences are taken up for study, in-depth, and feature-based analysis.

This work has studied and presented the analysis of how artificial intelligence is helpful in smartly operating the different types of autonomous vehicles, integrating IoT with autonomous vehicles, and handling the operation, coordination, communication, decisional systems, and data handling processes.

Furthermore, the use of advanced technologies and artificial intelligence in autonomous vehicles is explored in this work.

2.2. Survey Data Collection and Analysis. This section discusses the process of article collection, analysis, filtering, and survey preparation. Figure 1 shows the complete process in detail. The essential phases of this process are briefly explained as follows:

Step 1: in the first step, articles are collected using Google Scholar and reputed publisher's search engines. This mainly includes Elsevier, IEEE, Springer, ACM, Wiley, Taylor and Francis, IET, Hindawi SAGE, and MDPI. To search an article, keyword-based search is applied that mainly include "artificial intelligence for autonomous vehicles," "artificial intelligence for unmanned vehicles," "artificial intelligence and advanced technologies for autonomous vehicles," "surveys on artificial intelligence and autonomous vehicle," "IoT and artificial intelligence for autonomous vehicles," "artificial intelligence in autonomous driving," "autonomous vehicles," "autonomous underwater vehicles," "machine learning and autonomous vehicles," "autonomous vehicles applications," and "autonomous electric vehicles."

Step 2: after selecting the articles, the key findings, advantages, disadvantages, and significant challenges that still need to be addressed are observed. The key findings include (i) autonomous vehicles and their classification, (ii) artificial intelligence role in autonomous vehicles or autonomous electric vehicles, (iii) role of artificial intelligence in autonomous driving or decision systems, (iv) ways to integrate artificial intelligence with autonomous systems, (v) intelligent tools and frameworks, (vi) training and testing autonomous levels and systems using artificial intelligence, (vii) artificial intelligence in on-road object detection and vehicle control system, and (viii) artificial intelligence and green energy solution for autonomous systems.

Step 3: after studying the key findings, the articles were classified into four categories, implementation, survey, discussion, and tutorial. An implementation-based article contains the integration of artificial intelligence in simulating or implementing autonomous experiences. The survey articles' category includes the significant studies of artificial intelligence in autonomous systems. Discussion and tutorial articles provide a detailed explanation and classifications of autonomous vehicles and the importance of artificial intelligence and autonomous vehicles.

3. Theoretical Background

This section introduces the recent studies on artificial intelligence and its application in autonomous vehicles. Details of similar approaches are briefly discussed in subsequent sections. Furthermore, this section discusses the comparative analysis of studies, surveys, and developments to observe the advantages, disadvantages, and future directions.

3.1. Recent Artificial Intelligence Trends for Autonomous Vehicles and Driving Systems. This section explores current advancements, surveys, and practices in artificial intelligence

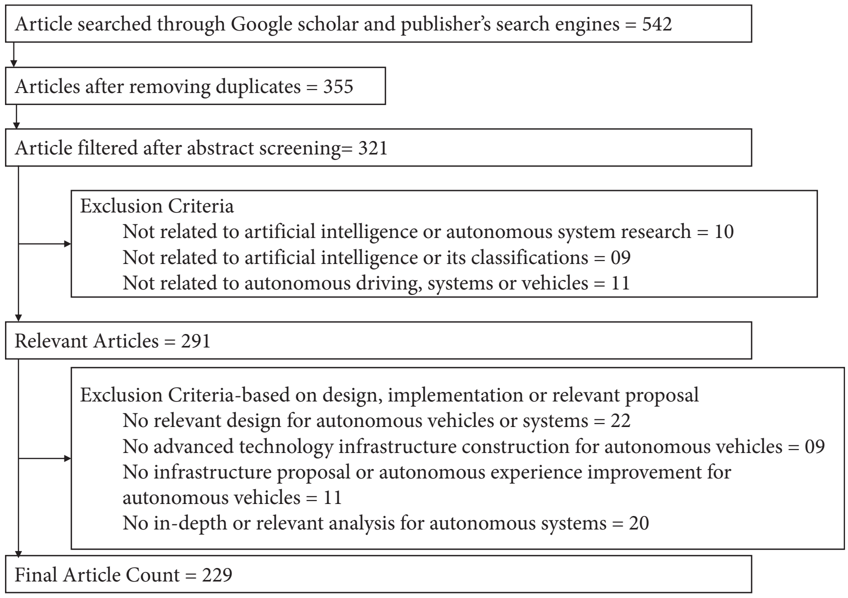


Figure 1: Autonomous vehicles' safety and security.

technology for AVs. This highlights the importance of AI for AVs. The details are as follows.

Khayyam et al. [36] discussed to integrate artificial intelligence with autonomous systems. Here, various types of artificial intelligence, their importance, and details of autonomous driving are explained. Furthermore, the development of the industrial revolution with the integration of artificial intelligence and IoT are discussed to address the high-performance embedded system in the autonomous industry. Additionally, the importance of cloud and edge computing in independent infrastructure is concerned, keeping the center's technical challenges such as delay, bandwidth, and security. Ma et al. [30] conducted an in-depth analysis of artificial intelligence in autonomous vehicles. In observations, it has been found that the current practices of using artificial intelligence in autonomous vehicles are limited to object detection and tracking. The object includes the traffic signs, on-road vehicles, on-road moveable or stationery items, and pedestrians. Here, the critical challenges to artificial intelligence for autonomous applications such as (i) sensor integration and performance issues to artificial intelligence and autonomous systems, (ii) complexities and uncertainties to autonomous and associated complex systems and recent developments, (iii) fine-tuning and optimization approaches, (iv) hardware concerns, and (v) artificial intelligence-integrated opportunities and future research directions are discussed. Cunneen et al. [37] surveyed to elaborate the use of artificial intelligence in various systems of autonomous vehicles. Here, the primary focus is drawn towards using artificial intelligence-integrated conceptual framing that supports governance and regulation. So far, little attention is drawn towards the conceptual frame. This work has discussed the role of conceptual structure in anticipatory governance. This role increases the accuracy and impact of safety concepts and directions in autonomous systems. This work is more of theoretical development and can be extended to discuss the conceptual framing in various applications and use-cases.

Table 1 shows the comparative analysis of artificial intelligence-integrated autonomous systems surveys. This comparative analysis is performed mainly over fake intelligence-related domains relevant to autonomous systems or vehicles. These surveys discuss various challenges, solutions, and application scenarios. For example, attack and defense analysis is examined to identify the significant cyberattack scenarios to autonomous vehicles and systems.

3.2. Literature-Based Research Challenges to Autonomous Vehicles and Related Studies. This section explores the recent research challenges to autonomous vehicles. Details are presented as follows [21–25, 37–41].

Autonomous vehicles offer better driving decisional spectrum that avoids intoxication, distraction, fatigue, and nability to make timely decisions. All of these factors are associated with the ability of the technologies to outperform the human driving decisions abilities [37]. Thus, advancements in technology to avoid errors and give real-time responses are significant challenges for AI-integrated autonomous vehicles. Various research works have discussed the importance of the safety and performance metrics of autonomous vehicles. These metrics should include sensor error, programming bugs, unanticipated events and entities, cyberattack and threat probabilities, and hardware failures. Development of these metrics and analyzing these metrics in a real-time environment are essential to address in the future. Table 2 highlights the comparative analysis of autonomous driving systems.

There are various categories of cyberattacks, including attacks over control systems, driving system components, vehicle-to-everything network communications, and risk assessment and survey systems. The primary attack

Table 1: Comparative analysis of artificial intelligence-integrated autonomous vehicle system surveys.

Author	Year	A	B	C	D	E	F	G	H	I	J	Key findings	Challenges and future directions
Khayyam et al. [36]	2020	✓	×	✓	✓	✓	✓	×	×	×	✓	In this research work, various artificial intelligence and autonomous vehicle experiences are discussed. The autonomous vehicles and their developments are associated with industrial revolution, cloud and edge computing environments, IoT networks, and sensors. Furthermore, the research directions in integrating artificial intelligence with autonomous vehicles are explored.	This survey is a short discussion over artificial intelligence field and autonomous vehicles. This work can be extended to address the use of advanced technologies (such as blockchain, serverless computing, and unmanned aerial vehicle control) with autonomous vehicles. Furthermore, the computational complexities, security, and advanced technology integration can be studied in detail.
Ma et al. [30]	2020	×	✓	✓	✓	×	✓	✓	✓	×	✓	This is an in-depth survey covering the importance of artificial intelligence-integration with autonomous vehicles. Here, attentions are drawn to discuss the properties of autonomous vehicles, challenges, and comparative analysis of existing systems in this direction. Furthermore, the solution or architectures are proposed or discussed to address the existing challenges in similar systems.	This work can be extended to propose solutions or conducted in-depth studies over challenges such as complexities and uncertainties. The complexities and uncertainties can be resolved with advanced systems such as artificial intelligence-integrated decisional support system. Likewise, other issues can be addressed.
Cunneen et al. [37]	2019	×	✓	✓	✓	×	✓	✓	×	×	×	This is a detailed survey addressing the artificial intelligence and autonomous vehicle concerns. The role of artificial intelligence in decisional making and decisions limitations for autonomous vehicles is discussed. Furthermore, the shortcomings and safety discussions in artificial intelligence-intergrated autonomous vehicles are discussed.	This is a theoretical development and discussion over role of artificial intelligence for autonomous vehicles. This work can be extended to include the discussions over complexities and amplifications of emerging technologies in autonomous systems. Furthermore, risks associated with conceptual frameworks for deployment of autonomous vehicles are another important aspect to explore in detail.
Osório and Pinto [38]	2019	×	✓	✓	✓	×	×	✓	×	×	×	This work has discussed the uncertainty handling, unpredictability control and decision-making power of artificial intelligence for autonomous vehicles. Here, risks associated with successful manipulation of high levels of uncertainty are discussed.	This is theoretical development and discussion over role of aritificial intelligence in information uncertainty and manipulability. The importance is drawn towards how the quality of decision processing artificial intelligence in improving the efficiency and welfare is important. However, this work can be extended to discuss the system complexities and manipulation proof of artificial intelligence for pretecting the lives and welfare of society.
Li et al. [39]	2018	×	✓	✓	✓	✓	×	✓	✓	✓	×	This work has surveyed the issues of traffic in various countries and discussed the role of connected and autonomous vehicles to tackle these issues. The key benefits and need of artificial intelligence are discussed. In artificial intelligence, special focus is drawn towards the role of deep and reinforcement learning for autonomous vehicles.	This work can be extended to include more use cases that integrate advanced technologies such as blockchain, IoT, edge, fog, cloud, and serverless computing. Furthermore, more algorithmic approaches in artificial intelligence can be explored.

Table 1: Continued.

Author	Year	A	B	C	D	E	F	G	H	I	J	Key findings	Challenges and future directions
Kunze et al. [40]	2018	✓	×	✓	✓	✓	×	✓	×	×	×	This work has surveyed artificial intelligence-enabled long-term autonomy systems and features. The major areas where artificial intelligence plays an important role include navigation and mapping, knowledge representation and reasoning, perception, planning, learning, and interaction.	This work has discussed the different domains and brief about related work in these domains. The work can be extended to have in-depth discussions, identifying challenges in existing proposals, and develop discussions over possible solutions to challenges.
Kim et al. [41]	2021	×	✓	✓	✓	×	×	×	×	✓	✓	This work has identified the attack and defense strategies from 151 articles and conducted a comprehensive survey. The comprehensive survey investigates the autonomous vehicles in three major categories including autonomous control system, autonomous driver system and associated components, vehicle-to-everything network and communications.	Cyberattacks and defense strategies are explored which have grave impact over human life and safety. This work can be extended to discuss the complexities of proposed systems and related cyberattacks and defense strategies. Furthermore, attack and defense scenarios can be explored for in-depth analysis.

A: short survey, B: long and in-depth survey, C: autonomous driving or assistance, D: artificial intelligence, E: machine, deep, or reinforcement learning, F: artificial intelligence and IoT network for autonomous systems/driving/vehicles, G: artificial intelligence applications for autonomous, H: artificial intelligence and sensors for autonomous vehicles, I: artificial intelligence-based data analysis in autonomous systems, and J: other advanced technologies (such as cloud/edge/fog computing, blockchain, and serverless).

categories that need to be explored and researched include sensor attacks, mobile application-based vehicle information system attacks, IoT infrastructure-based attacks, physical attacks, and side-channel attacks. Furthermore, artificial intelligence is used in cybersecurity in attack identification. Autonomy architecture is another interesting aspect. In architecture, autonomous systems integrating sensors and actuators, control functions, vehicular monitoring environment, external control factors, speed, visibility, and object detection are critical subsystems to observe and explore.

With the increase in autonomous vehicles, the communication overheads will also increase. This causes delay or loss of packets which indirectly decreases the performance or increases the error in communication. Autonomous vehicles and their implementation are critical to human life.

The limitations of existing works are that extensive analysis of recent developments such as use of deep learning and IoT are not covered. Furthermore, discussion on intelligent tools and software are essential that is not included in existing works. Furthermore, developments of efficient simulation are required. Object detection, path planning, sensors, and use of cloud computing should be improved to develop autonomous vehicles.

4. Autonomous Driving Decision Systems

In the past, substantial progress in the development of control theory and the implementation of its findings are observed for an extended period. Increasingly, products come with built-in computers that make decisions. Recent years have seen a significant increase in the use of walking robots. Robots that look like humans are perceived as friendlier and more accepted in society. This strategy can also be employed in self-driving automobiles. The idea of giving the illusion of human traits to robots is gaining traction. The types of robots that can be developed based on the drive type include [45] (i) wheeled robots, which are employed primarily for light work (for example, following a particular line or path), (ii) manned or unmanned tracked or crawling robots, which can maneuver in either man-made or natural settings, (iii) programmable robots that are capable of performing a specified sophisticated task in a natural and industrial setting, and (iv) combination of robots for different tasks. Autonomous driving is both a bold idea and a very feasible technological accomplishment. Many technical companies (including Audi, Ford, Tesla, Renault, Waymo, and ride-sharing firms Lyft and Uber) are battling to overcome technological hurdles and allow an altogether new style of driving that will surprise and thrill consumers. This section explores the importance of various technologies that are associated with autonomous driving systems. Details are presented as follows.

4.1. Advanced Technologies and Autonomous Driving Systems. Autonomous vehicles are becoming more intelligent thanks to recent advances in artificial intelligence and deep learning. Current AI techniques are used in most contemporary self-driving car components [43]. Driverless vehicles are complex systems for moving people or cargo. Like introducing AI-powered autonomous automobiles on public highways, introducing AI-powered autonomous vehicles on public roadways presents many challenges. Using the current framework and explainability of neural networks, it is tough to demonstrate the functional safety of these vehicles. To make use of deep learning methods, you will need enormous training datasets and plenty of

Table 2: Comparative analysis of autonomous driving systems or experiences.

Author	Year	A	B	C	D	E	F	G	H	I	J	Key findings	Challenges and future directions
Grigorescu et al. [42]	2020	×	✓	✓	✓	×	✓	✓	✓	×	×	This work has discussed the importance of deep learning in autonomous driving. Here, a set of challenges in autonomous driving systems are discussed that can be overcome with deep learning and artificial intelligence approaches.	This work can be extended to discuss the role of deep learning while integrating it with other autonomous driving assisting infrastructure. This includes advanced infrastructure aspects including IoT, cloud, and blockchain technologies.
Ning et al. [43]	2021	×	✓	✓	×	×	✓	✓	×	×	✓	This work has researched and categorised the present state of automated driving, and devised a taxonomy for self-driving vehicles. Furthermore, this work has proposed an hybrid human and artificial intelligence architectural concept. Autonomous driving was also summarized by the design of the vehicle itself. As with self-driving car technology, this work developed a taxonomy of autonomous driving technologies. We valued human information integrity and machine-human interaction above simple driver replacement.	This work can be extended to include safety standards and discussions. The proposed hybrid architecture includes the safety monitoring system which can be extended with other advanced technologies including drones and cloud computing environments. Furthermore, data security and privacy can be handled with blockchain technology as well. Further performance issues can be explored with advanced network such as 5G networks.
Kumar et al. [44]	2021	×	×	✓	×	×	✓	✓	×	✓	×	This work mainly discusses the use of drones in autonomous systems. Furthermore, the anticollision strategies for drone movement and traffic monitoring are discussed. Results are analyszed by varying the drones and on-road vehicles.	This work is relevant to real-time deployment and observations of autonomous systems. However, connection between drones and autonomous vehicles need to be explored in detail.
Kim et al. [45]	2020	✓	×	✓	×	×	✓	✓	×	✓	×	This research looked at system settings, components, operations, and real scenarios for autonomous vehicles, smart UAVs, and drones. The main research issues and security concerns for AI-based attacks are addressed.	This work is a short survey discussing the important technological aspects and their role in autonomous vehicles, driving and systems. This work can be extended to have in-depth discussions over technological aspects.
Wang et al. [46]	2020	×	✓	✓	×	×	×	×	×	✓	✓	This blockchain-based architecture supporting safety and security to autonomous vehicles and its networks.	Work can be extended to include smart-contract for different systems and subsystems for better credibility.

A: short survey, B: long and in-depth survey, C: artificial intelligence-based proposal/discussion, D: deep learning, reinforcement learning, F: AI-enabled motion control systems for autonomous vehicles, G: safety systems for AI-integrated autonomous driving, H: computational, hardware, and deployment challenges supporting AI technology, I: advanced technological solutions (drone, blockchain, and IoT), and J: AI-integrated safety or cyberattack protection systems/proposal or discussion.

processing capacity. This article presents an overview of deep learning for autonomous vehicle use. When designing AI-based self-driving vehicles, understanding the requirements and capabilities of the system serves as a blueprint. Grigorescu et al. [42] thoroughly discuss the deep learning models utilized in autonomous vehicle driving. Recurrent and convolutional neural networks and deep reinforcement learning are included in AI-based self-driving architectures and it is further elaborated in Section 13.9. The sampled driving approach starts with these tactics, which serve as the foundation for how individuals perceive, plan, and behave in the situation. Modular perception-planning-action pipeline end-to-end systems are used for deep learning techniques. The research described here unveils deep learning and AI techniques for autonomous driving. Ning et al. [43] provide a taxonomy of current independent driving designs. After that, a proposal is made to integrate hybrid human-artificial intelligence into a semiautonomous driving system. This work has proposed a theoretical architecture based on hybrid human-artificial intelligence for improved usage. With this architecture, it is easy to categorize and overview potential technologies while illustrating benefits. In the proposal, research challenges associated with autonomous driving are also discussed.

Artificial intelligence and drone-based system to monitor on-road driving: Kumar et al. [44] discussed the importance of drones and Internet of Vehicles (IoV) for traffic monitoring. It has been observed that traffic cameras are among the drawbacks of incomplete data collection, restricted medical assistance, and inability to follow vehicles after an accident. Artificial intelligence-integrated object detection and the drone-based system collects and transmits data about commuters, traffic patterns, and vehicle activity to various agencies for traffic planning. The authors have proposed software-defined networking (SDN) controlled drone networks to reduce control overhead and effectively handle on-road vehicle observation scenarios. Kim et al. [45] have investigated system settings, components, operations, and actual circumstances for significant application types, including autonomous vehicles, intelligent UAVs, and drones. This study has also provided instances and scenarios where autonomous vehicles can be used in public and private places with different viewpoints and circumstances. The primary research problems and security concerns about future AI-based attacks have been thoroughly discussed.

Artificial intelligence, machine learning, and cloud computing and autonomous driving: Yaqoob et al. [47] present a cross-domain solution for the Cognitive Internet of Vehicles (C-IoV) based on global AI fog computing and IoT AI service architecture. Furthermore, it explores the C-IoV for autonomous driving from the viewpoints of what, where, and how to compute. This work has used the Internet of Vehicles real-time task deployment to illustrate how the proposed approach works better than the existing alternatives. This work has presented a multilayered architecture for infrastructure assistance to autonomous vehicles and systems. Machine learning, cloud computing, fog computing, and IoT layer processing are proposed for autonomous vehicles.

Data security and autonomous driving systems: Ren et al. [48] reiterated that AVs will simplify driving, reducing driver fatigue and traffic accidents. The major credit goes to advances in artificial intelligence and Internet of Things; autonomous driving has come long. Despite its numerous benefits, it also brings new challenges, chief among them security. The authors analyze the security concerns of autonomous driving from many angles, focusing on how they are experienced, navigated, and managed. We describe the dangers and the associated defensive measures. Define emerging security risks, including deep learning-based self-driving cars. Ren et al. identified three kinds of possible assaults against current AVs and provided defensive strategies for each. Ren et al. investigated AVs' future, self-driving cars based on deep learning algorithms, and the new security risks. Ren et al. examined deep learning model security risks such as system faults, adversarial examples, model privacy, and hardware security. Singandhupe and La [49] present that robotics, augmented and virtual reality, and self-driving cars are interested in SLAM. SLAM collects information about the environment and then estimates the robot's location. While SLAM has been existed for over 30 years, it is responsible for the decade's self-driving cars. Singandhupe and La, in a concise manner, describe how SLAM techniques have contributed to the automotive industry. Singandhupe and La first sought to examine the various localization techniques available and to evaluate the state-of-the-art methodology. Finally, Singandhupe and La addressed the concerns about autonomous vehicle security and how this matters. Singandhupe and La discussed several SLAM techniques for autonomous driving using the KITTI dataset. Singandhupe and La tried to categorize SLAM techniques utilizing Lidar-based or Stereo-based odometry. Singandhupe and La aim to highlight the security flaws in autonomous driving systems. Many academics have created and studied attacks, making it a highly intriguing topic for future research. Singandhupe and La have developed an approach that uses graph-based SLAM algorithms and focuses on the KITTI dataset to close the current research loop better. Singandhupe and La also wish to focus on integrating and validating state-of-the-art deep learning methods to SLAM since they may be helpful for data analysis. Kun et al. [50] discussed that automation, in and of itself, requires connected vehicles, but this comes with its own set of problems. When car-to-car communication is implemented, the system may keep people from running into each other and enable unauthorized access to personal data. There is a possibility that the collector will reuse every activity in the vehicle. The function of vehicle user interfaces in this legal framework varies widely between countries. As our technological solutions and legal frameworks influence consumer acceptance and user experience, these solutions and frameworks will have a significant impact. Interoperability issues, however, are not often discussed by the company or in the literature. To meet rising customer expectations, there is a need to build better user experiences on safer hardware and software infrastructures. In recent times, it has been observed that both in-car applications and sophisticated user interfaces have improved. Transportation is transforming. They are connected to the outside world and depend on computer power to conduct autonomous driving. Decades of precise, explicit management are giving way to frequent, intervention-based control. With these new developments, the community now faces new challenges and looks forward to new opportunities. We anticipate that vehicle designers and researchers will increase safe and affordable transportation that allows passengers to work and play while traveling. Wang et al. [46] discussed that autonomous driving could revolutionize transportation networks by making roads safer, making people more comfortable, and giving cars more intelligence. In autonomous driving, AVSNs may disseminate data in essential applications such as safety and entertainment. Autonomous driving, however, changes based on time, place, and queue constraints. Thus it is challenging to direct CAVs to disseminate significant amounts of information inside autonomous vehicle networks. On the other hand, attackers could share inaccurate information to mislead the network, putting CAVs at risk for security and privacy issues. Sophisticated blockchain-based autonomous systems provide secure content transmission and also offer an economic incentive approach. The blockchain-enabled autonomous system architecture will serve to safeguard content distribution. Wang et al. evaluated CAV and RSU trustworthiness using task-based and credit-based reputation models. To inspire CAVs to give credible information, the researcher examines the influence of reputation and task rewards. While encouraging roadside units, to be honest, furthermore, authors have also designed a novel proof of reputation consensus method for blockchain-enabled autonomous vehicle networks. The architecture proposed does better in terms of dependability and security than existing approaches.

Artificial intelligence, IoT, and Autonomous Driving Systems: according to Khayyam et al. [36], many intelligent methods and technologies are being used to improve decision-making abilities with the advent of autonomous vehicles. Connecting AI and IoT for AV enables more dynamic and resilient control systems in environments. In addition to cloud hosting, new edge computing paradigms such as latency, network bandwidth, and security are difficulties for AVs. As a basis for future AI-based AV development, Khayyam et al. explore the architecture of an AI-based AV using edge computing. Du et al. [51] stated that anonymity in federation learning enables a community to gather, share, and analyze large quantities of data from numerous sources without revealing the original data. With computing capacity, multiple learning agents may be used to improve learning efficiency while also preserving the privacy of data owners. One of the reasons the federated learning business is on the rise is because privacy is a big issue. Future IoT systems will include numerous devices and privacy-sensitive data needing rapid connectivity, processing, and storage. It is possible that federated learning could serve as a solution to these problems. Du et al. started with the latest scientific study on federated learning and how it may be used for wireless IoT. Then, it is discussed that how important federated learning is in building a vehicle-based IoT and other possible associated avenues.

5. Safety Standards and Challenges in Autonomous Vehicles

Autonomous vehicles (AVs) is an active research area from last two decades. The rapid growth of vehicles on the road has increased the chance of traffic accidents that is considered a severe problem to the public and society. Human error factors such as inappropriate judgments, interruption, and exhaustion can be the reason for fatalities and accidents [52]. Hence, AV can be a solution to enhance vehicle safety and minimize traffic accidents and human driving errors. AV utilizes advanced technologies such as Electronic Controlled Units, path planning, Global Positioning System, 3D mapping, and light detection and ranging to reduce human driving mistakes, enhance safety, and optimize traffic flow [53]. Safety and security are the challenging tasks in AV to where significant research contributions are required.

Figure 2 demonstrates the safety and security of AV. Electrical and Electronics safety systems and mechanical safety systems are considered safety issues in AV's safety system. In contrast, cyber and physical security systems are identified as security issues in AV's security system.

Security of AV concentrates on defending the vehicle from deliberate attacks, and the safety of AV focuses on guarding the vehicle against incidental collapse [54]. A multisensor AV can pre-sense the attack conditions and handle accordingly. Here, AV can avoid the attack or

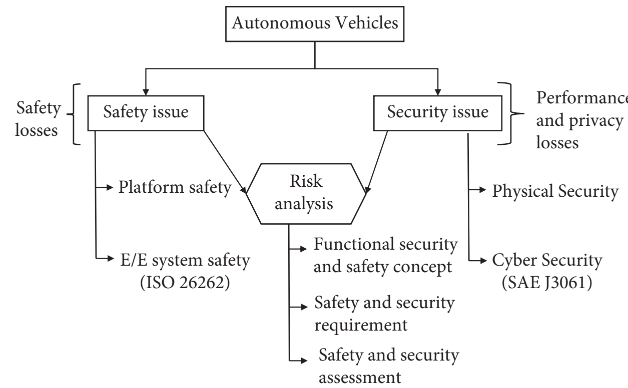


Figure 2: Autonomous vehicles' safety and security.

accident by changing its directions as well. These features are possible with integration of advanced technologies such as AI/ML, IoT, and Big Data analysis. The international standard ISO 26262 defined the operational security of Electrical and Electronics systems in AV [55]. ISO 26262 set of standards adapts to the International Electrotechnical Commission 61508 series of measures to deal with sector-specific electrical and electronic systems requirements in road vehicles. The international standard SAE J3061 defines the operational security in conventional vehicles [56]. SAE J3061 describes a process architecture of a cyber-physical vehicle system's security lifecycle. Standard SAE J3061 introduced a framework in which a communication bridge is established between cybersecurity and safety phases to integrate vehicle security and safety. Howeve, how to combine security and safety analysis is missing in this standard. In the literature [57–59], the issues related to alignment for the cyber-physical system have been addressed.

Six levels of driving automation are described in SAE J3061 [15]. It delivers a classification with complete descriptions of all six groups (0 to 5), from without automation to fully functional automation, against the backdrop of vehicles and their function on roads. Every level of driving automation has additional safety and operational requirements. Moreover, various levels will face a more significant number of possible challenges, hazards, and risks. To ensure functional safety and evaluate failures, HARA is considered as a standardized process and found in ISO 26262 [60]. In addition to this work, the authors [61] introduced a HARA technique by utilizing ASIL at level 4 for AV. ASILs are recursively improved to obtain specific safety objectives for vehicles. To assess the hazards or threats in AV, ASIL is utilized and considered as a critical point. In [62], the authors used a fault tree in HARA, similar to an attack tree, where three nodes represent failure events. STRIDE is a threat model that can identify and classify possible threats to a system [63].

To conduct a systematic analysis of system architectures, the authors combine two techniques, STRIDE [63] and HARA [60], and proposed a new approach named SAHARA [64]. The STRIDE technique performs the security analysis, while the HARA approach of ISO 26262 conducts safety analysis. The co-analysis of security and safety is also served by US2 [65], similar to SAHARA. For an attack, the security level is quantified first after that analyzes the safety hazards by US2. If an attack is introduced in safety hazards, then countermeasure of safety and security is required. Otherwise, countermeasure of protection is necessary.

The co-analysis of security and safety is also considered in [66] and introduced a technique that employed the standing methods, such as GORE. The results from security and safety analysis are considered by this approach to make the goal tree for addressing the necessities with correlated vulnerabilities. A new system theory-based hazard analysis approach is introduced in [67] to analyze the risk. It considers safety to be a control issue instead of a failure issue. This work is extended in [68] and introduced STPA-Sec. As a result, the authors outline a novel system thinking technique for safety to protect the complicated system against cyberattacks. In [69], the authors utilized STPA [67] and STPA-Sec [68] approaches and introduced a new methodology named as STPA-SafeSec to assure system security and safety by using highly effective mitigation approaches. Before selecting the appropriate mitigation approaches, the STPA-SafeSec method unified all the security and safety considerations and prioritized the system's most crucial component. Due to this analysis, the system can identify the potential loss due to particular security or safety exposure.

Several researchers have employed deep learning to enhance safety. In [42], the author explained application of deep learning in autonomous driving and reasoning about the safety such as (i) recognizing the consequences of potential errors and (ii) recognizing the more extensive system's meaning. In [70], the author utilized the convolutional neural network technique to determine the pedestrian. The task of this system is to detect the object with sufficient distance. After that, the system will manage the speed and braking system. The author describes safety as epistemic uncertainty, risk, and harm caused by unintended consequences [71]. After that, analysis is performed on the convenience of optimizing the empiric mean training cost and choice of the cost function. In [72], the authors described the accident issues due to machine learning approaches and specified inadequate artificial intelligence systems' harmful and unintended behavior. For accidental risk, the authors described five research problems and further categorised them into a specific area. Several significant areas of AV have been described as open problems such as data are considered as oil in the AV; the amount of data collected by an AV regularly is approximately petabytes; and it presents difficulties on the training procedure's parallelization along with storage resources:

(i) In safety-critical systems, the usage of the deep learning approach is still an open problem. Few efforts are there to bring functional safety and computational intelligence communities closer. For example, time-series analysis of AV and its movement need deep learning for in-depth and accurate prediction of its feasibility. Deep learning is helpful in predicting the futuristic trends of AVs as a close system to operate efficiently. Deep learning, one of the most important technologies, has made the realisation of self-driving cars a reality. For example, it may be used to provide answers to physics questions or to recognize photos in Google Lens, predict the behavior of vehicle movements, identify the roadside objects with more accuracy, among other things. It is a very adaptable tool that may be used to practically any situation without restriction.

(ii) It will be difficult to accurately localize, categorize, and detect objects in the external world to mitigate perception errors. Perception error is one of the challenging tasks of AV safety.

(iii) An accurate, stable, and effective decision-making system should be designed to respond to the surrounding environment promptly and adequately. To reduce the decision error, comprehensive and rigorous software and hardware system testing should be performed.

(iv) To avoid failure, observe the behavior of the system in different-different scenarios and situations.

(v) Cybersecurity for AV is the biggest concern for researchers. How securely wireless communication can be performed. Security and safety are significant concerns that can considerably influence the public's attitude towards the rising AV technology.

(vi) The performance of AI techniques mainly depends on the correctness of the sensor data as input signals. The input of AI techniques is affected by sensor issues.

(vii) Vehicle-to-everything (V2X) technology enables cars to connect with roadside units, vehicles, etc. Protection in privacy and secure communication among parties are still significant concerns for academia and industry people in AV.

(viii) Software updates are taking too much time because the line of code is increasing day by day. An over-the-air mechanism has been introduced to overcome this problem, but many attacks are reported during the software updates.

6. Artificial Intelligence in IoT-Enabled Autonomous Vehicles

The role of Internet of Things (IoT) is significant in Industry 4.0 revolution [36]. This is due to the fact that intelligent autonomous devices communicate for better value chain. Industry 4.0 is focused on improving the business process. IoT is very essential for business process in Industry 4.0. The combination of AI and IoT will enable the researchers and organizations to achieve fully autonomous Level 5. IoT collects data, and AI analyzes the collected data to convert this into relevant information for decisions. IoT becomes smarter using AI synergy [73]. In AV, data generation, processing, and communication are required. Furthermore, traffic congestion and path planning information is sent frequently. IoT provides the capability to vehicles to send and receive data as objects without human intervention.

Speech recognition and NLP are applications of artificial intelligence. Here, AI-based algorithms can be used to train the speech recognition system and read the messages written alongside the roadside units. Thus, speech and NLP are considered as well-known explanation of AI-based systems. Autonomous vehicles are developed these days which can follow the instructions by recognizing speech and text from base stations. These instructions are forwarded to Autonomous vehicles using IoT sensors. Furthermore, AI can be applied in IoT-enabled Autonomous vehicle to reduce traffic congestion [74]. Traffic signals and various devices collect information about traffic using IoT and information is sent to the AI-based predictive model for decision-making. Furthermore, updated path information can be sent to autonomous vehicle. Artificial intelligence and IoT can handle complex data which are generated form a large number of devices [75].

Various sensors and devices are connected to the IoT ecosystem, as depicted in Figure 3. There is requirement of connected and shared architecture that can communicate information in real time. For example, device information should be sent in real time and fast processing so that decision can be made. The advantage is that communication between devices and AV is efficient. Furthermore, various parts of AV are connected to a central point that sends and receives data. This will allow functioning AV effectively. There are four components in IoT-based autonomous vehicle platforms [36]:

(i) Sensors and hardware components send and receive data from the vehicle to the vehicle or the base station.

(ii) Communication network where data will be sent and received.

(iii) Big Data is a collection of Volume, Velocity, and Variety data. There is a need for Big Data technologies to process large-scale data.

(iv) Cloud where data will be saved so that it can be distributed to various objects.

There are various layers of data transfer between IoT devices. These data communication can be between vehicle to vehicle, vehicles to other devices. The decision-making by autonomous vehicles is based on inputs from various channels [76]. IoT devices that are connected send and receive essential data that can be analyzed by autonomous vehicles only if decision-making is based on AI such as neural networks or rule-based. The predictive model decides the output about line keeping, path planning, and object detection based on data from various sources. AI-based sensors are essential in AV, but in addition, IoT provides information about road conditions, weather, and specific area from connected devices in real-time. In smart cities, AI-based AV can be connected to the ecosystem for better path planning.

In Figure 4, communication of various sensors and devices with AV is depicted. Autonomous vehicle Cameras, LiDAR, GPS, and network information are sent to the IoT cloud. Information is sent to various devices, base stations, and network infrastructure. Real-time data are possible by the use of the IoT cloud for better decision-making. IoT can be essential for AI-based AVs in the following phases:

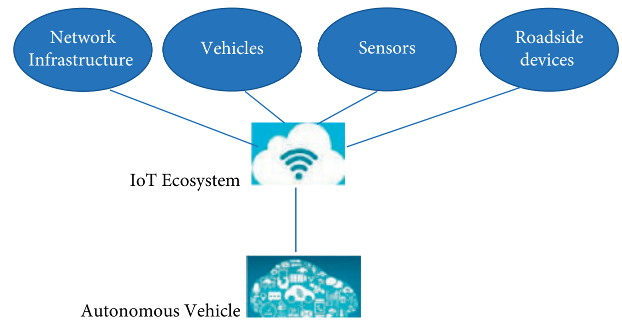


Figure 3: IoT ecosystem for AI-enabled autonomous vehicles.

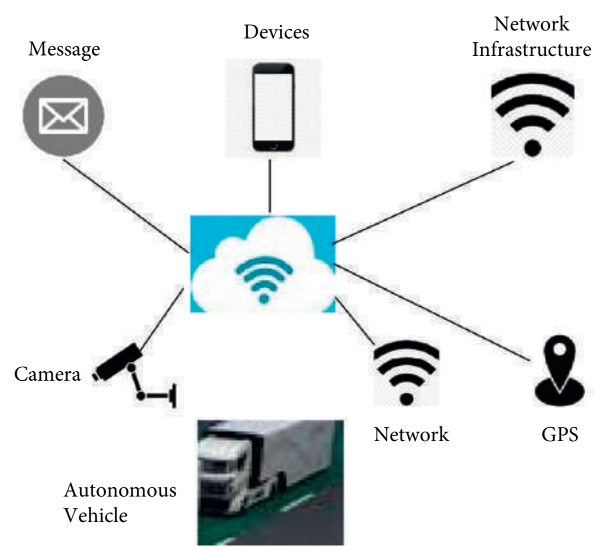


Figure 4: AI-enabled autonomous vehicles.

Data collection: artificial intelligence-based AVs require a large amount of data for training. Data should be relevant and in real-time. IoT devices can provide this in the ecosystem.

Path planning: path planning is based on Monoeuvre planning used for high-level decisions, and Trajectory planning used for path from one state to another. In these planning strategies, IoT is essential to provide real-time data for efficient path planning.

Act: in this phase, object detection and weather information-related response is achieved. If data collection from IoT devices is more and path planning is efficient, this phase will be processed effectively.

In [77], the significance of intelligent transportation for IoT-based AVs is highlighted. If maximum tasks can be implemented in vehicles, it will save computational and data transfer time.

7. Research Challenges in Artificial Intelligence-Enabled Autonomous Vehicles

Autonomous vehicles can decide path planning and motion control based on a predictive model. There is a need for an improved AI-based model for AVs. In real-time architecture, each component needs to be addressed. For instance, recognizing a scene requires object detection and object tracking [78]. There is a lack of start-to-end depiction in current AV architectures [79]. The architecture of AVs should be able to handle system faults and manage scalability. Real-time architecture is required as AVs have to perceive surroundings with communicating with other vehicles in real-time. AI-based techniques can achieve this. The main agents in AVs are infrastructure and devices which should coordinate to perform accurately [80].

Automation levels are classified by the SAE on a scale from 0 to 5, where 0 signifies no automation and 5 signifies full performance. Companies and researchers are putting a lot of effort to achieve Level 5 [81]. SAEJ 3016 defines component classes required in architecture as follows:

(i) Operational: in this class, the focus is on vehicle control.
(ii) Tactical: in this class, path planning and object detection, and tracking is planned.
(iii) Strategic: destination planning.

AI has improved AV design, development, validation, and real-time monitoring significantly. Perception, path planning, and decision-making can be achieved effectively by using AI. AI is used in AVs as follows:

(i) Autonomous vehicles decide paths based on a predictive model.
(ii) Autonomous vehicles learn from history to decide speed and path.
(iii) The efficiency of the transportation system is improved.
(iv) Intelligent use of real-time data provided by various sensors.

The issues in AI-enabled autonomous vehicles are elaborated as follows.

7.1. AI-Based Model Issues. There are three steps in the AI model for autonomous vehicles-data collection, path planning, and act [36]. In data collection, road, vehicles, and nearby object information is collected by various sensors. In path planning, the safe path from point A to point B is selected by AI techniques. In the act phase, decisions are finalized based on previous stages. If more data are analyzed, more accuracy will be obtained. The main issues faced by AI-based autonomous vehicles are checking road conditions and large-scale object detection. Highly scalable and fault-tolerant technologies are required for autonomous vehicles [47].

A limited amount of labeled training data is a real issue for AI in autonomous vehicles [82]. Training data validation is an open issue that can be addressed by data characterization and data collection [83]. Classification is also tricky on large distances. Data are not reliable in conditions where the sensor was not working fine. Inconsistent and complex data training is improper, which may provide incorrect output during validation and monitoring time.

The autonomous vehicle system was based on a rule-based controller [84]. Traditional machine learning models cannot be directly applied due to spatial and temporal data [85]. Deep learning-based models are suitable for a complex and nonlinear dataset. Deep learning can be deployed on new scenarios based on decision rules by knowledge. Deep learning provides better accuracy in less time. Furthermore, self-optimization based on complex data can be achieved by using deep learning. However, deep neural network architecture requires large-scale data to reduce variance [76]. In deep learning and machine learning architectures, parameter tuning for autonomous vehicles is computationally expensive. The reason is that there is a lack of information about how hidden layers and parameters are set up for autonomous vehicles. The number of layers selected in deep learning is a significant issue. If the number of layers is less, training is inadequate, and overfitting may occur if the number of layers is large. The solutions to problems can be coordinate descent, random search, and grid search.

7.2. Hardware Issues. The processing of sensor devices requires high processing speed and capacity. High computing devices rely on GPUs, CPUs, and FPGA [30]. Traditional CPUs cannot perform the processing required for AI. Thus, several researchers use GPUs for AVs development. The limitation of GPUs is that GPUs consume ten times more power as compared to FPGA. Google developed TPU, which serves 15–20 times better than GPUs [86].

Price and performance issues are associated with hardware. This is the reason that embedded systems are integrated into autonomous vehicles due to portability and energy efficiency. Several autonomous vehicles companies use LiDAR or high-resolution cameras for detecting and recognizing objects. LiDAR provides 3D images, whereas the camera provides 2D photos. LiDAR is used in Audi's Research vehicle, Google: Toyota Prius, Volvo: (Stoklosa, Cars), Apple's Lexus SUVs, and IR camera is used in BMW750i xDrive, Apple: Lexus SUVs [30]. LiDAR offers high-resolution 360-degree images but is vulnerable to weather conditions. The main issue with the use of LiDAR is its cost. Researchers have found a solution for this, LiDAR is used for training images, and image data are used for validation purposes [42]. Perception is problematic in complex areas. Various sensors are used, which results in the heterogeneous dataset, which is challenging to analyze.

7.3. Other Issues. Lack of good intelligent software is also an issue in AI-based autonomous vehicles [42]. Software that can predict with accuracy based on the unlabeled dataset is essential. Furthermore, more roads that are covered by maps are needed. In developing countries, roads are not covered on maps which is not easy for path planning in autonomous vehicles [43]. In AI-enabled AVs, machines have complete control, so the issue is to design a system in the ASIL [44, 45].

8. Intelligent System Software and Tools for Autonomous Vehicles

Autonomous vehicles contain a lot of sensors that feed input in computing systems [87]. Intelligent and reliable software is required to process information from various sensors and decision-making. There are software and tools available specifically for the design and development phase where the model is trained on large numbers of 2D and 3D images and simulators. Furthermore, validation, runtime monitoring, and analysis of the trained model are necessary for a controlled manner. Specific software is available for this phase.

Software systems for autonomous vehicles should work like biological systems [88]. Multilayer architecture should be incorporated into this software. Traditional AI-based system capability is limited as compared to fuzzy logic and neural network-based systems. In [88], several types of system software such as Java Expert System Shell, Fuzzy Logic in Integrated Learning, Subsumption Architecture, and Autonomous Robotic Architecture are described. It is observed by researchers that several types of intelligent software are based on rule-based and computational intelligence.

In Figure 5, it is depicted that SysWeaver and SysAnalyzer are used to design and develop various modules and layers. TROCS and AutoSim are used for analysis and validation. Tools used in autonomous vehicles are as follows:

(i) SysWeaver: it is a model-based design for integrating hardware and software components. Traditional programming language-based software cannot quickly achieve fault tolerance and reliability. These can be captured by model-based design [87]. It is designed by [89]. The system generates code when the model is configured for interfaces. Application agents, protocol agents, and state managers are software components. The timing model is based on rate monotonic scheduling.

(ii) Autism: it is used for various scenarios such as lane change, etc. It is an emulator that can interact with the vehicle and allows the vehicle to sense virtual surroundings.

(iii) SysAnalyzer: this tool is used to schedule various module timelines synchronously. It can also provide backup.

In [90], Eclipse IDE is used to implement an autonomous car. MATLAB and C++ were used for software development. In [91], Dynacar software is used for vehicle modeling. Various software/languages for autonomous vehicles are as follows:

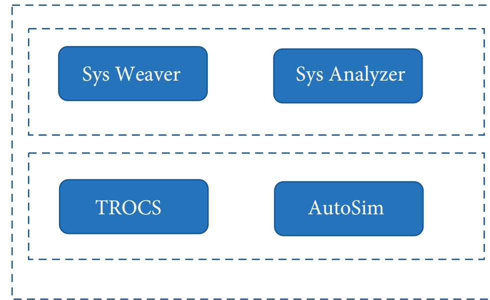


Figure 5: Tools for autonomous vehicle.

(i) OpenPilot is open-source software to improve existing driving assistance. It is developed by comma.ai. Various applications such as lane centring and drive monitoring of OpenPilot are used in autonomous vehicles. Several companies are using OpenPilot for improving autonomous vehicles.

(ii) Carla (https://carla.org/) is open-source software for research in autonomous driving. Various functionalities such as flexible API and baselines are available. CARLA 0.9.11 is a recent version.

(iii) Flow (https://flow-project.github.io/): this open-source framework is developed by Mobile sensing lab members at UC Berkeley. Deep reinforcement learning is used for custom traffic scenarios.

(iv) Point-Cloud library (http://pointclouds.org/): this library is used for managing point-cloud data. Furthermore, the Euclidean distance-based algorithm can be implemented by the use of this library.

(v) OpenCV: this library is used for image processing. Several APIs are available to process images. Feature selection and object detection can be implemented by this library which is essential for autonomous vehicles. Lane detection, edge detection on images, region of interest, and road sign recognition are the applications of OpenCV in autonomous vehicles.

(vi) Java Expert System Shell (JESS) (http://www.jesruls.com/): JESS is the rule-based engine that supports forward-chaining and backward chaining. PKD android is developed using JESS. The inputs are sent using JESS and NLG functions. JessDE platform is used, which is similar to Eclipse IDE. JESS is service based on network that is implemented as hardware. It is also used for implementing Autonomous Car Assistance.

(vii) FuzzyClips and FuzzyJ: FuzzyCLIPS is developed in Isaac language, which is rule-based for geometric values. FuzzyJ is a Java-based API that is used for fuzzy logic systems.

(viii) AuRA: Autonomous Robotic Architecture is a hybrid-based framework. In the deliberative

Table 3: Summary of various software/tools for autonomous vehicles.

Software/tool	Design techniques	Language/development
SysWeaver	Model-based	Models/Couplers
SysAnalyzer	Scheduling	Models
AutoSim	3D graphics, simulator	Simulator
Flow	Deep reinforcement learning	Python
OpenCV	Image processing, machine learning, object detection	C++
JESS	Symbolic AI	Java
FuzzyJ	Fuzzy logic	Isaac
AuRA	Neural network, genetic algorithm	LISPs

component, a plan sequence is included. In the reactive part, the rum-time controller is included.

In [92], various virtual environments are highlighted. AirSim [93], ASM, CarMaker, OpenDS, PreScan, Racer, and VDrift are summarized based on the latest release, accessibility, platform, use-case, and programming languages. In Table 3, software and tools used for autonomous vehicles are outlined based on design techniques and language used.

9. Artificial Intelligence-Enabled Testing Techniques for Autonomous Vehicles

AVs have taken the transportation by a storm. The promises which it entails surely outweigh the challenges faced in bringing this technology to the masses and making it commercially viable. AVs are designed and developed using integration and interoperability of multiple intelligent systems driven by machine learning and deep learning algorithms. Almost all of the major car manufacturers such as Daimler with their MBUX, Hyundai with their Smart Sense, Audi's MMI Virtual Cockpit, and many more, in addition to the involvement of the tech companies such as Watson by IBM, Google, and Nvidia, are realizing AI's impact on the services offered and are transitioning towards the development and nurturing of AI [94].

The testing techniques that are most used today, e.g., miles driven and frequency of human intervention, are insufficient to fully advocate the safety of an autonomous vehicle [95]. Such techniques are misleading and cannot fully satisfy the safety requirements of an autonomous vehicle. The faulted assumptions can lead to failure of the autonomous vehicle system [96]. Since the autonomous vehicle itself uses a lot of AI technologies for different decisions, the quality of those decisions cannot be left to manual testing because of two reasons:

(i) Systems with AI-enabled components can have a high density of errors due to the algorithmic bias and faulty predictive algorithms. The prediction of failure is so nondeterministic that makes the entire AI-enabled system so hard to test and verify [97].

(ii) Non-AI-enabled testing might leave a lot of people induced errors which itself might break the whole concepts of automation [98].

To solve these issues, we explored different ways of testing autonomous vehicles. This section first reviews the operational testing of autonomous vehicle consisting of full functional testing and validation. After that, it assesses the AI-enabled testing techniques because autonomous vehicle is a master amalgamation of AI-based technologies. AI-enabled techniques can shorten the testing and verification time for vehicle manufacturers and how it can be boon for making these more secure.

Autonomous vehicles seem to be coming from the sci-fi world into the real world suddenly. In the past 15 years, scientists and engineers have been working hard to make it a reality. However, Auto manufacturers are struggling to fine-tune AI algorithms that form the brain of the AV through metallic arms of obstacles and environments. The use of multidisciplinary sensors such as LiDAR enforces significantly different testing requirements not related with Vehicle movement but with respect to accuracy of measurement of these devices [99]. As 5 G is being rolled out, it opens a new world of possibilities for autonomous vehicle industry [100]. To take care of these different testing requirements, the testing of autonomous vehicles will need to move from functional testing of components to a fully autonomous testing.

We need to be cognizant not only before the production and development of these vehicles but also during the whole lifecycle of the component involved. For example, for the demand of high level of parallel, time-critical, and fault-tolerant computing, FPGAs are suitable as they are programmable and customizable and can process high volumes of data in parallel on a single chip [101]. These chips need to be in working for at least 10 years. Nobody has tested the lifespan of these chips over a decade in outside road conditions. Hence, we need a comprehensive testing approach and techniques to take care of operational as well as software scenarios to ensure the quality of these autonomous vehicles.

In Figure 6, the components of autonomous vehicle are depicted. The testing of GPS, Radar, Sensors, and computing unit components are required for better functioning of AV.

9.1. Operational Testing Approaches. Autonomous vehicles are believed to be safe with the researchers' belief that the number of car accidents will be reduced. However, some of the autonomous vehicles crashes have attracted attention all over the autonomous vehicle industry [102]. The autonomous vehicles will spread across the world, so as the testing of autonomous vehicles on public roads. This will require a regulatory approach to the autonomous technology [103].

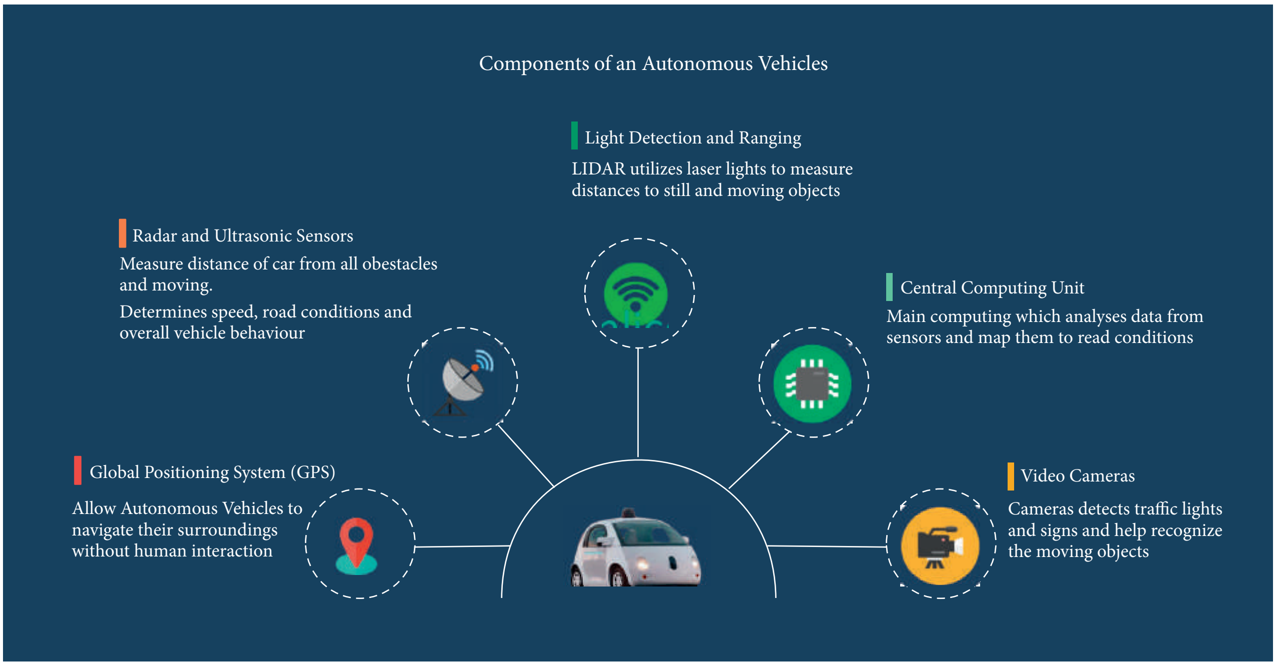


Figure 6: Components of autonomous vehicle.

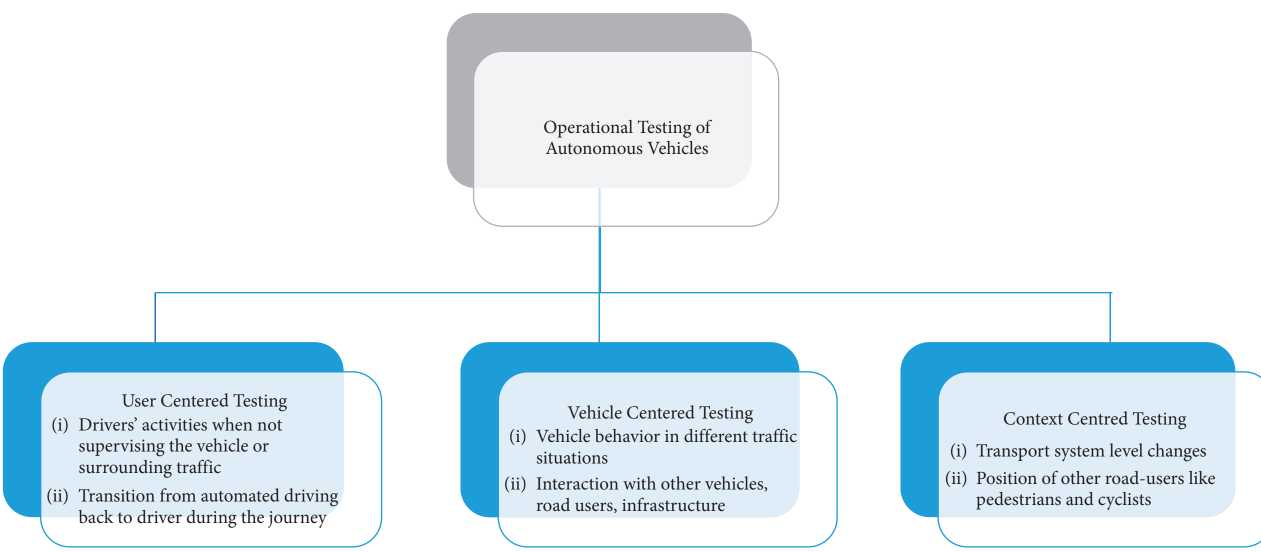


Figure 7: Autonomous vehicle operational testing.

To keep quality checks on an autonomous vehicle production, it needs to undergo a high-fidelity operational testing.

Few governments such as Taiwan have introduced regulatory frameworks for the testing of autonomous vehicles [104]. US and Chinese AV manufacturers have been testing the autonomous vehicles since long. Only Waymo has driven more than 20 million miles of autonomous driving at the time of writing. A Chinese company, WeRide has driven a total running distance of 2.6 million km using autonomous vehicle since its inception in 2019 (https://www.am.miraeasset.com.hk/insight/race_china_autonomous_vehicle/).

The operational testing of these vehicles can be divided broadly into vehicle centered testing, user centered testing, and context centered testing.

The various techniques of operational testing of AV are presented in Figure 7. An increased dependence on simulated and operational testing seems unavoidable to measure safety and reliability. Several standards such as IEC61508 and EN50129 include several parts of statistical evaluation from operational testing [105]. However, the autonomous vehicle core system relies heavily on machine learning and artificial intelligence algorithms. Despite intense research, there is no established operational testing process or tool

Table 4: Tools for testing of AV.

Tool	Underlying technique	Impact area	Languages
Facebook Infer	AI/ML	Automatically identify code quality issues, regressions, security vulnerabilities in AI/ML algorithms	Java or C/C++
Testim.io	Deep learning	Fast authoring with code flexibility boosts coverage	JavaScript
PaddleCV	Deep learning	Rich official model library, covering various visual tasks	Python
mltest	Deep learning	Machine learning testing framework for TensorFlow	Python
Torch-test-case	DL	Machine learning testing framework for PyTorch	Python
Functionize	NLP	Low code testing solution	NLP-based testing

which can satisfactorily validate the correct design. We need a different approach to testing to deal with this VUCA world which is going to change the face of humanity forever.

9.2. AI-Enabled Testing. There are thousands of algorithms with millions of lines of code are written in a single autonomous vehicle which will be deciding the next move of the vehicle in real time. This requires a complete suit of automation functional tests on source code. The testing practices used today will require another level of automation in terms of automatically created test cases and the mapping of test cases to requirements.

Telemetry usage data, especially errors in real time, are sent back to the manufacturers. Manufacturers use this real-time telemetry data to improve their software and send the system updates over the air. There is no need for every mobile owner to go back to the manufacturer or a dealer to fix it until there is anything serious. The car manufacturers will need a continuous testing facility throughout the life-cycle of an autonomous vehicle. The simulator- and algorithm-based automated testing can also be integrated. For example, the Udacity simulator testing which creates different randomly, manually modified scenes to identify the failures across the system.

Since the core AI- and ML-based systems have millions of hyperparameters to adjust which makes the normal testing out of question. We need to use AI-based testing tools which can adjust these parameters automatically based on the telemetry data collected from the vehicles over a span of time. In fact, real-time testing scenarios will be driving the next upgrade of firmwares which needs to be put in the autonomous systems. For example, the object detection systems will need to be continuously improved based on the real-time telemetry data for which the ML algorithms were not being trained for. As the real-time telemetry data will increase, so would be the ability of embedded ML algorithms for decision-making. According to our research, currently there is not a single fully compliant testing system which can make these adjustments in real time and can make the autonomous vehicle more secure [106].

9.3. AI Tools and Techniques Used. The fault detection in machine learning applications is like finding a needle in hay stack because there are no standard practices of creating a test oracle to verify the correctness of the algorithms used [107]. Among the vehicle simulators, we can use a variety of tools. The most known is probably a Driving Simulator product from IPG makes use of AR and the vehicle-in-the-loop testing methodology (https://ipg-automotive.com/products-services/test-systems/driving-simulators/#augmented-reality-with-vil). It allows the tester to visualize the different objects in real time with the help of AR glasses.

Another test methodology called the "Hybrid Testing," was developed in the scope of the EU-H2020 project INFRAMIX. This testing enables the evaluation of a real vehicle in a virtual scenario in an enclosed proving ground. The testing is usually performed with simulated traffic components and sensor signals, to make the environment simulating to real-life [105].

Sometimes because of the issues in the camera devices, result in a false induction and hence an empty photo. This might end up in generating an abundance of data in form of images [108]. Hence the machine learning algorithms used in autonomous vehicles might process a lot of unwanted data. A tool such as Zilong software (freely available at under BSD License) might help too. Vehicle identification is a crucial technique in autonomous vehicle operations while running on road. The testing goal should be to generate a test data of all the vehicle images captured by different cameras under various viewing angles. This will allow the testing of different vehicle identification algorithms in an efficient manner with different test input images. Vehicle companies should use vehicle re-identification (re-ID) techniques which can help in reducing the object identification load [109]. In Table 4, tools for AV testing are elaborated.

10. Autonomous Electric Vehicle and Its Applications

With rapid industrialization and recent development in the automobile sector, the need for fossil fuel drastically increased. Due to most gasoline-based vehicles used in routine transportation operations, GHE is grown and exploited in the natural environment. Hence, there is a need to save natural environmental conditions for saving the life of human beings. Therefore, the transformation of gasoline-based vehicles to electric vehicles and autonomous vehicles is essential. The electric vehicle has used the sources of electrical energy for driving it. Hence, it will save nature against GHE and protect human beings against the exploited environmental conditions.

AV and AEV are driverless vehicles that are simple to drive, safe, and comfortable in operation. Most of the driver functions in ordinary vehicles are performed automatically in AV and AEV with the help of intelligent sensors, intelligent controllers, onboard computers, recent hardware and software applications, novel algorithms, etc. AV is proper for physically disabled and elderly people to live their life independently. Hence, the quality of life of the ordinary person will be enhanced due to decreasing the GHE and its independent operation. Imagine that one of the directors of the movie has gone to shoot. Still, he forgets the movie's script and other correlated important things or any person gone to do the shopping. He forgets his debit and credit cards, money, etc. The AV could be capable enough of bringing the missing items quickly by considering the abovementioned generalized uses of AEV.

10.1. Specialized Applications of AV/AEV

(i) Public Transportation

AV was introduced initially in the public transportation system in the driverless mode of operation. Nowadays, modern trends in public transportation are helpful in the cosmopolitan region for the tourists, own citizens, etc. Transportation is a big challenge in crowded, cramped, and cluttered areas in various cities. Still, due to the introduction of autonomous electric vehicles (AEVs), it is possible to manage the issues in crowded places.

(ii) Autonomous Underground Vehicle

One of the examples is a fully automated underground vehicle developed in Denmark. Its performance is encouraged to a resident of Denmark for its further utilization in a transportation system.

(iii) Autonomous Electric Tram

The first automated electric tram was designed and developed by Siemens in Germany. In 2018, the first test drive of the tram was conducted in Germany for a distance of seven kilometers. The use of smart devices, such as smart cameras, intelligent sensors, and intelligent software-based LiDAR systems, is helpful to the tram to drive in crowded areas of various cities without any obstacles. Due to the intelligent algorithm, pram in front of the tram, and intelligent monitoring and controlling system, a tram will operate very safely even in crowded areas. At the occurrence of any obstacle, the pram will be taking care of it with the help of other auxiliaries' apparatus, and the journey begins immediately after removing the barrier. During the long and short distance journey, the trams maintained safety throughout the trip, automatically stopped the tram at the desired destination, and immediately began for a different destination. Tram responds immediately to crossing animals, human beings, other moving vehicles, different types of objects, and any other obstacles.

(iv) Autonomous Microbus

The testing of an autonomous microbus was completed in Finland in 2018. The primary objective of microbus is to reduce public transportation's load and utilize the available resources to minimize the GHE. The microbus operated for approximately seven months from 8.30 a.m. to 4.30 p.m. and completed about 4 to 7 journeys during working each hour. It is handy for the shorter distance, transporting the employees of nearby industries, citizens, etc. The main aim of this microbus is to motivate people to avail themselves of this bus to control pollution by 2022 and save the environment.

(v) Automated Robotics Bus

Again in Finland, another invention of an autonomous vehicle was introduced, called Automated Robotics Bus. It was also called GACHA. It is an automated shuttle operating in any weather conditions. It was the coordination of Japan-Finland efforts. This bus is capable enough of a driverless mode of operation with accurate obstacle detection, accurate navigation, and positioning. It is 2.5 meters wide, 5 meters long, and its height is about 3 m. It is a four-wheeled vehicle that operates at 45 km/hour speed and can cover a distance of 110 km, and the option of wireless and wired charging is possible to it. It carries 18 people in it, such as 11 people in seating mode and seven standing ways. It is clean, safe, and amicable to bring the remote peoples together in Finland during the winter season. It is suitable for all weather conditions and easily navigates in cloudy lousy weather conditions such as rains, storms, and fogs.

(vi) Fully Automated SEDRIC

The Volkswagen group initially launched SEDIRC Car under an autonomous level of 5. It is simple, well electrified, well digitally networked, safer, and sustainable. Due to being digitally interfaced, it is available at any interval, such as hiring a taxi. In 2017, voice commands and control button-based operating cars were launched in the motor show of Geneva. Due to the absence of a steering wheel, paddles, etc., it provides sufficient space and sufficient comfort during the journey. The journey information is mentioned in its display such as the length of distance in km, speed, time required to reach, safety, and traffic congestion.

(vii) Automated Electric Volvo Bus

A fully automated Volvo electric bus was designed and developed in 2019 in association with Singapore

University. It has a carrying capacity of 75 seats with a driverless mode of operation. Obstacles detection and control are obtained by using LIDAR 5 intelligent sensors. Automated Electric Volvo bus offers high flexibility, safety, compactness, reliability, sustainability, and high efficiency. Hence shortly, this bus will be reflected in public transportation.

(viii) Autonomous Electric Helicopter

VSR700 is one of innovated prototype Autonomous Electric Helicopters invented in 2020 by Airbus under the heavy test drive in France. It is designed and developed for operating alongside various naval assets. The objective is to empower the ships, enhance their scope by using intelligent sensors in association with helicopters, and enhance the information collection scenario from ship perspectives. Autonomous Helicopters are doing the job of surveillance of their targets' information and confirm the destination of reaching the ships at desired locations. Sustainability is enhanced in modern ships and autonomous helicopters by using faster intelligent sensors.

(ix) Autonomous Smart Truck

A fully automated electric truck was designed and developed in 2016 by the name Otto. Without a human driver, it operates with the help of the LIDAR system. These modern trucks are minimizing accidents and utilized for delivering heavy goods and services. In addition to this, Vera as Volvo autonomous electric truck is designed and developed for carrying goods from various destinations such as industries, dockyards, mines, ports, storage yards, and warehouses and has very efficient, safer, clean, and sustainable ways than ordinary trucks. Using intelligent cameras and other sensing devices, these Vera trucks are smartly operating, positioning, detecting, and controlling in more innovative ways and decreasing waiting periods and pollution. Hence, their performance increases technically and economically.

(x) Google Self-Driving Waymo

The testing of Waymo vehicles such as trucks and cars was completed in various weather conditions and road conditions in California. Driverless mode of operations is considered using computer-integrated cockpit and various sensing and controlling devices. It provides security and safety during the journey with information about other vehicles nearby.

(xi) Fully Autonomous Shuttle

In England, a fully autonomous Shuttle was designed and developed in 2017 by Harry's name and tested in London. In the UK, places where the lack of public transportation or no buses, no trains nearby the various locations for public transpiration, decided to enhance the public transportation more smartly. Hence, these shuttles are used in such areas to improve the efficiency of transport. It is acquired near about 5 to 6 people and covers the distance of 12 kms. It is operated using intelligent sensors, intelligent cameras, LiDAR, and other smart monitoring and control systems.

(xii) Autonomous Metro Train

It is a fully automated train design and developed by China in 2020 for the country Turkey. It is operating at a speed of 130 km/hr. It can carry about 1200 passengers with 4 to 5 carriages.

(xiii) Nuro's Fully Automated Vehicle

It is helpful for elders, the physically disabled, etc. It is also beneficial for transporting goods from one place to another place. It was developed in 2018 for delivering goods in a driverless manner [110].

(xiv) Autonomous Underwater Vehicle

It is used in marine earth science and is popular in the technical and defense sector also. The primary function of this vehicle is to obtain an improved image of the seafloor with a very high resolution from the vessel's surface. The different types of underwater vehicles are marine robots, hybrid automated underwater vehicles (AUV), bluefin Hovering AUV, AUV Urashima, hyper dolphin, and solar-powered autonomous vehicles II (SAUV) [111].

(xv) Autonomous Vehicles for Agriculture and Mining

Autonomous vehicles are used in the agriculture sector for various farming processes and used in mining operational tasks. Different types of agriculture and mining autonomous vehicles are autonomous agriculture tractors, unmanned ground vehicles used for smart farms, mining vehicles such as mining trucks, mining automated machines, etc.

(xvi) Automated Rover

It is an autonomous vehicle utilized for indoor and outdoor applications. It is an unmanned vehicle used where human intervention is not easily possible in various conditions. In those applications, self-detection and diagnosis of faults are the leading features of Rover [112].

11. Power Train Energy Management and Machine Learning Applications in AEV

The power train is defined as the generation of electric power with the help of different sets of components and subsystems in the EV to drive the wheels of the EV and move the vehicle from one place to another. The power train of an IC Engine vehicle is complex rather than an EV. Ordinary IC Engine vehicles have more than 100 moving components are present and out of which engine is the main component of a power train. Similarly, the various subcomponents and subsystems are axles, more comprehensive cooling systems, differential transmission systems, drive shaft control systems for emission, etc. are used in EV. In the EV/AEV power train, 65% fewer subcomponents are used than the IC Engine vehicle power train.

The power train of EV/AEV consists of the following features: battery bank, DC to AC converter, controller for motor, electric drive motor, smart onboard charger, battery management system, DC to DC converter, intelligent temperature monitoring system, intelligent body control module, etc. These components are elaborated in the next section.

11.1. Essential Components of the Power Train of AEV/EV.

Battery Bank: its function is to store the energy in chemical form during the charging mode of operation and release electrical energy during its discharging mode of operation. It consists of different types of lithium ion cells used in series or parallel or combines hybrid ways.

Converter (DC to AC): DC output power obtained from the battery bank is converted into AC, and this AC power is utilized for driving the electric motor.

Motor Controller: it is also called a power train controller. It controls the desired rated speed and frequency of power feed to the motor. So that maintains the acceleration and related speed according to information of driver communication through acceleration and brakes.

Electric drive motor: it is utilized for the movement of vehicles. It converts battery-based electrical energy into shaft power movement of wheels of vehicles through its transmission system. Similarly, regenerative braking can be used under this mode of operation.

Charger on board: charging point of AC supply is converted into DC supply. Using a control system controls the technical parameters of the battery banks, such as current through the battery bank.

In addition to the above primary components, various hardware and software systems are present in EV/AEV power train systems such as electronic control unit, battery management system, thermal control, body control unit, and DC to DC converter, which are used in AEV. Data exchange and data processing are conducted under various software programs integrated with the EV power train system. Many electronics control units are used in AEV for performing a particular function.

Uniformity of equal voltage levels in all lithium ion battery cells is maintained using a battery management system (BMS). It is routine monitoring and controlling a cell's voltage to avoid malfunction and protect the system. The stable balancing of cells is obtained by using BMS and enhanced efficiency of the battery bank. It is also communicated very properly with EVSE, different electronic control units to maintain the rated parameters at the charging points. Various subsections of AEV/EV are getting the power by using the battery. Still, each subsection, such as mirror control, Horne, parking light, wipers, and lights, required a different voltage. Hence, the DC to DC converter issued herewith fulfills their voltage needs besides the standard voltage levels.

A temperature control system monitors and controls the rated or optimum temperature of the power train system in AEV. So that avoids if any inconvenience during the normal running conditions of AEV. A body control unit also monitors and controls routine operations of AEV such as vehicle access control, mirror control, and power windows controls.

11.2. Power Train Efficiency in AEVs/EV and ICEVs. Power train efficiency of AEV is a ratio of power required to a vehicle to complete the drive cycle to its consumption of fuel energy. The comparative analysis of AEVs and ICEVs is mentioned in Figure 8. The energy input of ICEV is 100%. Out of which adequate energy is 15%, the rest of the percentages are consumed by various losses such as Idling loss 17%, energy loss 62%, and driveline losses 6%. In AEV, by considering the energy input of 100%, the adequate energy is about 80%, and the rest of the losses are only 20%, i.e., electrical losses are 145, and driveline losses are 6%.

11.3. Significance of Machine Learning and Deep Learning in the Operation of Autonomous Electric Vehicles. As per the global scenario, 1.40 billion road accidents are occurring each year, and a leading cause of accidents is the crashing of vehicles due to human mistakes and error. Hence, due to autonomous vehicles, the percentage of accidents decreases and saves human beings' lives. The cost of delivery is reducing due to the driverless mode of operation, and the vehicle's performance drastically rises. Machine learning (ML) can be used in the autonomous vehicle for the Advanced Driver Assistance System (ADAS) function to enhance a vehicle's entire performance. ML performs the various roles in the routine operation of autonomous electric vehicles are as follows.

Classification of obstacles, objects, and their intelligent detection: in existing vehicles, smart sensors, high definition cameras, LiDAR, Radar, etc., technology-based intelligent devices are used for the detection, classification of various obstacles, and objects. The results obtained from this system are satisfactory, but there are chances to get the wrong category of things due to the slight difference in pixel of images. There is the chance of accidents being created due to the wrong interpretation of images, and evil actions may happen. Due to the proper involvement of the intelligent, trained ML model in existing autonomous vehicles, the system's perception can be enhanced, precisely identifying the obstacles or objects. So, the accuracy of detection of objects is improved with safety and security using the ML model in AEV. Also usage of deep learning (DL) intelligent software developed the intelligent algorithms for training the neural network (NN) system in AEV. Using an image processor of different objects accurately classifies and detects, and accordingly, the vehicles react for further actions such as lane detection, stay in highway lanes, and path prediction very accurately. Distinguishing between human beings on highways, animals, other vehicles, lamp posts, etc. can be efficiently and accurately possible using ML and NN.

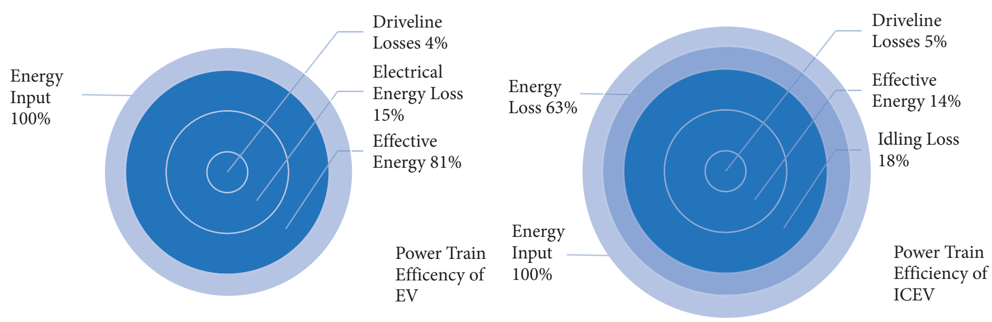


Figure 8: Efficiency of ICEV and AEV/EV [113].

The speed of moving objects, directions, free spaces, etc. quickly understands by ML.

Power Train in AEV using ML and DL: the various real-time data points are produced in the power train. By applying ML to these data points, the function of battery management, controlling of motors, etc. are improving. According to available power train data changes, ML offers the flexibility of boundary conditions as per the ages of the vehicle's system. Even changing operating conditions, the ML-based system has sufficient computing capability and is helpful even in real-time surrounding environmental conditions. The system is capable enough to identify the irregularities and provides regular information about warnings, maintenance, failure of motor controls, etc.

Security, safety, and reliability of AEV using ML: ML ensures the accurate operation of vehicles and avoids various accidents. ML also prevents accidents due to failures of different smart devices such as sensors, Radar, Cameras, and LiDAR. The data of multiple subsystems, such as state of charge, temperature control, speed, range, and battery level, are recorded. Furthermore, it is analyzed to conclude the performance of the AEV subsystem, such as motor performance and health index of AEV. The indicating system quickly concludes whether the AEV/EV operating is the average or abnormal mode of operations.

Identification of hacking, cyberattacks, and privacy in AEV-related data: using networking and intelligent computerized protected system with ML to ensure the security and confirm the detection of cyberattacks, hacking, etc. and overcome these problems quickly. Data privacy is easily maintained by using ML. Optimization of energy consumption in power train-based AEV/EV is obtained by combining Big Data from various sensors used in AEV/EV and ML.

12. Autonomous Driving Subsystems in AEV

Electric vehicles require multidisciplinary technologies such as electrical engineering, chemical engineering, and automobile/mechanical engineering. Furthermore, the Electrical Engineering system requires electric machines, power electronics, control systems, energy, battery management, and charging. Mechanical/Automobile Engineering involves gearing differential, chassis, suspension braking, steering, etc. The knowledge of IC Engines is also required in HEV. Chemical Engineering involves knowledge of batteries and different kinds of chemical features and knowledge of fuel cells. Battery and fuel cells are energy sources, and it also requires the knowledge of fuels such as liquid and gases, which is helpful for EV development.

12.1. Electric Vehicle Subsystems and Configurations. It is classified into two types: (i) converted electric vehicle (retrofitting) and (ii) purpose-built.

(i) Converted Electric Vehicle/Retrofitting

Converting an existing diesel engine or petrol engine based vehicle to electric vehicle in place of IC Engine similar rated electrical motor is fitted and the rest of components are kept the same without any change. This type of EV design is simple, and it can use IC engine-based used vehicles of 15 to 20 years. This kind of EV is popular only when the cost to a customer per kilometer of driving is less in the converted EV than diesel engine/petrol engine EV. This is not a high-performance EV

(ii) Purpose-Built

All modern EVs are purpose-built. Purpose-built EV means the body and frame of the vehicle are nearly designed such that it takes into the set ration the structural requirements of the EV, and it also uses all the flexibility that the EV system offers. In IC Engine-based vehicles, the power flow or the energy flow is done mechanically. It uses bolted frames and rigid systems to transfer energy from one system to another. However, in EV, the power flow is done using electric wires, which are very flexible. It allows the distinction of different components of an EV throughout the vehicle, and energy transfer can be done using flexible wires. Hence, distinction flexibilities are very high in purpose-built EV. Type of the propulsion system used in EV is also a deciding factor. There may be gears or gearless; some may use a differential, others may not, some may use the single motor, and others may use multiple or dual motors. So, depending on the type of EV, the design of EV has to be done. It cannot be the same for all kinds of configurations. The type of energy sources used in an EV decides the design of EV a lot, so if a single battery-based vehicle is designed, it has to be

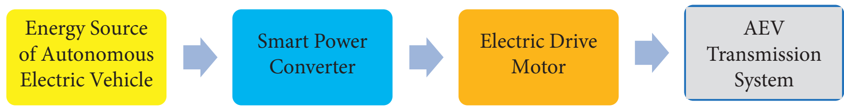


Figure 9: Propulsion system of EV.

suitable. If a multibattery system, it has to be done in another way. So, batteries can be used in the chassis of the vehicle, on the top side of the vehicle, or in the vehicle's luggage space. Hence, all these possibilities are there, and they can be connected using a wiring arrangement. The design must be different when using a fuel cell in a fuel electric vehicle because the fuel to be stored is hydrogen. It requires a lot of auxiliary systems. Charging system is essential. The charging system for different types of energy uses may be different. It can be a board charger; it can be an IPT kind of system where secondary coils are to be installed in the vehicle. All these requirements have to be taken into considerations while designing a purpose-based EV.

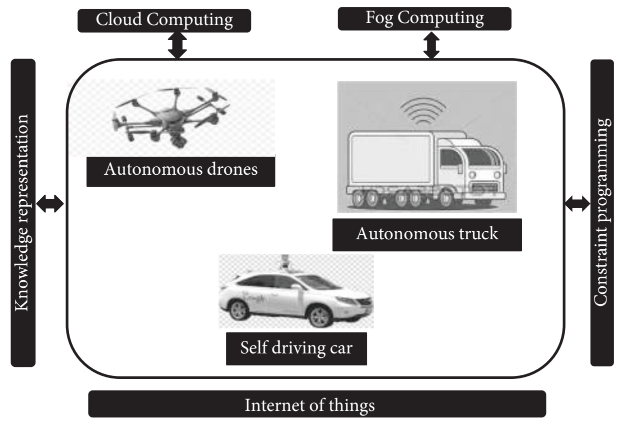


Figure 10: Advanced technologies for autonomous vehicles.

12.2. Components of EV System. The essential component of the EV system is an electrical propulsion system. Under this system power converter, controller, power electronics, motor transmission, gears, and differential gears are used. The performance of EVs is increased by optimizing these subcomponents. EV will get higher performance operation with minimum energy. The motor is designed to have high power density; it has high torque density and efficiency in wide speed and torque ranges. Power electronics are generally created at a high switching frequency. Loss-making components such as gears and the differential can be avoided, but employing complicated control is the job of a complex control system or the controller.

12.3. The Propulsion System of EV. The movement of EV is obtained by using a propulsion system. Figure 9 shows the electric vehicle propulsion system [114]. Initially, energy is extracted from the energy sources such as conventional and nonconventional renewable energy sources. The raw power is processed and converted from one level to another by using different intelligent converters.

The stable energy supply is feeding to the electric drive motor, and the rotation of the engine is utilized for driving the wheels of EV with the help of a transmission system; finally, EV is starting to rotate. The propulsion system of EV is depicted in Figure 9.

12.4. Autonomous Electric Vehicle (AEV) Driving Subsystem. It is a complex system consisting of various driving subsystems such as object sensing, perception, and decision-making. Also, it consists of a robotics operating system, various hardware, the platform for cloud computing, devices for data storage, modeling and simulation, ML- and DL-based different training models, high definition mapping, and novel algorithms.

It collects the raw data from various sensors and extracts essential information from sensors using algorithms subsystems. This algorithm information further gets the need for reliability and real-time data. The cloud platforms offer the offline computation of data and store the data in a different storage system of AEV using the medium of clouds. It is possible to test various types of novel algorithms and update mapping at a high definition range and offer intelligent recognition, following tracing with a particular decision of model.

AEVs are considered the future of vehicles, whereas the intelligent grid appears to be the grid of the future. Vehicle to Grid (V to G) is the link between these two technologies, and both get benefitted from it. Much research is going on to make electronics sensors in EVs more compact, rugged, and cheaper. Development of charging infrastructure with required EVSE should be significantly considered for safe and controlled energy transfer to EVs. Customer acceptance can be enhanced by increasing desired safety standards, reliability, durability, and efficiency of battery chargers with reduced charger cost. The modernization of the power system accelerates the utilization of EVs in terms of V to G technology. In an innovative grid environment, EVs become a possible solution to balance the power fluctuations due to the intermittent nature of RES [115–119].

13. Advanced Technologies and Autonomous Vehicles

This section discusses advanced technologies that play a vital role in the enhancement of autonomous vehicles. The technologies such as Internet of Things (IoT), cloud computing, autonomous drones, constraint programming, and knowledge representation, along with artificial intelligence, are explored in this section. Figure 10 illustrates the key technologies for autonomous vehicles.

13.1. Artificial Intelligence, IoT, and Autonomous Vehicles. The convergence of AI and IoT have emerged as an essential domain towards enhancing human QoL. The automotive industry has begun to adopt digital systems and applications from product services to customers. In recent decades, artificial intelligence and IoT have perpetuated the development of connected autonomous vehicles independent of human interventions as drivers. Significant enhancements in servicing technologies, control systems, and high computing capability have empowered the development and performance of autonomous vehicles. The service values such as safety, cost, fuel efficiency, user comfort, and in-vehicle quality of experience are more focused.

The primary objective of IoT is to digitally sense, measure, analyze, and decision-making in a real-world scenario. These digital devices are interconnected globally through Internet as a backbone network to achieve extensive scalability. El-Hassan et al. [120] discussed the low-cost sensor-based intelligent systems for detecting road obstacles, collision avoidance strategy, traffic signal identification, lane identification, lane monitoring, and halt responses. The authors discussed the challenges between innovative prototype systems and real-world road systems for automotive vehicles. Wang et al. [121] addressed the control theory analysis for automotive vehicles over the brilliant system performance such as control, stabilization, and reachable components of the automotive system. The comparative study was conducted between autonomous vehicles and human drivers under a simulated environment for a mixed traffic scenario. Safavi et al. [122] addressed autonomous vehicle health forecasting using the Internet of Things and artificial intelligence. The sensors are the critical part of the intelligent system of autonomous vehicles; however, these sensors may fail to function properly due to various dynamic factors. To the multiple sensor failures, the authors proposed a neural network-based framework that involves sensor fault detection, faulty sensor isolation, faulty sensor identification, and forecasting of sensor health. Furthermore, the authors elaborate on the forecasting categories, including monotonic system life prediction and nonmonotonic behavior prediction. In Table 5, advanced technologies are summarized based on Cloud, Fog, and Edge computing.

13.2. Artificial Intelligence, Cloud Computing, and Autonomous Vehicles. Fog computing is a paradigm shift in a computing platform that brings cloud computing facilities nearer to the edge system. Delay sensitive applications such as vehicular communication, data analytics, and data processing are carried at the proximity of the edge devices. Fog computing eliminates the delay and unnecessary network hoping. Sookhak et al. [123] discussed the need for fog vehicular computing to augment computational power and offloading data for storage. The authors proposed a fog vehicular computing framework consisting of four layers: edge network layer, service layer, core network layer, and cloud layer. The edge network layer consists of an embedded system and intelligent things. The service layer performed field area network service and multiedge services through fog computing servers. The core layer performs IP protocol, security, QoS, and broadcasting. The cloud layer consists of data centers and cloud computing systems. Kong et al. [126] proposed the offloading of LiDAR sensor measurements from autonomous vehicles to the edge cloud servers for processing and analysis. The sensors generated environment data were transmitted to a lamp post for sharing with other passing autonomous vehicles.

13.3. Artificial Intelligence, Drones, and Autonomous Vehicles. IoT-enabled drone-based application has widely perpetuated into the parcel delivery system. The integrated truck delivery approach and support from the drone systems have overcome the limitation in both delivery systems. The drones have computational resource limitations such as battery power and low payload. At the same time, the truck delivery system has the demerits of long hauling duration and lack of interior area coverage for parcel delivery. Wang et al. [138] discussed the combination of drone and truck-based parcel delivery systems. The authors proposed a framework for a simultaneous truck drone parcel delivery system. Three independent parcel delivery systems, namely, truck parcel delivery system, hybrid truck drone, and standalone drone parcel delivery system, have been explored in detail. The authors proposed scheduling and routing algorithms for the hybridized truck drone parcel delivery system. Sa et al. [139] presented an efficient framework for hybridized truck drone-based LMD. The collaborative routing strategy for truck routing along with a fleet of drones was discussed.

The estimation of efficient truck parking from where the drone can fly to deliver the parcel to customers was proposed. The collaborative routing is framed as an optimization problem using mixed linear integer mathematical model formulation [140]. The objective of this optimization problem is to minimize the delivery makespan to the last-mile customers. A greedy randomized metaheuristic-based feasible solution for a large-size problem was proposed. Fotouhi et al. [141] proposed a cost-effective visual-inertial (VI) odometry-based autonomous drone (VTOL) system. These VTOL-based autonomous drones are widely utilized for building infrastructure inspection, aerial surveillance, precision agriculture, and aerial cinematography [142]. These tasks require high performance in controller mechanism, low latency, obstacle avoidance, precise decision in landing and take-off, object tracking and picking and maneuver, and path planning.

Authors contributed to develop open-source software for system identification, calibrating parameters, and state estimation in different dynamic environments. Moon et al. [143] presented the various challenges towards autonomous drone racing technology. The authors analyzed the possibility of waypoint sequence estimation for the autonomous drone. Further high- and low-level navigations in the indoor and outdoor environment were studied. Patrik et al. [144] addressed autonomous drone systems for parcel delivery using the GNS. The medical aid delivery for patients in a remote natural calamitic scenario was considered for the study. The autonomous drone was assigned with the task

Table 5: Summary of key reference on advanced technologies for autonomous vehicles.

Research work	Year	Objective	Advanced technology A	B	C	D	Method
El-Hassan [120]	2020	IoT low-cost sensor-based AV monitoring prototype	Y				Developed method based on LiDAR sensor for sensing environment AVs. Results show that the performance of sensor is better on roads.
Wang et al. [121]	2020	On-road traffic monitoring and control mechanism for AVs	Y				Proposed real-time road traffic monitoring for AVs. Experiment results validate that CAVs can be used for smooth traffic flow
Safavi et al. [122]	2021	Multi IoT sensor-based health and fault forecasting in AVs	Y				Proposed architecture for health and fault forecasting based on multiple sensors on IoT platform
Sookhak et al. [123]	2017	IoT data augmentation for vehicular services on the cloud computing		Y	Y		Developed mechanism for sensor data offloading on the cloud for AVs
Kong [124]	2020	AV sensor data offloading and computation on cloud computing		Y			Developed resource allocation for AV sensor data computation on cloud platform
Khayyam et al. [36]	2020	Integrated approach of AI & IoT for AVs	Y				Reviewed on various approaches towards combined AI and IoT for AVs
Nanda et al. [125]	2020	Internet-based AVs	Y				Surveyed on communication protocols, security, and privacy issues
Kong et al. [126]	2017	Millimetre wave IoT wireless device communication for cloud computing support	Y	Y			Designed mmWave-based IoT system for object recognition in real time
Garg et al. [127]	2017	Hybrid approach of mobile computing, software-defined network, and cloud computing for AVS	Y	Y			Hybrid method of distributed network-based vehicle management along with software-defined networks
Coutinho and Boukerche [128]	2019	Methodology for vehicular data offloading on to the cloud infrastructure for AVs	Y	Y			Reviewed on content delivery on cloud for AVs and discuss on limitation in exiting methodologies of connected AVs
Moustafa et al. [129]	2017	Fog computing for handling AV video data	Y		Y		Developed reverse mechanism for video content deliver on fog network for AVs
Wang et al. [114]	2019	Distributed fog network for connected AVs	Y		Y		Designed a cruise control based on connected fog network based for AVs
Thakur and Malekian [130]	2017	Internet of vehicles and fog computing-based vehicle congestion monitoring	Y		Y		Reviewed and developed vehicle congestion detection mechanism as part of intelligent transport system for AVs
Du et al. [131]	2020	Distributed cooperative fog network for data handling of AVs	Y		Y		Simulation of trajectory detection based on light gated recurrent unit (Li-GRU) neural network algorithm
Hou et al. [132]	2016	Vehicle infrastructure monitoring using fog computing approach			Y		Reviewed on vehicle communication, connectivity and mobility management.
Feng et al. [133]	2018	Edge computing and ACO framework for AVs	Y			Y	Developed edge computing and ACO-based communication services
Guo et al. [134]	2017	Mobile edge network for AV data offloading.	Y			Y	Designed and studied offloading on mobile edge network for AV sensor data
Sun et al. [135]	2019	User experience enhancement through adaptive allocation of resources on edge computing for AVs	Y			Y	Developed algorithm for dynamic resource allocation on edge for AVs
Baidya et al. [136]	2020	AV applications and services based on vehicular edge computing	Y			Y	Developed applications and services using edge network for AV
Yang et al. [137]	2020	AV accident data analysis based on edge computing framework	Y			Y	Designed ML model for AV accident data analysis and forecasting the health of AVs

A: IoT, B: cloud computing, C: fog computing, and D: edge computing.

Table 6: Research on autonomous vehicles by different companies.

Year	Manufacturing company	Contribution by the companies
1920	Houdina Radio control, Chandler Motor car	Houdina introduced Radio-controlled cars in 1925. In 1926, Chandler Motor car introduced an antenna that operated small electric motors which directed every movement of the vehicle.
1930	General Motors	Radio-controlled cars driven by electromagnetic fields accompanied by circuits inserted in roadways.
1950	General Motors Firebird, Radio Corporation of America	Electronic guide systems using wires were introduced by the General Motors Firebird.
1960	Citroen DS, Bendix Corporation, Stanford University, etc.	Citroen DS introduced magnetic cables embedded in the road in 1960. In 1970, Bendix Corporation introduced AV's powered and controlled by buried cables with roadside communicators depending on computer messages. Stanford developed a small wheeled robot.
1980	Defense Advanced Research Projects Agency, Mercedes Benz, etc.	Mercedes Benz in the year 1980 developed a vision-guided robotic van. In the same decade, DARPA-introduced an Autonomous Land driven Vehicle for the first time which used Lidars, computer vision, and robotic control.
1990	VaMP,Vita-2, Mercedes Benz, Jaguar Cars, Park Shuttle, People Mover, etc.	In 1994, twin robots of VaMP and Vita-2 of Daimler Benz introduced AV's passing in lane system and lane changes with autonomous passing cars. Jaguar cars funded Lucas Industries for making parts for semiautonomous cars.Park Shuttle was the first promoted driver less people mover that used artificial reference points inserted into the road surface to cross check its position.
2000	National Institute of Standards and Technology, DARPA, Radio-frequency identification, Royal Academy of Engineering, Toyota, aluminium division of Rio Tinto, Google	In 1998, Toyoto was the first to introduce Advance Cruise Control (ACC) on a production vehicle disclosing a laser-based system for its luxury sedan version.
2003	Tesla	Various models are developed by Tesla such as Model S, Model 3, Model Y, and Model X. Tesla Semi, Cybertruck, and Roadster are also in development phase.
2017-18	Apple	Apple's self-driving car effort, Project Titan, lost around 200 employees in January 2019. Apple bought Drive.ai five months later. Apple is purportedly buying the company to hire its people rather than buy its technology. With its self-driving vehicle fleet, Apple spent 2018. The California Department of Motor Automobiles registered 70 vehicles in September 2018. Apple's fleet logged 80,739 km of autonomous driving between April 2017 and November 2018.
2020	Audi	Ford makes its debut with hands-free technology in a car. The European Union has authorized the deployment of the A8's self-driving capability on public roads. The former Tesla Autopilot manager now serves as the CTO of Audi's Autonomous Intelligent Driving (AID) business.
2020	Autoliv	An airbag in the form of a cocoon is being developed in conjunction with Autoliv and will be integrated into the seat frame. In the case of an accident, it would protect drivers from flying debris or unbuckled passengers in the backseat. A prototype of the car was shown off at the AutoMobili-D exhibition in Detroit, Michigan, in January 2019.

such as object deductions and destination position reachability using GPS. The authors also proposed an auto drone navigation algorithm based on the positional deviation between the actual and desired landing positions.

13.4. Artificial Intelligence, Knowledge Representation, and Autonomous Vehicles. Gregor et al. [140] addressed situational awareness by ontology framework for an autonomous vehicle in the manufacturing industry. The semantic representation is essential for reasoning systems and internal state machines to achieve the goal of the desired tasks. Pellkofer and Dickmanns [145] proposed an ontology to perceive the autonomous vehicle environment and robot telemetry. The work also discussed the knowledge graph for IoT robotic domine in the intelligent automotive production intralogistics environment. Asmar et al. [146] examined the multifocal dynamic visual system for an autonomous vehicle. The advanced vision system consisted of the camera on a high bandwidth pan-tilt holder that performed the active gazing for the independent system. The authors addressed both static and dynamic knowledge representation. The static knowledge included a digital map of the real-world knowledge repository about the apriority performance parameters. In Table 6, advanced research on AV by various companies is presented.

The dynamic knowledge included computers, processes, scene trees, and sequence of tasks representing the mission objects [147]. The proposed system consists of decision-making units that performed three tasks, namely, behavior decision for vehicle gazing, behavior decision for maneuver, and centrally coordinated behavior for decision-making. Zhao et al. [148] addressed the knowledge representation for autonomous vehicle driving environment in a machine-readable format. Ontologies were proposed for safe driving based on road maps, driving lanes, and surrounding driving. The authors proposed core ontologies for the enhanced driver assistance control system. The proposed ontology included map ontology, control ontology, and car ontology. The map ontology describes the road network with the roads, lanes, markings, road intersections, and traffic signal status. The control ontology described the driving action, driving state, and maneuver path of the autonomous vehicle based on the GPS. The car ontology contained the details about sensors, vehicle engine status, the vehicle's exact location, and the vehicle's speed.

13.5. Artificial Intelligence, Machine Learning, and Internet of Things for Autonomous Heavy Vehicles. The concept of autonomous vehicles, where manual driving is not required, has gained many in this busy life. It has made many automotive manufacturers exploit every opportunity in developing autonomous vehicles. The technologies such as artificial intelligence, machine learning, and IoT have raised hopes for autonomous vehicles. This evolution leads to the enhancement of data analysis and prediction processes and procedures. Artificial intelligence has gained a wide range of scope in various autonomous sectors [149]. Till now, driving assistance systems such as proximity sensors and ADAS are experienced. Now, a step ahead with machine learning and IoT concepts, the future is driving towards autonomous vehicles. Competition in the current vehicle industry forced companies to adapt to the rapidly changing environments with technologies, improved features, safety, automation, and data transfer. The AI and IoT in combination will enhance the change for self-driving autonomous vehicles (AVs). This article lets us know how the latest developments of AI and IoT will assist in the quest for automated vehicles [149].

AI, ML, and IoT are the fields of computer science used to develop intelligent devices and intelligent machines. These Intelligent machines such as AVs respond and react like human brains through ML and AI. In the goal for evolving at full automation (i.e., self-driving), it is prominent to know how AI works in hand with AV. Automated vehicles use significant amounts of input data from sensors and intelligent devices. These sensors of AV provide inputs such as time frame, movement detection, navigation directions, image recognition, voice, and word recognition, multiple touch recognition, virtual assistance, vehicle speed, vehicle acceleration and decelerations, mileage information, fuel status, vehicle location, and position [114, 150].

In October 2010, Segway Incorporated and General Motors jointly advanced a two-seat electric car with new features such as self-vehicle parking, crash avoidance, and vehicle patrol. In 2011, Volkswagen group commenced HAVEit, having features such as Radar systems, cruise control, side observation for safer lane-changing, and TAP mode to maintain a particular distance from other vehicles. In 2014, Nissan's Infiniti Q50 introduced a virtual steering column. In 2018, Google planned to release self-driving cars with all the features such as lane-changing and hassle-free parking, with all the Adaptive Cruise control options [151].

13.6. AI in Autonomous Vehicles. AI in autonomous vehicles is applied in the following phases:

(i) Information Collection

AVs are built with multiple sensors and intelligent devices such as Radar sensors, cameras to capture images, and brilliant communication cables to produce a considerable amount of data from vehicle and vehicle surroundings. This information has the lane information, road signals, road signs, surrounding vehicles movement tracking and vulnerable road user's data, parking location details, and traffic status. This information is then sent and further processed.

(ii) Path Planning

This bulk data from AV systems will be stored and clubbed with past data from earlier rides in a database known as Big Data. AI agents act on this Big Data to produce sorted and meaningful algorithms by strategy control.

(iii) Act

The decisions made by AI agents are used to detect objects, traffic, parking areas, and bicycles; pedestrians make the AV reach the destination safely. AVs are also equipped with function controls such as steering control, gestures, and speech recognition. AI agents are responsible for making final decisions in demanding driving situations.

13.7. Challenges in AI-Driven Automated Vehicles

(i) Sensor issues

Sensors of the AV play a significant role in the automation process. Sensors can be classified mainly in 3 ways. Firstly, by using the already existing sensors, i.e., speed sensor, acceleration sensor, fuel sensor, steering angle sensor, etc.

Secondly, positioning sensors of the vehicle, i.e., GPS. Thirdly, surrounding sensors such as markings on the road, inclination, signboards, weather updates, surrounding vehicles detection, and vulnerable user's detections.

(ii) Complexity and uncertainty

Complexity involves dealing with vast amounts of information gathered from sensors and training the data model. Uncertainty occurs during sensor data collection; there may be noise that makes the input errors given to the sensors.

(iii) Complex model tuning issues

Deep learning, machine learning, and reinforcement learning methods are used in AVs. As a result, complex data models are generated, and then the parameter calibration for these models becomes complex. End-user has to develop a suitable tuning model by costly trial and error method. For example, we use supervised learning algorithms in automated vehicles and suppose if the trained data set and the input dataset are entirely different in some situations such as the traffic on the lanes, which is unpredictable, here comes a problem of complex model tuning issues. Training the datasets of metropolitan, cities, semi-urban and rural areas also involves complex model tuning issues. The passing of information from trained datasets to test datasets also becomes a great challenge for the artificial intelligence technical approach in automated vehicles.

(iv) Solving the hardware problem

Multiple computing systems are interconnected in AVs. Different computing models were proposed, such as multicore systems including CPUs, heterogeneous systems, and distributed computing systems are used in AV. The significant issues with GPU, CPU, and programmable gate arrays are programmed to change image processing and computer graphics. All these are used in real-time testing applications, and the cost becomes high for commercial deployment. Hence, there is a need for advanced hardware implementation.

13.8. Statistical Learning Methods in Autonomous Vehicles. Considering different types of accidents such as rear crashes that occur frequently, driving style plays an essential role in designing ADAS Systems and Vehicle control systems. In ADAS systems, inputs for different driving styles are considered statistical methods such as acceleration, relative distance, and relative velocity [152]. Some statistical techniques used to find additional driving assistance are collision risk surrogates, trajectory feature extraction, discrete wavelet transform, and discrete Fourier transform [150]. In Cooperative Adaptive Cruise Control vehicles, we use statistical models to calculate real-time inconsistency in-vehicle communication, and kinematics laws are considered. According to an article by [149] Wang and Li, the safety of an autonomous vehicle depends on the driver's performance and road crash tests performed in a suitable environment. By utilizing the data of automated vehicle crash details, statistical methods, logistic regression, and data classification are achieved.

In recent years, the emergence of connected autonomous vehicles are noticed. According to Yan [153], carrying sensors and connected vehicles can increase energy adaptability, better routing, and less traffic on roads. To calculate the usage fuel and discharge of fuel used unsupervised learning methods are applied on the real-world datasets of autonomous vehicles. Using unsupervised learning techniques, a new way for segregating driving conditions concerning velocity and acceleration has been applied on real-time AV datasets that work effectively [154]. As a reference from an article by Wanchfeld, unsupervised methods and statistical methods are to be applied to achieve autonomous vehicles' safety on-road testing. A linear dynamic system and a mixture of a linear dynamic system for context-aware robot system and expectation minimization were used to learn the model [155]. An optimal unsupervised algorithm was introduced to increase the fastness of the response, and hierarchical, K-means, and Gaussian matrix models were used to optimize the path for vehicles [156]. Vishnukumar et al. proposed a novel method using AI core-based machine and deep learning algorithms for real-time applications such as T&V and Advanced Driver Assistance System (ADAS) to improve their efficiency [157]. Mishra et al. proposed an AI-based camera to monitor the occupants in cabins and their behavior and also discussed wave power-based autonomous vehicles to enhance the facilities in various fields [158, 159]. In 2021, Malik et al. [160] introduced a new concept vehicle as a service to reduce the CO_2 effect on the environment.

Regression algorithms are used in the cases of prediction. These algorithms are used in automated vehicles to predict and maintain a relationship between the image and its position [161]. The output of usage of this algorithm is an image, its place, and its presence. Some of the regression algorithms used in self-driving vehicles are neural network regression, decision forest regression, etc. In ADAS, the data collected from all the sensors consists of different datasets that require filtering data from raw or irrelevant data. Hence, it forms a necessity for the classification of data that uses pattern recognition. Category of data helps in reducing the dataset. The SVM and HOG are widely used for component analysis. A supervised model was designed to avoid unwanted intervention while driving [162]. Trasnea et al. used the GrisSim to learn deep learning, reinforcement learning, and genetic algorithms to maximize the speed [163].

This concept includes creating and automating mathematical models and algorithms that can optimize the ability to perform particular tasks. Machine learning proceeds from examining survey models, operations and research, and statistics and explores the data. The primary task of a machine learning algorithm in the autonomous vehicle is frequently capturing and analyzing changes in the surrounding environment. Major tasks are as follows:

(i) Object Detection

Takumi [164] proposed multispectral images as input information for object detection in traffic. These are composed of RGB images, middle infrared photos, and multilateral information. Multispectral datasets are used for object detection in traffic. Liu et al. [165] discussed that multispectral detection pedestrian is required for the safety and existence of certain autonomous driving features using ConvNet fusion architectures, which combine two ConvNets on different DNNs stages, which attain better performance. Kuznetsova described a method for real-time object detection using hybrid viola-jones cascade with the conventional neural network [166]. Object detection is the most important technique for autonomous vehicles. The nearby vehicles, traffic lights, and signals should be detected and recognized. Localization and classification is achieved by object detection.

(ii) Object Identification/Recognition

Furqan et al. proposed a method for object identification naming decision tree and decision fusion based recognition system which combines two feature sets of RGB pixel values and nonlinear points from each pixel from the dataset [167]. Lidar-based viewpoints can detect the objects of any transition, and tracking can be achieved more effectively [168]. This technique involves dividing, partition making, clustering, and monitoring.

(iii) Object Classification

Yoshioka et al. presented object classification in the real world based on ReadAda Algorithm [169]. LiDAR 3D point object clouds improve object classification accuracy to 90%, distinguishing objects, persons, and electric poles on the path [170].

(iv) Object Localization

Localization is a crucial phenomenon for developing autonomous vehicles, especially in metropolitan areas [171]. A stereo camera is used to distinguish a long-standing object from an electric pole in an environment. The particle filter approach is used for localization for vigor and sensor fusion. Vision-based localization would work more effectively in an object localization process when the data are unable to retrieve from any hardware component in its failure, and the data can be retrieved by a single camera [172].

(v) Prediction of Moment

An unexpected change in the surroundings, signboards, traffic, shape of the lanes, and vehicle condition can drastically impact the behavior system of an autonomous vehicle [173]. The movement prediction has a statistical behavior that is resolved by various machine learning and deep learning techniques. With the study of multiple autonomous vehicles behavior concerning time and distance, the position can be predicted [174].

13.9. Deep Learning and Deep Reinforcement Learning Methods for Autonomous Vehicles. Deep learning comes under machine learning. In deep learning, inputs are taken from images, text, and sound and segregated. These models have more accuracy (sometimes more than humans) [175]. Models are trained by using multiple layers of input data. Deep learning enables them to notice a stop sign or differentiate a user from an electric pole. Conventional neural network is a deep learning technique used for image classification and feature extraction from training models. This technique can be used for automating the feature extraction process and image recognition. Reinforcement learning is a promising key in diving strategies, movement and action of automated vehicles, and perception planning. The deep learning neural network is more beneficial over the conventional machine learning technique [175]. According to Lee et al., deep learning techniques are used for autonomous vehicles in following a lane without taking many lane departures. These deep learning techniques are also used to set specific angle positions for steering [176]. Reinforcement learning methods are used for maintenance and controlling various aspects of connected autonomous vehicles. In [177], the authors proposed a hybrid approach of QEN and a TSK-FIS to control the tuning parameters for fuzzy control. Gu et al. have proposed a hybrid method deep reinforcement algorithm combined with the feedback control technique to improve the performance [156].

(i) Deep Reinforcement learning

Reinforcement learning is a technique in machine learning that enables the generation of a series of decisions. Deep Reinforcement learning further enhances reinforcement learning by using deep learning and multilayered neural networks. Deep reinforcement learning techniques are used in pipelined structures to train the models of deep neural networks associated with autonomous vehicles [178]. These deep reinforcement techniques are used for acquiring sensor amalgamation and spatial characteristics.

Diplomatic decision-making is a critical aspect of advanced driving systems that involves several challenges, such as uncertainty in other drivers' behaviors and the trade-off between safety and smartness. To avoid this type of situation, we use deep reinforcement learning techniques. An ultrasonic sensor calculates the distance from a target object by discharging ultrasonic sound waves and then converts them to electronic signals. A vehicle with ultrasonic sensors can detect conditions in its area; autonomous vehicles need to work on Big Data. An ultrasonic sensor needs to get data from thousands of connected vehicles, which is required for building better algorithms [36].

Accessory, which is crucial for the Advanced Driving Support System (ADAS), is a camera. It is used for vehicle parking, lane departure warning, and detecting real-time obstacles. This image has an array of pixels. Computer vision algorithms convert images by converting low-level to high-level information images [36]. Unlike other sensors, Radar has a remarkable ability to transmit signals irrespective of poor weather conditions such as fog, rain, and snow and will not hinder even during poor light. These have overall signal perception from a vehicle. Radar has a better backup performance added to lidar and camera. Radar's output includes an object list containing speed, location, acceleration, motion type, and boundary information.

Similar to WiFi, dedicated DSRC is wireless communication. DSRC has a high data transfer rate among vehicles. DSRC is highly secured as well. These are used for both vehicles to vehicle and vehicle-to-infrastructure communications. The scope of DSRC is seen very high because of its low latency and high and secure transmission. This type of communication can be used to pay at parking slots and tolls, identify the curve approach on lanes, alert the driver, and alert construction sites.

Challenges in DSRC and AV:

(i) DSRC spectrum or band should not affect the vehicle to infrastructure performance.

(ii) Make sure that the driver responds accurately to the vehicle to infrastructure warnings.

(iii) Maintaining and managing data security.

(iv) Labeling the variables related to potential responsibility issues posed by vehicles to infrastructure communication systems.

Precise estimation of accurate position in automated vehicles becomes crucial in terms of driver safety and comfort. It is more critical at junctions and intersections to avoid accidents. Initially, GPS is used for this position tracking, but it may have some issues such as signal loss and 3 to 4 meters of accuracy. To avoid this, the use of pose sensors came into existence. Generally, in autonomous vehicles, these sensors are embedded into the road infrastructure system where the vehicle's external orientation (translation) is the output to be obtained. These sensors are placed at intersections and junctions at some heights such as surveillance cameras. According to Khayyam, an autonomous vehicle can have a right of 6 degrees of pose such as position (x, y, z) and angles while rolling ($ß$, Ý). These AV pose sensors are used to calculate GPS in the movement of position receiver, acceleration parameters, and dynamic vehicle movements.

13.10. IoT in Autonomous Vehicles. IoT can connect devices to the Internet for sharing data. All the autonomous vehicles are interconnected to send and receive data from applied sensors and intelligent devices of surrounding pedestrians and cyclists around, nearby traffic sensors, parking locations, etc.

Following components are associated with IoT in autonomous vehicles:

(1) Intelligent devices and sensors are the essential components and collect multiple pieces of information.

(2) Mobile networks (3 G/4 G/5 G) and WiFi technologies.

(3) All the data which are collected should be interchanged, stored, and processed.

(4) Cloud services in AVs: here, the software as a service is provided by the cloud.

(5) An advantage of IoT is intelligent control over vehicles, GPS, information services, etc.

IoT will support autonomous vehicles and transform the automobile industry, and in turn, the automobile industry will give a considerable boost to IoT. IoT will create an edge between auto manufacturers and software developers. IoT not only will transform the automobile industry but will also trigger a power struggle between automakers as the incumbent players on one side and software developers on the other side.

13.11. Recent Autonomous Vehicles Development Using Artificial Intelligence and Machine Learning.

(i) AV sensor technology 2020

LiDAR systems will become crucial for autonomous vehicles. It works with visual sensors, ultrasonic sensors, and Radar systems to communicate with other vehicles. LiDAR sensors can continuously watch 360 degrees, and it also provides accurate depth information. One of the essential LiDAR sensors is Velodyne-64HDL 64E. A LiDAR sensor works by emitting the laser light, and it measures how long it takes to reach the sensor. Sensors and Radars were used before, but now the LiDARs came into the picture in making the most efficient Automated Vehicles.

(ii) Vehicle-to-Infrastructure (V2I) Communication

In Vehicle-to-Infrastructure communication, data are exchanged between the vehicles and road infrastructure by using wireless transfer. Road infrastructure such as lane markings and traffic signals provide information to vehicles wirelessly and vice versa. Road infrastructure will have a significant impact on successful self-driving vehicles. V2I communication becomes brighter by adopting smart sensors in road signs, intelligent traffic signaling systems, and intelligent speed control units to endeavor smoother autonomous vehicle flow. New, creative, and intelligent infrastructure is required to stay in a digital ecosystem.

Few emerging technologies can be used to improve road safety, and mobility is like

(1) Advanced road markings can be visible to both humans and machines in any road condition.

(2) Smart signs also provide directional signage that both humans and machines can see in any road condition. Retro-reflective symbols offer more accuracy in navigation and faster decision-making for drivers as well as machines.

(3) Wireless communication is also required for driver safety, automation, and improved mobility.

(iii) Advanced driver assistance systems (ADASs)

Driving assistance systems are evolving more and more as per the need of the day. ADAS or advanced driver assistance systems will enhance the driver effectively and confidently in terms of safety. Standard ADAS systems include front collision warning, lane departure warning, adaptive cruise control, parking assistance systems, and many more. Blindspot detection systems, night vision systems, and other features enable the drivers to commute in a safer and better way for the future. Likewise, Mercedes are very keen to adopt these features to promote and to enhance safety for their customers.

14. Conclusion and Future Directions

The scientific community now accepts autonomous vehicles and autonomous driving as feasible solution due to advancement in AI. With artificial intelligence, autonomous vehicles and driving systems may make a choice that propels the industry into a new era of rapid development. Despite this, artificial intelligence has significant limitations, limiting the growth of autonomous driving. This work has conducted a comprehensive survey over artificial intelligence in autonomous vehicles, systems, and driving experiences. In observations, it is found that there is a lack of safety standards for autonomous systems, and AI is an important concept while designing the safety standards for futuristic autonomous systems. Furthermore, a comparative analysis of various studies on autonomous systems shows that integrating two or more advanced technologies (blockchain, IoT, cloud computing, fog computing, edge computing, and artificial intelligence) is required to make autonomous systems a reality. Here, the focus is drawn on how artificial intelligence monitors the vehicle's activities and movements. Intelligent tools are necessary for autonomous vehicle design and development. In this work, various latest release of tools and frameworks are analyzed in context of design techniques and programming languages used. The operational testing is essential for effective functionality of AVs. Thus, various testing techniques employed by organizations and researchers are highlighted in this work. The limitations of existing testing techniques are also discussed.

14.1. Future Directions. Various important future research directions in this field are briefly explained as follows [179–184]:

(i) An intentional attack on the AI system that interferes with its operation may put autonomous vehicles in danger of being destroyed. Attacks against stop signs, such as placing stickers on them to make them more challenging to identify, are two examples of such attacks. As a result of these modifications, artificial intelligence may erroneously detect objects, resulting in the autonomous vehicle behaving in a way that puts humans in danger. Thus, there is a need to explore the RFID or IoT-based solutions that use artificial intelligence to solve these challenges.

(ii) It is observed that self-driving cars will revolutionize our lives. There is a need that legislators must create legislation that benefits the country's economy and socially. Studies examine AVs' potential to become a "killer app" with dramatic consequences. AVs will have substantial impacts over time, even if they are still in development. Thus, there is a need to study the safety precautions before accepting them in real environments.

(iii) Deep neural networks (DNNs) enable self-driving cars to learn how to move around their surroundings independently. Human brains are similar to DNNs because of this: both learn via trial and error. There is no hard and fast rule regarding autonomous driving and how many DNNs are required. Thus, there is a need to conduct an in-depth study in the future.

(iv) A real-autonomous driving on-road environment requires millions of interactions between vehicles, people, and devices. To handle such an extensive infrastructure, there is a need for a high-end infrastructure which may be costly. Thus, there is a need to study how artificial intelligence can efficiently utilize the infrastructure for smooth autonomous experiences.

(v) In future, more intelligent tools and software should be developed to implement better path planning and object detection in autonomous vehicles. Data communication should be of more velocity as real-time decisions are to be made.

(vi) In autonomous systems, the machine learning system monitors machine activity to predict problems. The solution reduces unplanned downtime costs, extends asset life, and increases operational efficiency. There is a need to identify the best machine learning algorithms and approaches that monitor a machine or its activities. This task can be explored in the future.

(vii) Early diagnosis of vasculature via fundus imaging may be able to prevent retinopathies such as glaucoma, hypertension, and diabetes, among others, from developing [185–187]. The overall purpose of this study is to create a new way for combining the benefits of old template-matching techniques with those of more current deep learning methods in order to achieve greater efficiency. A U-shaped fully connected convolutional neural network is used to train the segmentation of vessels and backgrounds in pixels of images (U-net). Likewise, other advanced technologies such as blockchain and quantum can be explored for AVs

[188, 189]. The wireless sensor network is used in autonomous vehicles for information communication [190–193].

Abbreviations

AI: Artificial intelligence
AUV: Automated underwater vehicles
ASIL: Automotive safety integrity levels
AEV: Autonomous electric vehicle
AV: Autonomous vehicle
AVSN: Autonomous vehicle social networks
CAV: Connected autonomous vehicle
DSRC: Dedicated short-range communications
DL: Deep learning
DNN: Deep neural network
DARPA: Defense advanced research projects agency
EV: Electric vehicle
EVSE: Electric vehicle supply equipment
ELROB: European land-robot
FPGA: Field programmable gate array
GNS: Global navigation satellite system
GPS: Global positioning system
GORE: Goal-oriented requirements engineering
GPU: Graphics processing unit
GHE: Greenhouse gas emissions
HARA: Hazard analysis and risk assessment
HAVEit: Heavily automated vehicles for intelligent transport
HOG: Histogram of gradient descents
IEEE: Institute of electrical and electronics engineers
IC: Integrated circuit
IDE: Integrated development environment
IA: Intelligent Automation
ISO: International organization for standardization
IoT: Internet of things
IoV: Internet of vehicle
JESS: Java expert system shell
LMD: Last-mile delivery system
LiDAR: Light detection and ranging
ML: Machine learning
MMI: Man machine interface
MBUX: Mercedes benz user experience
NLP: Natural language processing
QEN: Q estimator network
QoL: Quality of life
QoS: Quality of services
SAE: Society of automotive engineers
SDN: Software-defined network
SLAM: Spatiotemporal localization and mapping
SVM: Support vector machines
STPA-sec: Systems theoretic process analysis
TSK-FIS: Takagi-sugeno-type fuzzy inference system
TPU: Tensor processing unit
V2X: Vehicle-to-everything
V2I: Vehicle-to-infrastructure
VTOL: Vertical take-off and landing
VUCA: Volatile, uncertain, complex, ambiguous.

Data Availability

No data were used to support the findings of this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was supported by Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2022R195), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

References

[1] A. Mukhtar, L. Xia, and T. B. Tang, "Vehicle detection techniques for collision avoidance systems: a review," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 5, pp. 2318–2338, 2015.
[2] S. Pendleton, H. Andersen, X. Du et al., "Perception, planning, control, and coordination for autonomous vehicles," *Machines*, vol. 5, no. 1, p. 6, 2017.
[3] L. M. Clements and K. M. Kockelman, "Economic effects of automated vehicles," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2606, no. 1, pp. 106–114, 2017.
[4] P. Gao, H.-W. Kaas, D. Mohr, and D. Wee, "Automotive revolution–perspective towards 2030: how the convergence of disruptive technology-driven trends could transform the auto industry," *Advanced Industries*, McKinsey and Company, Chennai, India, 2016.
[5] M. Grazia Speranza, "Trends in transportation and logistics," *European Journal of Operational Research*, vol. 264, no. 3, pp. 830–836, 2018.
[6] H. S. Mahmassani, "50th anniversary invited article-autonomous vehicles and connected vehicle systems: flow and operations considerations," *Transportation Science*, vol. 50, no. 4, pp. 1140–1162, 2016.
[7] U. Lindqvist and P. G. Neumann, "The future of the internet of things," *Communications of the ACM*, vol. 60, no. 2, pp. 26–30, 2017.
[8] P. G. Neumann, "Risks of automation," *Communications of the ACM*, vol. 59, no. 10, pp. 26–30, 2016.
[9] S. Parkinson, P. Ward, K. Wilson, and J. Miller, "Cyber threats facing autonomous and connected vehicles: future challenges," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 11, pp. 2898–2915, 2017.
[10] J. Złotowski, K. Yogeeswaran, and C. Bartneck, "Can we control it? Autonomous robots threaten human identity, uniqueness, safety, and resources," *International Journal of Human-Computer Studies*, vol. 100, pp. 48–54, 2017.
[11] M. Buehler, K. Iagnemma, and S. Singh, *The 2005 DARPA grand challenge: the great robot race*, Vol. 36, Springer, , New York, NY, USA, 2007.
[12] M. Buehler, K. Iagnemma, and S. Singh, *The DARPA urban challenge: autonomous vehicles in city traffic*, Vol. 56, Springer, , New York, NY, USA, 2009.
[13] F. E. Schneider and D. Wildermuth, "Results of the european land robot trial and their usability for benchmarking outdoor robot systems," in *Proceedings of the Conference towards Autonomous Robotic Systems*, vol. 6856, pp. 408-409, Berlin, Germany, May 2011.

[14] J. Xin, C. Wang, Z. Zhang, and N. Zheng, "China future challenge: beyond the intelligent vehicle," *IEEE Intell. Transp. Syst. Soc. Newslett*, vol. 16, no. 2, pp. 8–10, 2014.

[15] Sae International, *Surface Vehicle Recommended Practice (R) Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles*, SAE International, Pittsburgh, PA, USA, 2018.

[16] F. M. Favarò, N. Nader, S. O. Eurich, M. Tripp, and N. Varadaraju, "Examining accident reports involving autonomous vehicles in California," *PLoS One*, vol. 12, no. 9, Article ID e0184952, 2017.

[17] Tesla, "Tesla's Fully Self Driving Timeline — Tesla Forums," 2018, https://teslamotorsclub.com/tmc/tags/full-self-driving/.

[18] L L C, *Waymo Safety Report: On the Road to Fully Self-Driving*, https://ltad.com/resources/waymo-safety-report-2021.html, 2018.

[19] X. Di and R. Shi, "A survey on autonomous vehicle control in the era of mixed-autonomy: from physics-based to AI-guided driving policy learning," *Transportation Research Part C: Emerging Technologies*, vol. 125, Article ID 103008, 2021.

[20] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (VOC) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.

[21] T. Lin, M. Maire, S. Belongie et al., "Microsoft COCO: common objects in context," *European conference on computer vision*, vol. 8693, pp. 740–755, 2014.

[22] S. Zhang, R. Benenson, M. Omran, J. Hosang, and B. Schiele, "Towards reaching human performance in pedestrian detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 973–986, 2018.

[23] X. Chen, K. Kundu, Y. Zhu, H. Ma, S. Fidler, and R. Urtasun, "3D object proposals using stereo imagery for accurate object class detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 5, pp. 1259–1272, 2018.

[24] J. Fang, Y. Zhou, Y. Yu, and S. Du, "Fine-grained vehicle model recognition using A coarse-to-fine convolutional neural network architecture," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 7, pp. 1782–1792, 2017.

[25] B. Li, "3D fully convolutional network for vehicle detection in point cloud," in *Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1513–1518, Vancouver, BC, Canada, September 2017.

[26] H. S. Lee and K. Kim, "Simultaneous traffic sign detection and boundary estimation using convolutional neural network," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 5, pp. 1652–1663, 2018.

[27] H. Luo, Y. Yang, B. Tong, F. Wu, and B. Fan, "Traffic sign recognition using a multi-task convolutional neural network," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 4, pp. 1100–1111, 2018.

[28] M. Bach, S. Reuter, and K. Dietmayer, "Multi-camera traffic light recognition using a classifying Labeled Multi-Bernoulli filter," in *Proceedings of the 2017 IEEE Intelligent Vehicles Symposium (IV)*, pp. 1045–1051, Los Angeles, CA, USA, June 2017.

[29] K. Behrendt, L. Novak, and R. Botros, "A deep learning approach to traffic lights: detection, tracking, and classification," in *Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1370–1377, Singapore, June 2017.

[30] Y. Ma, Z. Wang, H. Yang, and L. Yang, "Artificial intelligence applications in the development of autonomous vehicles: a survey," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 2, pp. 315–329, 2020.

[31] D. Bojarski, Del Testa, D. Dworakowski, B. Firner, and B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, and U. Muller, J. K. Zhang, X. Zhang, J. K. Zhao, and K. Zieba, *End to End Learning for Self-Driving Cars*, https://arxiv.org/abs/1604.07316, 2016.

[32] M. Bojarski, P. Yeres, A. Choromanska et al., *Explaining How a Deep Neural Network Trained with End-To-End Learning Steers a Car*, https://arxiv.org/abs/1704.07911, 2017.

[33] G. Notomista and M. Botsch, "A machine learning approach for the segmentation of driving maneuvers and its application in autonomous parking," *Journal of Artificial Intelligence and Soft Computing Research*, vol. 7, no. 4, pp. 243–255, 2017.

[34] K. Akermi, S. Chouraqui, and B. Boudaa, "Novel SMC control design for path following of autonomous vehicles with uncertainties and mismatched disturbances," *International Journal of Dynamics and Control*, vol. 8, no. 1, pp. 254–268, 2018.

[35] P. F. Alcantarilla, S. Stent, G. Ros, R. Arroyo, and R. Gherardi, "Street-view change detection with deconvolutional networks," *Autonomous Robots*, vol. 42, no. 7, pp. 1301–1322, 2018.

[36] H. Khayyam, B. Javadi, M. Jalili, and R. N. Jazar, "Artificial intelligence and internet of things for autonomous vehicles," *Nonlinear Approaches in Engineering Applications*, vol. 1, pp. 39–68, 2020.

[37] M. Cunneen, M. Mullins, and F. Murphy, "Autonomous vehicles and embedded artificial intelligence: the challenges of framing machine driving decisions," *Applied Artificial Intelligence*, vol. 33, no. 8, pp. 706–731, 2019.

[38] A. Osório and A. Pinto, "Information, uncertainty and the manipulability of artificial intelligence autonomous vehicles systems," *International Journal of Human-Computer Studies*, vol. 130, pp. 40–46, 2019.

[39] J. Li, H. Cheng, H. Guo, and S. Qiu, "Survey on artificial intelligence for vehicles," *Automotive Innovation*, vol. 1, no. 1, pp. 2–14, 2018.

[40] L. Kunze, N. Hawes, T. Duckett, M. Hanheide, and T. Krajnik, "Artificial intelligence for long-term robot autonomy: a survey," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 4023–4030, 2018.

[41] K. Kim, J. S. Kim, S. Jeong, J.-H. Park, and H. K. Kim, "Cybersecurity for autonomous vehicles: review of attacks and defense," *Computers & Security*, vol. 103, Article ID 102150, 2021.

[42] S. Grigorescu, B. Trasnea, T. Cocias, and G. Macesanu, "A survey of deep learning techniques for autonomous driving," *Journal of Field Robotics*, vol. 37, no. 3, pp. 362–386, 2020.

[43] H. Ning, R. Yin, A. Ullah, and F. Shi, "A survey on hybrid human-artificial intelligence for autonomous driving," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–16, 2021.

[44] A. Kumar, R. Krishnamurthi, A. Nayyar, A. K. Luhach, M. S. Khan, and A. Singh, "A novel Software-Defined Drone Network (SDDN)-based collision avoidance strategies for on-road traffic monitoring and management," *Vehicular Communications*, vol. 28, Article ID 100313, 2021.

[45] H. Kim, J. Ben-Othman, L. Mokdad, J. Son, and C. Li, "Research challenges and security threats to AI-driven 5G virtual emotion applications using autonomous vehicles, drones, and smart devices," *IEEE Network*, vol. 34, no. 6, pp. 288–294, 2020.

[46] Y. Wang, Z. Su, K. Zhang, and A. Benslimane, "Challenges and solutions in autonomous driving: a blockchain approach," *IEEE Network*, vol. 34, no. 4, pp. 218–226, 2020.

[47] I. Yaqoob, L. U. Khan, S. M. A. Kazmi, M. Imran, N. Guizani, and C. S. Hong, "Autonomous driving cars in smart cities: recent advances, requirements, and challenges," *IEEE Network*, vol. 34, no. 1, pp. 174–181, 2020.

[48] K. Ren, Q. Wang, C. Wang, Z. Qin, and X. Lin, "The security of autonomous driving: threats, defenses, and future directions," *Proceedings of the IEEE*, vol. 108, no. 2, pp. 357–372, 2020.

[49] A. Singandhupe and H. M. La, "A review of SLAM techniques and security in autonomous driving," in *Proceedings of the 2019 Third IEEE International Conference on Robotic Computing (IRC)*, Naples, Italy, February 2019.

[50] A. L. Kun, S. Boll, and A. Schmidt, "Shifting gears: user interfaces in the age of autonomous driving," *IEEE Pervasive Computing*, vol. 15, no. 1, pp. 32–38, 2016.

[51] Z. Du, C. Wu, T. Yoshinaga, K.-L. A. Yau, Y. Ji, and J. Li, "Federated learning for vehicular internet of things: recent advances and open issues," *IEEE Open Journal of the Computer Society*, vol. 1, pp. 45–61, 2020.

[52] S. Singh, "Critical reasons for crashes investigated in the national motor vehicle crash causation survey," 2015, https://trid.trb.org/view.aspx?id=1346216&source=post_page.

[53] J. Cui, G. Sabaliauskaite, L. S. Liew, F. Zhou, and B. Zhang, "Collaborative analysis framework of safety and security for autonomous vehicles," *IEEE Access*, vol. 7, Article ID 148672, 2019.

[54] J. Cui and S. Sabaliauskaite, "On the alignment of safety and security for autonomous vehicles," in *Proceedings of the IARIA CYBER*, pp. 1–6, Singapore, June 2017.

[55] "International Organization for Standardization (Iso), *ISO-26262: Road Vehicles - Functional safety*, ISO, Geneva, Switzerland, 2016.

[56] Sae J3061 Vehicle Cybersecurity Systems Engineering Committee, *Cybersecurity Guidebook for Cyber-Physical Vehicle Systems*, SAE International, Warrendale, PA, USA, 2016.

[57] A. Banerjee, K. K. Venkatasubramanian, T. Mukherjee, and S. K. S. Gupta, "Ensuring safety, security, and sustainability of mission-critical cyber-physical systems," *Proceedings of the IEEE*, vol. 100, no. 1, pp. 283–299, 2012.

[58] L. Piètre-Cambacédès and M. Bouissou, "Cross-fertilization between safety and security engineering," *Reliability Engineering & System Safety*, vol. 110, pp. 110–126, 2013.

[59] T. Novak and A. Treytl, "Functional safety and system security in automation systems - a life cycle model," in *Proceedings of the 2008 IEEE International Conference on Emerging Technologies and Factory Automation*, pp. 311–318, Hamburg, Germany, September 2008.

[60] Google, *Road Vehicles-Functional Safety*, https://www.googleadservices.com/pagead/aclk?sa=L&ai=DChcSEwjZ4bDg9fn3AhXXfCsKHeVSDYEYABAAGgJzZg&ae=2&ohost=www.google.com&cid=CAASJeRopVIgmiZ7d1mANMmKF_jW4-quFYcaMouwMrcMShbDVX_fv28&sig=AOD64_2YDCdrg5QNymRADIJYwSHs8YZZNw&q&adurl&ved=2ahUKEwjEo6rg9fn3AhX_aGwGH, 2016.

[61] T. Stolte, G. Bagschik, A. Reschka, and M. Maurer, "Hazard analysis and risk assessment for an automated unmanned protective vehicle," in *Proceedings of the 2017 IEEE Intelligent Vehicles Symposium (IV)*, pp. 1848–1855, Los Angeles, CA, USA, June 2017.

[62] E. Ruijters and M. Stoelinga, "Fault tree analysis: a survey of the state-of-the-art in modeling, analysis and tools," *Computer Science Review*, vol. 15-16, pp. 29–62, 2015.

[63] M. Corporation, "The Stride Threat Model," 2005, https://docs.microsoft.com/en-us/previous-versions/commerce-server/ee823878(v=cs.20)?redirectedfrom=MSDN.

[64] G. Macher, H. Sporer, R. Berlach, E. Armengaud, and C. Kreiner, "SAHARA: a security-aware hazard and risk analysis method," in *Proceedings of the In 2015 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pp. 621–624, Grenoble, France, March 2015.

[65] J. Cui and G. Sabaliauskaite, "US$$2$$2: an unified safety and security analysis method for autonomous vehicles," *Advances in Intelligent Systems and Computing*, vol. 886, pp. 600–611, 2018.

[66] C. Ponsard, G. Dallons, and P. Massonet, "Goal-oriented Co-engineering of security and safety requirements in cyber-physical systems," in *Proceedings of the International Conference on Computer Safety Reliability, and Security*, pp. 334–345, Springer, Trondheim, Norway, September 2016.

[67] N. Leveson, *Engineering a Safer World : Systems Thinking Applied to Safety*, MIT Press, Cambridge, UK, 2011.

[68] W. Young and N. Leveson, "Systems thinking for safety and security," in *Proceedings of the 29th Annual Computer Security Applications Conference*, pp. 1–8, New Orleans, LA, USA, December 2013.

[69] I. Friedberg, K. McLaughlin, P. Smith, D. Laverty, and S. Sezer, "STPA-SafeSec: safety and security analysis for cyber-physical systems," *Journal of Information Security and Applications*, vol. 34, pp. 183–196, 2017.

[70] S. Burton, L. Gauerhof, and C. Heinzemann, "Making the case for safety of machine learning in highly automated driving," *International Conference on Computer Safety Reliability, and Security*, vol. 10489, pp. 5–16, 2017.

[71] K. Varshney, "Engineering safety in machine learning," in *Proceedings of the 2016 Information Theory and Applications Workshop (ITA)*, pp. 1–5, IEEE, La Jolla, CA, USA, February 2016.

[72] D. Amodei, C. Olah, J. Steinhardt, P. Christiano, J. Schulman, and D. Mané, "Concrete Problems in AI Safety," 2016, https://arxiv.org/abs/1606.06565.

[73] S. G. Tzafestas, "Synergy of IoT and AI in modern society: the robotics and automation case," *Robotics & Automation Engineering*, vol. 31, no. 5, pp. 1–15, 2018.

[74] S. Soomro, M. H. Miraz, A. Prasanth, and M. Abdullah, "Artificial intelligence enabled IoT: traffic congestion reduction in smart cities," in *Proceedings of the Smart Cities Symposium 2018*, Bahrain, April 2018.

[75] D. Minovski, C. Ahlund, and K. Mitra, "Modeling quality of IoT experience in autonomous vehicles," *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 3833–3849, 2020.

[76] W. Schwarting, J. Alonso-Mora, and D. Rus, "Planning and decision-making for autonomous vehicles," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 1, pp. 187–210, 2018.

[77] B. V. Philip, T. Alpcan, J. Jin, and M. Palaniswami, "Distributed real-time IoT for autonomous vehicles," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 2, pp. 1131–1140, 2019.

[78] S. Kato, E. Takeuchi, Y. Ishiguro, Y. Ninomiya, K. Takeda, and T. Hamada, “An open approach to autonomous vehicles,” *IEEE Micro*, vol. 35, no. 6, pp. 60–68, 2015.

[79] R. Novickis, A. Levinskis, R. Kadikis, V. Fescenko, and K. Ozols, “Functional architecture for autonomous driving and its implementation,” in *Proceedings of the 2020 17th Biennial Baltic Electronics Conference (BEC)*, pp. 1–6, IEEE, Tallinn, Estonia, October 2020.

[80] M. Martínez-Díaz and F. Soriguera, “Autonomous vehicles: theoretical and practical challenges,” *Transportation Research Procedia*, vol. 33, pp. 275–282, 2018.

[81] S A E, *“(Sae), Society of Automotive Engineers*, Springer, Warrendale, PA, USA, 2014.

[82] B. Okumura, M. R. James, Y. Kanzawa et al., “Challenges in perception and decision making for intelligent automotive vehicles: a case study,” *IEEE Transactions on Intelligent Vehicles*, vol. 1, no. 1, pp. 20–32, 2016.

[83] P. Koopman and M. Wagner, “Challenges in autonomous vehicle testing and validation,” *SAE International Journal of Transportation Safety*, vol. 4, no. 1, pp. 15–24, 2016.

[84] B. Paden, M. Cap, S. Z. Yong, D. Yershov, and E. Frazzoli, “A survey of motion planning and control techniques for self-driving urban vehicles,” *IEEE Transactions on intelligent vehicles*, vol. 1, no. 1, pp. 33–55, 2016.

[85] A. Miglani and N. Kumar, “Deep learning models for traffic flow prediction in autonomous vehicles: a review, solutions, and challenges,” *Vehicular Communications*, vol. 20, Article ID 100184, 2019.

[86] N. Jouppi, C. Young, N. Patil et al., “In-datacenter performance analysis of a tensor processing unit,” in *Proceedings of the 44th annual international symposium on computer architecture*, pp. 1–12, Toronto, ON, Canada, June2017.

[87] A. Bhat, S. Aoki, and R. Rajkumar, “Tools and methodologies for autonomous driving systems,” *Proceedings of the IEEE*, vol. 106, no. 9, pp. 1700–1716, 2018.

[88] L. Long, S. Hanford, O. Janrathitikarn, G. Sinsley, and J. Miller, “A review of intelligent systems software for autonomous vehicles,” in *Proceedings of the IEEE Symposium on Computational Intelligence in Security and Defense Applications*, Honolulu, HI, USA, April 2007.

[89] D. De Niz, G. Bhatia, and R. Rajkumar, “Model-based development of embedded systems: the sysweaver approach,” in *Proceedings of the Twelveth IEEE Real-Time and Embedded Technology and Applications Symposium*, San Jose, CA, USA, April 2006.

[90] C. Berger and B. Rumpe, “Engineering autonomous driving software,” 2014, https://arxiv.org/ftp/arxiv/papers/1409/1409.6579.pdf.

[91] R. Lattarulo, J. Pérez, and M. Dendaluce, “A complete framework for developing and testing automated driving controllers,” *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 258–263, 2017.

[92] Y. Kang, H. Yin, and C. Berger, “Test your self-driving algorithm: an overview of publicly available driving datasets and virtual testing environments,” *IEEE Transactions on Intelligent Vehicles*, vol. 4, no. 2, pp. 171–185, 2019.

[93] S. Shah, D. Dey, C. Lovett, and A. Kapoor, “Airsim: high-fidelity visual and physical simulation for autonomous vehicles,” *Field and Service Robotics*, vol. 5, pp. 621–635, 2018.

[94] S. Ravi, “Ces 2019: The Rise of AI in Automotive - Microsoft Industry Blogs,” https://cloudblogs.microsoft.com/industry-blog/automotive/2019/01/09/ces-2019-the-rise-of-ai-in-automotive/.

[95] G. Jahangirova, A. Stocco, and P. Tonella, “Quality Metrics and Oracles for Autonomous Vehicles Testing,” in *Proceedings of the 2021 Fourteenth IEEE Conference on Software Testing, Verification and Validation (ICST)*, April 2021.

[96] G. A. Lewis, S. Bellomo, and A. Galyardt, “Component Mismatches Are a Critical Bottleneck to Fielding AI-Enabled Systems in the Public Sector,” 2019, https://arxiv.org/abs/1910.06136.

[97] I. Ozkaya, “What is really different in engineering AI-enabled systems?” *IEEE Software*, vol. 37, no. 4, pp. 3–6, 2020.

[98] W. Xu, “From Automation to Autonomy and Autonomous Vehicles: Challenges and Opportunities for Human-Computer Interaction,” *Magazine*, vol. 28, no. 1, pp. 48–53, 2020.

[99] A. Ullrich and M. Pfennigbauer, “Advances in lidar point cloud processing,” in *Proceedings of the Laser Radar Technology and Applications International Society for Optics and Photonics*, Baltimore, Maryland, USA, May 2019.

[100] C. Mc Goldrick, V. Rabsatt, and M. Gerla, “Independent Active Ageing-The Role of 5G and Autonomous Vehicles,” *IEEE COMSOC MMTC E-Letter*, vol. 10, no. 4, Piscataway, N, J, USA, 2015.

[101] V. Chaudhary, “Reconfigurable computing for smart vehicles,” in *Proceedings of the Smart Cities*, pp. 135–147, Springer, Tetouan Morocco, October 2018.

[102] S. He, “Who is liable for the UBER self-driving crash? Analysis of the liability allocation and the regulatory model for autonomous vehicles,” in *Proceedings of the Autonomous Vehicles*, pp. 93–111, Springer, Singapore, December 2021.

[103] A. J. London and D. Danks, “Regulating autonomous vehicles,” in *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 216–221, New York, NY, USA, December 2018.

[104] S. Van Uytsel, “Testing autonomous vehicles on public roads: facilitated by a series of alternative, often soft, legal instruments,” in *Proceedings of the Autonomous Vehicles*, pp. 39–64, Springer, Singapore, December 2021.

[105] X. Zhao, K. Salako, L. Strigini, V. Robu, and D. Flynn, “Assessing safety-critical systems from operational testing: a study on autonomous vehicles,” *Information and Software Technology*, vol. 128, Article ID 106393, 2020.

[106] S. Ali, “Self-Driving Cars: Automation Testing Using Udacity Simulator,” *International Research Journal of Engineering and Technology (IRJET)*, vol. 8, no. 4, 2021.

[107] X. Xie, J. W. K. Ho, C. Murphy, G. Kaiser, B. Xu, and T. Y. Chen, “Testing and validating machine learning classifiers by metamorphic testing,” *Journal of Systems and Software*, vol. 84, no. 4, pp. 544–558, 2011.

[108] W. Wei, G. Luo, J. Ran, and J. Li, “Zilong: a tool to identify empty images in camera-trap data,” *Ecological Informatics*, vol. 55, Article ID 101021, 2020.

[109] J. Hou, H. Zeng, L. Cai, J. Zhu, J. Chen, and K.-K. Ma, “Multi-label learning with multi-label smoothing regularization for vehicle re-identification,” *Neurocomputing*, vol. 345, pp. 15–22, 2019.

[110] J. Pisarov and G. Mester, “The use of autonomous vehicles in transportation,” *Tehnika*, vol. 76, no. 2, pp. 171–177, 2021.

[111] N. Andra-Teodora, “Survey of autonomous vehicles in scientific applications,” *Mircea Cel Batran, Naval Academy, Scientific Bulletin*, vol. 20, no. 2, pp. 52–56, 2017.

[112] P. Srinivas, “Literature survey on autonomous rover,” *Turkish Journal of Computer and Mathematical Education*, vol. 12, no. 13, pp. 82–88, 2021.

[113] A. Jain, “NPTEL Electric Vehicle Part I,” 2018, https://nptel.ac.in/courses/108102121.

[114] Z. Wang, Y. Guo, Y. Gao, C. Fang, M. Li, and Y. Sun, "Fog-based distributed networked control for connected autonomous vehicles," *Wireless Communications and Mobile Computing*, vol. 2020, Article ID 8855655, 11 pages, 2020.

[115] K. Bhadane, P. Sanjeevikumar, and B. Khan, M. Thakre, A. Ahmad, T. Jaware, D. P. Patil, A. S. Pande, A comprising study on modernization of electric vehicle subsystems, challenges, opportunities and strategies for its further development," in *Proceedings of the 2021 4th Biennial International Conference on Nascent Technologies in Engineering (ICNTE)*, January 2021.

[116] K. V. Bhadane, T. Jaware, D. P. Patil, and A. Nayyar, "Wind energy system grid integration and grid code requirements of wind energy system," *Control and Operation of Grid-Connected Wind Energy Systems*, Springer, , pp. 247–273, Cham, Switzerland, March 2021.

[117] M. S. Ballal, K. V. Bhadane, R. M. Moharil, and H. Suryawanshi, "A control and protection model for the distributed generation and energy storage systems in microgrids," *Journal of Power Electronics*, vol. 16, no. 2, pp. 748–759, 2016.

[118] K. Bhadane, M. Ballal, and R. Moharil, "Enhancement of distributed generation by using custom power device," *Journal of electronic science and technology*, vol. 13, no. 3, pp. 246–254, 2015.

[119] K. V. Bhadane, M. S. Ballal, and A. Nayyar, D. P. Patil, T. H. Jaware, H. P. Shukla, A comprehensive study of harmonic pollution in large penetrated grid-connected wind farm," *Mapan*, vol. 36, no. 4, pp. 729–749, 2020.

[120] F. T. El-Hassan, "Experimenting with sensors of a low-cost prototype of an autonomous vehicle," *IEEE Sensors Journal*, vol. 20, no. 21, Article ID 13131, 2020.

[121] J. Wang, Y. X. Q. Zheng, J. Wang, and K. Li, "Controllability analysis and optimal control of mixed traffic flow with human-driven and autonomous vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 12, pp. 7445–7459, 2020.

[122] S. Safavi, M. A. Safavi, H. Hamid, and S. Fallah, "Multi-sensor fault detection, identification, isolation and health forecasting for autonomous vehicles," *Sensors*, vol. 21, no. 7, 2547 pages, 2021.

[123] M. Sookhak, F. R. Yu, Y. He et al., "Fog vehicular computing: augmentation of fog computing using vehicular cloud computing," *IEEE Vehicular Technology Magazine*, vol. 12, no. 3, pp. 55–64, 2017.

[124] P.-Y. Kong, "Computation and sensor offloading for cloud-based infrastructure-assisted autonomous vehicles," *IEEE Systems Journal*, vol. 14, no. 3, pp. 3360–3370, 2020.

[125] A. Nanda, D. Puthal, J. J. P. C. Rodrigues, and S. A. Kozlov, "Internet of autonomous vehicles communications security: overview, issues, and directions," *IEEE Wireless Communications*, vol. 26, no. 4, pp. 60–65, 2019.

[126] L. Kong, M. K. Khan, F. Wu, G. Chen, and P. Zeng, "Millimeter-wave wireless communications for IoT-cloud supported autonomous vehicles: overview, design, and challenges," *IEEE Communications Magazine*, vol. 55, no. 1, pp. 62–68, 2017.

[127] S. Garg, K. Kaur, S. H. Ahmed, A. Bradai, G. Kaddoum, and M. Atiquzzaman, "MobQoS: mobility-aware and QoS-driven SDN framework for autonomous vehicles," *IEEE Wireless Communications*, vol. 26, no. 4, pp. 12–20, 2019.

[128] R. W. L. Coutinho and A. Boukerche, "Guidelines for the design of vehicular cloud infrastructures for connected autonomous vehicles," *IEEE Wireless Communications*, vol. 26, no. 4, pp. 6–11, 2019.

[129] H. Moustafa, E. M. Schooler, and J. McCarthy, "Reverse CDN in Fog Computing: the lifecycle of video data in connected and autonomous vehicles," in *Proceedings of the 2017 IEEE Fog World Congress (FWC)*, pp. 1–5, IEEE, Santa Clara, CA, USA, November 2017.

[130] A. Thakur and R. Malekian, "Fog computing for detecting vehicular congestion, an internet of vehicles based approach: a review," *IEEE Intelligent Transportation Systems Magazine*, vol. 11, no. 2, pp. 8–16, 2019.

[131] H. Du, S. Leng, F. Wu, X. Chen, and S. Mao, "A new vehicular fog computing architecture for cooperative sensing of autonomous driving," *IEEE Access*, vol. 8, Article ID 10997, 2020.

[132] X. Hou, Y. Li, M. Chen, D. Wu, D. Jin, and S. Chen, "Vehicular fog computing: a viewpoint of vehicles as the infrastructures," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 6, pp. 3860–3873, 2016.

[133] J. Feng, Z. Liu, C. Wu, and Y. Ji, "AVE: autonomous vehicular edge computing framework with ACO-based scheduling," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 12, Article ID 10660, 2017.

[134] H. Guo, J. Liu, and J. Zhang, "Computation offloading for multi-access mobile edge computing in ultra-dense networks," *IEEE Communications Magazine*, vol. 56, no. 8, pp. 14–19, 2018.

[135] X. Sun, J. Zhao, X. Ma, and Q. Li, "Enhancing the user experience in vehicular edge computing networks: an adaptive resource allocation approach," *IEEE Access*, vol. 7, Article ID 161074, 2019.

[136] S. Baidya, Y.-J. Ku, H. Zhao, J. Zhao, and S. Dey, "Vehicular and edge computing for emerging connected and autonomous vehicle applications," in *Proceedings of the 57th ACM/IEEE Design Automation Conference*, San Francisco, CA, USA, July 2020.

[137] B. Yang, X. Cao, X. Li, C. Yuen, and L. Qian, "Lessons learned from accident of autonomous vehicle testing: an edge learning-aided offloading framework," *IEEE Wireless Communications Letters*, vol. 9, no. 8, pp. 1182–1186, 2020.

[138] D. Wang, P. Hu, J. Du, P. Zhou, T. Deng, and M. Hu, "Routing and scheduling for hybrid truck-drone collaborative parcel delivery with independent and truck-carried drones," *IEEE Internet of Things Journal*, vol. 6, no. 6, Article ID 10483, 2019.

[139] I. Sa, M. Kamel, and M. Burri, M. Bloesch, R. Khanna, and M. Popović, J. Nieto, R. Siegwart, "Build your own visual-inertial drone: a cost-effective and open-source autonomous drone," *IEEE Robotics and Automation Magazine*, vol. 25, no. 1, pp. 89–103, 2018.

[140] R. Gregor, M. Lutzeler, M. Pellkofer, K.-H. Siedersberger, and E. D. Dickmanns, "EMS-Vision: a perceptual system for autonomous vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 3, no. 1, pp. 48–59, 2002.

[141] A. Fotouhi, M. Ding, and M. Hassan, "Understanding autonomous drone maneuverability for Internet of Things applications," in *Proceedings of the 2017 IEEE 18th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, June 2017.

[142] C. Huang, F. Gao, and Jie Pan, Z. Yang, W. Qiu, P. Chen and X. Yang, S. Shen, K.-T. Cheng, ACT: "an autonomous drone cinematography system for action scenes," in *Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA)*, May 2018.

[143] H. Moon, J. Martinez-Carranza, and T. Cieslewski, M. Faessler, D. Falanga, A. Simovic, D. Scaramuzza, S. Li, M. Ozo, C. De Wagter, G. de Croon, S. Hwang, S. Jung, H. Shim, H. Kim, M. Park, T.-C. Au, and S. J. Kim, Challenges and implemented technologies used in autonomous drone racing," *Intelligent Service Robotics*, vol. 12, no. 2, pp. 137–148.

[144] A. Patrik, G. Utama, A. A. S. Gunawan et al., "GNSS-based navigation systems of autonomous drone for delivering items," *Journal of Big Data*, vol. 6, no. 1, pp. 53–14, 2019.

[145] M. Pellkofer and E. D. Dickmanns, "Behavior decision in autonomous vehicles," *Intelligent Vehicle Symposium*, vol. 2, pp. 495–500, 2002.

[146] E. Asmar, B. Syrine Chelly, and M. Färber, *AWARE: An Ontology for Situational Awareness of Autonomous Vehicles in Manufacturing*, 2020.

[147] M. Buechel, G. Hinz, F. Ruehl, H. Schroth, C. Gyoeri, and A. Knoll, "Ontology-based traffic scene modeling, traffic regulations dependent situational awareness and decision-making for automated vehicles," in *Proceedings of the 2017 IEEE Intelligent Vehicles Symposium (IV)*, pp. 1471–1476, Los Angeles, CA, USA, June 2017.

[148] L. Zhao, I. Ryutaro, M. Seiichi, and S. Yutaka, "Core Ontologies for safe autonomous driving," in *Proceedings of the International Semantic Web Conference*, 2015.

[149] S. Wang and Z. Li, "Exploring the mechanism of crashes with automated vehicles using statistical modeling approaches," *PLoS One*, vol. 14, no. 3, Article ID e0214550, 2019.

[150] F. Alotibi and M. Abdelhakim, "Anomaly detection for cooperative adaptive cruise control in autonomous vehicles using statistical learning and kinematic model," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 6, pp. 3468–3478, 2021.

[151] K. Bimbraw, "Autonomous cars: past, present and future - a review of the developments in the last century, the present scenario and the expected future of autonomous vehicle technology," in *Proceedings of the 12th International Conference on Informatics in Control, Automation and Robotics*, Colmar, France, July 2015.

[152] H. M. Eraqi, Y. Abouelnaga, M. H. Saad, and M. N. Moustafa, "Driver distraction identification with an ensemble of convolutional neural networks," *Journal of Advanced Transportation*, vol. 2019, Article ID 4125865, 12 pages, 2019.

[153] C. Yan, *Fuel Economy and Emission Testing for Connected and Automated Vehicles Using Real-world Driving Datasets*, University of Michigan, Ann Arbor, CA, USA, 2018.

[154] W. Wachenfeld and H. Winner, "The new role of road testing for the safety validation of automated vehicles," in *Automated Driving*Springer, Cham, Switzerland, 2017.

[155] B. Abegaz, "ASDVC-A self-driving vehicle controller using unsupervised machine learning," in *Proceedings of the IEEE International Conference on Environment and Electrical Engineering and 2020 IEEE Industrial and Commercial Power Systems Europe*, Madrid, Spain, June 2020.

[156] W. Gu, X. Xu, and J. Yang, "Path following with supervised deep reinforcement learning," in *Proceedings of the 2017 4th IAPR Asian Conference on Pattern Recognition (ACPR)*, pp. 448–452, Nanjing, China, November 2017.

[157] H. Vishnukumar, B. Butting, C. Müller, and E. Sax, "Machine learning and deep neural network—artificial intelligence core for lab and real-world test and validation for ADAS and autonomous vehicles: AI for efficient and quality test and validation," in *Proceedings of the 2017 Intelligent Systems Conference (IntelliSys)*, pp. 714–721, IEEE, London, UK, September 2017.

[158] A. Mishra, J. Kim, D. Kim, J. Cha, and S. Kim, "An intelligent in-cabin monitoring system in fully autonomous vehicles," in *Proceedings of the 2020 International SoC Design Conference (ISOCC)*, Yeosu, Korea (South), October 2020.

[159] A. Dallolio, B. Agdal, A. Zolich, J. A. Alfredsen, and T. A. Johansen, "Long-endurance green energy autonomous surface vehicle control architecture," in *Proceedings of the Oceans 2019 Mts/ieee Seattle*, Seattle, WA, USA, October 2019.

[160] S. Malik, H. Khattak, Z. Ameer, U. Shoaib, H. Tayyab Rauf, and H. Song, "Proactive scheduling and resource management for connected autonomous vehicles: a data science perspective," *IEEE Sensors Journal*, vol. 21, 2021.

[161] N. E. A. Motamedidehkordi, "Modeling tactical lane-change behavior for automated vehicles: a supervised machine learning approach," in *Proceedings of the IEEE International Conference on Models and Technologies for Intelligent Transportation Systems*, Naples, Italy, June 2017.

[162] V. Romero-Cano, J. Nieto, and G. Agamennoni, "Unsupervised motion learning from a moving platform," in *Proceedings of the IEEE Intelligent Vehicles Symposium*, Gold Coast, Australia, June 2013.

[163] B. Trasnea, L. Marina, A. Vasilcoi, C. Pozna, and S. Grigorescu, "GridSim: a vehicle kinematics engine for deep neuroevolutionary control in autonomous driving," in *Proceedings of the 2019 Third IEEE International Conference on Robotic Computing*, Naples, Italy, February 2019.

[164] K. E. A. Takumi, "Multispectral object detection for autonomous vehicles," in *Proceedings of the on Thematic Workshops of ACM Multimedia*, pp. 35–49, New York, NY, USA, October 2017.

[165] J. Liu, S. Zhang, S. Wang, and D. N. Metaxas, "Multispectral deep neural networks for pedestrian detection," 2016, https://arxiv.org/abs/1611.02644.

[166] E. Kuznetsova, E. Shvets, and D. Nikolaev, "Viola-Jones based hybrid framework for real-time object detection in multispectral images Eighth International Conference on Machine Vision," in *Proceedings of the Eighth International Conference on Machine Vision (ICMV 2015)*, Barcelona, Spain, December 2015.

[167] A. Furqan, R. Mehmood, and I. Katib, "D2TFRS: an object recognition method for autonomous vehicles based on RGB and spatial values of pixels," in *Proceedings of the International Conference on Smart Cities, Infrastructure, Technologies and Application*, vol. 224Cham.

[168] R. Dominguez, E. Onieva, J. Alonso, J. Villagra, and C. González, "LIDAR based perception solution for autonomous vehicles," in *Proceedings of the International Conference on Intelligent Systems Design and Applications*, November 2011.

[169] M. Yoshioka, N. Suganuma, K. Yoneda, and M. Aldibaja, "Real-time object classification for autonomous vehicle using LIDAR," in *Proceedings of the 2017 International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS)*, Okinawa, Japan, November 2017.

[170] V. U. Maheswari, G. Varaprasad, and S. V. Raju, "Local directional maximum edge patterns for facial expression recognition," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 5, pp. 4775–4783, 2020.

[171] R. Spangenberg, D. Goehring, and R. Rojas, "Pole-based localization for autonomous vehicles in urban scenarios," in *Proceedings of the 2016 IEEE/RSJ International Conference on*

Intelligent Robots and Systems (IROS), Daejeon, Republic of Korea, October 2016.
[172] S. Zahedian, K. Sadabadi, and A. Nohekhan, “Localization of autonomous vehicles: proof of concept for a computer vision approach,” 2021, https://arxiv.org/abs/2104.02785.
[173] S. Paravarzar and B. Mohammad, “Motion prediction on self-driving cars: a review,” 2020, https://arxiv.org/abs/2011.03635.
[174] S. Mozaffari, O. Y. Al-Jarrah, M. Dianati, P. Jennings, and A. Mouzakitis, “Deep Learning-Based Vehicle Behavior Prediction for Autonomous Driving Applications: A Review,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 1, pp. 33–47, 2020.
[175] M. Lee, K. Y. Han, J. Yu, and Y.-S. Lee, “A new lane following method based on deep learning for automated vehicles using surround view images,” *Journal of Ambient Intelligence and Humanized Computing*, 2019.
[176] B. Kővári, F. Hegedüs, and T. Bécsi, “Design of a reinforcement learning-based lane keeping planning agent for automated vehicles,” *Applied Sciences*, vol. 10, no. 20, 7171 pages, 2020.
[177] S. Ninan, B. Gangula, M. von Alten, and B. Sinderman, “Who owns the road? The IoT-connected car of today—and tomorrow,” 2015, http://dupress.com/articles/internet-of-things-iot-in-automotive-industry.
[178] A. E. Sallab, M. Abdou, E. Perot, and S. Yogamani, “Deep reinforcement learning framework for autonomous driving,” *Electronic Imaging*, vol. 29, no. 19, pp. 70–76, 2017.
[179] S. Sharma, S. Gupta, D. Gupta et al., “Deep learning model for the automatic classification of white blood cells,” *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 7384131, 13 pages, 2022.
[180] S. Bhatia, S. Alam, M. Shuaib, M. Hameed Alhameed, F. Jeribi, and R. I. Alsuwailem, “Retinal vessel extraction via assisted multi-channel feature map and U-net,” *Frontiers in Public Health*, vol. 355.
[181] T. Palanisamy, D. Alghazzawi, S. Bhatia, A. Abbas Malibari, P. Dadheech, and S. Sengan, “Improved energy based multi-sensor object detection in wireless sensor networks,” *Intelligent Automation & Soft Computing*, vol. 33, no. 1, pp. 227–244, 2022.
[182] S. Juneja, G. Dhiman, S. Kautish, W. Viriyasitavat, and K. Yadav, “A perspective roadmap for IoMT-based early detection and care of the neural disorder, dementia,” *Journal of Healthcare Engineering*, vol. 2021, Article ID 6712424, 11 pages, 2021.
[183] A. Balakrishnan, R. Kadiyala, G. Dhiman et al., “A personalized eccentric cyber-physical system Architecture for smart healthcare,” *Security and Communication Networks*, vol. 2021, Article ID 1747077, 36 pages, 2021.
[184] S. Atakishiyev, M. Salameh, H. Yao, and R. Goebel, *Explainable Artificial Intelligence for Autonomous Driving: A Comprehensive Overview and Field Guide for Future Research Directions*, https://arxiv.org/abs/2112.11561, 2021.
[185] S. Bhatia, S. Alam, M. Shuaib, M. Hameed Alhameed, F. Jeribi, and R. I. Alsuwailem, “Retinal vessel extraction via assisted multi-channel feature map and U-net,” *Frontiers in Public Health*, vol. 10, 2022.
[186] P. Kaur, S. Harnal, R. Tiwari et al., “Recognition of leaf disease using hybrid convolutional neural network by applying feature reduction,” *Sensors*, vol. 22, no. 2, 575 pages, 2022.
[187] S. Alam, S. T. Siddiqui, A. Ahmad, R. Ahmad, and M. Shuaib, “Internet of things (IoT) enabling technologies, requirements, and security challenges,” in *Proceedings of the Advances in Data and Information Sciences*, pp. 119–126, Springer, Singapore, January 2020.
[188] A. Samad, S. Alam, S. Mohammed, and M. U. Bhukhari, “Internet of vehicles (IoV) requirements, attacks and countermeasures,” in *Proceedings of the 12th INDIACom; INDIACom-2018; 5th International Conference on “Computing for Sustainable Global Development” IEEE Conference*, New Delhi, India, March 2018.
[189] M. Shuaib, N. H. Hassan, S. Usman et al., “Self-sovereign identity solution for blockchain-based land registry system: a comparison,” *Mobile Information Systems*, vol. 202217 pages, Article ID 8930472, 2022.
[190] S. Singh, A. Malik, R. Kumar, and P. K. Singh, “A proficient data gathering technique for unmanned aerial vehicle-enabled heterogeneous wireless sensor networks,” *International Journal of Communication Systems*, vol. 34, no. 16, 2021.
[191] A. Sharma and P. K. Singh, “UAV-based framework for effective data analysis of forest fire detection using 5G networks: an effective approach towards smart cities solutions,” *International Journal of Communication Systems*, 2021.
[192] S. Jain, N. J. Ahuja, P. Srikanth et al., “Blockchain and autonomous vehicles: recent advances and future directions,” *IEEE Access Blockchain and Autonomous Vehicles: Recent Advances and Future Directions*, 2021.
[193] A. Kumar, D. Augusto de Jesus Pacheco, K. Kaushik, and J. J. P. C. Rodrigues, “Futuristic view of the internet of quantum drones: review, challenges and research agenda,” *Vehicular Communications*, vol. 36, Article ID 100487, 2022.

RESEARCH ARTICLE

PGWO-AVS-RDA: An intelligent optimization and clustering based load balancing model in cloud

Raghavender Reddy Kothi Laxman, Amit Lathigara, Rajanikanth Aluvalu, Uma Maheswari Viswanadhula

First published: 27 June 2022 | https://doi.org/10.1002/cpe.7136

Read the full text > PDF TOOLS SHARE

Summary

Load balancing and task scheduling in cloud have gained a significant attention by many researchers, due to the increased demand of computing resources and services. For this purpose, there are various load balancing methodologies are developed in the existing works, which are mainly focusing on allocating the tasks to Virtual Machines (VMs) based on their priority, order of tasks, and execution time. Still, it facing the major difficulties in finding the best tasks for allocation, because the sequence of patterns are normally used to categorize the relevant tasks with respect to the load. Thus, this research work intends to develop an intelligent group of mechanisms for efficiently allocating the tasks to the VMs by finding the best tasks with respect to the scheduling parameters. Initially, the user tasks are given to the load balancer unit, where the Probabilistic Gray Wolf Optimization (PGWO) technique is used to find the best fitness value for selecting the tasks. Then, the Adaptive Vector Searching (AVS) methodology is utilized to cluster the

Related Information

Recommended

Task scheduling using artificial bee foraging optimization for load balancing in cloud data centers

Geetha Muthsamy, Suganthe Ravi Chandran

Computer Applications in Engineering Education

Service-level agreement–aware scheduling and load balancing of tasks in cloud

Kaippilly Raman Remesh Babu, Philip Samuel

Software: Practice and Experience

Load balancing in cloud computing

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 5422428, 18 pages
https://doi.org/10.1155/2022/5422428

Research Article

Hybrid Optimized GRU-ECNN Models for Gait Recognition with Wearable IOT Devices

K. M. Monica,[1] R. Parvathi,[1] A. Gayathri,[2] Rajanikanth Aluvalu,[3] K. Sangeetha,[4] and Chennareddy Vijay Simha Reddy[5]

[1]*School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, Tamilnadu, India*
[2]*Department of Computer Science and Engineering, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Chennai, India*
[3]*Chaitanya Bharathi Institute of Technology, Hyderabad, Telangana, India*
[4]*Department of Computer Science and Engineering, Kebri Dehar University, Kebri Dehar, Ethiopia*
[5]*Department of Computer Science, Middlesex University, London, UK*

Correspondence should be addressed to K. Sangeetha; sangeethak@kdu.edu.et

Received 22 February 2022; Revised 12 April 2022; Accepted 20 April 2022; Published 13 May 2022

Academic Editor: Shakeel Ahmad

With the advent of the Internet of Things (IoT), human-assistive technologies in healthcare services have reached the peak of their application in terms of diagnosis and treatment process. These devices must be aware of human movements to provide better aid in clinical applications as well as the user's daily activities. In this context, real-time gait analysis remains to be key catalyst for developing intelligent assistive devices. In addition to machine and deep learning algorithms, gait recognition systems have significantly improved in terms of high accuracy recognition. However, most of the existing models are focused on improving gait recognition while ignoring the computational overhead that affects the accuracy of detection and even remains unsuitable for real-time implementation. In this research paper, we proposed a hybrid gated recurrent unit (GRU) based on BAT-inspired extreme convolutional networks (BAT-ECN) for the effective recognition of human activities using gait data. The gait data are collected by implanting the wearable Internet of Things (WIoT) devices invasively. Then, a novel GRU and ECN networks are employed to extract the spatio-temporal features which are then used for classification to realize gait recognition. Extensive and comprehensive experimentations have been carried out to evaluate the proposed model using real-time datasets and also other benchmarks such as whuGait and OU-ISIR datasets. To prove the excellence of the proposed learning model, we have compared the model's performance with the other existing hybrid models. Results demonstrate that the proposed model has outperformed the other learning models in terms of high gait classification and less computational overhead.

1. Introduction

In recent years, activity recognition (AR) has witnessed exponential growth in in different domains such as healthcare [1], home automation [2], and even criminal activity detection. These methods are adopted aiming both at improving the quality of living and allowing people to stay without any support from others [3]. In the health care system, these AR systems are burgeoning technology mainly designed to detect the patient's mobility in rehabilitation therapy and to monitor physical performance after undergoing treatment with great expectations of improving his/her living quality as much as possible.

However, activity data remain more complex, which paid the way for the open research to design the intelligent human activity recognition system. Initially, simple binary sensors are used to design the recognition system [4, 5]. More recently, the Internet of Things (IoT) has been used to collect and analyze human activities and gestures [6, 7]. These devices are used as wearable devices that can be continuously used indoors or outdoors while ensuring the privacy and security of the data.

Owing to their pervasiveness and embedded sensor diversity, wearable IoT devices have been commonly used to develop AR recognition systems [8–10]. In this development, wearable IoT devices have the capability to capture and process activities and behaviors that are termed as gait signals. Accelerometers and gyroscopes are considered to be the most frequently used sensors equipped in WIOT devices to capture and transmit the gait sequences that can be used for further monitoring. Therefore, these devices have allowed for the extraction of diverse gait information from the person's movement that can be used to recognize physical activities related to health care applications.

Hence, the WIOT devices are considered as most important data capturing unit in AR systems. The collected data are then used to build the effective recognition systems. Magnificent development in AR systems is done by using the conventional machine learning algorithms such as Decision Trees [11–13], the Hidden Markov models [14–16], and support vector machines (SVMs) [17–19] have been deployed to achieve the higher rate of recognition. Since these methods are trapped in lower-dimensional data space, handling the larger data require the more efficient learning models to achieve higher performance.

Recently, studies are migrating towards deep learning algorithms to handle the larger amounts of data in an effective manner. Deep learning algorithms such as convolutional neural networks (CNNs) [20, 21] and recurrent neural networks (RNNs) [22, 23] play an undisputed role for developing AR systems. Additionally, the hybrid deep learning methods [24–26] are also gaining the brighter light of research in designing AR systems, but these collected gait data need transformation to influence the deep learning algorithms to obtain better classification with reduced computational cost. Hence, the hybrid combination of algorithms is required mandatorily to perform the data transformation and achieve high performance with low complexity.

In this context, this paper proposes a new hybrid algorithm, which ensembles the CNN layers with gated recurrent units and BAT-inspired classification networks. The user-defined CNN is used to extract the spatial features, whereas GRU is used to extract the temporal features. These features are then fed to complexity-aware BAT-inspired classification networks to achieve a better classification of AR with low complexity overhead.

1.1. Contribution

(1) This paper focuses on the development of novel testbeds based on wearable IoT devices for the effective collection of raw gait data.

(2) This paper also proposes a methodology for restructuring the raw data suitable to train the deep learning algorithms for better performance.

(3) This paper proposes a hybrid deep learning algorithm for effective feature extraction with less computational cost and a high gait recognition rate.

(4) Finally, the paper presents the excellence of the proposed methodology by conducting experiments using other benchmark datasets and comparing the performance with other existing deep learning-based AR systems.

The rest of the paper is organized as follows: Section 2 presents the related works proposed by more than one authors. The data collection unit, data preprocessing, and the proposed hybrid model are presented in Section 3. The dataset descriptions, experimentations, results, findings, and analysis are presented in Section 4. Finally, the paper is concluded in Section 5 with future enhancements.

2. Related Works

Abdullah et al. adopted a neural network for diagnosing the human abnormalities using their walking styles, which are detected at lower limbs. These real-time samples are extracted through the Levenberg-Marquardt method, and their artifacts are removed using the Butterworth filters in order to train the neural network effectively. The gait data are observed from 5-subjects at distinct speeds 2.4, 3.2, and 5.4 kmph and in total 45 instances are utilized for evaluation [27]. Though the proposed NN achieved better accuracy for tested data, it is not suitable for dynamic movements, and the tested data range are very low.

On the HuGaDB dataset, Saleh et al. used the three supervised machine learning models for human activity recognition: random forest, Navie Bayes, and IB1 classifiers. This HuGaDB contains data on standing, sitting, running, and walking, which is monitored using accelerometers and gyroscopes. Random forest outperformed the other two learning models in terms of classification accuracy while requiring less setup time [28]. Moon et al. introduced a multimodel gait identification classifier based on the convolutional and recurrent neural networks combined with a support vector feature extractor [29].

Jiang and Yin used the Short-time Discrete Fourier Transform (STDFT) to create a time-frequency-spectral image from time-serial signals in [30]. After that, CNN is used to process the image in order to recognize basic daily movements such as walking and standing. Using a mix of time-frequency-spectral characteristics and CNNs, Laput and Harrison [31] built a fine-grained hand activity sensing system. They were able to classify 25 atomic hand activities performed by 12 participants with a 95.2 percent accuracy. The spectral properties can be employed not only for wearable sensor activity recognition but also for activity recognition without the usage of a device. For learning modality-specific temporal properties, Ha and Choi [32] proposed a new CNN structure with distinct 1D CNNs for different modalities. Other types of CNN variants are being studied as part of the development of CNNs for efficiently integrating temporal characteristics.

Shen et al. [33] used the gated CNN to recognize everyday activities from audio signals and found it to be more accurate than the naïve CNN. Long et al. used residual blocks to create a two-stream CNN structure that can handle several time scales. On three benchmark datasets, Guo et al. [34] developed an ensemble technique of numerous deep LSTM networks that outperformed individual networks. Aside from RNN structure variations, other researchers looked into distinct RNN cells. For example, instead of using LSTM cells, Yao et al. [35] built an RNN using gated recurrent units (GRUs) and used it to activity recognition. However, some research has found that different types of RNN cells do not perform significantly better than the traditional LSTM cell in terms of classification accuracy [36]. Wang et al. [37] used the CNN and an LSTM to create a classifier that could automatically extract difficult characteristics from sound data and recognize gestures. For different scales of local temporal feature extraction, Xu et al. [38] used the sophisticated Inception CNN structure, whereas GRUs were used for efficient global temporal representations.

To assess more complex temporal hierarchies, Yuta et al. [39] used a dual-stream ConvLSTM network, with one stream handling shorter time lengths and the other longer time lengths. Guo et al. [40] proposed that MLPs be used to generate a base classifier for each sensory modality, and that ensemble weights be assigned at the classifier level to incorporate all classifiers. The authors not only evaluated recognition accuracy while creating the basis classifiers but also stressed variety by inducing diversity metrics. As a result, the diversity of different modalities is retained, which is important for overcoming over-fitting difficulties and enhancing overall generalization capacity.

3. Proposed System

3.1. System Overview. The proposed framework has four main phases, namely: (i) Data collection unit; (ii) Data preprocessing and filtering; (iii) Spatial and Temporal feature extraction using the proposed architecture; (iv) Classification phase. The block diagram of the proposed framework is shown in Figure 1.

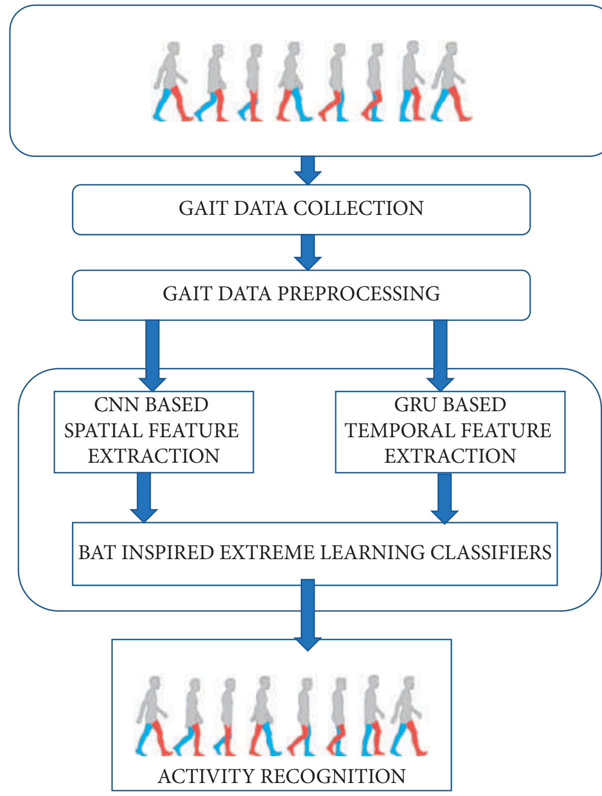


Figure 1: Block diagram for the proposed methodology.

3.2. Materials and Methods

3.2.1. Data Collection Unit. To collect the experimental data, 29 volunteers with body weights ranging from 25 kg to 64 kg were selected. The participants were all healthy without any neurological disorders and had no physical injuries to their legs or feet, which may have affect the walking gait phase detection. With the advancement of Internet of Things (IoT) devices, this work used six battery-powered IoT devices to collect the corresponding inertial information. Figure shows the placement of the six IoT devices on the participants. To collect the inertial data from the lower limbs, MICOTT boards are used as the main IoT devices, which consist of 8-BIT NODEMCU as the main CPU interfaced with the 10-BIT SPI (Serial Peripheral Interface) based MCP3008 analog channels and ESP8266 WIFI transceivers. ADXL435 three-axis accelerometers and BMG250 three-axis gyroscopes are interfaced with MICOTT boards to collect inertial information from both limbs of participants. Micropython programming was deployed in the board to collect data and transmit them to the cloud. The series of Li-On batteries with operating voltage of 3.3 V is used to power up the board and can be replaced as the batter drains its total power.

During the experimentation, all participants were required to walk normally on the treadmill at different speed ranging from 0.66 m/s to 1.3 m/s for at least 180 s. All the participants were requested to walk normally for 2 minutes at each speed. The experimental data were collected for every 3 minutes and the data collected were transmitted to the cloud for further processing. Besides, to evaluate the excellence of the proposed algorithm, we have used other public benchmark datasets such as the whuGait and OU-ISIR datasets, and details of datasets are discussed in Section 4. Figure 2 presents the data collection procedure used in the proposed methodology.

3.2.2. Data Preprocessing Process. The stored data sample in the cloud contain multiple features from the six IoT devices, and each data includes acceleration and angular velocity data in the X, Y and Z directions. The sequences of the data sample are denoted by the following equation:

$$y = \{s1, f2, s2, f1, s3, f3\}, \quad (1)$$

where y is the total data sample, $s1$, $s2$, and $s3$ are accelerometer data, and $f1$, $f2$, and $f3$ are angular velocity data, which are stored in cloud. As mentioned in the above equation, combined data are stored in the cloud, which need the segmentation and extraction that can be used for the

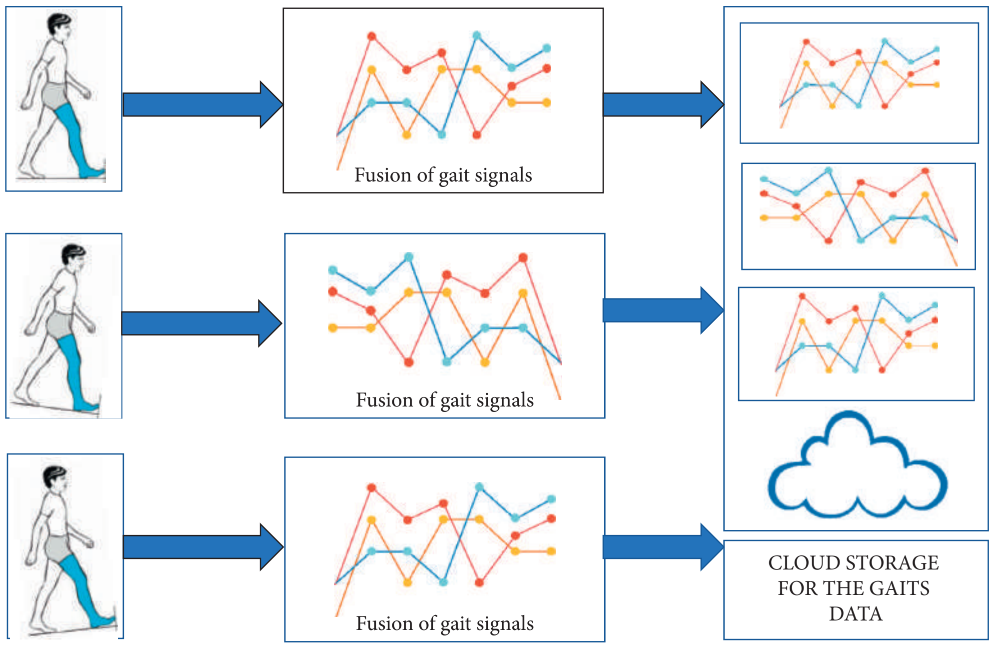


Figure 2: Data collection system used in the proposed methodology.

better classification. The data collected in the cloud are downloaded offline and data preprocessing steps are used for effective data separation and extraction. To achieve less computational complexity with high accuracy of segmentation, this paper uses the novel Pearson correlation sliding window technique [41], which combines the Pearson correlation coefficient [42] and Sliding Window techniques. The value of P plays an important role in the data extraction, in which different thresholds are used for effective data extraction over a period of time. Figure 3 presents the preprocessed data after applying the proposed technique.

3.2.3. Proposed the Hybrid Deep Learning Model. As the analysis of the walking ability of the individual models with the fused features, we find that the integration of the different learning models can lead to better gait signal recognition and classification with less complexity. Hence, we intend to design the hybrid ensemble of the deep and machine learning models to learn the combined spatio-temporal feature effectively, which tends to the way of high accuracy and less computational complexity. The complete architecture of the proposed hybrid model is shown in Figure 4.

3.2.4. CNN-Based Spatial Feature Extraction. This paper uses the CNN layers are core spatial feature extractors, which can act as the input to the dense learning layers, which are based on the optimized extreme learning machines. First, we briefly explain the concept of CNN architectures, which act as the main spatial feature extractor. The convolutional neural network (CNN) is a biologically propelled advancement of the multilayer perceptron (MLP).

As shown in Figure 5, CNN by connecting various convolution layers and max-pooling tasks. Information is handled through these profound layers to deliver the element maps, which are at last changed into an element vector by going through an MLP. This is alluded to as a fully-connected layer (FC) that performs classification and detection. For an effective spatial feature extractor, this paper uses six-convolutional layers in which the preprocessed collected data are given as the inputs. The CNN layers used in this paper are presented in Table 1.

The ReLU function is used as activation function in the network. To reduce the risk of the gradient vanishing problem, we used the batch-normalization process right after the fourth and fifth convolutional layers. The convolutional feature maps for the input x are denoted by using the following equation:

$$F = f\left(x, \{W1, B1, \beta(\text{Relu})\}\right), \quad (2)$$

where $W1$ is weight matrix of the layers, $b1$ is networks' bias weights, and $\beta(\text{Relu})$ is ReLU activation function. We train the network by initializing the weights randomly with a learning rate of 0.01 and momentum of 0.9.

3.2.5. GRU-Based Temporal Feature Extraction. The most important structure used for the temporal feature extraction is the GRU module, which receives the data collected from the IoT-cloud systems. Figure 6 shows the structure of the GRU network used in the paper.

The GRU network consists of two gates and is considered faster than the LSTM and RNN models [43]. Where x_t is the input feature at the current state, y_t is the output state, h_t is the output of the module at the current instant, Z_t and r_t are update and reset gates, $W(t)$ is weights, and $B(t)$ is bias weights at current instant. The mathematical expression for extracting the feature maps is given in the following equation:

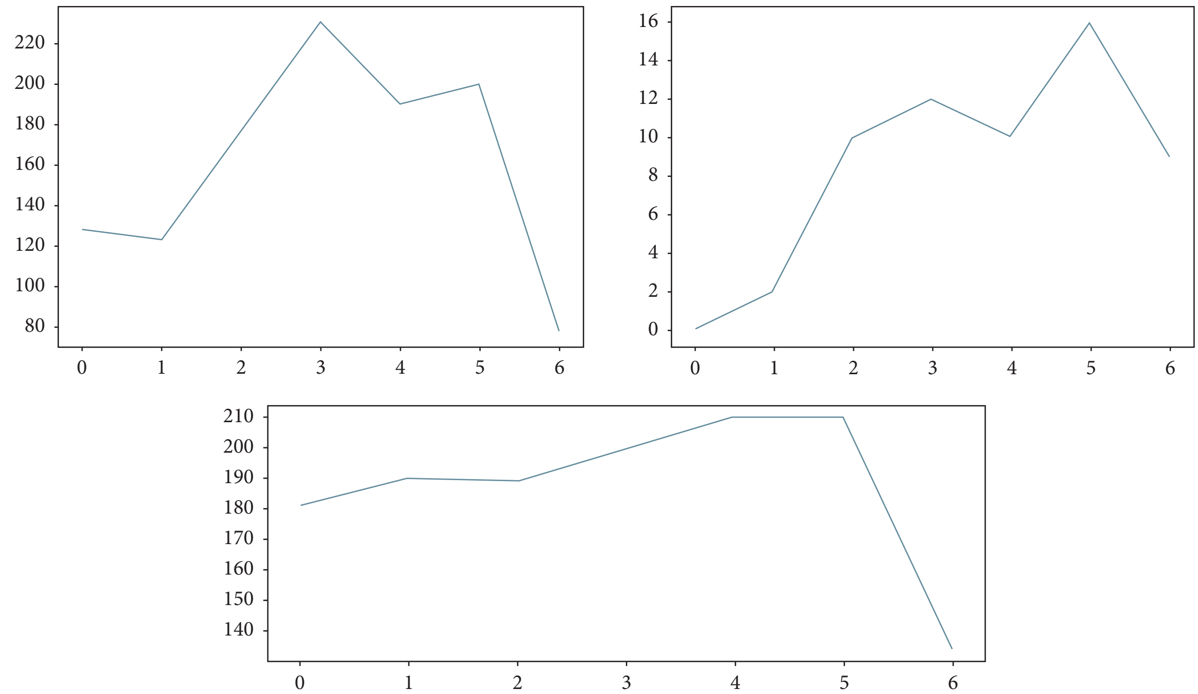


FIGURE 3: Data preprocessing after applying the proposed technique of data separation.

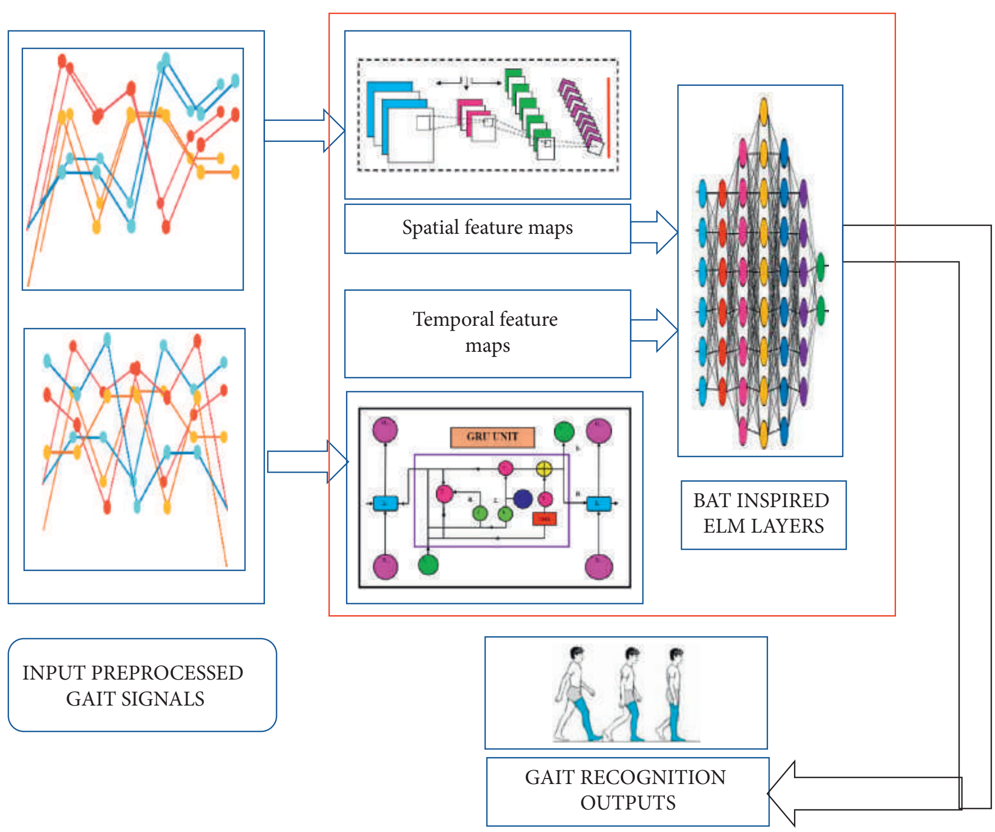


FIGURE 4: Proposed Architecture for the Hybrid Feature Extraction and Classification layer.

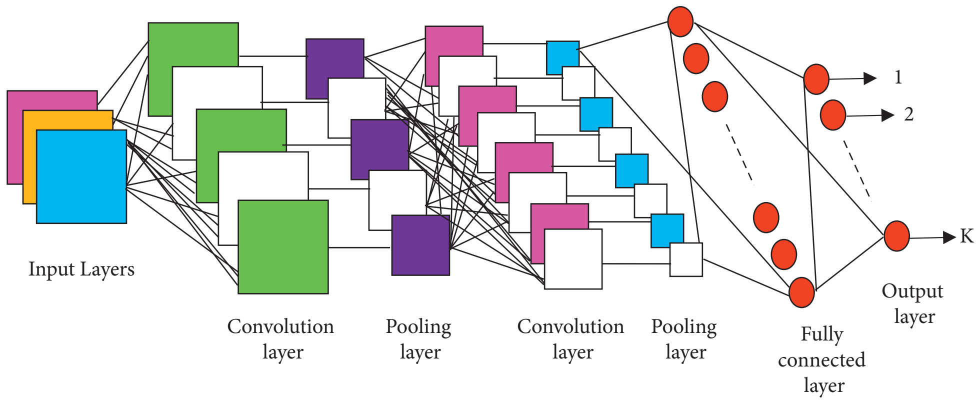


FIGURE 5: Schematic representation of convolutional neural networks.

TABLE 1: Parameters of CNN spatial feature extraction.

Sl. No	No of convolutional layers	Stride length	No of layers
1	Conv(2d) -Layer-1	2	3 × 3
2	Max-pooling layers-1		2 × 2
3	Conv(2d) -Layer-2	2	3 × 3
4	Max-pooling layers-2		2 × 2
5	Conv(2d) -Layer-3	2	2 × 2
6	Max-pooling layers-3		1 × 1
7	Conv(2d) -Layer-4	2	2 × 2
8	Max-pooling layers-4		1 × 1
9	Conv(2d) -Layer-5	2	2 × 2
10	Max-pooling layers-5		1 × 1
11	Conv(2d) -Layer-6	2	1 × 1
12	Max-pooling layers-6		1 × 1

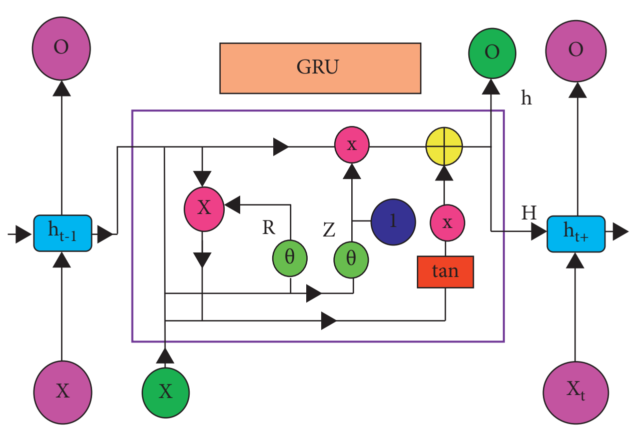


FIGURE 6: GRU network for Temporal Feature Extractor.

$$P = \mathrm{GRU}\left(\sum_{t=1}^{n}\left[x_t, h_t, z_t, r_t\,(W(t), B(t), \eta(\tan nh))\right]\right). \quad (3)$$

3.2.6. Classification Layers. Next, we further propose an optimized single feed forward network, which uses the principle of extreme learning machine to train the spatio-temporal features obtained from the previous layers. In order to have less computational complexity, this research work uses the extreme learning network with auto-tuning property whose optimization is done by the BAT-inspired principles. The detailed description of the proposed classification layer is given as follows:

(1) ELM Decision and Classification layer: ELM is a kind of neural network that utilizes single hidden layers and works on the principle of auto-tuning property. ELM exhibits better performance, high speed, and less computational overhead when compared with the other learning models such as support vector machines (SVM), bayesian classifier (BC), K-nearest neighborhood (KNN), and even Random Forest (RF).

This kind of neural network utilizes the single hidden layers, in which the hidden layers do not require the tuning mandatorily. Compared with the other learning algorithms such as support vector machines (SVM) and Random Forest (RF), ELM exhibits better performance, high speed, and less computational overhead. ELM uses the kernel function to yield good accuracy for better performance. The major advantages of the ELM are minimal training error and better approximation. Since ELM uses auto-tuning of the weight biases and nonzero activation functions. The detailed working mechanism of the ELM is discussed in [44]. The input features maps of the ELM are denoted by the following equation:

$$X = F(F, P), \quad (4)$$

where X is the fused spatio-temporal features obtained from the CNN and GRU layers, F is the CNN's spatial feature and P is the GRU temporal feature.

The output ELM function is denoted by the following equation:

$$Y(n) = X(n)\beta = X(n)X^T\left(\frac{1}{C}XX^T\right)^{-1}O. \quad (5)$$

The overall training of ELM is given by the following equation:

$$S = \alpha\left(\sum_{n=1}^{N}(Y(n), B(n), W(n))\right), \quad (6)$$

where $X(n)$ is input fused feature maps, β is temporal matrix, which is solved by the Moore–Penrose generalized inverse theorem, denoted by X^T, C is constant, and B and W are weights and bias factors of the network with the sigmoidal activation function. The proposed network is trained with these features using the sigmoidal activation function. To resolve the computational problems, this paper adds the BAT-inspired optimizers to tune the hyper-parameters of the proposed ELM classifiers. The working mechanism of the BAT-inspired ELM is discussed as follows.

(2) BAT Inspired ELM Layers: this section describes the working mechanism of the BAT algorithm over ELM layers to provide better classification.

(3) Bat Algorithm- an Overview: the standard mega- bat calculation depended on the echolocation or bio-sonar attributes of microbats. In light of the echo cancelation calculations, Yang [45] (2010) built up the bat calculation with the accompanying three glorified guidelines:

(1) All bats use echolocation to detect separation, and they likewise "know" the distinction between sustenance/prey and foundation obstructions in some mystical manner.
(2) Bats look for prey by flying at a random velocity v_i at position x_i with a fixed frequency fmin, changing wavelength λ, and loudness A0. They can consequently modify the wavelength (or recurrence) of their transmitted pulse and alter the rate of pulse emission r 2 [0, 1], based on the nearness of their objective.
(3) In spite of the fact that the loudness can fluctuate from numerous points of view, we expect that the loudness shifts from an extensive (positive) $A0$ to a minimum constant value Amin.

Each bat Motion is associated with the velocity v_{it} and initial distance x_{it} with the "n" number of iterations in a dimensional space or search space. Among all the bats, the best bat has to be chosen depends on the three rules, which are stated above. The updated velocity vit and initial distance x_{it} using the three rules are given below in the following equation:

$$f_i = f\min \pm (f\max - f\min)\beta, \\ x_{it} = x_{it} - 1 + v_{it}, \tag{7}$$

where $\beta \in (0,1)$ fmin is the minimum frequency = 0 and $f$max is the maximum frequency, which initially depends on the problem statement. Each bat is initially allocated for the frequency between the $f$min and $f$max. Consequently, bat calculations can be considered as a frequency tuning calculation to give a reasonable blend of investigation and exploitation. The emission rates and loudness basically give mechanism to programmed control and auto-zooming into the district with promising solutions.

To get a better solution, it is fundamental for the variety of the loudness and the pulse emission. Since the loudness normally diminishes once a bat has discovered its prey, while the rate of pulse emission expands, the loudness can be picked as any estimation of accommodation, between Amin and Amax, accepting Amin = 0 implies that a bat has quite recently discovered the prey and briefly quit transmitting any stable.

3.2.7. Advantages of Bat Algorithms. The major advantages of BAT algorithms are as follows:

(1) High Efficiency than PSO, GA, and other heuristic algorithms [46]
(2) Faster and more versatile search space than SGD [47]

Motivated by the advantages of the BAT algorithm, we proposed the new hybrid integration of the BAT algorithm and the ELM training network for better gait classification.

3.2.8. BAT-Inspired ELM Layers. As discussed in Section 3.2.4, the simple bat algorithms are used to optimize the weights of ELM networks. In this case, bat's prey searching mechanism is used as the main term to optimize the weights and hidden layers of ELM. Initially, these hyper parameters are selected randomly and passed to the ELM training network. The fitness function of the proposed network is given by equation (9). For each iteration, hyper parameters are calculated by using equations (7) and (8). The iteration stops when the fitness function matches equation (9).

$$\text{Fitness Function} = \text{Max}\,(\text{Accuracy}, \text{Precison}, \text{REcall}, \text{Specificty\&F1} - \text{Score}). \tag{8}$$

Once the inputs weights are optimized by the BAT algorithm, the proposed classification layer classifies the gait activities with high speed and less computation. The working mechanism of the proposed classification layers is presented in Algorithm 1. The training network uses 30 epochs, batch size of 40 with 150 hidden layers and 0.001 learning rate.

4. Section -IV

4.1. Experimentation and Evaluation Metrics. Table 2 presents the experimental parameters used for training the proposed network. Furthermore, we have calculated the performance metrics such as accuracy, precision, recall, specificity, and F1-score using different datasets. Additionally, we have calculated the AUC (Area under ROC) and confusion matrix to prove the superiority of the proposed model. The mathematical expression used for calculating the performance metrics is presented in Table 3. Higher scores of the metrics indicate better performances. To solve the network's overfitting problem and improve the generalization problem, the early stopping method [48] is used in the paper. This method can be used to end the proposed network training when the validation performance shows no improvement for N consecutive times. The

```
(1)  Input = Bias weights, Hidden layers, Epochs, Learning Rate
(2)  Output: Gaits/Human Activity Recognition
(3)  Randomly assign the bias weights, hidden layers, epochs, learning rate
(4)  Initialize the Loudness, Frequency, Distance, No of bats and Velocity
(5)  While (true)
(6)    Calculate the ELM 's output using (5)
(7)    Calculate the Fitness function using equation (9)
(8)    For t = 1 to Max_iteration
(9)      Assign the bias weights and input layers by (6) and (7)
(10)     Calculate the fitness function using equation (9)
(11)     If (Fitness function = = Maximum Accuracy)
(12)       Go to Step 17
(13)     Else
(14)       Go to Step 8
(15)     End
(16)   End
(17)   If (output value ≤ 1)
(18)     / Normal Activity is determined
(19)   Else if (output value ≤ 2&& output value > 1)
(20)     / Activity-2 is determined
(21)   Else if (output value ≤ 3&& output value > 2)
(22)     / Activity-3 is determined
(23)   Else
(26)     Go to step 8
(27)   End
(28) End
(29) End
```

Algorithm 1: Pseudo Code for the Proposed Optimized ELM layers.

Table 2: Training Parameters used for the Proposed Hybrid Model.

Sl. no	Detailed parameters	Specifications
01	No of Epochs	200
02	Batch Size	100
03	Learning Rate	0.0001
04	Training data	70
05	Testing data	30

complete model was developed using open source TensorFlow version 2.1.0 with Keras as backend and implemented on a PC workstation with Intel Xeon CPU, NVIDIA Titan GPU, 16 GB RAM, and 3.5 GH_Z operating frequency.

4.2. Performance Evaluation of the Proposed Model Using the Different Datasets. In this part, we conducted experiments using real-time and benchmark datasets. We have calculated ROC and confusion matrix of the proposed network model using different datasets.

Figure 7 shows the ROC curves of the proposed model using different gait datasets. It is obvious that the proposed model has shown the 0.9880 AUC for raw data collected, 0.980 AUC for whuGait, and 0.9780 AUC for OU-ISIR datasets. The proposed network has shown constant performance for real-time datasets and public datasets also. Figure 8-shows the confusion matrix of the proposed model using different dataset. Figure 8 shows the confusion matrix of the proposed model under datasets. It is evident that from

Table 3: Mathematical expressions for the performance metrics' calculation.

Sl. no	Performance metrics	Mathematical expression
01	Accuracy	$(TP + TN)/(TP + TN + FP + FN)$
02	Recall	$TP/(TP + FN) \times 100$
03	Specificity	$TN/(TN + FP)$
04	Precision	$TN/(TP + FP)$
05	F1-Score	$2.((\text{Precison} * \text{Recall})/(\text{Precision} + \text{Recall}))$

TP is True Positive Values, TN is True Negative Values, FP is False Positive and FN is False negative values.

Figure 7, the proposed model maintains the uniform performance even with different datasets. The performance metrics of the proposed algorithm with the different datasets have been depicted in Table 4. From Table 4, it is found that the proposed network has exhibited higher performance using real-time datasets and whuGait datasets. It is also found that the proposed model has shown slight edge of peak performance when handling the OU-ISIR datasets.

4.3. Comparative Analysis of the Proposed Model with the Other Existing Models. To prove the superiority of the algorithm, performance of the proposed model is calculated and evaluated against the existing the hybrid deep learning algorithms such as TL-LSTM [49], 2D-CNN-LSTM [50], DCLSTM [51], Q-BTDNN [52], ATTENTION + CNN [53], CNN + GRU [54], and CNN + SVM [55].

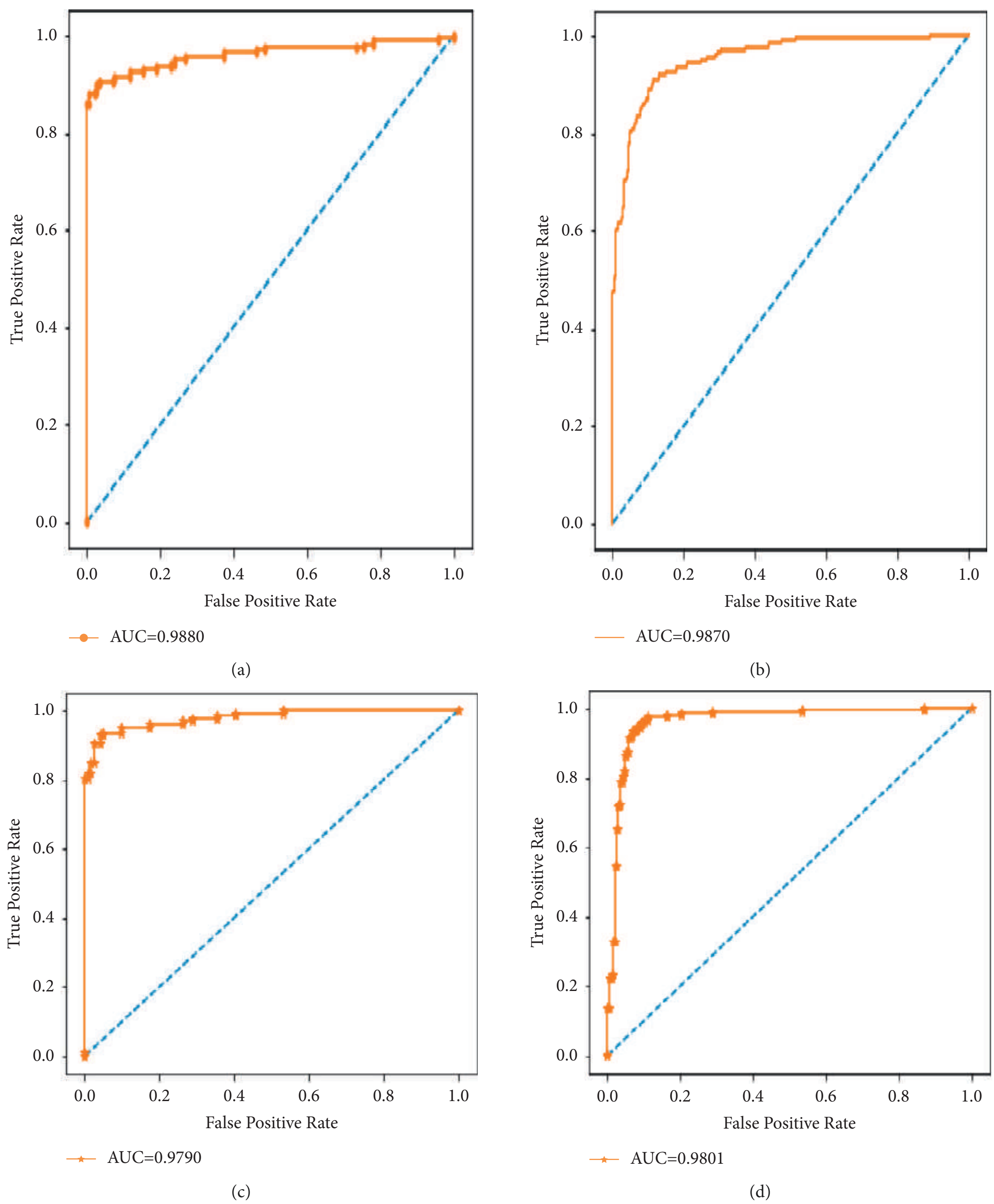
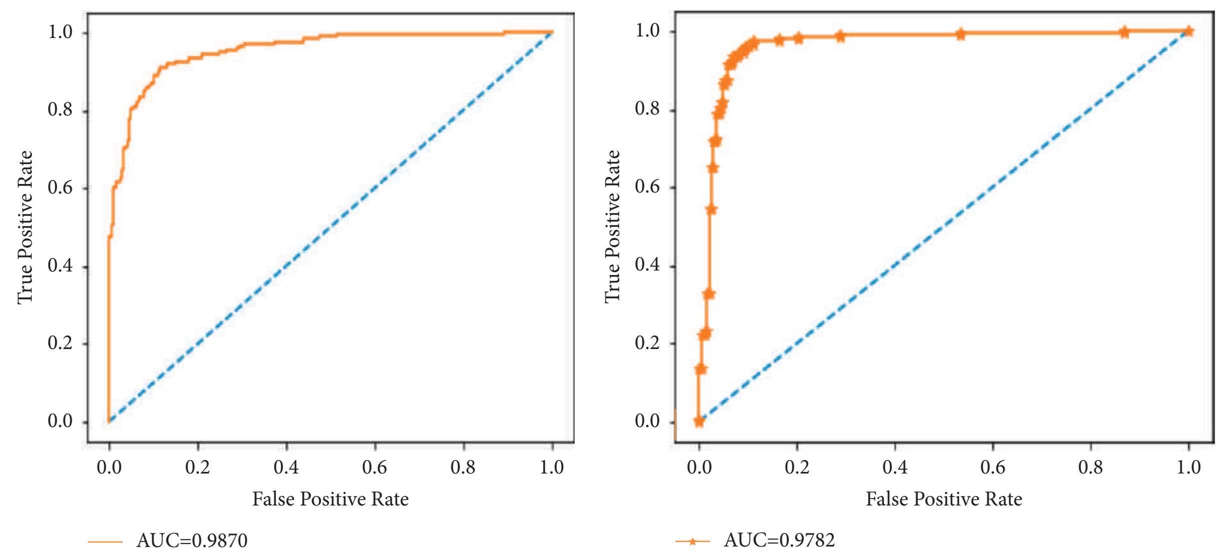


Figure 7: Continued.



(e) (f)

Figure 7: ROC curves of the proposed methodology. (a) Real time Datasets; (b) Dataset-1(whuGait); (c) Dataset-2; (d) Dataset-3; (e) Dataset-4; (f) OU-ISIR datasets.

A.L	1	2	3	4	5	6	7	8	9	10
1	98.8%	0	0	1	0.5	0.4	0	0	0	0
2	0	98.90%	0	0	0.4	0.5	0	0	0	0
3	0	0	98.89%	1	0.5	0.4	0	0	0	0
4	0	0	0	98.89%	0.5	0.4	0	0	0	0
5	0	0	0	0	98.90%	0.4	0	0	0	0
6	0	0	0	0	0.5	98.89%	0	0	0	0
7	0	0	0	0	0.5	0.4	98.9%	0	0	0
8	0	0	0	0	0.5	0.4	0	99.0%	0	0
9	0	0	0	0	0.5	0.4	0	0	98.89%	0
10	0	0	0	0	0.5	0.4	0	0	0	98.91%

(a)

Figure 8: Continued.

A.L	1	2	3	4	5	6	7	8	9	10
1	98.8%	0	0	1	0.5	0.4	0	0	0	0
2	0	98.90%	0	0	0.4	0.5	0	0	0	0
3	0	0	98.89%	1	0.5	0.4	0	0	0	0
4	0	0	0	98.89%	0.5	0.4	0	0	0	0
5	0	0	0	0	98.90%	0.4	0	0	0	0
6	0	0	0	0	0.5	98.89%	0	0	0	0
7	0	0	0	0	0.5	0.4	98.9%	0	0	0
8	0	0	0	0	0.5	0.4	0	99.0%	0	0
9	0	0	0	0	0.5	0.4	0	0	98.89%	0
10	0	0	0	0	0.5	0.4	0	0	0	98.91%

(b)

A.L	1	2	3	4	5	6	7	8	9	10
1	98.8%	0	0	1	0.5	0.4	0	0	0	0
2	0	98.90%	0	0	0.4	0.5	0	0	0	0
3	0	0	98.89%	1	0.5	0.4	0	0	0	0
4	0	0	0	98.89%	0.5	0.4	0	0	0	0
5	0	0	0	0	98.90%	0.4	0	0	0	0
6	0	0	0	0	0.5	98.89%	0	0	0	0
7	0	0	0	0	0.5	0.4	98.9%	0	0	0
8	0	0	0	0	0.5	0.4	0	99.0%	0	0
9	0	0	0	0	0.5	0.4	0	0	98.89%	0
10	0	0	0	0	0.5	0.4	0	0	0	98.91%

(c)

Figure 8: Continued.

A.L	1	2	3	4	5	6	7	8	9	10
1	98.8%	0	0	1	0.5	0.4	0	0	0	0
2	0	98.90%	0	0	0.4	0.5	0	0	0	0
3	0	0	98.89%	1	0.5	0.4	0	0	0	0
4	0	0	0	98.89%	0.5	0.4	0	0	0	0
5	0	0	0	0	98.90%	0.4	0	0	0	0
6	0	0	0	0	0.5	98.89%	0	0	0	0
7	0	0	0	0	0.5	0.4	98.9%	0	0	0
8	0	0	0	0	0.5	0.4	0	99.0%	0	0
9	0	0	0	0	0.5	0.4	0	0	98.89%	0
10	0	0	0	0	0.5	0.4	0	0	0	98.91%

(d)

A.L	1	2	3	4	5	6	7	8	9	10
1	98.8%	0	0	1	0.5	0.4	0	0	0	0
2	0	98.90%	0	0	0.4	0.5	0	0	0	0
3	0	0	98.89%	1	0.5	0.4	0	0	0	0
4	0	0	0	98.89%	0.5	0.4	0	0	0	0
5	0	0	0	0	98.90%	0.4	0	0	0	0
6	0	0	0	0	0.5	98.89%	0	0	0	0
7	0	0	0	0	0.5	0.4	98.9%	0	0	0
8	0	0	0	0	0.5	0.4	0	99.0%	0	0
9	0	0	0	0	0.5	0.4	0	0	98.89%	0
10	0	0	0	0	0.5	0.4	0	0	0	98.91%

(e)

Figure 8: Continued.

Label	Recognition of Activity	Non -recognition of Activity
Recognition of Activity	98.9%	1.2%
Non - recognition of Activity	1.3%	98.7%

(f)

Figure 8: Confusion matrix for the Proposed Hybrid Model. (a) Real time Datasets; (b) Dataset-1(whuGait); (c) Dataset-2; (d) Dataset-3; (e) Dataset-4; (f) OU-ISIR datasets.

Table 4: Performance metrics of the proposed model using different datasets.

Datasets	Performance metrics				
	Accuracy	Precision	Recall	Specificity	F1-score
Real-time Datasets	0.989	0.987	0.986	0.989	0.9902
Datasets-1	0.9889	0.985	0.984	0.978	0.983
Dataset-2	0.9890	0.9856	0.989	0.9902	0.989
Dataset-3	0.9890	0.9879	0.990	0.9901	0.992
Dataset-4	0.9891	0.9890	0.990	0.99	0.988
OU-ISIR datasets	0.990	0.989	0.982	0.992	0.990

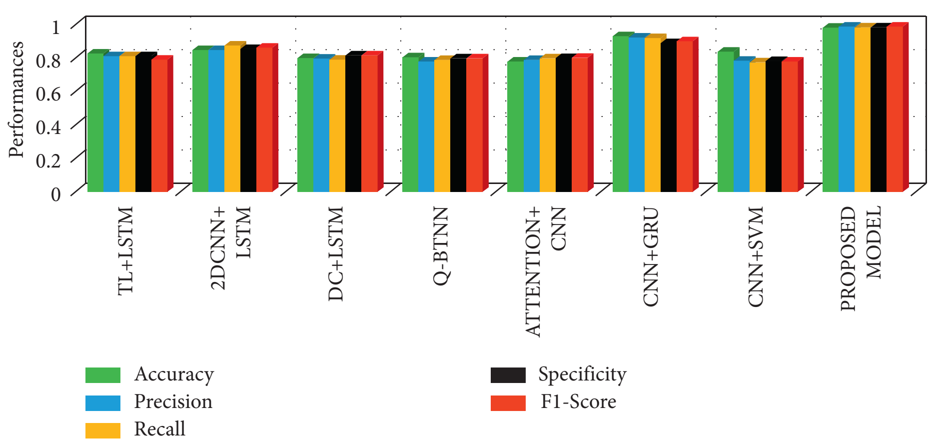


Figure 9: Performance analysis of the different hybrid models using real-time datasets.

Figures 9–14 show the performance of the different hybrid learning models using the IoT-based real-time datasets. It is found that the proposed hybrid model and CNN + GRU have exhibited the same performance in AR detection systems. But still, the proposed model has shown little edge over the CNN + GRU learning model and outperforms the other learning models in the detection of gaits. Figures show the comparative analysis of the different learning models using whuGait datasets. The proposed model has shown greater performance than the other existing learning models. The performance of the different learning models using OU-ISIR datasets are shown in Figures 9–14. From Figures 9–14, it is clear that the inclusion of the BAT-inspired ELM models along with spatio-temporal feature extraction has shown its excellence over the other learning models. From the above experiments, it is clear that the proposed model has shown the better AR rate even with multiple datasets.

4.4. Computational Complexity. The computational complexity of the proposed technique is represented by big-o-

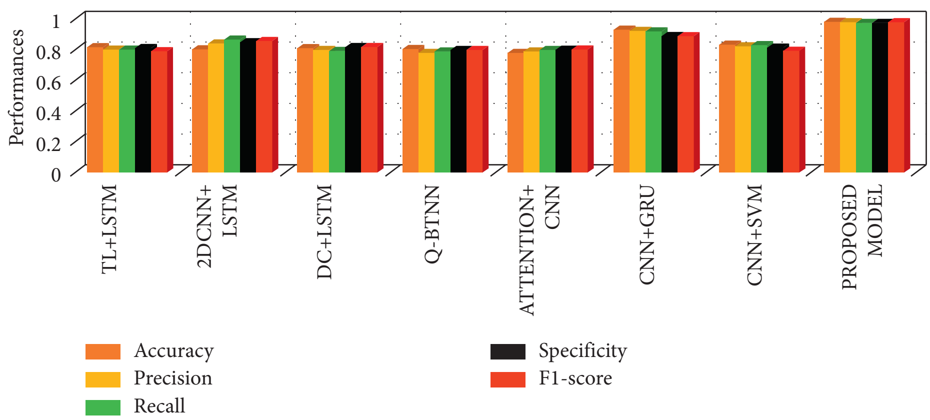


Figure 10: Performance analysis of the different hybrid models using WhuGait Dataset-1.

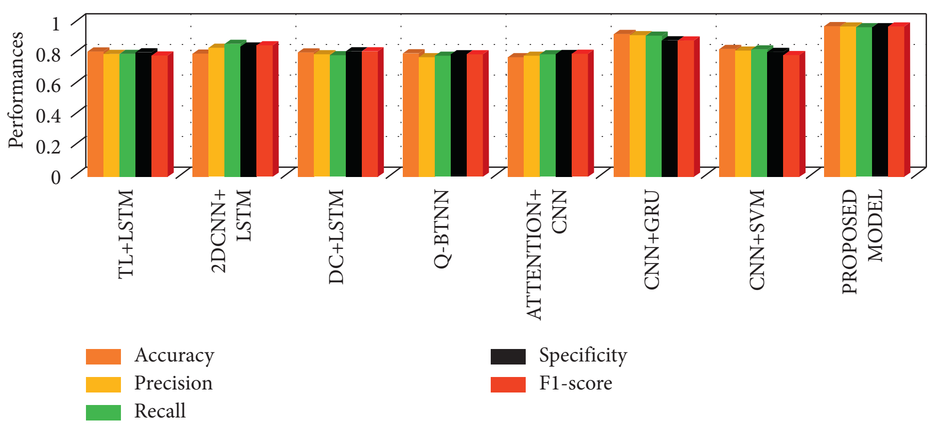


Figure 11: Performance analysis of the different hybrid models using WhuGait Dataset-2.

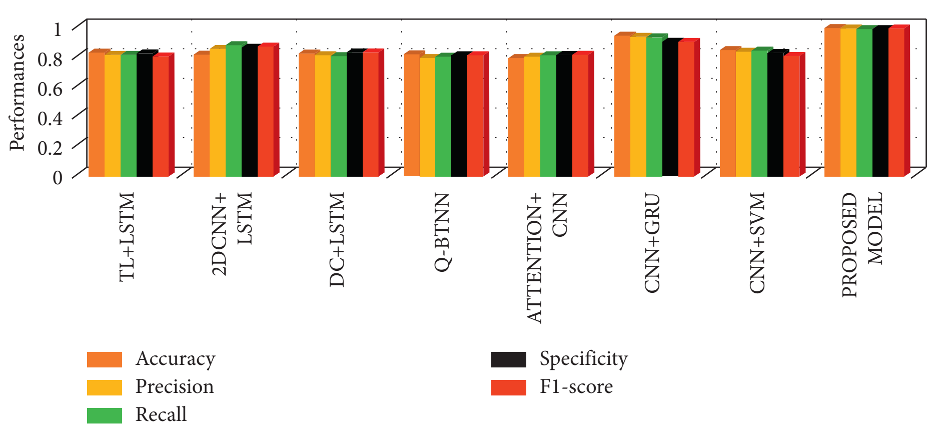


Figure 12: Performance analysis of the different hybrid models using WhuGait Dataset-3.

notations. The different CNN algorithms used for evaluation and complexity analysis are presented in Table 5. The mathematical expressions for calculating the computational complexity using Big-O-Notation are given by the following equation:

$$\text{Time Complexity} = O(\text{Convolutaional layers} * \text{Pooling Layers} * \text{Training Networks}). \quad (9)$$

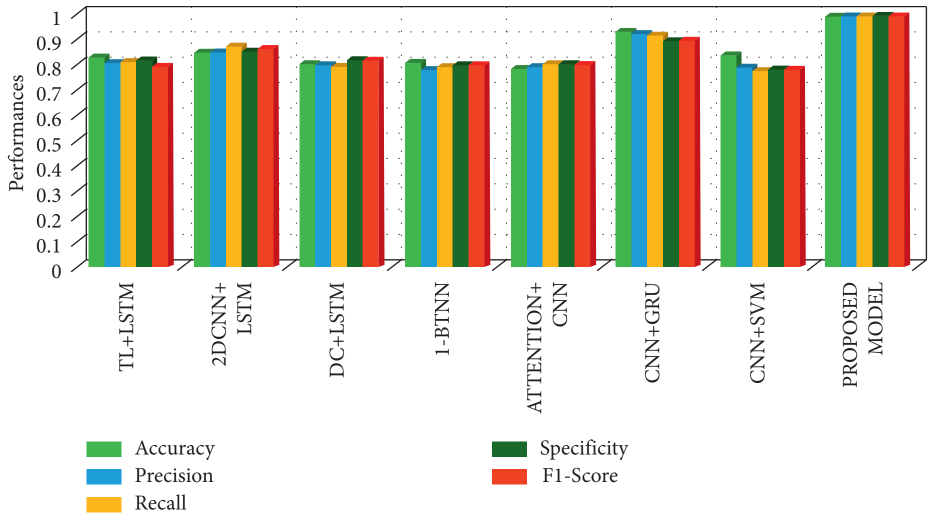


Figure 13: Performance analysis of the different hybrid models using WhuGait Dataset-4.

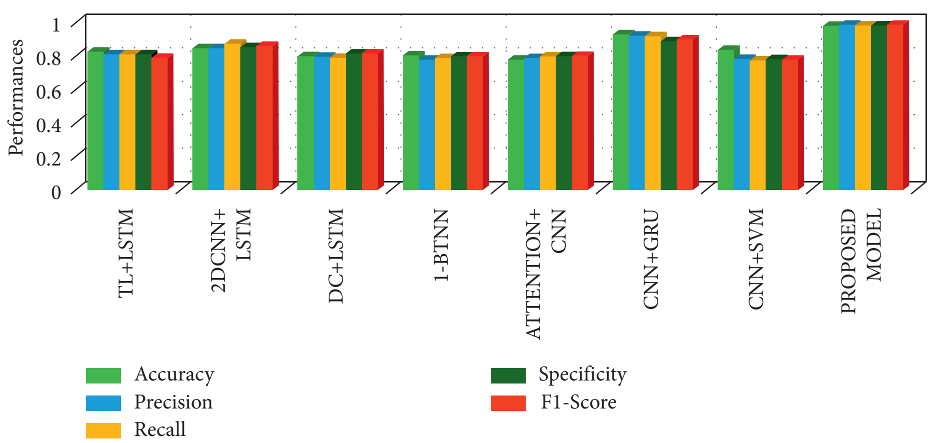


Figure 14: Performance analysis of the different hybrid models using OU-ISIR datasets.

Table 5: Computational complexity analysis between the hybrid and proposed model.

Algorithm Details	No. of layers required	Big-O-Notations
CNN (without Optimization)	No. of Convolutional layers = 6 No. of Polling layers = 06 No. of training layers = k	O (n6, n6, nk)
2DCNN-LSTM	No. of Convolutional layers = 05 No. of Polling layers = 05 No. of training layers = $k+3$	O (n5, n5, nk + 3)
CNN-SVM	No. of Convolutional layers = 06 No. of Polling layers = 06 No. of training layers = $k+5$	O (n6, n6, nk + 5)
CNN + GRU	No. of Convolutional layers = 06 No. of Polling layers = 06 No. of training layers = $k+6$	O (n6, n6, nk + 6)
Attention CNN	No. of Convolutional layers = 06 No. of Polling layers = 06 No. of training layers = $k+10$	O (n6, n6, nk + 10)
Proposed architecture	No. of Convolutional layers = 06 No. of Polling layers = 06 No. of training layers = k – 5	O (n5, n5, nk – 5)

*k = Maximum Number of required Training Layers.

From Table 5, it is found that BAT optimized classification layer has produced less computational complexity, which is even 10% lesser than the other existing algorithms.

5. Section -V

5.1. Conclusion and Future Scope. In this paper, a novel GRU fused CNN feature extractor with the BAT-inspired classification layer is formed for better recognition of human gaits that can be used for health care applications. The real-time datasets were collected using the wearable IoT(W-IoT) devices and stored in the cloud for further monitoring and processing. For an efficient classification, these data were restructured using the Pearson correlated sliding windowing method. Then, these restructured data are fed into the two layers of the deep learning model one is user-defined CNN, which is used to extract the spatial features and the other is GRU, which is used to extract the temporal features. Finally, these spatio-temporal features are then feed into the proposed BAT-inspired optimized classifiers to have better gait recognition. The extensive experimentation is carried out using the real-time datasets along with the public datasets such as whuGait and OU-ISIR benchmarks. Results demonstrated the proposed model has shown better recognition rate and less computational cost than the other existing hybrid learning models.

For future work, we would further implement the proposed gait recognition system over the limited hardware resource even on a smartphone. Besides, performance metrics, other parameters such as energy consumption, resource constraint parameters, and computing capability also to be considered for better implementation in real-world scenarios. Furthermore, our gait recognition model can extend its application toward the human behaviors prediction, which can play a vital role in psychology and crime investigation domains.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request (Sangeethak@kdu.edu.et).

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the present study.

Authors' Contributions

K. M. Monica conceptualized the study, performed data curation, performed formal analysis, developed the methodology, provided the software, and wrote original draft. R. Parvathy did supervision, reviewed and edited the article, did project administration, and performed visualization. A. Gayathri performed visualization, investigation, and formal analysis and provided the software. Rajanikanth Aluvalu performed data Curation, performed investigation, and provided resources software. K. Sangeetha performed supervision, reviewed and edited the article, and performed visualization. Chenna Reddy Vijaya Simha Reddy provided the software, performed validation, wrote the original draft, developed the methodology, and supervision.

References

[1] J. Bravo, L. Fuentes, and D. L. de Ipiña, "Theme issue: "ubiquitous computing and ambient intelligence"," *Personal and Ubiquitous Computing*, vol. 15, no. 4, pp. 315-316, 2011.

[2] J. Bravo, R. Hervás, J. Fontecha, and I. González, "m-Health: lessons Learned by m-Experiences," *Sensors*, vol. 18, no. 5, p. 1569, 2018.

[3] P. Rashidi and A. Mihailidis, "A survey on ambient-assisted living tools for older adults," *IEEE Journal of Biomedical and Health Informatics*, vol. 17, no. 3, pp. 579–590, 2013.

[4] E. De-La-Hoz-Franco, P. Ariza-Colpas, J. M. Quero, and M. Espinilla, "Sensor-based datasets for human activity recognition - a systematic review of literature," *IEEE Access*, vol. 6, pp. 59192–59210, 2018.

[5] J. Medina-Quero, S. Zhang, C. Nugent, and M. Espinilla, "Ensemble classifier of long short-term memory with fuzzy temporal windows on binary sensors for activity recognition," *Expert Systems with Applications*, vol. 114, pp. 441–453, 2018.

[6] R. Ali-Hamad, A. Salguero, M. H. Bouguelia, M. Espinilla, and M. Medina-Quero, "Efficient activity recognition in smart homes using delayed fuzzy temporal windows on binary sensors," *IEEE J. Biomed. Health Inform.*, 2019.

[7] F. Ordóñez and D. Roggen, "Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition," *Sensors*, vol. 16, no. 1, p. 115, 2016.

[8] M. Kose, O. D. Incel, and C. Ersoy, "Online human activity recognition on smartphones," in *Proceedings of the 2nd International Workshop on Mobile Sensing*, Beijing, China, April 2012.

[9] C. A. Ronao and S.-B. Cho, "Deep convolutional neural networks for human activity recognition with smartphone sensors," *Neural Information Processing*, in *Proceedings of the International Conference on Neural Information Processing*, pp. 46–53, Istanbul, Turkey, November 2015.

[10] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, "A public domain dataset for human activity recognition using smartphones," in *Proceedings of the European Symposium on Artificial Neural Networks (ESANN), Computational Intelligence and Machine Learning*, Bruges, Belgium, April 2013.

[11] O. D. Lara and M. A. Labrador, "A survey on human activity recognition using wearable sensors," in *IEEE Communications Surveys & Tutorials*vol. 15, no. 3, pp. 1192–1209, IEEE, 2013.

[12] R. San-Segundo, H. Blunck, J. Moreno-Pimentel, A. Stisen, and M. Gil-Martín, "Robust human activity recognition using smartwatches and smartphones," *Engineering Applications of Artificial Intelligence*, vol. 72, pp. 190–202, 2018.

[13] F. Attal, S. Mohammed, M. Dedabrishvili, F. Chamroukhi, L. Oukhellou, and Y. Amirat, "Physical human activity recognition using wearable sensors," *Sensors*, vol. 15, no. 12, pp. 31314–31338, 2015.

[14] A. Malaisé, P. Maurice, F. Colas, F. Charpillet, and S. Ivaldi, "Activity recognition with multiple wearable sensors for industrial applications," in *Proceedings of the Eleventh International Conference on Advances in Computer-Human Interactions*, Rome, Italy, March 2018.

[15] S. Allan, B. Henrik, B. Sourav et al., "Smart devices are different: assessing and mitigating mobile sensing heterogeneities for activity recognition," in *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems, ACM*, pp. 127–140, NY, USA, October 2015.

[16] A. Subasi, M. Radhwan, R. Kurdi, and K. Khateeb, "IoT based mobile healthcare system for human activity recognition," in *Proceedings of the 2018 15th Learning and Technology Conference (L&T)*, pp. 29–34, Jeddah, Saudi Arabia, February 2018.

[17] M. Kheirkhahan, S. Nair, A. Davoudi et al., "A smartwatch-based framework for real-time and online assessment and mobility monitoring," *Journal of Biomedical Informatics*, vol. 89, pp. 29–40, 2019.

[18] Y. Liu, L. Nie, L. Han, L. Zhang, and D. S. Rosenblum, "Action2activity: recognizing complex activities from sensor data," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Boston, MA, USA, June 2015.

[19] K. M. Masum, A. Barua, E. H. Bahadur, M. R. Alam, M. A. U. Z. Chowdhury, and M. S. Alam, "Human activity recognition using multiple smartphone sensors," in *Proceedings of the 2018 International Conference on Innovations in Science, Engineering and Technology (ICISET)*, pp. 468–473, Chittagong, Bangladesh, October 2018.

[20] Y. Tian, X. Wang, W. Chen, Z. Liu, and L. Li, "Adaptive multiple classifiers fusion for inertial sensor based human activity recognition," *Cluster Computing*, vol. 22, no. S4, pp. 8141–8154, 2019.

[21] A. Peinado-Contreras and M. Munoz-Organero, "Gait-based identification using deep recurrent neural networks and acceleration patterns," *Sensors*, vol. 20, no. 23, p. 6900, 2020.

[22] W. Shen and T. Tan, "Automated biometrics-based personal identification," *Proceedings of the National Academy of Sciences*, vol. 96, no. 20, pp. 11065-11066, 1999.

[23] R. He, X. Wu, Z. Sun, T. Tan, and C. N. N. Wasserstein, "Learning invariant features for NIR-VIS face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, pp. 1761–1773, 2018.

[24] K.-T. Nguyen, T.-L. Vo-Tran, D.-T. Dinh, and M.-T. Tran, "Gait recognition with multi-region size convolutional neural network for authentication with wearable sensors," in *Proceedings of the International Conference on Future Data and Security Engineering (FDSE)*, Ho Chi Minh City, Vietnam, December 2017.

[25] J. Hannink, T. Kautz, C. F. Pasluosta, K. G. Gasmann, J. Klucken, and B. M. Eskofier, "Sensor-based gait parameter extraction with deep convolutional neural network," *IEEE J. Biomed. Health Inform.*vol. 21, pp. 85–93, 2016.

[26] Q. Zou, Y. Wang, Q. Wang, Y. Zhao, and Q. Li, "Deep learning-based gait recognition using smartphones in the wild," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 3197–3212, 2020.

[27] A. Alharbi, K. Equbal, S. Ahmad, and H. Alyami, "Human gait analysis and prediction using the levenberg-marquardt method," *Journal of Healthcare Engineering*, vol. 2021, p. 11, Article ID 5541255, 2021.

[28] A. Saleh and T. Hamoud, "Analysis and best parameters selection for person recognition based on gait model using CNN algorithm and image augmentation," *J Big Data*, vol. 8, p. 1, 2021.

[29] J. Moon, N. A. Le, N. H. Minaya, and S.-I. Choi, "Multimodal few-shot learning for gait recognition," *Applied Sciences*, vol. 10, no. 21, p. 7619, 2020.

[30] W. Jiang and Z. Yin, "Human activity recognition using wearable sensors by deep convolutional neural networks Proceedings of the 23rd ACM international conference on Multimedia," *Acm*, pp. 1307–1310, 2015.

[31] G. Laput and C. Harrison, "Sensing fine-grained hand activity with smartwatches," in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, vol. 338, Glasgow Scotland Uk, May 2019.

[32] S. Ha and S. Choi, "Convolutional neural networks for human activity recognition using multiple accelerometer and gyroscope sensors," in *Proceedings of the 2016 International Joint Conference on Neural Networks*, pp. 381–388, Vancouver, BC, Canada, May 2016.

[33] Y.-H. Shen, K.-X. He, and W.-Q. Zhang, "SAM-GCNN: a gated convolutional neural network with segment-level attention mechanism for home activity monitoring," in *Proceedings of the 2018 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, pp. 679–684, Louisville, KY, USA, December 2018.

[34] H. Guo, L. Chen, L. Peng, and G. Chen, "Wearable sensor based multimodal human activity recognition exploiting the diversity of classifier ensemble," in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing. ACM*, pp. 1112–1123, Heidelberg Germany, September 2016.

[35] S. Yao, S. Hu, Y. Zhao, A. Zhang, and T. Abdelzaher, "Deepsense: a unified deep learning framework for time-series mobile sensing data processing," pp. 351–360, 2017, https://arxiv.org/abs/1611.01942.

[36] K. Greff, R. K. Srivastava, J. Koutnik, B. R. Steunebrink, and J. Schmidhuber, "LSTM: a search space odyssey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2222–2232, 2017.

[37] Y. Wang, J. Shen, and Y. Zheng, "Push the limit of acoustic gesture recognition," in *Proceedings of the IEEE Transactions on Mobile Computing (2020)*, Toronto, ON, Canada, July 2020.

[38] C. Xu, D. Chai, J. He, X. Zhang, and S. Duan, "InnoHAR: a deep neural network for complex human activity recognition," *IEEE Access*, vol. 7, no. 2019, pp. 9893–9902, 2019.

[39] Y. Yuta, J. Nozaki, K. Hiroi, K. Kaji, and N. Kawaguchi, "Activity recognition using DualConvLSTM extracting local and global features for SHL recognition challenge," in *Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers. ACM*, pp. 1643–1651, Singapore, September 2018.

[40] B. I. N. Guo, U. A. N. Chen, D. A. L. I. N. Zhang, and Y. U. Zhiwen, "Deep learning for sensor-based human activity recognition: Overview, challenges and opportunities," *ACM*, vol. 37, no. 4, p. 40, 2018.

[41] S. Qiu, H. Wang, J. Li et al., "Towards wearable-inertial-sensor-based gait posture evaluation for subjects with unbalanced gaits," *Sensors*, vol. 20, no. 4, p. 1193, 2020.

[42] A. Balakrishnan, K. Ramana, G. Dhiman et al., "Multimedia concepts on object detection and recognition with F1 car simulation using convolutional layers," *Wireless Communications and Mobile Computing*, vol. 2021, pp. 1–21, Article ID 5543720, 2021.

[43] K. Jun, S. Lee, D.-W. Lee, and M. S. Kim, "Deep learning-based multimodal abnormal gait classification using a 3D skeleton and plantar foot pressure," *IEEE Access*, vol. 9, pp. 161576–161589, 2021.

[44] C. Jothikumar, K. Ramana, V. D. Chakravarthy, S. Singh, and I. H. Ra, "An efficient routing approach to maximize the lifetime of IoT-based wireless sensor networks in 5G and beyond," *Mobile Information Systems*, vol. 2021, pp. 1–11, Article ID 9160516, 2021.

[45] X.-S. Yang, "A new metaheuristic bat-inspired algorithm," *In Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)*, pp. 65–74, Springer, Berlin, Heidelberg, 2010.

[46] A. Balakrishnan, R. Kadiyala, G. dhiman et al., "A Personalized Eccentric Cyber-Physical System Architecture for Smart Healthcare," *Security and Communication Networks 2021*, 2021.

[47] T. R. Gadekallu, D. S. Rajput, M. P. K. Reddy et al., "A novel PCA-whale optimization-based deep neural network model for classification of tomato plant diseases using GPU," *Journal of Real-Time Image Processing*, vol. 18, no. 4, pp. 1383–1396, 2021.

[48] V. Guimarães, I. Sousa, and M. V. Correia, "A deep learning approach for foot trajectory estimation in gait analysis using inertial sensors," *Sensors*, vol. 21, no. 22, pp. 7517–7522, 2021.

[49] X. Wang, Y. Zhang, H. Zhang, and Y. Li, "Radio frequency signal identification using transfer learning based on LSTM," *Circuits, Systems, and Signal Processing*, vol. 39, no. 5, 2020.
[50] D. Kreuzer and M. Munz, "Deep convolutional and LSTM networks on multi-channel time series data for gait phase recognition," *Sensors*, vol. 21, no. 3, 2021.
[51] P. K. R. Maddikunta, T. R. Gadekallu, R. Kaluri, G. Srivastava, R. M. Parizi, and M. S. Khan, "Green communication in IoT networks using a hybrid optimization algorithm," *Computer Communications*, vol. 159, pp. 97–107, 2020.
[52] Y. Nancy Jane, H. Khanna Nehemiah, and K. Arputharaj, "A Q-backpropagated time delay neural network for diagnosing severity of gait disturbances in Parkinson's disease," *Journal of Biomedical Informatics*, vol. 60, pp. 169–176, 2016.
[53] H. Huang, P. Zhou, Y. Li, and F. Sun, "A lightweight Attention-based CNN model for efficient gait recognition with wearable IMU sensors," *Sensors*, vol. 21, no. 8, p. 2866, 2021.
[54] N. Dua, S. Singh, and V. B. Semwal, "Multi-input CNN-GRU based human activity recognition using wearable sensors," *Computing*, vol. 103, no. 3, pp. 1–18, 2021.
[55] T. Liu, X. Ye, and B. Sun, "Combining convolutional neural network and support vector machine for gait-based gender recognition," in *Proceedings of the 2018 Chinese Automation Congress (CAC)*, pp. 3477–3481, Xi'an, China, November 2018.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 5092962, 14 pages
https://doi.org/10.1155/2022/5092962

Research Article

Performance Evaluation of SeisTutor Using Cognitive Intelligence-Based "Kirkpatrick Model"

Ninni Singh,[1] **Vinit Kumar Gunjan**,[1] **Ramana Kadiyala**,[2] **Qin Xin**,[3] **and Thippa Reddy Gadekallu**[4]

[1]*Department of Computer Science and Engineering, CMR Institute of Technology, Hyderabad, Telangana 501401, India*
[2]*Department of Information Technology, Chaitanya Bharathi Institute of Technology, Hyderabad, Telangana, India*
[3]*University of the Faroe Islands, Faroe Islands, Denmark*
[4]*Department of Information Technology, Vellore Institute of Technology, Vellore, Tamilnadu, India*

Correspondence should be addressed to Thippa Reddy Gadekallu; thippareddy.g@vit.ac.in

Received 9 February 2022; Accepted 1 April 2022; Published 16 April 2022

Academic Editor: Shakeel Ahmad

The classroom learning environment facilitates human tutors to interact with every learner and get the opportunity to understand the learner's psychology and then provide learning material (access learner prior knowledge and well align the learning material as per learner requirement) to them accordingly. Implementing this cognitive intelligence in intelligent tutoring system is quite tricky. This research has focused on mimicking human tutor cognitive intelligence in the computer-aided system of offering an exclusive curriculum to the learners. The prime focus of this research article is to evaluate the proposed SeisTutor using Kirkpatrick's four-phase evaluation model. Experimental results depicting the enhanced learning gain through intelligence incorporated SeisTutor as against the intelligence absence are demonstrated.

1. Introduction

Understanding "Seismic Interpretation" has been considered as important and valuable for petroleum exploration because it provides a sophisticated way of delineating the Earth's subsurface (in the form of seismic snap), understanding the seismic snaps, and then concluding the presence of hydrocarbon or not. This technique can be applied again and again while capturing, analyzing, interpreting, and predicting the presence of hydrocarbon amassing [1].

The interpretation of seismic snaps is a challenging task for a seismologist. Seismologists utilize their expertise to interpret seismic images. However, one possibility is that different geologists interpret the same seismic snap differently during interpretation. This vagueness is due to a lack of thumb rules—the interpretation expertise gained by experience over past years. Thus, novice geologists in this field took many years to gain these expertises. It will be paramount for them if these skills are offered to them at the beginning of their interpretation career. To accomplish this, exploratory learning would be the best strategy for learning. In this learning, an individual has control over the learning process. It means the learner chooses his/her topics, level of difficulty, learning pace, and so forth [2]. To effectively meet this learner-centric requirement, an intelligent tutoring system can offer personalized learning experiences to learners. Intelligent tutoring system (ITS) is an artificial intelligence (AI) technique that provides the learner exclusive learning material, aligned and gathered as per learner grasping ability and preferred media of learning.

SeisTutor has been developed for learning "Seismic Data Interpretation" as a subject domain of this research. This research aims to provide pastoral care to learners by cost-effectively offering one-to-one, customized learning material. SeisTutor delivers a personalized learning environment. It brings personalization in identifying tutoring strategy (based on pretest (learning style test and domain knowledge test)), exclusive curriculum design, and observation of learner psychological state of mind during learning sessions. The feature of personalization is built on many aspects, such as accuracy of predicting tutoring strategy (based on pretest), curriculum design, and psychological parameters. The only technique to determine the performance of the personalization facility of SeisTutor is to appraise the system in actual circumstances (learners who are learning the course material). Appraise acts as a critical element for quality assurance because it enables the learner to give their valuable feedback on learning experience and on learning content, which further helps to understand the learner perspective and makes the learning better.

The evaluation of SeisTutor was accompanied in the 2018–2019 academic year. The objective of the assessment is to examine how effectively SeisTutor personalized itself to fulfill the learner's needs and whether SeisTutor helps enhance the learning gain for learning the "Seismic Data Interpretation" domain. To accomplish this, the following program is outlined:

(1) Perform schematic literature review on how to evaluate the efficacy of learning.
(2) Generate an assessment for determining the effectiveness of learning.
(3) Perform analysis to identify how efficiently learners learn with SeisTutor.

This article is organized into five sections. It begins with a schematic literature review on evaluation on learning with ITS (Section 2). Further, Section 3 introduces the architecture of SeisTutor in detail, which includes the general description and mathematical justification of its functionalities. Section 4 illustrates the experimental analysis performed to evaluate the effectiveness of SeisTutor systematically. Section 5 elaborates the comparative analysis with current work, and, in Section 6, we finally conclude the paper with the implication of this work.

2. Background and Preliminaries

Evaluating and validating an ITS is a challenging task due to the lack of standard agreement and procedure.

The primarily used prototype for evaluating the training program of ITS is the prototype established by Donald Kirkpatrick [3–6]. Many researchers have revised this prototype, but its basic architecture is still the same [7].

Kirkpatrick's prototype comprises four stages of evaluation shown in Figure 1 and briefly discussed in Figure 2.

Computing results is the best way to quantify the effectiveness of any learning program, but it is challenging to conduct. McEvoy and Buller [13] put a statement regarding evaluation that not all learning programs focus on impacting the learning performance of a learner; instead, they can perform for a purpose. Other researchers used the Kirkpatrick model as a base model. Philips [14] introduced the fifth level on Kirkpatrick model named Return of Investment (ROI), used to measure the effectiveness of learning or investment.

The authors in [15] considered Kirkpatrick's model as a base model for measuring the effectiveness of a learning program. Their suggestion is to set the initial objective (from the learning program) and then monitor the fulfillment of objectives after the learning program. References [16, 17] disapproved Kirkpatrick's model by giving the following reason:

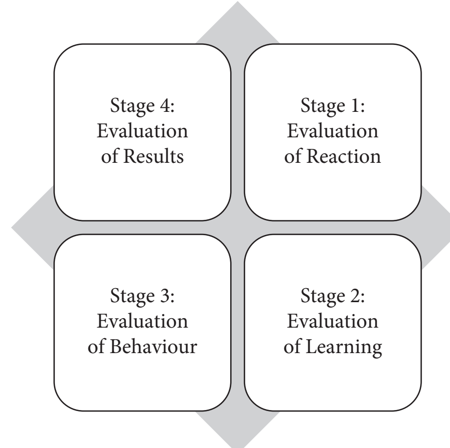


Figure 1: Kirkpatrick's four stages of evaluation.

(i) Offline Test (Written Test) lacks validity and reliability in quantifying knowledge, skill, and attitude (KSA).
(ii) 100% response rate is idealistic.
(iii) Control groups are not feasible in the learning program context.

Furthermore, educational organizations have recommended considering the merits and demerits of various evaluation prototypes and methodologies to build an organizational-specific evaluation prototype that fulfills their requirements [18]. In addition to this, it has been suggested that the evaluation prototype emphasizes both education processes and their outcomes [19]. Valid, reliable, inexpensive, and acceptable are the features of an ideal evaluation prototype. Moreover, the evaluation prototype may encompass quantitative, objective, subjective, and qualitative methods. Thus, evaluation results are advantageous for determining the learning attainments of the learning program [17, 20, 21].

The conclusion drawn from the literature recommends that the learning program include three levels of Kirkpatrick's prototype. Kirkpatrick level 3 and level 4 are very challenging to observe in an educational learning program [22] because while level 1 and level 2 can be quantified during an ongoing learning session, level 3 and level 4 require postassessment analysis. Level 4 needs the rigorous observation of the inference of the learning program. For evaluation, there is no articulated framework. As per [23], continuous feedback and instruction help the learner to achieve best in learning skills Storch and Tapper [24]; Polio et al. [25] Elliot and Klobucar [26]; Aryadoust et al. [27].

3. SeisTutor Functionality

This section illustrates the SeisTutor architecture and briefly discusses functionality incorporated in SeisTutor. SeisTutor

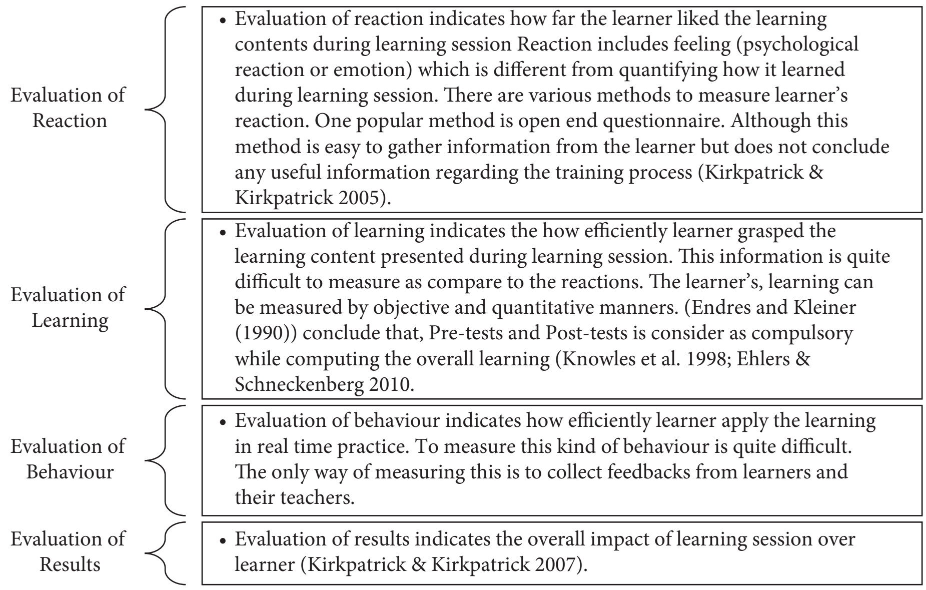


Figure 2: Brief illustration of Kirkpatrick's four stages of evaluation [8–12].

is an ITS explicitly designed for the "Seismic Data Interpretation" subject domain. SeisTutor recommends learning contents as per learner performance in pretest (prior knowledge assessment test). In addition, SeisTutor keeps track of the learner's behavior (psychological state) during the entire learning session and conducts a test to determine the degree of understanding of the topic. SeisTutor is adaptive, that is, content and link-level. Adaptive content level indicates that the learners with different performance in pretest (prior knowledge test) get different learning material. Before indulging learners in the learning session, SeisTutor enables learners to go through a pretest. There are two kinds of assessments performed in pretest: a learning style test and a prior knowledge test. The current focus of this research is on the prior knowledge test. SeisTutor observes the learner's performance during the tests and aligns the learning material accordingly.

For link elimination, ideally, the curriculum covers all the links of subtopics to learn; however, the links eliminated are not in the determined curriculum for the learner. SeisTutor continually observes and stores the learner's action, performance, and behavior. This information is further utilized for making intelligent strategic plans for recommendation and evaluation work.

4. SeisTutor Architecture

SeisTutor follows all the guidelines for implementing an intelligent tutoring system. The critical functional model of an ITS is the domain model, learner model, pedagogy model, and learner interface. SeisTutor architecture comprises learner model, pedagogy model, domain model, and learner interface (architecture paper). The following subsection briefly illustrates the various components involved in SeisTutor.

4.1. Domain Model. The domain model is described as the cluster of concepts. Here concept terminologies indicate the single topic. Other terminologies are used in the different research papers, such as knowledge element, object, learning outcome, and attribute. In the current context, concepts possess a prerequisite relationship to each other. Each concept is further segregated into learning units. SeisTutor utilizes unit variant technologies to attain content-level adaptation. Content-level adaptation indicates that the system has alternative units and recommends the learning units based on learner grasping and learning style.

4.2. Pedagogy Model. The pedagogical model consists of various rules and logic that build a knowledge infrastructure essential for adapting the learning materials as per the learner's.

Curriculum planner: The planner generates a curriculum as a sequence of learning units to be covered during the learning session [28].

Learning assessment: Learning assessment is the process of determining the learner's learning process. The accuracy of this adjudging model acts as a critical factor that affects the adaptation practice.

Understanding assessment: Understanding assessment identifies the learner's degree of understanding of the concept. SeisTutor makes any decision by referring to all the models like determining the curriculum for the learner and offering the learning content as per learner grasping level and learning style.

4.3. Learner Model. The learner model captures the learner's activity during the learning session and stores and updates the learner's information for making the decision. This information is fruitful for the system to adapt based on learner characteristics (grasping level and learning style). SeisTutor captures three characteristics of a learner.

Learner demographic information: Learner demographic information includes learner's basic details, such as name, learner username, and e-mail id. This information is used to create the learner profile and collected when the first-time learner signs up with SeisTutor.

Psychological state: It recognizes the learner's emotions during the tutoring session. Recognizing the psychological state of mind is essential because it helps determine how far the learner liked the learning contents during the learning session. SeisTutor determines six emotions, that is, happy, sad, neutral, surprised, afraid, and angry.

This **psychological state** recognition module gets triggered as soon as the learner starts the learning session, as shown in Figure 3. Initially, participants belong to both studies and undergo the initial assessment phase (pretest); after pretest, their learner profile and learning style are determined. The I2A2 learning style prototype is utilized to ascertain the learning style. Based on the tutoring strategy (study 2 and study 1) and course coverage plan, the learner and pedagogy model determine the relevant learning mode (study 1) and represent the learning contents delivered using the learner interface. Thus, SeisTutor examines the participant's attainment in pretest and predicts the individualized tutoring strategy (study 2 and study 1) and custom-tailored curriculum for the applicants (study 1).

As illustrated in Figure 4, the CNN-based emotion recognition module is instantiated as soon as the learner begins the learning session. This emotion recognition module takes the snap of the learner via webcam, which acts as an input to the CNN-based emotion recognition module. This module determines the learner's psychological (emotion) state, which is gauged for future analysis (phase 1: evaluation of reaction). The gathering psychological (emotion) state is repeated until the learner completes all the learning contents (topics) associated with all the weeks.

4.4. Min-Max Normalization. The learner classification parameters (prior knowledge test (0–5), postassessment test (0–5), learning gain (0–5), and learner emotion (0–100)) of each learner were used and normalized using the method called Min-Max. It converts a value of $Y = \{y_1, y_2, y_3, \ldots, y_n\}$ and converges in the range of $[A, B]$. The formula for score standardization is specified below, where A is the lowest range and B is the highest range. In our case $[A, B]$ is [0, 10];

$$Z = \left\{ \frac{y_i - \text{Lowestvaluein}Y}{\text{Highestvaluein}Y - \text{Lowestvaluein}Y} \right\} * (B - A) + A. \quad (1)$$

Learner performance evaluates the learner's performance by quantifying the learner's learning by organizing quizzes and tests.

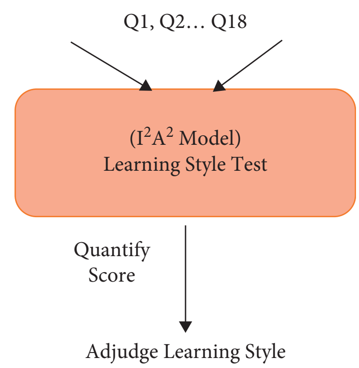


Figure 3: Flow of learning style test (LST) questionnaire.

4.5. Quiz and Understanding Test Representation. Learner performance has been quantified based on one parameter, that is, the number of correct responses. As soon as the learner finishes the learning content of each week, he or she has to give a quiz and understanding test. One quiz is associated with every week (shown in Figure 5). Each quiz contains five questions, and each question contains only one hint. Hints appear to the learner based on the learner's request to seek help to solve the question. SeisTutor asks the learner to summarize the learned concepts in understanding the test. Based on the user entered information, dictionary-based sentimental analysis is performed. The result gives a score out of 100, which tells the learner's overall understanding of the concepts.

To determine the overall learning gain, SeisTutor has pretest and posttest assessment scores. The average learning gain is computed (using the following equation):

$$\text{Learning\_Gain} = \left(\text{PostTest\_Score}_L - \text{PreTest\_Score}_L\right). \quad (2)$$

4.5.1. Learning History. It keeps track of the learner's activity, such as login time and total time spent, and its interaction details during the entire tutoring sessions.

SeisTutor utilizes this information to make necessary action which further helps to make the whole learning process effective.

To start tutoring with SeisTutor, the learner needs to register himself/herself first (see Figure 6). After registration, SeisTutor creates a learner account and instructs the learner to give a pretest. Pretest comprises two tests:

(1) Prior knowledge test

(2) Learning style test

4.6. Prior Knowledge Test. The prior knowledge test is the preliminary test used to identify the domain's elementary knowledge to know the learner's initial learning level. This test is responsible for analyzing the learner's knowledge about the domain. The domain model examined the test result and categorized the learners into three learning profiles, that is, "beginner, intermediate, and expert" (see Figure 7). A further outcome of this result is also responsible for determining the curriculum, which is exclusively

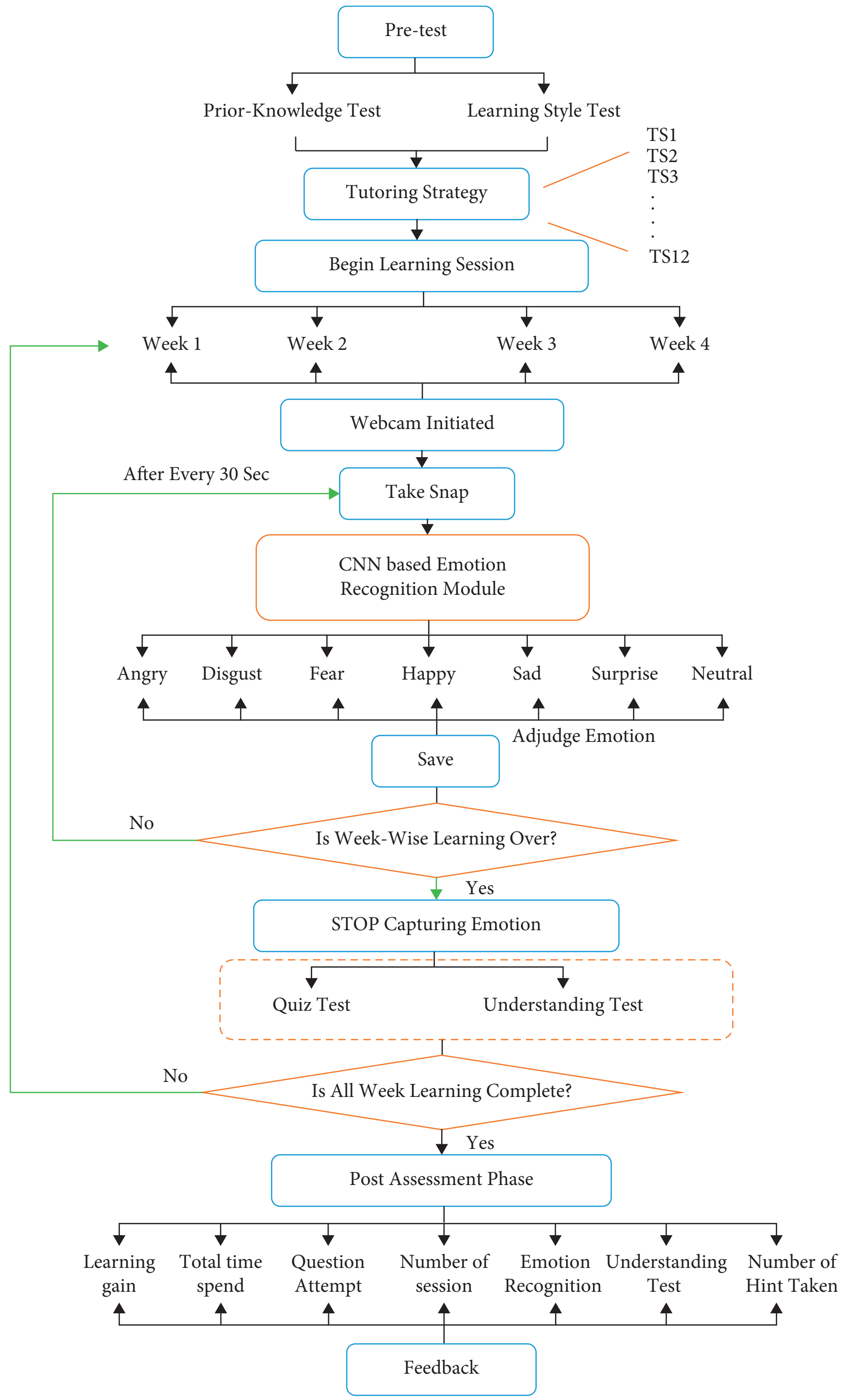


Figure 4: Flow diagram of CNN-based emotion recognition module.

designed for the learner. This test comprises twenty questions, and all the questions are verified by the domain expert of Seismic Data Interpretation.

4.7. Learning Style Test. This test is responsible for determining preferred media for learning. The research has noticed that learner performance is gradually increased if learning material is provided as per preferred learning media (see Figure 3). Thus, this test analyzes the learner-preferred media. The learner model examines the test result and categorizes the learners into four learning styles, that is, “Imagistic, Acoustic, Intuitive, and Active.”

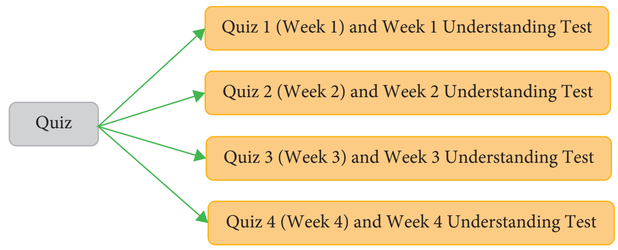


Figure 5: Quiz and understanding test representation.

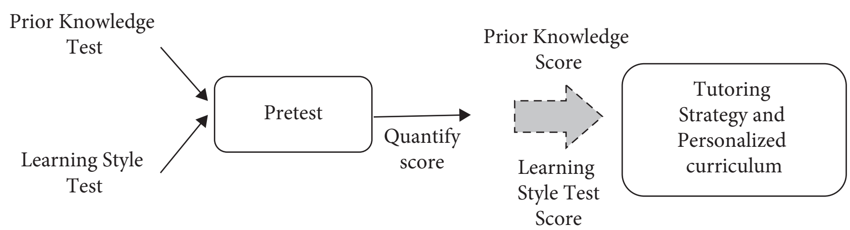


Figure 6: Flow diagram of learner characteristics model.

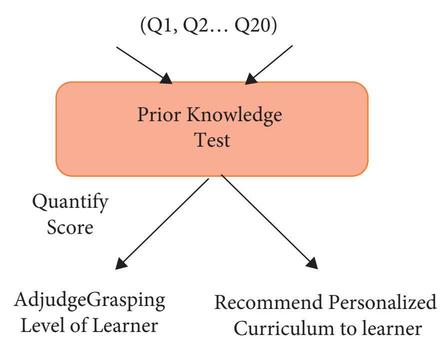


Figure 7: Flow of domain knowledge test (DKT).

Based on these two tests, the pedagogy model determines the tutoring strategy (the result of learning profile and learning style). A tutoring session then begins with the determined tutoring strategy. As soon as learning begins, psychological features get triggered, capturing learner emotions and saving the results in the database for future reference. After completion of every week, the learner is tested. Based on the test result, learner performance and degree of understanding are computed.

5. Results and Discussions

After tutoring, posttutoring assessments of learners are performed by SeisTutor (see Figure 8). This section depicts the analysis used to determine the impact of study 2's learning methods with SeisTutor. It exercises learning practices with features like personalized curriculum design and recognition of the psychological state of the learner during the learning process, and learner's degree of understanding of taught concepts is characterized as study 2. When it is not practiced, it is characterized as study 1. An experimental comparison of both study groups is tabulated (Table 1). These variances help to identify the discrepancy in the learning experiences.

5.1. Experimental Design and Methodology. The SeisTutor evaluation is the fundamental piece of the development of this framework. In order to quantify the adequacy and effectiveness of SeisTutor, assessment tests have been conducted. SeisTutor has been tested on a selected population of students, teachers, and both (teachers and students) from an anonymous university. A total of 60 learners volunteered in the evaluation process. Based on their compliance in the participation, a compliance agreement form was issued which demonstrates essential details related to the assessment process. It is a requisite for each applicant to give their approval for participation in the assessment process. Applicants were haphazardly assigned one of the groups. 32 applicants were in study 1, and the remaining were in study 2.

Out of 60 learners, 30% are learners pursuing graduation, 17% are graduates, 35% are postgraduates, and the rest's minimum qualification is doctorate (Ph.D.).

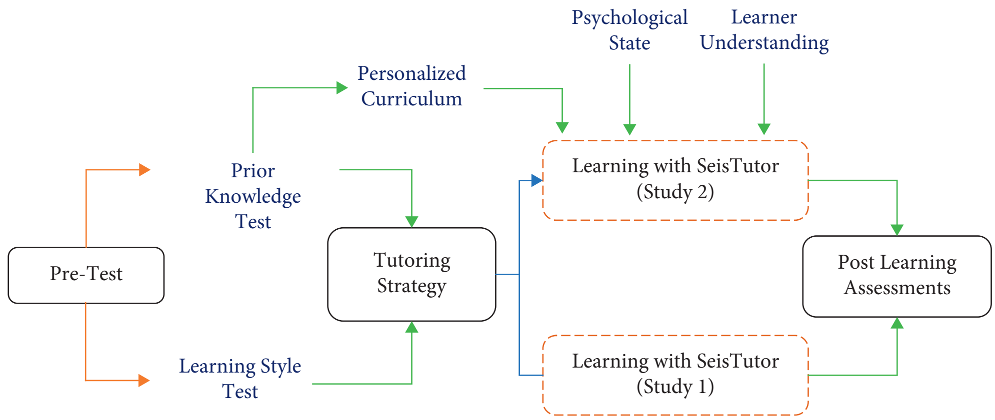


Figure 8: Flow of accomplishing the evaluation process.

Table 1: Feature of study 1 and study 2 evaluation groups.

	Study 2	Study 1
Personalized tutoring contents	Offer learning content (similar curriculum) based on tutoring strategy	Offer personalized learning content (different curriculum) based on determined prior knowledge level and tutoring strategy [29, 30]
Psychological state tracking	Psychological state of the learner is not tracked during ongoing learning session	Determine psychological state of the learner during ongoing learning session
Degree of understanding computation	Learner's understanding about the concept is not adjudged	Quantify learner's understanding about the concept

12% of learners were in the 18–20 age group, 18% of learners were in the 20–22 and 22–24 age groups, 5% of learners were in the 24–28 age group, 12% of learners were in the 28–32 age group, 22% of learners were in the 32–34, and the rest were in the above 34 age group (shown in Table 2). SeisTutor is explicitly created for the "Seismic Data Interpretation" domain. As a result, it is intended to be used by participants or learners belonging to the petroleum engineering and exploration domain. Thus, to quantify the effectiveness of SeisTutor, undergraduate learner (B.Tech/B.E. Petroleum Engineering), teacher (Petroleum Engineering Dept.), and others (government exploration industry) are taken into consideration (see Table 2).

The learners underwent a pretest as soon as they got registered with SeisTutor. Their learning style and grasping level (learning level) were adjudged as 28 learners as mentioned above were in study 2. Based on their responses in the pretest, the custom-tailored curriculum is determined, which is realigned and reorganized from the domain (content) capsule. SeisTutor examines every learner involved in study 2, identifies their psychological state (emotions) during the learning session, and quantifies the degree of understanding about the concepts. The remaining learners in study 1 follow the standard curriculum for learning, that is, contents in the same sequence (irrespective of their prior knowledge about the domain). Thus, all the learners follow the same learning path. The point to underline is that their pretest performance is not used for exclusive learning path recommendations. The learning session begins week-wise for both the study groups and subsequent postassessment tests.

5.2. Data Preparation. Before analysis, obtained data underwent data screening phase. In this phase, the elimination of missing values and data normalization is performed. For deducing conclusion about the effectiveness of learning through SeisTutor, learner's performances, that is, prior knowledge test (pretest), understanding test, psychological state results, and quiz test (posttest) during learning, are taken into consideration. SPSS version 25 was used for accomplished analysis.

5.3. Result Discussion. The evaluation of SeisTutor is performed using the Kirkpatrick evaluation model. As discussed in Section 2, the Kirkpatrick evaluation prototype comprises four phases shown in Figures 1 and 2. Table 3 describes the statistical methods and performance metrics used in the four levels of evaluation.

5.4. Kirkpatrick Phase 1: Evaluation of Reaction. Evaluation of reaction as its name indicates is the evaluation of the reaction (emotion) of the learner during learning or how far the learner likes the learning content and teaching process, that is, pedagogy. SeisTutor incorporates an emotion recognition module and an open-end questionnaire (learner feedback).

The Min-Max normalization is utilized to maintain the uniformity, which converges the original values in the scope of [0–10].

As mentioned in Table 1, the psychological state of the learner is determined only for the applicants involved in study 1. Thus, their descriptive states are shown in Table 4.

Table 2: Learner demographic characteristics.

Characteristic		Frequency (N = 60)	Percentage (%) (N = 60)
Gender	Male	35	58
	Female	25	42
Age	(18–20)	7	12
	(20–22)	11	18
	(22–24)	11	18
	(24–28)	3	5
	(28–32)	7	12
	(32–34)	13	22
	(>34)	8	13
Qualification	Diploma	0	0
	High and secondary school	18	30
	Graduation	10	17
	PG	21	35
	Ph.D.	11	18
Occupation	Student	18	30
	Teacher	11	18
	Both (student and teacher)	19	32
	Others	12	20

Table 3: Statistical and performance metrics considered for four-phase evaluation.

Kirkpatrick levels	Statistical methods	Performance metrics
Level 1: evaluation of reaction	Average mean score analysis	Emotion parameter
Level 2: evaluation of learning	Average mean score analysis Bivariate Pearson correlation	Learning gain Degree of understanding
Level 3: evaluation of behavior	Average mean score analysis (learner feedback)	Effectiveness Overall satisfaction Adaptation & personalization Satisfaction level on custom-tailored curriculum sequencing recommendation module
Level 4: evaluation of results	Paired-wise sample t-test (hypothesis analysis)	Pretest and posttest scores

From the stats shown in Table 4, the average mean score percentage among 28 applicants is 44% for emotion happy, 40% for emotion neutral, 36% for emotion angry, 32% for emotion surprised, 30% for emotion afraid, and 24% for emotion sad. Thus, one can deduce with confidence that, on average, learners are happy with the learning content and teaching process, that is, pedagogy.

Table 4: Descriptive statistics of psychological parameter of learner for study 1.

Emotions	Mean (/10)	Std. deviation	Mean (%)
Happy	4.4174	29.6357	44
Sad	2.4272	24.9175	24
Surprised	3.2275	28.2939	32
Afraid	3.0612	26.8571	30
Angry	3.6728	26.1069	36
Neutral	4.0389	26.6193	40

5.5. Kirkpatrick Phase 2: Evaluation of Learning. Evaluation of learning, as its name indicates, is the evaluation of how effectively learners grasp the learning content. SeisTutor conducts a small quiz and degree of the understanding test to adjudge the learner's overall learning.

The average learning gain of applicants involved in study 1 is 22% and in study 2 it is 12%. Thus, it is concluded that if learning material is offered as per the learner's inclination with an exclusively designed curriculum based on the learner's prior knowledge, then the proposed SeisTutor succeeds in enhancing the learner curiosity and interest, which indirectly enhances the overall learning gain (see Table 5 and Figure 9).

These tests were performed on learning gain and degree of understanding of study 1 through SeisTutor. Table 4 illustrates the progressive learning gain of 22% discovered among learners, those applicants who participated in study 1 of SeisTutor. Furthermore, this information (statistical) is used in analysis, that is, bivariate Pearson correlation.

Understanding tests were designed and conducted only for study 1 because it strongly proves the effectiveness of learning gain achieved by intelligent feature incorporated SeisTutor. Here, the correlation of learning gain with itself is

Table 5: Learner's learning gain.

Study cases	Number of participants (n)	Learning gain		
		Mean (/10)	Standard deviation	Mean (%)
Study 1	28	2.2170	1.02795	22
Study 2	32	1.2793	1.37034	12

one because a variable or parameter is perfectly interrelated. The Pearson correlation of learning gain with degree of understanding is 0.484, and the two-tailed significance, that is, P value, is less than 0.01 (see Tables 6 and 7). Thus, one can confidently say that learning gain and degree of understanding have statistically significant linear relationships (see Figure 10).

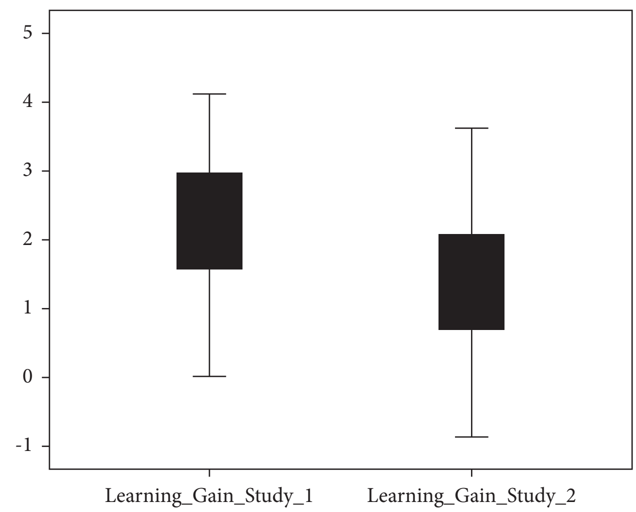


Figure 9: Learning gain of study 1 and study 2.

5.6. Kirkpatrick Phase 3: Evaluation of Behavior. Evaluation of behavior is quite hard to quantify. To measure this, SeisTutor collects feedback from the applicants. As the learner completes all the learning concepts of every week, SeisTutor requests the learner feedback. In this section, a conclusion from the learner feedback is drawn. Learners who have been the impeccable part of this evaluation of SeisTutor had a good perception of the system, and their feedbacks are very encouraging. It has appeared in their reactions whether they would like to recommend SeisTutor to others who need to take this study. Around 93% of the learners showed that they would recommend it to others, out of which 48% showed strong agreement, and the remaining 45% agreed on a recommendation as well (see Table 7). The overall satisfaction with SeisTutor was around 93%, out of which 45% were strongly satisfied and 48% were satisfied. It has also been observed that learners' studies became productive with SeisTutor (Table 8).

Few questions were asked on the impact of the intelligent features provided by SeisTutor and they are collected and summarized in Table 9. As some intelligent feature is not provided for study 2 applicants, 28 effective feedback from study 1 participants have been taken into consideration.

Most of the participants were happy with the tutoring strategy provided by the system, with 86% satisfaction, which includes 46% who were satisfied and 40% who were strongly satisfied. 85% of participants felt that learning by their own made them perform better, with 40% being strongly satisfied and 45% being satisfied. The participants were happy with the recommended exclusive curriculum by the system with 85% satisfaction, with 35% being satisfied and 50% being strongly satisfied. Most of the participants were happy with the recommended custom-tailored curriculum provided by the system, with 85% satisfaction, with 39% being satisfied and 46% being strongly satisfied. 92% of participants agreed that the understanding test at each week corresponds to the lesson taught; 39% strongly agreed, and 53% agreed. At last, 82% of students agreed with the psychological parameter accurately determined with SeisTutor: 39% strongly agreed to this and 43% agreed.

The overall impact of the support provided by SeisTutor on the learning process is assessed through the learner's feedback questionnaire answered by 60 participants (see Table 10). The analyzed results showed that 87% of the students are happy with SeisTutor supports, with 47% satisfied and 40% strongly satisfied. In addition, 78% of the students are happy with the system navigation support to find the needed information, with 43% satisfied and 35% strongly satisfied. At last, there was 80% satisfaction among the students, out of which 42% had a strong agreement and 38% agreed that the SeisTutor prelearning procedure was beneficial for learning.

The usefulness of the lesson components such as lesson explanations, revisions, presented quizzes, and the question hints in the learning process was evaluated in Table 11. The questionnaire feedback results show that 85% of students were happy with the content explained by SeisTutor, with 47% satisfied and 38% strongly satisfied. Moreover, 78% of students showed interest and agreed that the tutoring resources were adequate, with 35% strongly satisfied and 43% satisfied. It is clear that the quizzes and hints were realistic and focused with the learning contents provided by SeisTutor.

The impact of the interactive graphical user interface, content organization, and design features of SeisTutor in the learning process is evaluated through the learner questionnaire responses, described in Table 12. The questionnaire results revealed that there was 77% satisfaction with the interactive GUI and content organization of SeisTutor among the learners, with 40% strongly satisfied and 37% satisfied. There was 80% satisfaction with SeisTutor to compel and support to complete the quizzes and lessons among the learners, with 44% strongly satisfied and 36%

Table 6: Average mean score of learning gain and degree of understanding.

Study 1 parameters	No. of participants (n)	Learning gain		
		Mean (/10)	Standard deviation	Mean (%)
Learning gain	28	2.2170	1.02795	22
Degree of understanding	28	2.5467	1.31201	25

Table 7: Correlation matrix between learning gain and degree of understanding.

Parameters		Learning gain	Degree of understandings
Learning gain	Pearson correlation	1	0.484**
	Sig. (2-tailed)		0.009
	N	28	28
Degree of understanding	Pearson correlation	0.484**	1
	Sig. (2-tailed)	.009	
	N	28	28

**Correlation is significant at the 0.01 level (2-tailed).

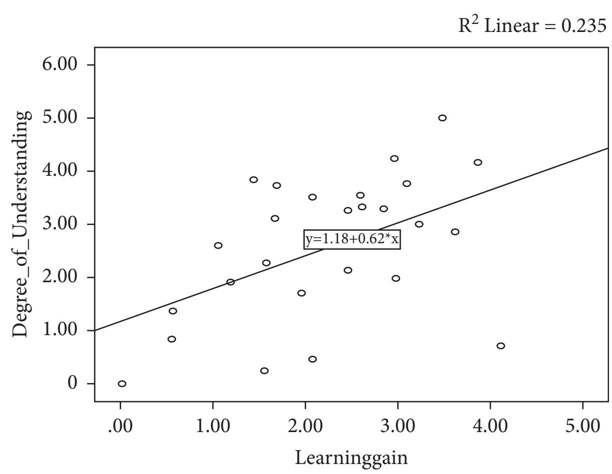


Figure 10: Linear relationship with learning gain and degree of understanding.

Table 8: Learner feedback on effectiveness of SeisTutor.

	Questions	Degree				
		Strongly satisfied	Satisfied	Neutral	Dissatisfied	Strongly dissatisfied
System effectiveness	What is your overall level of satisfaction with SeisTutor?	27	23	6	3	1
	The learning through this tutoring system (SeisTutor) was easy	27	26	5	1	1
	Did you feel that you were achieving learning outcomes?	30	21	6	3	0
	I would recommend a course through SeisTutor with no instructor help	29	24	3	5	0
	Would you recommend SeisTutor to individual who needs to take another course?	25	27	5	3	0
	Did SeisTutor support you to make your study productive?	28	27	3	0	2
	How well does this system deliver on your learning intentions?	31	21	5	2	1

Table 9: Learner feedback on adaptivity of SeisTutor.

	Questions	Degree				
		Strongly satisfied	Satisfied	Neutral	Dissatisfied	Strongly dissatisfied
Adaptivity/ personalization	Did SeisTutor satisfy you with dynamic creation of your learning profile?	27	21	7	1	4
	Were you convenient and satisfied with the tutoring strategy presented to you by SeisTutor?	24	19	9	6	2
	The information provided by SeisTutor is at a level that you understand	29	17	12	0	2
	The tutoring session was at the right level of difficulty for me	26	23	9	2	0
	As a learner, did you feel that your learning style was appropriately judged?	29	25	3	2	1
	Once tutoring begins and you were tutored, were your learning preferences sufficiently satisfied?	24	24	9	2	1
	Did the experience of learning by your own learning preference make you perform better?	24	21	6	7	2
	Based on your prior subject knowledge, has SeisTutor accurately determined exclusive curriculum for you?	14	8	2	3	1
	How satisfied are you with the exclusively determined curriculum?	13	7	4	3	1
	As a learner did you feel learning material enabled you to improve your ability to formulate and analyze the problem?	10	14	1	3	0
	Are you satisfied with the sequencing of learning content?	14	09	3	2	0
	Does sequencing of learning material relate with your prior knowledge? (give rating)	12	11	2	3	0
	Does learning content formulated under various learning levels and styles satisfactorily justify itself? (give rating)	17	7	2	0	2
	Has this learning session been successful in improving your knowledge in the subject domain? (give rating)	12	11	2	3	0
	Did this learning material fulfill your expectations?	11	13	2	1	1
	The understanding test at the end of each week corresponds to the lessons taught?	11	13	2	0	2
	SeisTutor compels and supports me to complete the quizzes, understanding test, and lessons ?	13	12	2	0	1
	The posttutoring evaluation system (week-wise understanding) as it exists is	14	10	1	2	1
	How do you rate the sequence of the lessons in the course?	18	8	0	0	2
	Has SeisTutor accurately determined your psychological (emotional) state during tutoring session? (give rating)	11	7	5	5	0
	Do you feel recognition of emotion during ongoing tutoring is indicative of empathy of the system?	13	12	2	0	1
	The course contents are relevant and well organized?	14	10	1	2	1

Table 10: Learner feedbacks on SeisTutor ongoing learning support.

Questions	Degree				
	Strongly satisfied	Satisfied	Neutral	Dissatisfied	Strongly dissatisfied
How are you satisfied with the system support?	24	17	11	6	2
The system navigation support enabled finding the needed information easily	21	17	9	11	2
Was the prelearning procedure available in SeisTutor helpful to you?	25	17	6	9	3
Were you able to understand the language used to explain the lessons in SeisTutor?	33	21	6	0	0
The tutoring was flexible to meet my learning requirements	30	21	7	2	0

Table 11: Learner feedback on learning material, quizzes, and overall SeisTutor support.

	Strongly satisfied	Satisfied	Neutral	Dissatisfied	Strongly dissatisfied
SeisTutor explained the content correctly	23	25	3	8	1
SeisTutor made the course as interesting as possible	31	19	9	1	0
The tutoring resources were adequate	21	19	7	9	4
The presentation of course content stimulated my interest during learning session	32	24	2	1	1
The course contents are relevant and well organized	29	25	3	2	1
SeisTutor supported me to understand the content which was found confusing?	27	26	6	1	0
Did the quiz at the end of each week correspond to the lessons taught?	28	27	3	1	1
The question-wise hints were helpful	27	26	6	0	1
Did the SeisTutor react decidedly to your necessities?	26	21	11	1	1
Was the learning provided sufficiently to take the quiz?	36	18	4	2	0
During ongoing tutoring, assessments are a fair test of my knowledge and learning preferences	32	21	5	2	0

Table 12: Learner feedback on learning material presentation and on overall SeisTutor look and feel.

Questions	Strongly satisfied	Satisfied	Neutral	Dissatisfied	Strongly dissatisfied
How satisfied are you with the look and feel (user interface design) of this system?	32	19	8	1	0
How satisfied are you with the account setup experience of this system?	31	21	5	3	0
How pleasing is the color scheme used in this system?	28	21	11	0	0
How user-friendly is this system? give a rating	27	21	8	2	2
SeisTutor compels and supports me to complete the quizzes and lessons	25	27	6	1	1
How satisfied are you with the organization/customization of contents feature of the system?	29	26	4	0	1

satisfied. Finally, students were happy with the account setup process with the system, which maintains the learners learning progress, grades, and basic account information.

The learner's overall evaluation of the SeisTutor showed that 82% of learners agreed that tutoring should begin based on the learner profile considering his/her learning style and prior knowledge. Most of the students were unaware of their learning style, and about 80% of students never knew about it. Most of the students liked the artificial intelligence features such as automatic selection of the tutoring strategies, dynamically assessing the learner attainment and flipping the tutoring plan or strategy.

The learner's feedback questionnaire responses were retrieved and analyzed in an accessible fashion. Some learners put their suggestions to improve the productivity of SeisTutor. Most of the suggestions were general and related to the improvement of the system, and few were pessimistic regarding improvement of the quality of learning contents, improving the quality of the video lessons, and hints provided by the system. At last, through the overall evaluation of SeisTutor, 87% of learners agreed that they improved their learning performance and outcomes.

5.7. Kirkpatrick Phase 4: Evaluation of Results. Evaluation of results illustrates the overall impact of learning on the learner. To quantify the effectiveness of learning, paired-wise sample *t*-test is performed on existing information, that is, pretest and posttest results of participants involved in both studies (study 1 and study 2). Here two cases are taken into consideration.

Case 1: A paired-sample *t*-test accomplished on study 1 incorporates the cognitive intelligence (see Table 1).

Hypothesis Case 1.0. Let us assume that the applicants involved in study 1 have homogenous mean scores in pretest and posttest.

Hypothesis Case 1.1. Let us assume that the applicants involved in study 1 did not have homogenous mean scores in pretest and posttest.

Case 2: A paired-sample *t*-test accomplished on study 2 incorporates the cognitive intelligence (see Table 1).

Hypothesis Case 2.0. Let us assume that the applicants involved in study 2 have homogenous mean scores in pretest and posttest.

Hypothesis Case 2.1. Let us assume that the applicants involved in study 2 did not have homogenous mean scores in pretest and posttest.

For study 1, the calculated T value (T_{Stats}, $p < 0.01$) is 11.410 (refer to Tables 13 and 14). The difference between posttest and pretest scores is 2.21786, and the calculated T_{Stats} value exceeded the T_{critical} value. Therefore, Hypothesis 1.0 is excluded. Tables 13 and 14 Indicate a considerable difference between pretest and posttest scores. For study 2, the calculated T value (T_{Stats}, $p < 0.01$) is 5.312 (refer to

Table 13: Statistical results of paired-sample t-test of study 1.

Comparison item	Learning mode			
	Mean	N	Std. deviation	Std. error of the mean
Posttest of study 1 applicants	3.9375	28	.39455	.07456
Pretest of study 1 applicants	1.7196	28	.99740	.18849

Table 14: Paired-sampled t-test results of study 1.

	Mean	Std. deviation	Std. error of the mean	95% confidence interval of the difference		T	df	Sig. (2-tailed)
				Lower	Upper			
Pair 1: posttest of study 1–pretest of study 1	2.21786	1.02856	.19438	1.81902	2.61669	11.410	27	.000

Table 15: Statistical results of paired-sample t-test of study 2.

Comparison item	Learning mode			
	Mean	N	Std. deviation	Std. error + Mean
Posttest of study 2 applicants	3.6525	32	0.58915	0.10415
Pretest of study 2 applicants	2.4053	32	1.39565	0.24672

Table 16: Paired-sampled t-test results of study 2.

	Mean	Std. deviation	Std. error of the mean	95% confidence interval of the difference		T	df	Sig. (2-tailed)
				Lower	Upper			
Pair 1: posttest of study 2–pretest of study 2	1.24719	1.32804	.23477	.76838	1.72600	5.312	31	.041

Tables 15 and 16). The difference between the posttest and pretest scores is 1.24719, and the calculated T_{Stats} value exceeded the T_{critical} value. Therefore, Hypothesis 2.0 is excluded. Tables 15 and 16 indicate a considerable difference between pretest and posttest scores. Both studies excluded the null hypothesis, implying that both provide useful training. On the other hand, this study determines which study has the most significant influence on improving learning gains. In order to deduce the conclusion, the objectives of both studies are compared. Study 1 achieved high scores compared to study 2. As a result, study 1 differs significantly from study 2 in posttutoring and pretest results, indicating that study 1 provides a more effective training program.

This analysis concludes that the intelligent incorporated SeisTutor used in study 1 outperforms SeisTutor used in study 2 in providing custom-tailored intended curriculum, identifying learner sentiments while learning, and computing the learner's overall degree of knowledge that meets the learner needs.

6. Conclusion

This article demonstrates the proposed personalized intelligent tutoring system, named SeisTutor. From the research, it has been noted that a learner receives repetitive learning content from learning, which indirectly disorients the learner. Thus, to address this issue, bug model has been utilized, which analyzes the bugs and recommends the custom-tailored curriculum to the learner. This technique helps to bring empathy to ITS. SeisTutor is not a passive tutor; it also analyzes the learner's behavior, that is, the psychological state of the learner during learning, which helps to understand the learner's experience with SeisTutor (about learning content). Experimental results reveal that SeisTutor utilized by participants in study 1 provides a customized learning sequence or path of learning material that endorses effective learning. Experimental analysis reveals the effective learning gain (when learner gets the custom-tailored sequenced learning material) of 44.34% compared to SeisTutor used in study 2 (not sequenced learning material). To evaluate the overall effectiveness of SeisTutor, Kirkpatrick's four-phase evaluation model is utilized. The analysis reveals that the participants involved in study 1 attain 44.4% while study 2 attains 24.8% and study 1 provides effective learning.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request (thippareddy.g@vit.ac.in).

Conflicts of Interest

The authors declare that they have no conflicts of interest to report regarding this study.

Authors' Contributions

Ninni Singh contributed to conceptualization, data curation, formal analysis, methodology, , and writing the original draft and provided the software; **Vinit Kumar Gunjan** contributed to supervision, reviewing and editing, project administration, and visualization; **Kadiyala Ramana** contributed to software, validation, writing the original draft, methodology, and supervision**;** **Qin Xin** contributed to visualization, investigation, and formal analysis, and provided software; **Thippa Reddy Gadekallu** contributed to data curation, investigation and provided resources and software.

References

[1] N. J. Ahuja and P. Diwan, "An expert system for seismic data interpretation using visual and analytical tools," *International Journal of Scientific Engineering and Research*, vol. 3, no. 4, pp. 1–13, 2012.

[2] K. Holzkamp, *Lernen: SubjektwissenschaftlicheGrundlegung*, Campus Verlag, Frankfurt, Germany, 1995.

[3] D. L. Kirkpatrick, "Techniques for evaluating training programs," *Training & Development Journal*, vol. 33, no. 6, pp. 78–92, 1979.

[4] J. Kirkpatrick and W. Kayser-Kirkpatrick, *The Kirkpatrick four levels: a fresh look after 55 years*, Ocean City: Kirkpatrick Partners, Newnan, Georgia, 2014.

[5] D. L. Kirkpatrick and J. D. Kirkpatrick, *Evaluating Training Programs: The Four Levels*, Berrett-Koehler Publication, San Francisco, SA, USA, 3rd edition, 2006.

[6] D. Kirkpatrick, *Evaluating Training Programs*, Berrett-Koehler, San Francisco, CA, USA, 1994.

[7] R. Bates, "A critical analysis of evaluation practice: the Kirkpatrick model and the principle of beneficence," *Evaluation and Program Planning*, vol. 27, no. 3, pp. 341–347, 2004.

[8] U. D. Ehlers and D. Schneckenberg, *Changing Cultures in Higher Education: Moving Ahead to Future Learning*, Springer, Heidelberg, Germany, 2010.

[9] A. Harden, G. Peersman, S. Oliver, M. Mauthner, and A. Oakley, "A systematic review of the effectiveness of health promotion interventions in the workplace," *Occupational Medicine*, vol. 49, no. 8, pp. 540–548, 1999.

[10] D. L. Kirkpatrick and J. D. Kirkpatrick, *The Transfer of Learning to Behavior: Using the Four Levels to Improve Performance*, BerrettKoehler Publication, San Francisco, CA, USA, 2005.

[11] D. L. Kirkpatrick and J. D. Kirkpatrick, *Implementing the Four Levels*, Berrett-Koehler Publication, San Francisco, CA, USA, 2007.

[12] M. S. Knowles, E. F. Holton, and R. A. Swanson, *The Adult Learner*, Gulf Publishing Company, Houston, TX, USA, 5th edition, 1998.

[13] G. M. McEvoy and P. F. Buller, "Five uneasy pieces in the training evaluation puzzle," *Training & Development Journal*, vol. 44, no. 8, pp. 39–42, 1990.

[14] J. J. Phillips, "A rational approach to evaluating training programs including calculating ROI," *The Journal of Lending and Credit Risk Management*, vol. 79, no. 11, pp. 43–50, 1997.

[15] G. J. Endres and B. H. Kleiner, "How to measure management training and development effectiveness," *Journal of European Industrial Training*, vol. 14, no. 9, pp. 3–7, 1990.

[16] T. C. Reeves and J. G. Hedberg, *Interactive Learning Systems Evaluation*, Educational Technology, Englewood Cliffs, NJ, USA, 2003.

[17] Y. Steinert, K. Mann, A. Centeno et al., "A systematic review of faculty development initiatives designed to improve teaching effectiveness in medical education: BEME Guide No. 8," *Medical Teacher*, vol. 28, no. 6, pp. 497–526, 2006.

[18] J. L. Fitzpatrick, J. R. Sanders, and B. R. Worthen, *Program Evaluation: Alternative Approaches and Practical Guidelines*, Allyn & Bacon, Boston, MA, USA, 3rd edition, 2004.

[19] B. Musal, C. Taskiran, Y. Gursel, S. Ozan, S. Timbil, and S. Velipasaoglu, "An example of program evaluation project in undergraduate medical education," *Education and Health*, vol. 21, no. 1, pp. 113–117, 2008.

[20] J. Morrison, "ABC of learning and teaching in medicine: Evaluation," *BMJ*, vol. 326, no. 7385, pp. 385–387, 2003.

[21] A. A. Malki, J. M. Al-Bareeq, and Al-Halili, "Evaluation of research writing workshop," *Bahrain Medical Bulletin*, vol. 25, no. 3, pp. 1–8, 2003.

[22] L. Praslova, "Adaptation of Kirkpatrick's four level model of training criteria to assessment of learning outcomes and program evaluation in Higher Education," *Educational Assessment, Evaluation and Accountability*, vol. 22, no. 3, pp. 215–225, 2010.

[23] J. Bitchener, S. Young, and D. Cameron, "The effect of different types of corrective feedback on ESL student writing," *Journal of Second Language Writing*, vol. 14, no. 3, pp. 191–205, 2005.

[24] N. Storch and J. Tapper, "The focus of teacher and student concerns in discipline-specific writing by university students," *Higher Education Research and Development*, vol. 19, no. 3, pp. 337–355, 2000.

[25] C. Polio, C. Fleck, and N. Leder, "If I only had more time:" ESL learners' changes in linguistic accuracy on essay," *Journal of Second Language Writing*, vol. 7, 1998.

[26] N. Elliot and A. Klobucar, "Automated essay evaluation and the teaching of writing," in *Handbook of Automated Essay Evaluation: Current Applications and New Directions*, M. D. Shermis& and J. Burstein, Eds., pp. 16–35, Routledge, New York, NY, USA, 2013.

[27] V. Aryadoust, P. Mehran, and M. Alizadeh, "Validating a computer-assisted language learning attitude instrument used in Iranian EFL context: an evidence-based approach," *Computer Assisted Language Learning*, vol. 29, no. 3, pp. 561–595, 2016.

[28] N. Singh, N. J. Ahuja, and A. Kumar, "A novel architecture for learner-centric curriculum sequencing in adaptive intelligent tutoring system," *Journal of Cases on Information Technology*, vol. 20, no. 3, pp. 1–20, 2018.

[29] N. Singh and N. J. Ahuja, "Bug model based intelligent recommender system with exclusive curriculum sequencing for learner-centric tutoring," *International Journal of Web-Based Learning and Teaching Technologies*, vol. 14, no. 4, pp. 1–25, 2019a.

[30] N. Singh and N. J. Ahuja, "Implementation and evaluation of intelligence incorporated tutoring system," *International Journal of Innovative Technology and Exploring Engineering*, vol. 8, no. 10C, 2019b.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 2060875, 14 pages
https://doi.org/10.1155/2022/2060875

Research Article

Multipath Transmission Control Protocol for Live Virtual Machine Migration in the Cloud Environment

Kadiyala Ramana,[1] **Rajanikanth Aluvalu**,[1] **Vinit Kumar Gunjan**,[2,3] **Ninni Singh**,[2] **and M. Nageswara Prasadhu**[4]

[1]*Department of Information Technology, Chaitanya Bharathi Institute of Technology, Hyderabad 500075, India*
[2]*Department of Computer Science and Engineering, CMR Institute of Technology, Hyderabad, India*
[3]*University of Science and Technology Chattogram, Bangladesh*
[4]*Department of Computer Science and Engineering, Annamacharya Institute of Technology and Sciences, Rajampet, Andhra Pradesh, India*

Correspondence should be addressed to Vinit Kumar Gunjan; vinit.gunjan@ustc.ac.bd

Received 11 March 2022; Accepted 2 April 2022; Published 22 April 2022

Academic Editor: Kuruva Lakshmanna

For mobile cloud computing (MCC), a local virtual machine- (VM-) based cloudlet is proposed to improve the performance of real-time resource-intensive mobile applications. When a mobile device (MD) discovers a cloudlet nearby, it takes some time to build up a virtual machine (VM) inside the cloudlet before data offloading from the MD to the VM can begin. Live virtual machine migration refers to the process of transferring a running Virtual Machine (VM) from one host to another without interrupting its state. Theoretically, live migration process must not render the instance being migrated unavailable during its execution. However, in practice, there is always a service downtime associated with the process. This paper focuses on addressing the need to reduce the service downtime in case of live VM migration in cloud and providing a solution by implementing and optimizing the multipath transmission control protocol (MPTCP) ability within an Infrastructure as a service (IaaS) cloud to increase the efficiency of live migration. We have also introduced an algorithm, the α-best fit algorithm, to optimize the usage of bandwidth and to effectively streamline the MPTCP performance.

1. Introduction

A virtual machine (VM) refers to an entity that does not exist physically, but is hosted on a hypervisor, and yet has all the capabilities of an equivalent physical machine [1]. This virtualization gives an added advantage in the form of the ability to migrate a VM from one host machine to another. Virtual machine migration may be done in the cloud for several reasons including load balancing and energy usage optimization. When a VM is used to handle data traffic within a cloud, a relatively freer host (destination host) can be chosen to migrate it to, as an attempt to relieve the load on the source host. Similarly, energy optimization within the cloud can be done by shutting off the nodes which do not host any occupied VMs [2].

In datacenter virtualization, live migration is a crucial technology and feature. With live migration, VMs can be transferred from one physical host to another with little to no impact on the availability of running applications. This ensures that running applications are not harmed by any physical server failures, significantly improving service availability. The TCP/IP protocol is used to send live migration communications via the Ethernet network that connects the cluster servers. The content that needs be migrated is mostly the CPU cache, memory, and buffers; however, the memory content takes up the majority of the time. The content of the CPU cache and buffers is almost insignificant in comparison to the RAM content, which is what most authors assume in live migration modeling.

In the early stages of its development, VM migration was static in nature. This is to say that the general practice was to suspend the VM state before the actual migration took place. This caused the instance to be unavailable throughout the migration process. The functioning of the VM was resumed after the entire VM data, that is, the memory pages and the processor state, was copied to the destination host. It is easily imaginable how much inconvenience this sort of a process can cause when implemented in cloud environment [3]. The users pay for the services they receive, and they expect those services to be available all the time. In such scenario, where a VM handling some or all of those services is rendered unavailable till the completion of the migration process, the user experience surely tends to degrade [4].

The solution was live VM migration. In this case, the VM is not unavailable for the entire duration of the migration but only a smaller fraction of it, which is termed as the service downtime. Theoretically, the machine continues to function as if nothing had occurred at the back end during the migration process. However, as shown by Sv¨ard et al. [5], running processes on the VM are interrupted even during its live migration. This is attributed to the service downtime. Therefore, even if the service downtime is small, to ensure good end user experience, there will always be a need to reduce it as much as possible.

When we talk about the functioning of an Infrastructure as a Service (IaaS) cloud, we automatically step into the realm of software-defined networking (SDN). SDN refers to the use of programmable software components that can be installed in a networking framework and still provide the same functionality as their physical hardware counterparts [6]. Modern clouds depend heavily on SDN to function. Though physical resources are required to host the cloud instances, the networking in the cloud combines elements which are both physical and software defined in nature. Without SDN, it would be nearly impossible to cope with the growing pressure on data networks as the number of mobile networking devices increase [7]. IaaS cloud platforms such as “OpenStack” use SDN to connect the instances to the physical networking resources. With SDN, networking becomes more manageable and less cumbersome. Moreover, we can add or remove elements from a software defined network according to our needs, which would otherwise require the installation and removal of physical networking devices.

In this paper, we have implemented multipath TCP, which is a recent project by the Internet Engineering Task Force (IETF) to facilitate faster data transfer within the cloud environment [8]. The VM data can be transferred with a higher speed between the host nodes in case of data centers if the effective bandwidth of the connection between them can be increased. This ability is provided by MPTCP by allowing us to use multiple paths for TCP data transfer at the same time. By the creation of TCP subflows, MPTCP makes use of multiple interfaces on the connected hosts for the transfer of data. Hence, theoretically, the aggregate bandwidth of each of the individual paths is available for data transfer. However, in reality, this is not the case. MPTCP requires some optimization techniques to be implemented effectively in the cloud. We introduce an algorithm, the α-best fit algorithm, for the optimization of MPTCP usage, and compare it to the subflow optimization offered by Joshi and Kataoka [9]. This algorithm provides a much better optimization scheme than the one already proposed [9] in terms of bandwidth usage. The generation of optimum number of subflows in [9] leads to wastage of bandwidth by generating suboptimal results, while the α-best fit algorithm minimizes this wastage, which is crucial in the cloud scenario where bandwidth resources to be dealt with are massive in scale.

There are three forms of live VM migration: precopy, postcopy, and hybrid-copy. The most resilient migration type is precopy live migration, which we focus on in this study. The source host memory content copy begins and continues to transfer until it reaches a halting circumstance in precopy migration. The VM will then stop on the source host and start on the target host. This is the most popular migration method used by commercial and open source hypervisors since it is the most dependable.

The remaining part of this paper is organized as follows: Section 2 explains the conventional algorithm that is in use for conducting live migrations, Section 3 explains MPTCP, and its effects on live migration of cloud instances, Section 4 explains the need for optimization of MPTCP implementation in cloud environments and puts forward our proposed approach to do so, Section 5 contains the details of implementation of MPTCP in the cloud environment, Section 6 embodies the results obtained, and Section 7 includes the discussion on the results and the conclusion.

2. Related Work

Multipath TCP is a TCP enhancement that allows for simultaneous transmission of data through many pathways from one end to the other. For example, multipath TCP allows many applications to transport and receive data over multiple interfaces, such as cellular and Wi-Fi, by creating a single TCP sub flow for each interface [10]. This subflow is used by the MPTCP scheduler to transmit and receive data. MPTCP's design has various advantages, including resource utilization, throughput, and a mild response to faults, as well as good route performance.

If numerous sub flows are available, a scheduler [11] selects the subflow with the least round-trip time (RTT) to transfer the data. During the transmission of the segment, the scheduler chooses the way with the shortest round-trip time among all subflows whose congestion window is not yet full. If there are multiple such paths, the scheduler creates a bias towards one of them. It also continues to deliver information on the particular subflow until the congestion window for that sub flow is full. In prior research, MPTCP was examined in a mobile situation. The authors investigated the impact of mobility on MPTCP in [12], and in [13], they offered various MPTCP options for use by cell phones for Wi-Fi handover. However, neither of the works investigates the heterogenetic nature of the path in lossy subflows. The authors of [14] gave a comparison of single TCP vs. multipath TCP. The authors of [15] used various scheduler algorithms to calculate the impact of scheduler architecture on performance. In [16, 17], different congestion mechanism techniques are compared.

The authors proposed scheduler algorithms in [18], which select subflows based on an estimate of the amount of traffic they can handle before becoming congested. By considering large network transfers and a restricted buffering quantity, the methodology demonstrates. Another unique scheduling policy offered by the authors in [19] was to keep a strategic distance from out-of-request parts. The authors do not explain why a segment is removed from the TCP buffer after it is transmitted by another subflow. The authors presented a delay-aware packet scheduler technique on the ns-2 simulator platform in [20]. In both delay and stable settings, the proposed technique analyses path heterogeneity. The performance of MPTCP over a mobile network was examined by the authors in [21]. The analysis took into account a variety of subflows as well as precise statistics such as round-trip time (RTT) and out-of-order delivery, but the authors did not account for lossy subflows. Lim et al. proposed a new scheduler method that sends buffers by monitoring available bandwidth on each subflow, but it does not take use of the data loss rate on each sub flow.

3. The Precopy Algorithm

This is the conventional method for carrying out live VM migrations. In this method, the memory pages of the VM are sent before its processor state to the destination node. The transfer of the memory pages occurs in several iterations. These iterations continue till a set number is reached, or a small enough writable working set (WWS) is obtained [22]. In each iteration, there are certain memory pages which are modified during the transmission, and these are called dirty pages. During each iteration, there is a possibility of a memory page being dirtied, and these dirty pages are transferred to the target node in the next iteration.

The duration of service downtime starts when either a WWS is obtained or the cap on the number of iterations is reached [23]. If either of the above conditions are satisfied, the VM at the source node is suspended, and all the remaining memory pages along with the dirty pages from the previous iteration are moved to the destination node. After this transfer, the VM is resumed at the destination node, and its source node copy is destroyed. Hence, it is obvious that the service downtime depends upon the number of memory pages to be sent, which also includes the dirty pages from the previous iteration [24].

The time taken to transfer the entire RAM is given by [22]

$$t_0 = \frac{V}{B}, \tag{1}$$

where V is the total memory size of the VM, and B is the available bandwidth.

We must also consider the page dirtying rate to get a measure of the time required for their transfer in each iteration during precopy. This time required for the transfer of the remaining memory pages, which also includes the pages dirtied in each iteration, is given by [22]

$$t_1 = \frac{R \times t_0}{B}, t_2 = \frac{R \times t_1}{B}, t_k = \frac{R \times t_{k-1}}{B}, \tag{2}$$

where R is the memory dirtying rate, and k is the total number of iterations.

The precopy process gets suspended when the k -$^{\text{th}}$ iteration occurs, and the stop conditions are satisfied. Following this, the VM at the source is suspended, and the remaining dirty pages, along with the processor state of the VM, are transferred to the destination node [22]. This takes about $t_k = R \times t_{k-1}/B$ amount of time.

Hence, the total time for the precopy and stop and copy is calculated as

$$T_0 = t_0 + t_1 + t_2 + \cdots + t_k = \frac{V}{B} \times \frac{1 - (R/B)^{k+1}}{1 - (R/B)}. \tag{3}$$

Therefore, the total migration time is given by

$$T_{\text{mig}} = \text{Pr}\, e - \text{migration overhead} + T_o + \text{Post-migration overhead}, \tag{4}$$

where the premigration overhead refers to the time required for the resource reservation, and the postmigration overhead consists of the commitment and activation stage of the precopy [22]. While these overheads are largely unavoidable, the value of T_o can be varied.

4. Multipath TCP (MPTCP)

MPTCP is a recent project by the Internet Engineering Task Force (IETF). This project enables data transfer via multiple TCP subflows that are sent over different TCP paths [8]. This capability is of great value if we consider its implementation in data centers. Multiple paths to send data imply more bandwidth for data transfer. In principle, this bandwidth should be equal to the aggregate of the bandwidth capacities of the individual paths [25].

As we can see in Figure 1, data is divided into chunks, and these chunks are transferred through different subflows over different paths. This means that the client and the server in a simple MPTCP connection have multiple IP addresses, which may or may not be in the same network [26]. In case of data centers, MPTCP can therefore be utilized to obtain the ability of fast data transfer; since, at any given time, more than one paths can be used simultaneously as opposed to TCP, which can be seen in Figure 2.

Now, consider the scenario in which the implementation of the precopy algorithm described in Section 2 in combination with the MPTCP capability is being carried out.

If we consider k number of iterations to occur for the transfer of dirty pages in case of a single active path for data transfer, as was the case in equations (2) and (3), then we can see that for an iteration i, the pages dirtied will be $R \times i$, and these will be transmitted in the $(i+1)$-$^{\text{th}}$ iteration.

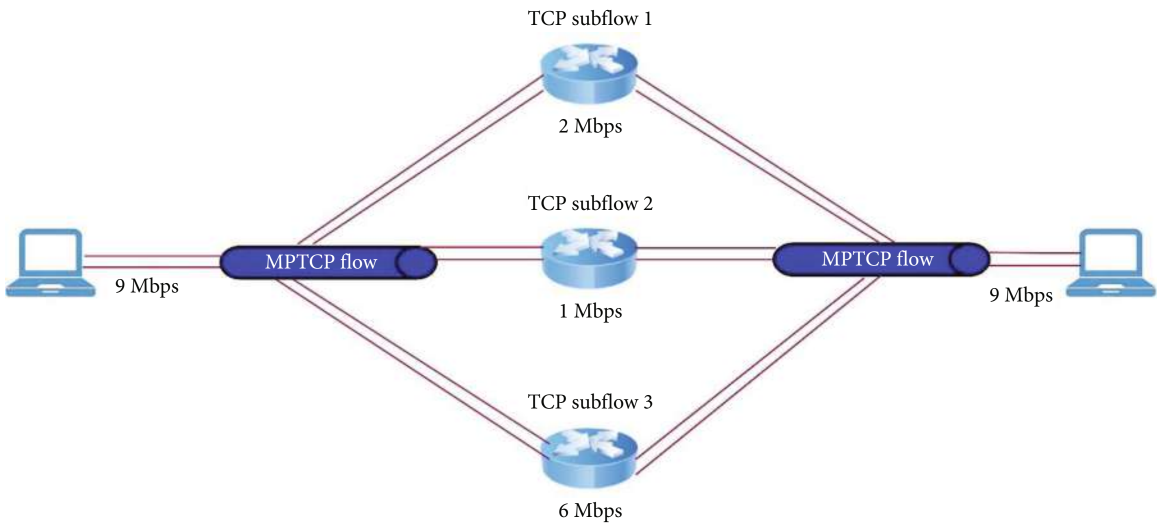


Figure 1: Division of MPTCP flow into TCP subflows.

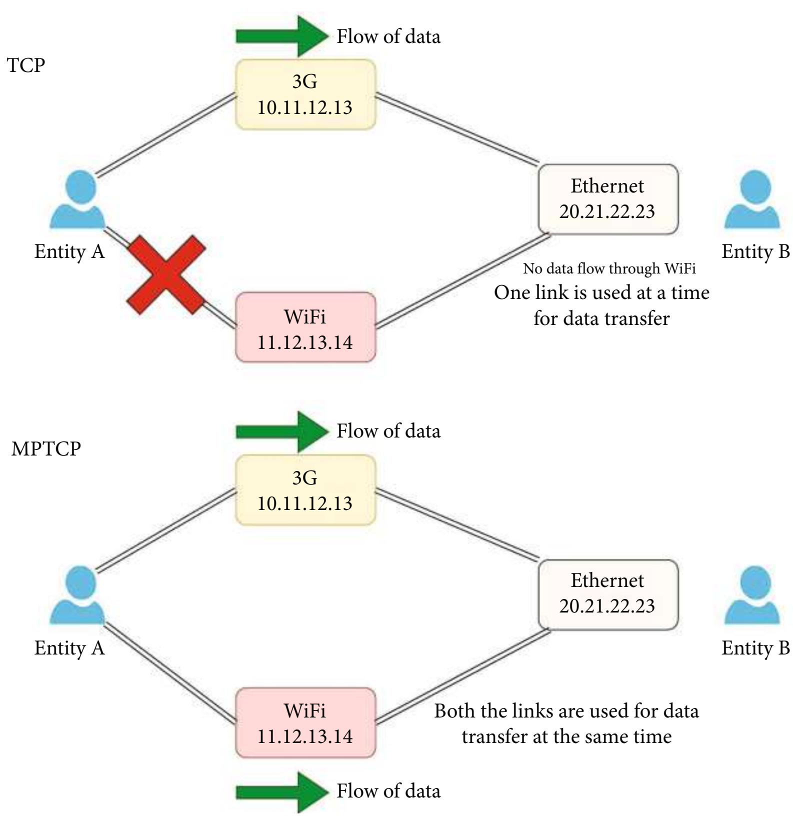


Figure 2: Comparison of MPTCP session with normal TCP session.

As stated by Nathan et al. [27], the page dirtying rate for the i-$^{\text{th}}$ iteration is given by the expression:

$$R = \frac{\text{number of pages dirtied in iteration } (\mathrm{i}-1) \times \text{page size}}{\text{time taken by iteration } (\mathrm{i}-1)}. \tag{5}$$

Now, the time taken for an iteration is dependent upon the bandwidth available, which is evident from equation (2). Hence, an increased bandwidth will reduce the page dirtying rate [28]. This implies that less pages will be dirtied in each iteration with the increase in effective bandwidth, and hence the time required for these iterations will decrease. Considering the page dirtying rate to be constant for a migration, then the advantage of added bandwidth is expected to reduce the time $t_{(i+1)} = R \times t_i / B'$, where B' is the new bandwidth.

This implies that though the iterations will require lesser time than in the case of a reduced bandwidth capacity, the actual number of iterations, k will remain the same.

This implies that the advantage MPTCP will provide due to increased aggregate bandwidth will be in the form of reduced transmission time rather than reduced number of iterations before the WWS is achieved. Also, as stated in [22], the time required after k iterations to transfer the WWS and the processor state is given by

$$t_k = \frac{R \times t_{k-1}}{B} \tag{6}$$

This time period required for the stop-and-copy phase will be reduced by an increase in bandwidth. Therefore, a decrease in service downtime is expected.

Assuming that we have an MPTCP connectivity between two host machines with x number of active paths for data transfer, let γ be the segment-wise bandwidth utilization factor, which is explained in the next section.

Hence, the aggregate effective bandwidth of the MPTCP connection will be $B' = x \times \gamma \times B$, where B is the bandwidth of each individual path in the MPTCP connection.

Using this new bandwidth, equations in Section 2 can be written for the MPTCP scenario as

$$t'_0 = \frac{V}{B'}, \tag{7}$$

$$t'_1 = \frac{R \times t'_0}{B'}, t_2 = \frac{R \times t'_1}{B'}, t'_k = \frac{R \times t'_{k-1}}{B'}, \tag{8}$$

$$T'_0 = t'_0 + t'_1 + t'_2 + \cdots + t'_k = \frac{V}{B'} \times \frac{1 - \left(R/B'\right)^{k+1}}{1 - \left(R/B'\right)}. \tag{9}$$

Using $B' = x \times \gamma \times B$ in equations 7and 8, it can be stated that for an iteration i,

$$t'_i = \frac{t_i}{(x\gamma)^{i+1}}. \tag{10}$$

From equation (10), it can be inferred that with the increasing number of iterations, the time required for transmission of data will decrease progressively in case of MPTCP connectivity between the hosts. When the value of "i" in equation (10) becomes equal to k, then the time period t'_k represents the time spent in the stop-and-copy phase [22], which is synonymous to the service downtime. The value x will always be positive and greater than one for an MPTCP connection, while γ will be a positive fraction representing the degree of utilization of the bandwidth resources.

On the other hand, equation (10) also provides the information that in case of TCP connection between the hosts, when $x = 1$, γ is still less than 1. This means that with each successive iteration, with increasing powers of γ, the performance of precopy for transmission of dirty pages will degrade.

Therefore, while TCP connectivity causes the performance to worsen with every iteration, MPTCP connectivity actually increases it with the increase in number of iterations till a set number of iterations, which will depend upon the product $(x\gamma)$. As long as the quantity $(x\gamma)$ remains greater than one, the performance will increase with increasing number iterations.

Rearranging equation 6 is as follows:

$$T'_0 = \sum_{i=0}^{k} \frac{t_i}{(x\gamma)^{i+1}}. \tag{11}$$

Therefore,

$$(x\gamma)^{k+1} T'_0 = \sum_{i=0}^{k} t_i (x\gamma)^{k-i}. \tag{12}$$

Dividing equation (9) by equation (3), we get

$$\frac{T'_0}{T_0} = \frac{1}{(x\gamma)^{k+1}} \times \sum_{i=0}^{k} (x\gamma)^{k-i} \tag{13}$$

Hence, it can be seen that the ratio T'_0/T_0 will always be less than one if the product $x\gamma > 1$. Therefore, the total migration time will also decrease with increasing number of active paths for data transfer.

This deduction implies that by increasing the number of active paths in an MPTCP connection, there will be a reduction in the service downtime to a negligible nonzero amount. However, this is not the case, and thus the MPTCP connection requires optimization for the reason that with increasing number of paths in an MPTCP connection, there is an increased possibility of network congestion at the points where the subflows reunite. This congestion lowers the value of the factor $(x\gamma)$, as the bandwidth utilization (γ) is drastically reduced in case of congestion.

Focusing on the value of γ, its relation to x and T_0 can be predicted to some extent using equation (11). Partially differentiating equation (11) with respect to γ,

$$\frac{\partial T'_0}{\partial \gamma} = \sum_{i=0}^{k} \frac{\partial}{\partial \gamma} \left(\frac{t_i}{(x\gamma)^{i+1}} \right) = \frac{-t_0}{x\gamma^2} - \frac{2t_1}{x^2\gamma^3} - \cdots - \frac{(k+1)t_k}{x^{k+1}\gamma^{k+2}}. \tag{14}$$

In terms of t_0, we have

$$\frac{\partial T'_0}{\partial \gamma} = \frac{t_0}{x\gamma^2} \left[-1 - \frac{2R}{Bx\gamma} - \cdots - \frac{(k+1)R^k}{B^k x^k \gamma^k} \right]. \tag{15}$$

This series can be converged, and the resulting equation is as follows:

$$\frac{\partial T_0'}{\partial \gamma} = \frac{-t_0}{x\gamma^2} \times \left[\frac{B^{-k}\gamma^{-k}R^{k+1}x^{-k}(k+1)R - B\gamma x(k+2) - B^2x^2\gamma^2}{(B\gamma x - R)^2}\right] \tag{16}$$

Figuring out the value of γ at which the value of T_o will be maximum or minimum can be achieved by finding the critical points by equating the partial differential in equation (16) to zero.

Substituting $\partial T_0'/\partial\gamma = 0$, equation (16) can be rewritten as

$$\left[\frac{B^{-k}\gamma^{-k}R^{k+1}x^{-k}(k+1)R - B\gamma x(k+2) - B^2x^2\gamma^2}{(B\gamma x - R)^2}\right] = 0. \tag{17}$$

$$\text{As} \frac{-t_0}{x\gamma^2} \neq 0. \tag{18}$$

Now, Let

$$\text{C1} = B^{-k}x^{-k}R^{k+2}(k+1), \tag{19}$$

$$\text{C2} = B^{1-k}x^{1-k}R^{k+1}, \tag{20}$$

$$\begin{aligned} \text{C3} &= B^2x^2. \\ C4 &= C2\,(k-2). \end{aligned} \tag{21}$$

Substituting the above values in equation (17),

$$\frac{C_1}{\gamma^k} - \frac{C_4}{\gamma^{k-1}} - C_3 = 0 \Rightarrow C_1 - C_4\gamma - C_3\gamma^k = 0. \tag{22}$$

Equation (22) is a polynomial of order k. Here, the value of γ can be represented by the series:

$$\begin{aligned} \gamma = \sum_{m=0}^{\infty}(-1)^m \frac{C_1^{mk-m+1}C_3^m}{C_4^{mk+1}m!}\prod_{j=0}^{m-2}(mk-j) = \frac{C_1}{C_4} - \frac{C_1^kC_3}{C_4^{k+1}} + \frac{2kC_1^{2k-1}C_3^2}{2!C_4^{2k+1}} \\ - \frac{3k(3k-1)C_1^{3k-2}C_3^3}{3!C_4^{3k+1}} + \frac{4k(4k-1)(4k-2)C_1^{4k-3}C_3^4}{4!C_4^{4k+1}} - \cdots \end{aligned} \tag{23}$$

This series will converge if the following condition is satisfied:

$$\left|\frac{C_1^{k-1}C_3}{C_4^k}\right| < \frac{(k-1)^{k-1}}{k^k}, \tag{24}$$

which translates to

$$\left|\left(\frac{Bx}{R}\right)^2 \times \frac{(k+1)^{k-1}}{(k-2)}\right| < \frac{(k-1)^{k-1}}{k^k}. \tag{25}$$

For this inequality, it can be clearly seen that $k \notin [0\,;2]$, because for $k \in [0\,;2]$, the inequality becomes meaningless. Therefore, $k \geq 3$ which is justified as practically the number of iterations is always higher than two. Hence, the series in equation (23) cannot be converged.

Now, the constraint on the value of γ is that $\gamma \in (0;\,1)$ If $C_1, C_3, C_4 > 0$ and $C_1 < C_3 + C_4$, there can be a unique value of γ, which will be the critical point for the equation (11). This can be accomplished by taking a sufficient number of terms so that we get a good approximation of the sum of the series in equation (23).

The page dirtying rate, as defined by Nathan et al. [27], is lesser than the value of (Bx), which is quite reasonable. Also, since k and x are positive integers, and $\gamma \in (0;\,1)$.

Therefore, using the values of C_1, C_2, C_3, and C_4 in equation (25), it can be seen that

$$\gamma = \frac{R}{Bx} \times \frac{k+1}{k-2}. \tag{26}$$

The higher terms in the series given by equation (23) are not taken into consideration, because of the fact that for $k \in (3,\infty)$, the value of higher terms will become negligible. At greater values of k, the power of the terms in the denominators of the terms in equation (23) will increase steeply. The first term yields a number that satisfies the condition of $\gamma \in (0,\,1)$, while inclusion of the higher terms does not affect its value significantly. This concludes that the value of k, given our constraints is as in equation ((26)).

Now,

$$\frac{\partial^2 T_0'}{\partial\gamma^2} = \frac{\partial^2}{\partial\gamma^2}\sum_{i=0}^{k}\left[\frac{t_i}{(x\gamma)^{i+1}}\right] = \sum_{i=0}^{k}\frac{\partial}{\partial\gamma}\left[\frac{-t_i(i+1)}{x^{i+1}\gamma^{i+2}}\right] = \sum_{i=0}^{k}\frac{t_i(i+1)(i+2)}{x^{i+1}\gamma^{i+3}}. \tag{27}$$

As all the variables in equation (27) are positive, and the critical point given by equation (26) is also a positive value, it can be inferred that the critical point is the point of maxima for the function T_0' with respect to γ with x and k constant.

Thus, it is explicit that for the value of γ given in equation (26), the migration time T_0' will be maximum. This proves that for MPTCP to be advantageous at a given number of active data paths between the hosts, the value of γ must be greater than the expression given by equation (26).

5. Optimizing MPTCP Capability

The MPTCP capability offers higher bandwidth for data transfer in clouds and data centers. In theory, by increasing the number of paths in an MPTCP session between two hosts, there should be significant improvement in the data transfer speed as an effect of increased effective bandwidth. One must therefore expect that the number of paths can be increased without any limit to obtain a nearly infinite bandwidth capacity for data transfer. However, it is not the case. While it is true that the increase in the number of paths leads to increased number of TCP subflows in an MPTCP session, their optimum usage is something that depends upon other factors.

As explained by Joshi and Kataoka [9], overutilization as well as underutilization of the MPTCP capability by creating fewer or a greater number of subflows than required leads to poor performance in terms of data transfer. [9] already provides an optimization algorithm which has been tested and implemented to provide an improved result than using MPTCP without any subflow optimization in terms of throughput.

The SFO given in [9] is modified to obtain the α-best fit algorithm, which is more suited for the cloud environment. Moreover, as the cloud environment deals with bandwidth resources that are in order of Gbps, the optimization of the SFO [9] for improved performance in terms of bandwidth utilization is needed.

The MPTCP subflow optimization as elaborated in [9] consists of three functions, the first function responsible for finding the feasible paths for the MPTCP session between the hosts, and the second function is responsible for calculating the optimum number of sub flows, while the third function deals with the assignment of paths to the MPTCP subflows.

SFO [9] uses the modified k-shortest path algorithm in order to calculate the set of shortest paths for data transfer between the hosts. While this approach suffices in the experimental topology considered in [9], it might lead to suboptimal bandwidth utilization in case of other available topologies for data centers and cloud environment. Hence, the first optimization we propose is to modify the path finder method to function not on the shortest k-path algorithm, but on maximum bandwidth algorithm, which gives priority to the path with the highest bandwidth capacity during selection.

The function responsible for the calculation of optimum number of sub flows has been modified by us to include the aspects of protection against path failure and reliability in performance along with optimization of the selection process.

5.1. The α–Best Fit Algorithm. Before discussing the α-best fit algorithm for the optimization of MPTCP subflows, it is essential to consider a factor that ought to be useful while we deal with substantial amount of bandwidth capacities, as is in the case of data centers. This factor is the segment-wise bandwidth utilization ratio (γ), including this factor will lead to a better reliability of the system, as it gives us a more realistic and practical view of the bandwidth resources. The inclusion of γ increases the accuracy of the algorithm while functioning in real time conditions.

This factor is calculated by the standard method, which involves using SNMP objects to know what fraction of the total bandwidth of a link is being considered. The bandwidth of a path is limited by the capacity of the minimum bandwidth link in its constituting links. This means that for a path consisting of n links, $\lambda_P = [\min(\lambda_i, \forall_i \in P)]_{i=1}^{n}$. Therefore, γ is calculated segment-wise.

The value of γ can be calculated as [29]

$$\gamma = \frac{\max(\Delta\text{IfInOctets},\Delta\text{IfOutOctets}) \times 8}{\Delta\text{time period between the polls} \times \text{IfSpeed}}, \quad (28)$$

where IfInOctets and IfOutOctets refer to the ingress and egress SNMP objects on a TCP link which are sent repeatedly after a fixed time interval. IfSpeed refers to the interface speed as stated by the specifications of the link.

Let S be the total size of the VM data to be transferred during the MPTCP session, τ be the time limit in which the live VM migration should be complete, α^d be the demand of bandwidth resources for the transfer of the VM, ONS be the optimum number of subflows, OP be the set of all the optimized paths available to put the sub flows on, FP be the set of all the feasible paths, D be the set of discarded paths, and λ be the bandwidth capacity.

The subflow optimization [9] provides a method to divide the MPTCP traffic into a suitable number of sub flows; however, they do not clarify the ramifications of the case when a particular path fails in the MPTCP connection. The comparison between α-best fit algorithm and subflow optimization algorithm [9] is represented in Table 1.

In order to address the scenario, in which a path moves to a "down" state, an additional clause has been devised in the α-best fit algorithm. This part of the algorithm makes use of a function named failure protection, which uses the set of all feasible paths (K) and the bandwidth capacity of the failed path to decide which path to redirect the subflow, which was initially on the now failed path, or to divide the subflow further into smaller subflows that are redirected onto other available paths in the set K. It is ensured that the algorithm functions in a way that causes minimum wastage of bandwidth owing to the importance and the order of magnitude of the bandwidth resources in the cloud environment.

6. Implementation

The implementation MPTCP on an OpenStack Mitaka IaaS cloud consisting of four nodes–one controller node, two compute nodes, and a block storage node, was done. The networking inside the cloud makes use of the OpenStack Neutron server residing on the controller node, and ML2 is used as the core plugin with Linux Bridge agent as the mechanism driver. The controller node has 16 GB RAM, 250 GB SDD, two network interfaces, while the compute nodes have 32 GB of RAM, 150 GB SDD, and four network interfaces each. The block storage node has 32 GB of RAM, 150 GB SDD on which the Operating System is hosted, and a 500 GB SDD to provide block storage resources to the OpenStack instances. The Operating System installed on all the four nodes is Ubuntu 14.04 LTS. Figure 3 shows the host networking implemented for deploying Openstack with MPTCP capability.

In order to measure the service downtime, instances are created, and they are pinged continuously from the controller node while the migration process from one compute host to another occurs. The time for which the instance remains unavailable is observed as the service downtime. A packet sniffer (Wireshark) is used for this purpose.

The service downtime is measured without using MPTCP (normal TCP connection) and then using MPTCP using two and then three paths between the hosts, respectively. Comparisons are made using the obtained results.

In order to test the proposed algorithm for optimization of MPTCP connection, we compare the results of the algorithm given in [9] to α-best fit algorithm's performance on randomly generated bandwidth values for a Multi Path TCP connection.

Table 1: Comparison between α-best fit algorithm and subflow optimization algorithm [9].

The α–best fit algorithm	Subflow optimization [9]
The maximum bandwidth path is given priority while selection of feasible paths.	The shortest path is chosen while deciding the set of feasible paths.
Discarded paths are stored as they can be used later to make maximum use of the available bandwidth.	Paths once discarded are never used again.
Either the path with bandwidth closest to the demand is used, or the subflow is divided into smaller subflows.	If the ONS is one, it might lead to the selection of a path with far greater bandwidth than the demand.
Protection against failure of a path is taken into account.	Protection against failure is not considered.
It is more reliable to function correctly in real time environment as bandwidth utilization factor is taken into account.	Bandwidth utilization factor is not considered, hence, its reliability in real time environment is limited.

1. Compute α^{d}=s/τ
2. Initialise D=ϕ; ONS=0; OP=ϕ; FP←K; Sort K in ascending order of bandwidths
3. Compute $\lambda_P \forall P \in K$
$\lambda_P = \gamma[\lambda_P]_{(i=1)}^{|FP|}$
4. Compute the total capacity of all feasible paths as
$\propto = \sum_{P=1}^{(|FP|)} \lambda_P$
5. if $\alpha^d < \alpha$ **then**
6. FP←FP-$P_{min,}$ Where P_{min} is the path ∈ FP with the minimum bandwidth
7. $D \longleftarrow D \cup P_{\min}$
8. **Compute the new bandwidth capacity**
$\alpha^s = \sum_{P=1}^{(|FP|)} \lambda_P$
9. **if** $\alpha^d < \alpha^s$ **then**
10. $\alpha \longleftarrow \alpha^s$
11. **Repeat** 5
12. while i=1 to |D **do**
13. $FP \longleftarrow FP \cup D_i$
14. $\alpha^{s'} = \sum_{P=1}^{(|FP|)} \lambda_P$
15. if $\alpha^d \geq \alpha^{s'}$ **then**
16. FP←FP-D_i
17. i++18. OP←FP, ON=|OP|
19. if ONS=1 & $\lambda_P \neq \alpha^d$ **then**
20. K←ϕ, K←D, D←ϕ
21. **goto 3**
22. return OP & ONS
23. if $[OP_i]_i^{|OP|}$ is down **then**, Where OP is the set of Optimal Paths returned from 22
24. Invoke failure_protection(K,λ(OP_i)), OP←OP-OP_i+OP_{index}, where index is a set returned by the function failure_protection
25. The function failure_protection can be elaborated as:
26. Function FAILURE_PROTECTION(K,λ(OP_i))
27. Initpos=0, K'←ϕ, index←ϕ, C=0
28. for j=1 to |K| **do**
29. if K_j=λ(OP_i) **then**
30. pos←j
31. $K' \longleftarrow K' \cup [P_k]_{k=1}^{j} : P_k \in K$
32. C←possible number of combinations in K' such that $\sum_{k=1}^{K'} = \lambda(OP_i)$
33. if C=0 **then**
34. index ←pos+1
35. else
36. Choose the combination with minimum number of paths in K'
37. index←positions of constituting in K'
38. return index

Algorithm 1: The α-best fit algorithm.

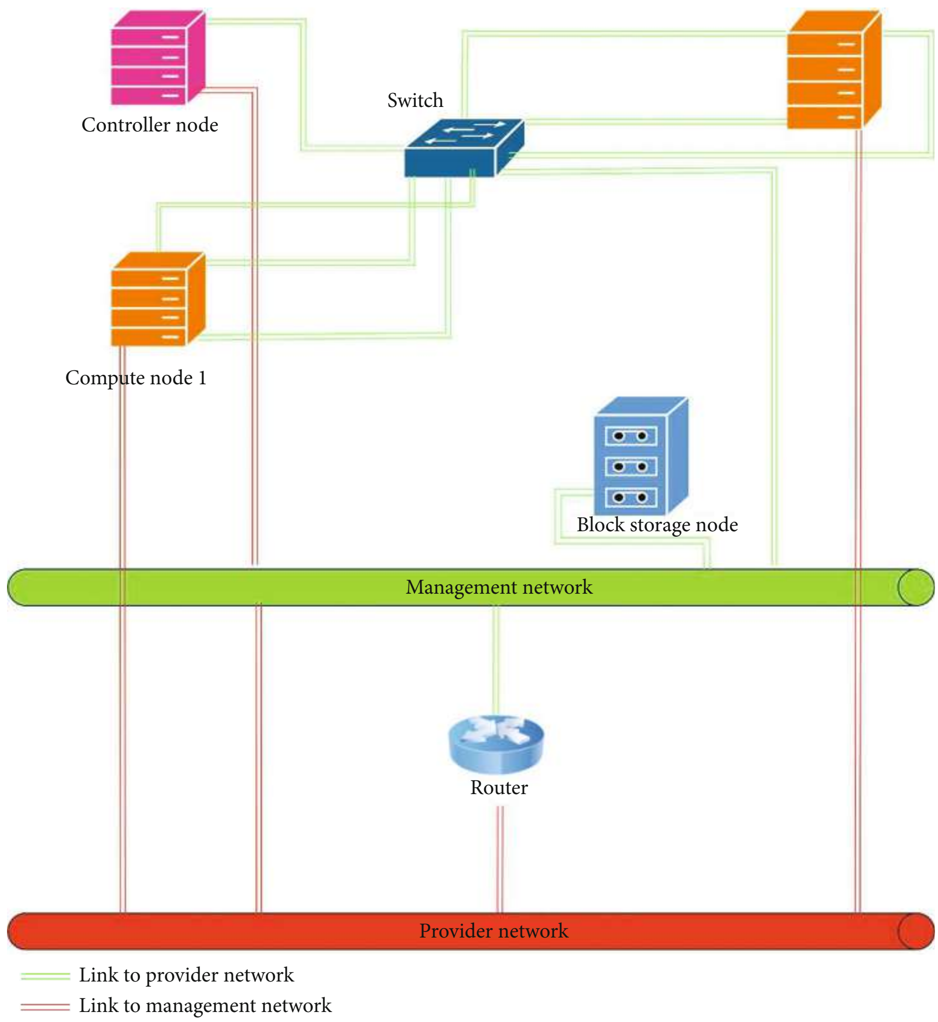


Figure 3: Host networking layout for implementation.

7. Results and Discussion

During the migration of an instance, the service downtime is detected using a packet sniffer (Wireshark). The instances are pinged every millisecond, and the time of their unavailability is measured in terms of the packets not received by them during the duration of the migration process.

It can be inferred from Figure 4 that when the migration of a CentOS 7 instance with 16 GB RAM is conducted, we have a drop for a significant amount of time in the number of ICMP packets transmitted to the OpenStack instance. This dip is shown graphically in the figure and pertains to the service downtime.

We have also measured the service downtime for 50 migrations of “Cirros” instances with different RAM specifications, and a graph displaying avg downtime vs. RAM of the instance being migrated is shown in Figure 5.

The results in Figure 5 show that with increasing RAM size of the instance, the service downtime is increased. This can be justified by considering that with the increase in RAM, the number of memory pages to be transferred in each iteration of the precopy algorithm increases as per equation (1). This leads to higher number of pages dirtied in every iteration. When the WWS is obtained, or the cap on the number of iterations is reached, with increasing RAM, the amount of data to be transferred increases as seen from equations (2) and (3). Hence, there is an increase in the service downtime.

7.1. Results of MPTCP Implementation without Optimization. After the measurement of service downtime in a single path TCP scenario, MPTCP capability is enabled, and the same procedure is carried out using two paths for the MPTCP communication between the compute hosts. The concept for performing this experiment is that with increased number of paths, the VM data will be divided into smaller streams/flows of data and transferred from both the paths at the same time, speeding up the data transfer. The results obtained provide the proof that the assumption made earlier is correct. While the average service downtime with a single path connectivity for data transfer between the compute hosts is 1.12 seconds, by increasing just one path, it is reduced to 0.58 seconds. Hence, it is observed that the service downtime has been reduced by 48% of the original value.

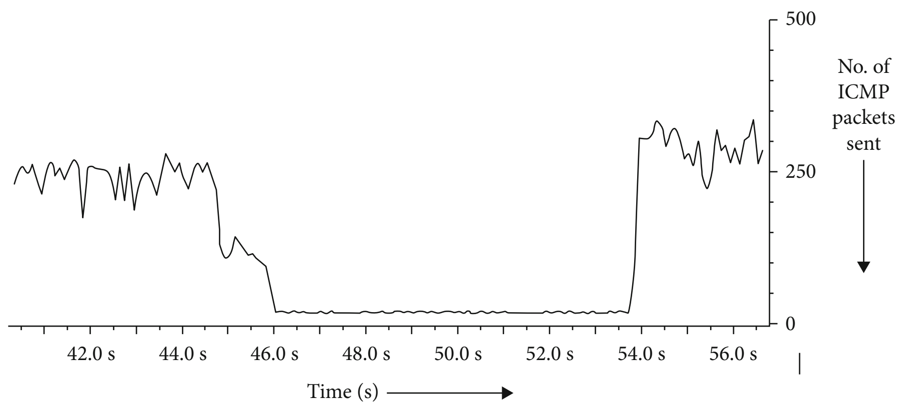


FIGURE 4: Depiction of service downtime in terms of ICMP packets sent during the migration process of an instance.

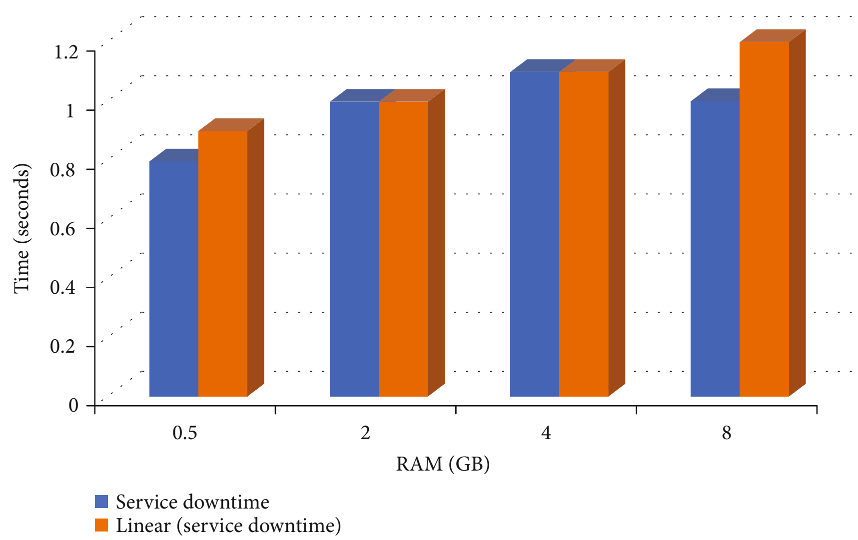


FIGURE 5: The variation of average service downtime with RAM for a "Cirros" instance.

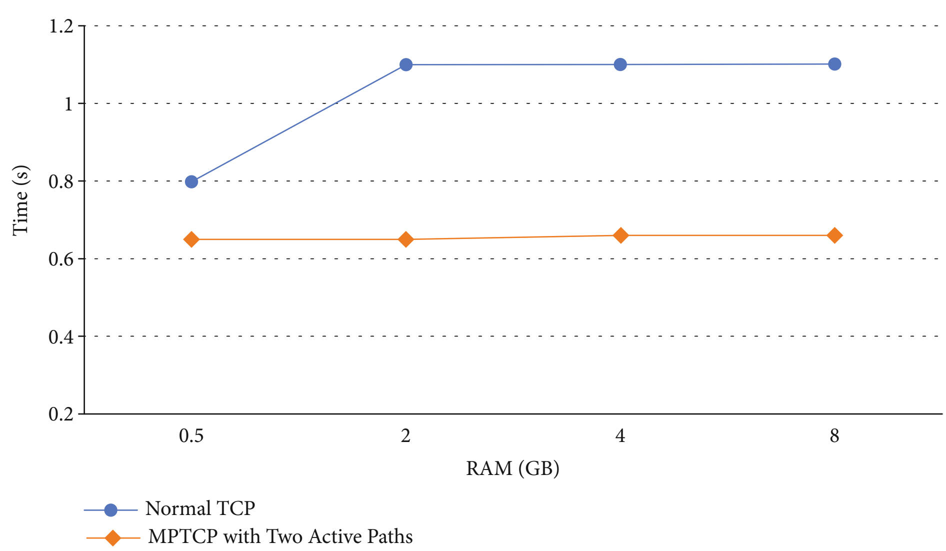
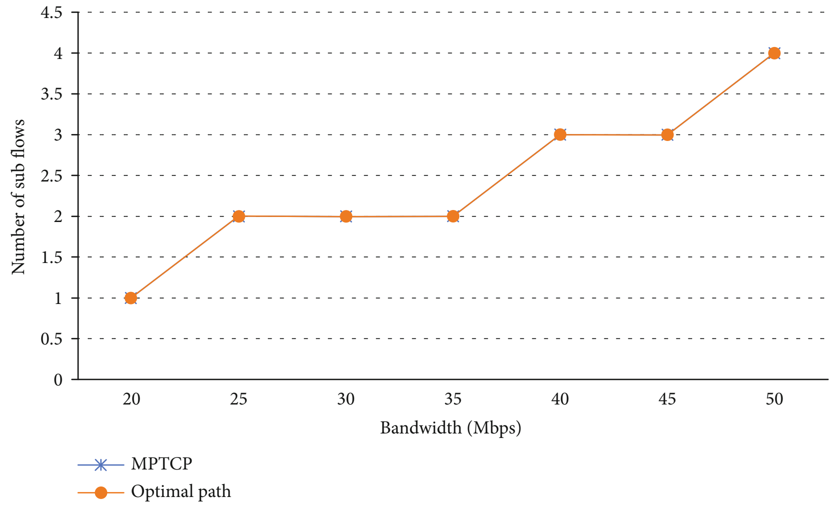
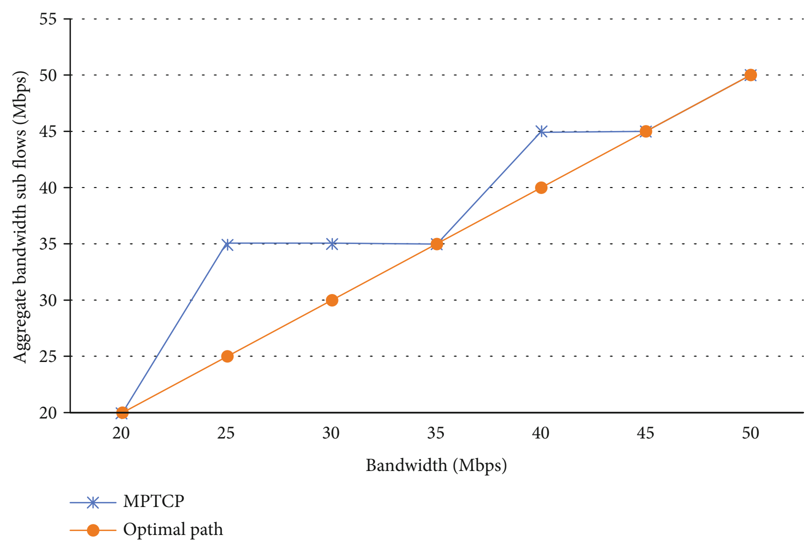


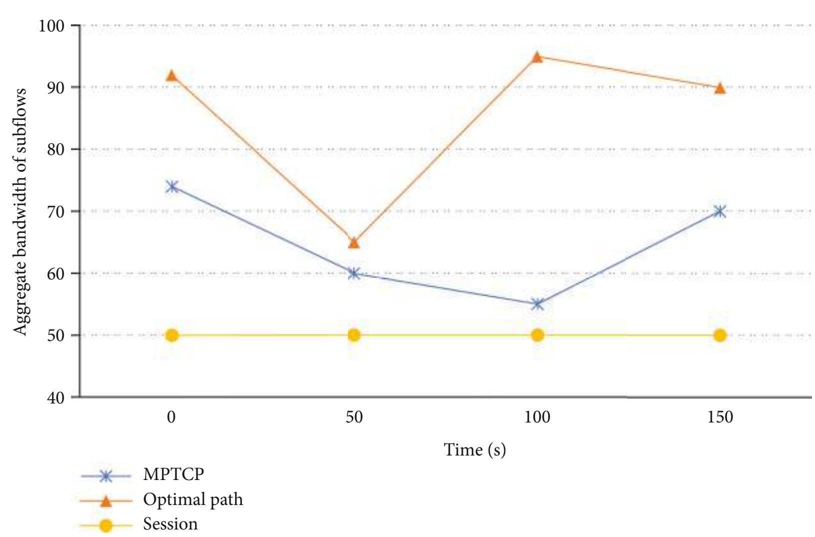
FIGURE 6: The comparison of variation of average service downtime with RAM for a "Cirros" instance using Normal TCP and MPTCP with two paths.



(a)

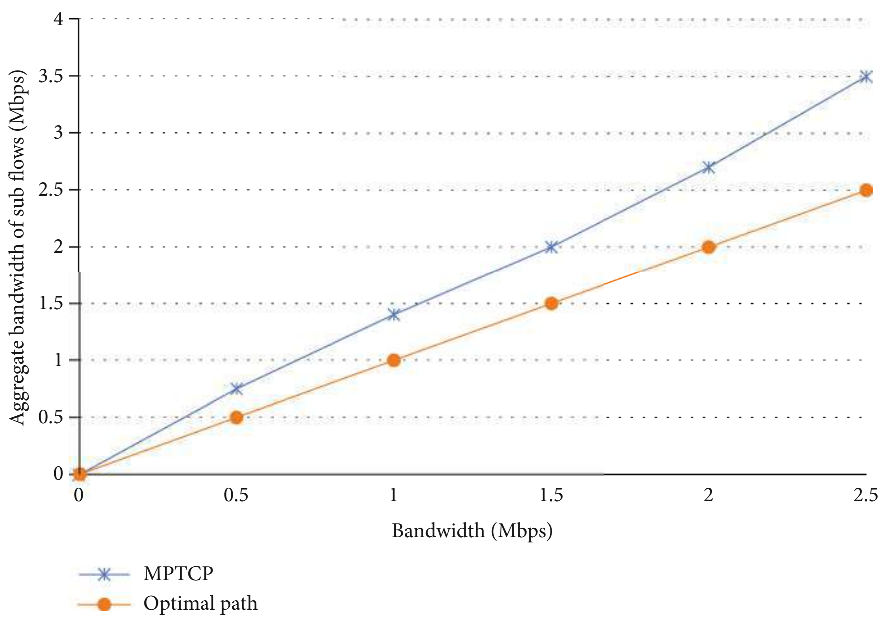


(b)

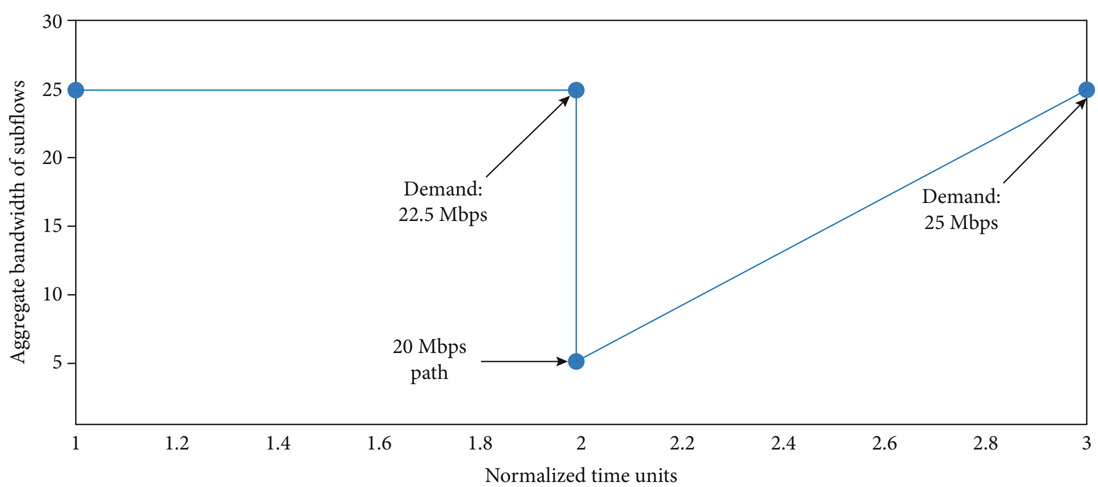


(c)

Figure 7: Continued.



(d)



(e)

FIGURE 7: Performance comparison of the α-best fit algorithm with SFO [9] in terms of bandwidth utilization. (a) Variation in the number of subflows with the increasing bandwidth demand (α). (b) Variation in the aggregate bandwidth ($\propto^s$) for a fixed number of subflows. (c) The variation of aggregate bandwidth of the sub flows with time. (d) Variation of the aggregate bandwidth of the subflows with the bandwidth demand. (e) The performance of the α-best fit algorithm in case of path failure.

Encouraged by the results in Figure 6, the number of MPTCP paths was increased to three. The service downtime of random migrations using single path TCP, MPTCP with two paths, and then with three paths has been plotted.

While there is a noted improvement with the increase in the number of paths by one, we observed that by adding the third path, the decrease in the service downtime is not satisfactory. The service downtime for random migrations using two path MPTCP connectivity and three path MPTCP connectivity has also been plotted.

The justification for this has been provided in Sections 3 and 4, which elaborate the functioning of the precopy mechanism with MPTCP and the need for optimization of MPTCP. The increase in the number of active paths to three for data transfer caused an increase in network congestion in the OpenStack management network, which reduced the bandwidth utilization. As the value of γ dropped, the factor $(x\gamma)$ dropped as well, thus accounting for the reduced performance as per equation (10).

The α-best fit algorithm proposed in Section 4 is tested for random bandwidth values of the available paths. We have compared the results obtained from the run of the proposed algorithm 4.1 to those obtained after the run of the algorithm given by Joshi and Kataoka for ONS computation [9].

7.2. Results from the α-Best Fit Algorithm. In order to demonstrate the comparison of the performance of the algorithm for optimization of MPTCP subflows proposed in this paper, and of the algorithm given in [9], tests were conducted in different aspects. The first aspect to be considered is the optimal number of subflows for different bandwidth demands. Evident from Figure 7(a), this is pretty much the same for both the algorithms. This means that for different values of $\propto^d$, the same number of subflows will be created using either of the algorithms.

Figure 7(b) shows the variation of the aggregate bandwidth of the subflows ($\propto^s$) with the demand of bandwidth ($\propto^d$) for a given value of ONS, thereby displaying the comparison between the α-best fit algorithm and SFO [9]. It can be inferred from the graph that our proposed algorithm delivers precisely the same amount of bandwidth as required during an MPTCP session.

Figure 7(c) shows the variation in the aggregate bandwidth of the subflows as a function of time. The MPTCP session bandwidth demand is assumed to be fixed at 50 Mbps. It can be seen that the range of variation of the aggregate bandwidth of the optimal paths is narrower, and the extremities are closer to the demand value in case of the α-best fit algorithm.

Figure 7(d) shows the relationship between $\propto^s$ and = $\propto^d$. As $\propto^d$ increases, $\propto^s$ increases in both the cases. However, in the algorithm given in [9], the rise is steeper in comparison with the α-best fit proposed algorithm. This means that the algorithm proposed in this paper adheres to the demand closer than the initial algorithm. Hence, the bandwidth usage is optimized.

In general, the utilization of bandwidth for the same demand in the α-best fit algorithm is more optimized than the initial algorithm. SFO [9] leads to wastage of bandwidth, as more bandwidth than required is taken up for the same data transfer. In case of cloud environments, this makes a huge difference, as the bandwidth is in the order of Gbps.

Finally, the action taken by the α-best fit algorithm in case of path failure is illustrated in Figure 7(e). Consider four paths with bandwidths 5, 10, 15, and 20 Mbps. If the demand is fixed to 25 Mbps, and the number of ONS is two, then initially, the 20 Mbps and 5 Mbps paths are chosen to put the subflows on. However, in case of the failure of the 20 Mbps path, the algorithm invokes the path_failure function and chooses the 15 Mbps and 10 Mbps paths to put the subflows on. Hence, the α-best fit algorithm offers a greater degree of optimization along with failure protection when compared to SFO [9].

8. Conclusion and Future Enhancements

As can be shown, MPTCP can be quite useful in a cloud setting for live VM migration. This capacity reduced service downtime, allowing for enhanced service quality. The results in Section 6 show that adding a single active data channel reduced service downtime by nearly half. In addition to several paths for data transfer between the host machines, the service downtime was decreased from 1.12 seconds to 0.58 seconds. Reduced service downtime enhances end-user experience and overall cloud performance in terms of responsiveness and reliability. We addressed the necessity to optimize MPTCP connections by providing an algorithm 4.1. A more elaborate approach was presented that takes into account not only the optimization of subflows but also the improvement of reliability and performance in case of path breakdown.

The proposed α-best fit approach can be used in conjunction with the precopy in the cloud to improve the efficiency of live VM migration. The α-best fit algorithm saves a lot of bandwidth by optimizing its consumption. The algorithm's failure prevention clause boosts its reliability. Also, the aggregate bandwidth of the α-best fit algorithm's subflows is less than the SFO's subflows [9]; yet, it meets the bandwidth requirements. Our results also show that the aggregate bandwidth of subflows from the α-best fit method changes within a narrow region near the bandwidth requirement.

The service downtime for live migration of instances can be further reduced by using a separate migration method in conjunction with ours. For example, adaptive memory compression [24], CR/TR motion [30], delta compression [5], and CPU scheduling for synchronization have all been added to precopy. These changes will reduce service downtime. Others like the postcopy algorithm [31] or a hybrid strategy that incorporates the benefits of both precopy and postcopy algorithms can also be employed in the cloud using MPTCP in future.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request (vinit.gunjan@ustc.ac.bd).

Conflicts of Interest

The authors declare that they have no conflicts of interest to report regarding the present study.

Authors' Contributions

Kadiyala Ramana contributed to the conceptualization, data curation, formal analysis, methodology, software, and writing-original draft. Rajanikanth Aluvala contributed to the supervision, writing—review and editing, project administration, and visualization. Vinit Kumar Gunjan contributed to the software, validation, writing-original draft, and methodology. Ninni Singh contributed to the supervision, writing—review and editing, funding acquisition, and visualization. M. Nageswara Prasadhu contributed to the data curation, investigation, resources, and software.

References

[1] M. Nelson, "Michael Nelson. Virtual machine migration," US Patent 7, 484, 2009.

[2] B. Sowmiya, E. Poovammal, K. Ramana, S. Singh, and B. Yoon, "Linear elliptical curve digital signature (LECDS) with Blockchain approach for enhanced security on cloud server," *Access*, vol. 9, pp. 138245–138253, 2021.

[3] M. Vashishtha, P. Chouksey, D. S. Rajput et al., "Security and detection mechanism in IoT-based cloud computing using hybrid approach," *Transactions*, vol. 11, no. 5–6, pp. 436–451, 2021.

[4] V. Rajanikanth Aluvalu, U. Maheswari, K. K. Chennam, and S. Shitharth, "Data security in cloud computing using Abe-based access control," in *Architectural wireless networks solutions and security issues*, vol. 196, no. 1pp. 47–62, Springer, Singapore, 2021.

[5] P. Svärd, B. Hudzia, J. Tordsson, and E. Elmroth, "Evaluation of delta compression techniques for efficient live migration of large virtual machines," *ACM SIGPLAN notices*, vol. 46, no. 7, pp. 111–120, 2011.

[6] K. K. Chennam, R. Aluvalu, and S. Shitharth, "An authentication model with high security for cloud database," in *Architectural wireless networks solutions and security issues*, vol. 196, no. 1pp. 13–25, Springer, Singapore, 2021.

[7] S. P. Rm, S. Bhattacharya, P. K. R. Maddikunta, S. R. K. Somayaji, K. Lakshmanna, R. Kaluri et al., "Load balancing of energy cloud using wind driven and firefly algorithms in internet of everything," *Journal of parallel and distributed computing*, vol. 142, pp. 16–26, 2020.

[8] A. Ford, C. Raiciu, M. Handley, and O. Bonaventure, "Tcp extensions for multipath operation with multiple addresses, draft-Ietf-Mptcp-Multiaddressed-09," in *Internetdraft*, p. 2012, Internet Engineering Task Force, 2012.

[9] K. D. Joshi and K. Kataoka, "Sfo: Subflow optimizer for mptcp in sdn," *2016 26th International Telecommunication Networks and Applications Conference*, 2016, pp. 173–178, Dunedin, New Zealand, Dec. 2016.

[10] C. Raiciu, D. Niculescu, M. Bagnulo, and M. J. Handley, "Opportunistic mobility with multipath TCP," in *Proceedings of the sixth international workshop on MobiArch*, pp. 7–12, Bethesda Maryland USA, 2011.

[11] C. Paasch, G. Detal, F. Duchene, C. Raiciu, and O. Bonaventure, "Exploring Mobile/Wi-fi handover with multipath TCP," in *Proceedings of the 2012 ACM SIGCOMM workshop on Cellular networks: operations, challenges, and future design*, pp. 31–36, Helsinki Finland, 2012.

[12] S. Deng, R. Netravali, A. Sivaraman, and H. Balakrishnan, "WiFi, LTE, or both? Measuring multi-homed wireless internet performance," in *Proceedings of the 2014 Conference on Internet Measurement Conference*, pp. 181–194, Vancouver BC Canada, 2014.

[13] B. Arzani, A. Gurney, S. Cheng, R. Guerin, and B. T. Loo, "Impact of path characteristics and scheduling policies on MPTCP performance," in *28th International Conference on Advanced Information Networking and Applications Workshops*, pp. 743–748, Victoria, BC, Canada, May 2014.

[14] R. Khalili, N. Gast, M. Popovic, U. Upadhyay, and J.-Y. Le Boudec, "MPTCP is not Pareto-optimal: Performance issues and a possible solution," in *Proceedings of the 8th International Conference on Emerging Networking Experiments and Technologies. ACM*, pp. 1–12, Nice France, 2012.

[15] D. Wischik, C. Raiciu, A. Greenhalgh, and M. Handley, "Design, implementation and evaluation of congestion control for multipath TCP," *8th USENIX Symposium on Networked Systems Design and Implementation*, vol. 11, p. 8, 2011.

[16] C. Raiciu, D. Wischik, and M. Handley, *Practical congestion control for multipath transport protocols*, University College London, London/United Kingdom, Technical Report/Representative, 2009.

[17] K. Noda and Y. Ito, "Proposal of novel MPTCP congestion control to suppress QoS fluctuation for WebQoE improvement," in *2018 IEEE 8th international conference on consumer electronics-Berlin*, pp. 1–3, Berlin, Germany, Sept. 2018.

[18] F. Yang, P. Amer, and N. Ekiz, "A scheduler for multipath TCP," in *2013 22nd International Conference on Computer Communication and Networks*, pp. 1–7, Nassau, Bahamas, Aug. 2013.

[19] F. Yang, Q. Wang, and P. D. Amer, "Out-of-order transmission for InOrder arrival scheduling for multipath TCP," *28th International Conference on Advanced Information Networking and Applications Workshops*, 2014, pp. 749–752, Victoria, BC, Canada, May 2014.

[20] N. Kuhn, E. Lochin, A. Mifdaoui, G. Sarwar, O. Mehani, and R. Boreli, "DAPS: Intelligent delay-aware packet scheduling for multipath transport," in *International Conference on Communications*, pp. 1222–1227, Sydney, NSW, Australia, June 2014.

[21] Y.-C. Chen, Y.-s. Lim, R. J. Gibbens, E. M. Nahum, R. Khalili, and D. Towsley, "A measurement-based study of multipath TCP performance over wireless networks," in *Proceedings of the 2013 Conference on Internet Measurement Conference. ACM*, pp. 455–468, Barcelona Spain, 2013.

[22] H. Chen, H. Kang, G. Jiang, and Y. Zhang, *Coordinating virtual machine migrations in Enterprise data centers and clouds*, Institute of Electrical and Electronics Engineers, New York, 2013.

[23] K. Kumar, K. R. Singamaneni, G. Dhiman, S. Singh, and B. Yoon, "A novel Blockchain and bi-linear polynomial-based QCP-ABE framework for privacy and security over the complex cloud data," *Sensors*, vol. 21, no. 21, p. 7300, 2021.

[24] H. Jin, L. Deng, S. Wu, X. Shi, and X. Pan, "Live virtual machine migration with adaptive, memory compression," in *2009 IEEE International Conference on Cluster Computing and Workshops*, New Orleans, LA, USA, Sept. 2009.

[25] K. Ramana, T. Krishna, C. Narayana, and M. P. Kumar, "Comparative analysis on cloud computing and service oriented architecture," *International Journal of Advanced Research in Technology*, vol. 1, no. 1, pp. 22–28, 2011.

[26] K. Ramana, "A survey on cloud computing and service oriented architecture," *VSRD International Journal*, vol. 1, no. 9, 2011.

[27] S. Nathan, P. Kulkarni, and U. Bellur, "Resource availability based performance benchmarking of virtual machine migrations," in *Proceedings of the 4th ACM/SPEC International Conference on Performance Engineering*, pp. 387–398, Prague Czech Republic, April 2013.

[28] T. Wang, Y. Quan, X. S. Shen, T. R. Gadekallu, W. Wang, and K. Dev, "A privacy-enhanced retrieval technology for the cloud-assisted internet of things," *IEEE transactions on industrial informatics*, p. 1, 2021.

[29] "How to calculate bandwidth utilization using snmp," pp. 10–26, 2005, http://www.cisco.com/c/en/us/support/docs/ip/simple-network-management-protocol-snmp/8141-calculate-bandwidth-snmp.

[30] H. Liu, H. Jin, X. Liao, H. Liting, and Y. Chen, "Live migration of virtual machine based on full system trace and replay," in *Proceedings of the 18th ACM international symposium on High performance distributed computing*, p. 101, Garching Germany, June 2009.

[31] M. R. Hines and K. Gopalan, "Post-copy based live virtual machine migration using adaptive pre-paging and dynamic self-ballooning," in *Proceedings of the 2009 ACM SIGPLAN/SIGOPS international conference on Virtual execution environments*, pp. 51–60, Washington DC USA, March 2009.

frontiers | Frontiers in Public Health

ORIGINAL RESEARCH
published: 07 June 2022
doi: 10.3389/fpubh.2022.769692

A Neural Network and Optimization Based Lung Cancer Detection System in CT Images

Chapala Venkatesh[1], *Kadiyala Ramana*[2*], *Siva Yamini Lakkisetty*[1], *Shahab S. Band*[3*], *Shweta Agarwal*[4] and *Amir Mosavi*[5,6,7*]

[1] Department of ECE, Annamacharya Institute of Technology and Sciences, Rajampet, India, [2] Department of IT, Chaitanya Bharathi Institute of Technology, Hyderabad, India, [3] Future Technology Research Center, College of Future, National Yunlin University of Science and Technology, Douliou, Taiwan, [4] SAGE University, Indore, India, [5] John von Neumann Faculty of Informatics, Obuda University, Budapest, Hungary, [6] Faculty of Civil Engineering, TU-Dresden, Dresden, Germany, [7] Institute of Information Engineering, Automation and Mathematics, Slovak University of Technology in Bratislava, Bratislava, Slovakia

OPEN ACCESS

Edited by:
Celestine Iwendi,
School of Creative Technologies
University of Bolton, United Kingdom

Reviewed by:
G Surya Narayana,
Vardhaman College of Engineering, India
Chennareddy Vijay Simha Reddy,
Middlesex University, United Kingdom
Venkata Subbaraju Dommaraju,
University of the Cumberlands, United States

***Correspondence:**
Shahab S. Band
shamshirbands@yuntech.edu.tw
Amir Mosavi
amir.mosavi@kvk.uni-obuda.hu
Kadiyala Ramana
ramana.it01@gmail.com

Specialty section:
This article was submitted to Digital Public Health, a section of the journal Frontiers in Public Health

Received: 02 September 2021
Accepted: 20 January 2022
Published: 07 June 2022

Citation:
Venkatesh C, Ramana K, Lakkisetty SY, Band SS, Agarwal S and Mosavi A (2022) A Neural Network and Optimization Based Lung Cancer Detection System in CT Images.
Front. Public Health 10:769692.
doi: 10.3389/fpubh.2022.769692

One of the most common causes of death from cancer for both women and men is lung cancer. Lung nodules are critical for the screening of cancer and early recognition permits treatment and enhances the rate of rehabilitation in patients. Although a lot of work is being done in this area, an increase in accuracy is still required to swell patient persistence rate. However, traditional systems do not segment cancer cells of different forms accurately and no system attained greater reliability. An effective screening procedure is proposed in this work to not only identify lung cancer lesions rapidly but to increase accuracy. In this procedure, Otsu thresholding segmentation is utilized to accomplish perfect isolation of the selected area, and the cuckoo search algorithm is utilized to define the best characteristics for partitioning cancer nodules. By using a local binary pattern, the relevant features of the lesion are retrieved. The CNN classifier is designed to spot whether a lung lesion is malicious or non-malicious based on the retrieved features. The proposed framework achieves an accuracy of 96.97% percent. The recommended study reveals that accuracy is improved, and the results are compiled using Particle swarm optimization and genetic algorithms.

Keywords: cancer, lung cancer, machine learning, artificial intelligence, deep learning, cancer detection

INTRODUCTION

The most well-known reason for death because of malignant growth is lung cancer. The second most habitually analyzed type of malignancy is lung cancer. Pneumonic nodules are apparent in the lung to evaluate metastases from different malignancies (1, 2). Computed tomography (CT) is the most significant image mode for assessing progress/crumbling and for observation and decision-making malignant lung growths. As a result of the precocious presentation of lung malignancy by CT, doctors can be suggested more productive treatments (3, 4). Guess and recuperating components for scattered sickness with precise malignancy stages are required for orderly and consoling treatment (5). The early conclusion of the period of lung malignancy is firmly connected to the patient's continuance rate (6). In clinical terms, the disease is known to be strange hyperplasia and significantly beyond what 200 sorts can influence the individuals (7). According to the ACS (American Cancer Society), lung malignancy is the main cause of death in both men and women in the United States. about a total of 2,28,820 new lung malignancy cases were estimated, with 1,35,720 deaths (8). It causes a larger number of deaths than other malignant tumors. Early recognition of tumorous lung nodules is the key factor for patient survival rate. When contrasted with chest X-ray imaging, CT perceives the tumorous nodules consistently at an underlying stage (9). Practically all radiologists use CT by exploring multiple pictures from a solitary patient. Thus, the exhaustion of the radiologists can prompt wrong analysis. Hence, the exact physical valuation measure is tedious and colossally inconsistent (10). A precise segment is noteworthy for the right valuation of nodule improvement and for the arrangement of malignant nodules (disease cells) from benign ones (non-disease cells). The reason for this work is to exactly recognize the nodules over the CT lung pictures.

The proposed instructional method pulls back, utlilizing a middle method to reduce confusion based on the CT image. Second, a crossover division approach is utilized to isolate the lung zone from its environmental factors. The proposed division strategy utilizes the Otsu thresholding to eliminate superfluous groups, consequently isolating the specific lung locales, and nodules of interest can be decisively characterized by cuckoo inquiry advancement. Third, the assortment of surface highlights for the particular nodule is undisturbed by parallel neighborhood examples at the feature stage. Finally, the highlights of the sectioned lung nodules are prepared by the CNN classifier to distinguish the lesions as malignant or non-malevolent.

RELATED WORK

In 2019, Ananya et al. (11) developed a multi-approach system for lung cancer categorization using genetics. They assessed false negatives and true positives for classification accuracy in this study, but not detected accuracy.

In 2019, Venkatesh et al. (12) developed an innovative approach to detect lesions based on a GA and LBP. This process achieves an accuracy of 90%.

In 2019. Preeti et al. (13) introduced a lung cancer detection framework based on the fuzzy c-mean clustering and SVM classifier techniques

In 2019, Senthil Kumar et al. (14) introduced an approach for detecting lung lesions using GCPSO. In this work, multiple optimization techniques are used to classify cancer in CT images. The process obtained a precision of 95%.

In 2018, Perumal et al. (15) proposed an ABC algorithm for malignancy recognition and classification. This guidance attained a 92% accuracy.

In 2017 Ammar et al. (16) established an early diagnostic architecture for genetically altered tumor detection. In this study, the authors achieved an accuracy rate of 84%.

In 2017 Kamil et al. (17) introduced a DWT-based lung lesion detection system. In this method by using subtraction and erosion techniques images are analyzed to remove the cancer region. This approach yielded an accuracy of 89%.

In 2016 Mukesh et al. (18) introduced a DWT-based method for assessing a high volume of tissues in chest X-ray images. Using this method, the authors were able to achieve an accuracy of 86%.

In 2014 Santos et al. (19) described an area development and Hessian matrix to identify minor respiratory lesions. The presented approach achieves a classification accuracy of 88.4%.

In 2014, Jinsa and Gunavathri (20) reported an ANN-based lesion categorization technique. They were able to classify with an accuracy of 93.3%.

The principal gap has indeed been extended due to the lack of research publications that demonstrate computations. Because of poor directionality, slower processing, greater calculation time, and complex computations, the methods suggested by the authors mentioned above are less effective in all cases. As a result, an interactive technique for identifying lung cancer in CT images is suggested in this article, which uses the otsu threshold-based Cuckoo search algorithm, Local Binary Pattern for image retrieval, and CNN for classification to conquer all of the shortfalls of the existing methods. By selecting the most cost-effective strategy, optimization algorithms tend to produce a solution for image processing processes.

MOTIVATION AND CONTRIBUTION

Lung cancer is confirmed by physicians after a thorough examination of CT scans, which requires a lot of time and is not always accurate. To create imagery as precise, operational, and efficient as possible, state-of-the-art optimization techniques and image processing approaches were required. The proposed technology will aid doctors in accurately identifying lung nodules at an early stage, as well as studying the internal anatomy. As a part of the contribution, some glitches related to lung cancer detection are discussed here. The region of interest is retrieved using Otsu thresholding and cuckoo search optimization, which is a novel approach to segmentation. This proposed partitioning approach requires only a few parameters to precisely separate nodules of varied sizes and shapes.

Proposed Methodology

Figure 1 depicts the prospective lung malignancy diagnostic procedure, which comprises five phases: (1) contrast enhancement and Noise reduction through pre-processing, (2) Otsu thresholding based cuckoo search algorithm to segment the lesion from its backgrounds, (3) retrieval of regions of concern, (4) retrieval of descriptors from segmented lung lesions, and, in the last phase, (5) SVM has been used to assess if the lesion was abnormal or normal. The next sections provide detailed descriptions of the above-mentioned phases.

Image Acquisition

It is the basic step before proceeding with other critical steps. It is a method for processing a digital image from a database (21). Numerous sorts of scanners, such as X-Ray, MRI, and CT, are used to obtain the images. The CT image was captured using a CT scanner. It is a type of scanning that creates cross-section scans for each pixel (22).

Pre-processing

The equations should be inserted in editable format from the equation editor. It is then procedure to improve image details. The basic idea is to suppress noise, which corrects undesired distortions and enhances the associated attributes of the image for subsequent processing (23). Because all techniques are sensitive to noise, efficient images pre-processed allow for better

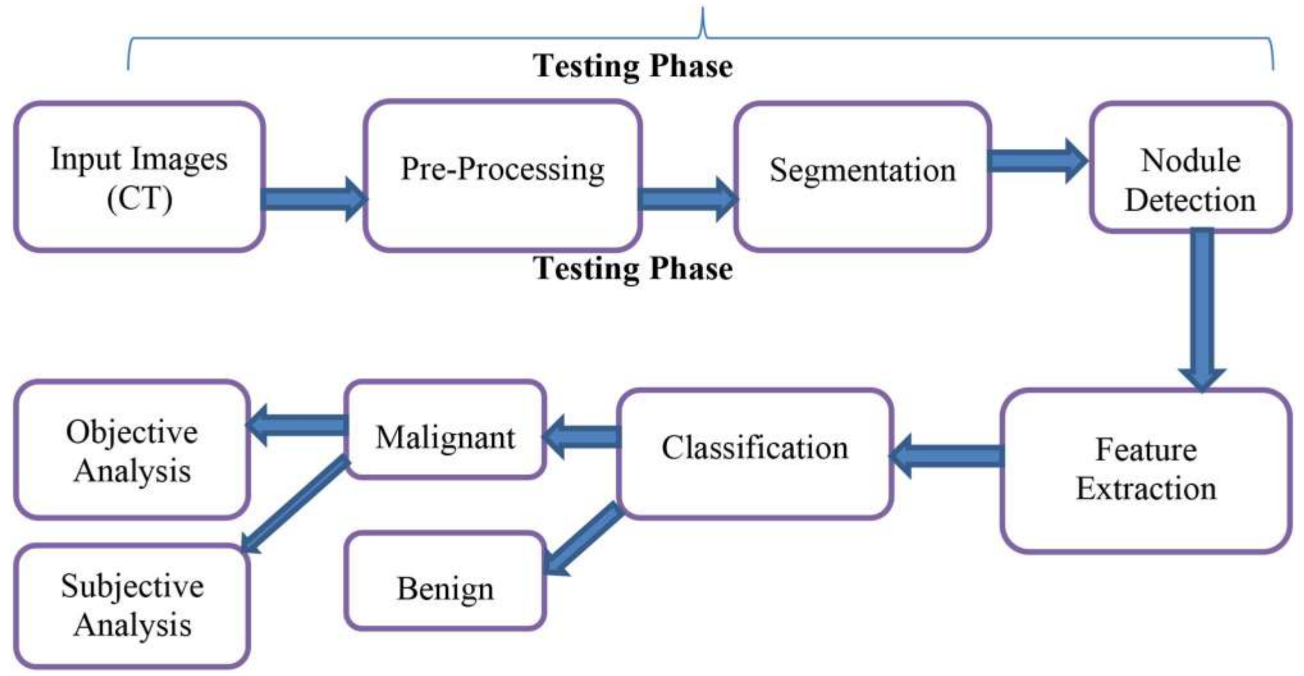


FIGURE 1 | Architecture of proposed method.

segmentation and, as a result, better classification. The size of the pixel area could be used to classify pre-processing procedures. Image enhancement employs these techniques. Enhancement operations operate on the image pixels of the neighborhood and the corresponding values of the neighborhood. Contribute quality to the images by decreasing noise and distortion (24).

$$\mathbf{f}(x) = \mathbf{a_0} + \sum_{n=1}^{\infty}\left(\mathbf{a}_n \cos\frac{n\pi x}{\mathbf{L}} + \mathbf{b}_n \sin\frac{n\pi \mathbf{x}}{\mathbf{L}}\right)$$

Median Filtering

Salt and pepper noise can be found on CT scans. The finest features are obscured by these impacts. By keeping the frontier of the image as fine as possible, this filtering lowers salt and pepper noise (12). This filter gathers information from a sample within a non-averaged window (25). The edges of the filter are better managed than those of other linear filters. The following equations are used to get the median value.

$$\mathbf{M}(g) = \sum_{k=1}^{n} |x_k - \mathbf{g}|. \quad (1)$$

$$\mathbf{g} = \mathbf{median}\,\{x_1, x_2, \ldots\ldots\ldots. x_n\}. \quad (2)$$

Otsu Thresholding Segmentation With Optimization

Otsu Thresholding

The goal of this strategy is to scour specified like-classes of pixels in a picture for the closeness of neighboring pixels in order to generate a concentrated image object. Separating the background and sub-regions in medical imaging is tough (26). The Otsu segmentation algorithm works better to "recognize" or "smear" the context contents of the front objects. It is an adaptive threshold binarization procedure proposed by OTSU in 1979. This procedure uses the highest within-class variance between the context and the target based on the rule of threshold assortment (27). It segments the image into the forefront and the contextual based on the characteristics of gray level values. If the finest threshold is attained, the gap between the two regions is the highest. The Otsu algorithm, in general, utilizes the greatest within-class variance. The larger the variance value, the wider the difference between the two areas, since variance is a useful determinant of uniform gray distribution. If some areas are wrongly segmented into contextual or if some contextual is segmented into areas, then the gap is too small between the two areas. As a result, if the variance among groups is higher, the likelihood of incorrect classification is lowered, resulting in cohesive segmentation.

The following is the main principle of OTSU-based threshold segmentation:

Let us call the gray values *g* and the number of pixels *nx*. Then

$$\mathbf{P} = \sum_{x=0}^{\mathbf{L}-1} n_x = n_0 + n_1 + n_2 + \ldots + n_{\mathbf{L}-1} \ldots \quad (3)$$

where g = 0, 1,..., L-1, and P indicate the number of pixels. Suppose C1 and C2 are the two kinds of pixels. C1 pixels have a range of [0,x] while C2 pixels have a range of [x+ 1,L1].

$$\sigma_{Gv}^2 = \sum_{g}^{L-1} (g - m_{Gv})^2. \quad (4)$$

$$\sigma_{Bv}^2 = P_1 (m_1 - m_{Gv})^2 + P_2 (m_2 - m_{Gv})^2 \quad (5)$$

The below-mentioned calculations are used to compute the mean intensities.

$$m_1 = \frac{1}{P} \sum_{g}^{x} g.P_g \quad (6)$$

$$m_2 = \frac{1}{P} \sum_{g=x+1}^{L-1} g.P_g \quad (7)$$

$$m_{Gv} = \sum_{g=0}^{L-1} g.P_g \quad (8)$$

where m 1 and m 2 are the C1 and C2 pixel average intensities, and m Gv is the global mean intensity. Lastly, the ratio τ , which is provided below, is used to determine the ideal threshold.

$$\tau = \frac{\sigma_{Bv}^2}{\sigma_{Gv}^2} \quad (9)$$

Cuckoo Search Optimization

The cuckoo generation function is projected using this approach, which reduces the implications. A large number of nests are accessible during the search procedure. The location of the cuckoo egg has been discovered as a novel solution (28). The steps in the search procedure are as follows. A cuckoo bird places one egg at a time in a randomly chosen nest. The parasite nests were static, and the number of eggs in the nests would increase until they reach their highest level. When the cuckoo's egg is spotted, the host bird seems to have the choice of chucking the egg away or scrapping the nest and forming a new one.

The Levy flight theory has improved the CS algorithm (29). This CS technique is used to calculate the appropriate threshold for eliminating the lung nodule.

The following analogy is incorporated into the proposed technique for optimum selection (**Algorithm 1**):

The new solution, which depicts the subcategory of thresholds, is depicted by the egg of a cuckoo. This is also utilized for segmentation of the lung nodule. The grade of eggs for each host nest is either 0 or 1, which mimics the segmentation procedure's threshold partition. Pa is the probability that a cuckoo's egg will be discovered by the host bird. Pa has a predefined threshold. It demonstrates the principle of removing the least relevant threshold subgroups and, as a result, removing these threshold values from further analysis.

Algorithm 1 | Cuckoo search algorithm.

Step1: Initialization parameters: n, Pa, & M where n=number of host nests; pa : probability of discovery of alien, M: maximum number of iterations

Step2: Generate initial n host, n_i^t

Step3: Evaluate f(n_i^t)

Step4: Generate a new solution $n_i^{t+1} = n_i^t + \alpha \oplus Lev'y\,(\gamma\,)$

Where the symbol ⊕ is entry-wise multiplication,

α >0 indicates the step size, Levy(γ)= $g^{-\gamma}$ (1< γ ≤3)

Step5: Evaluate $f(n_i^{t+1})$

Step6: Choose a nest n_j randomly

Step7: If $(n_j^t) > (n_j^{t+1})$ then Replace n_j^t *with* n_j^{t+1}

Step8: Confiscate a worse nest with Pa

Step9: Construct new nest using Levy flights

Step10: Retain the best solutions

Feature Extraction

The LBP operator was established to determine texture in the first place (30). By thresholding an image with the central pixel value and taking the result as a binary quantity, the operator applies a mark to each pixel (31). The picture of the Lung CT could be considered as a micro-pattern structure that the LBP operator can well portray. The steps for extracting the characteristics are outlined below.

- Divide the window looking at into cells.
- In a cell, collate each pixel with its neighbors.
- If the value of the center pixel is larger than the value of the neighbor pixel, assign "1"; otherwise, assign "0".
- A binary number is created by comparing all of the pixels.
- Lastly, over the cell, compute the histogram. The LBP value can be calculated using the expression $LBP_{X,Y}^{U}$ where U represents uniform pattern and X,Y indicate neighborhood.

$$LBP_{X,Y} = \sum_{0}^{m-1} s(P_s - P_c\,)2^s \quad (10)$$

$$s(t) = \begin{cases} 1 & t \geq 0 \\ 0 & t < 0 \end{cases} \quad (11)$$

P_c is the gray value of the center pixel, P s is the intensity value of the location pixels (m = 0, 1,......m-1), and m is the image element well within range R where R is higher than zero (R > 0), creating a regionally oriented neighborhood set. After identifying each pixel in a picture, a histogram is created to define the texture image (32, 33).

CLASSIFICATION

CNN belongs to DNN group which is comprised of numerous hidden layers, like RELU, fully linked, pooling and convolution layer, etc. CNN securities weights in the coevolutionary layer, which lowers the network latency, and enhances network performance (34). CNN's prominent features are prevalent weights, local networking, and neuronal 3D sizes. A feature map is created with a kernel by a convolution layer of diverse sub-regions of the input image (35). Then, a nonlinear function is added to the RELU layer to progress the convergence possessions when the error is small. The architecture of CNN is as shown in **Figure 2**.

In relation to complex layers, CNNs quite often incorporate pooling layers (36). They are principally used only for lessening the dimensions of the tensor and speeding up estimations (37). All such layers are simple. So, the image

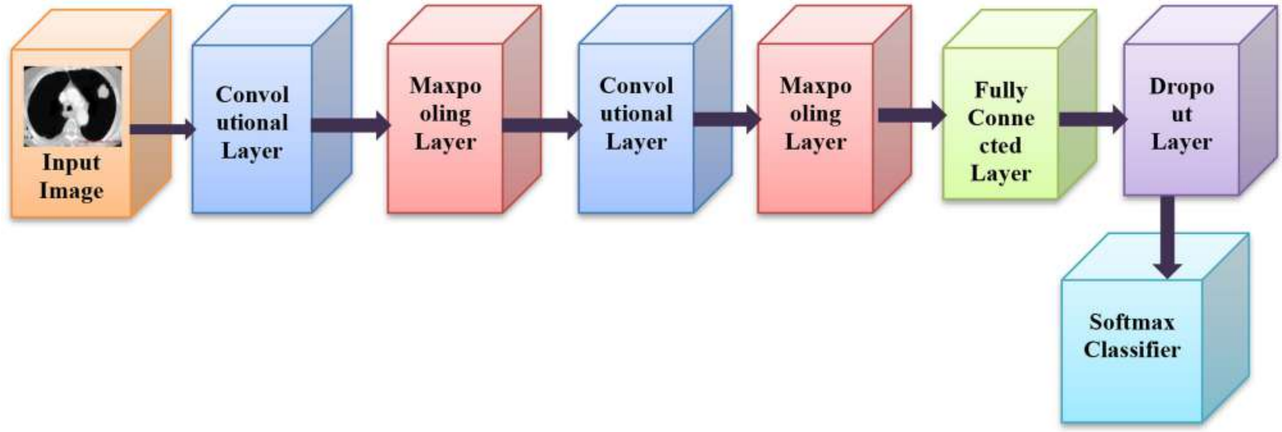


FIGURE 2 | Architecture of CNN.

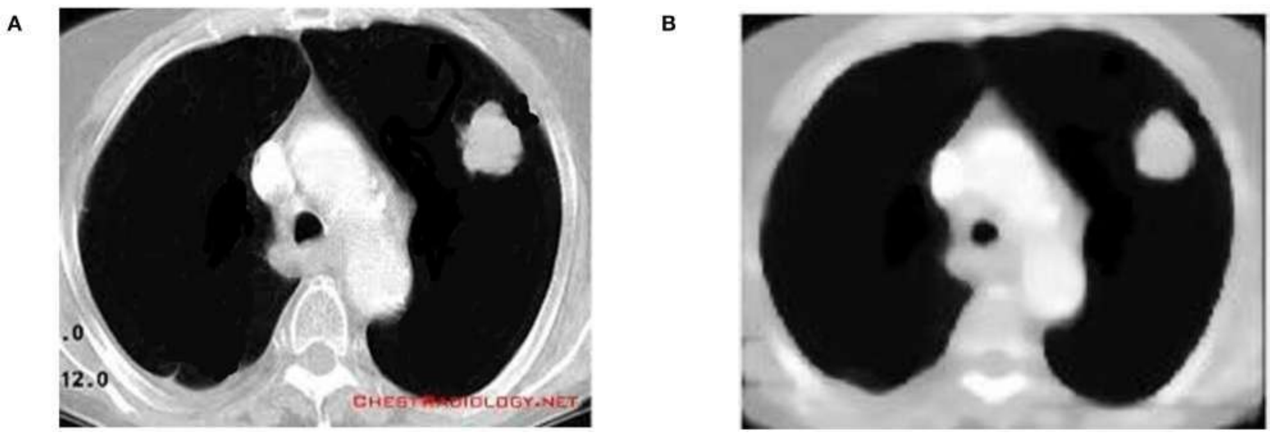


FIGURE 3 | (A) Input CT image. **(B)** Filtered output.

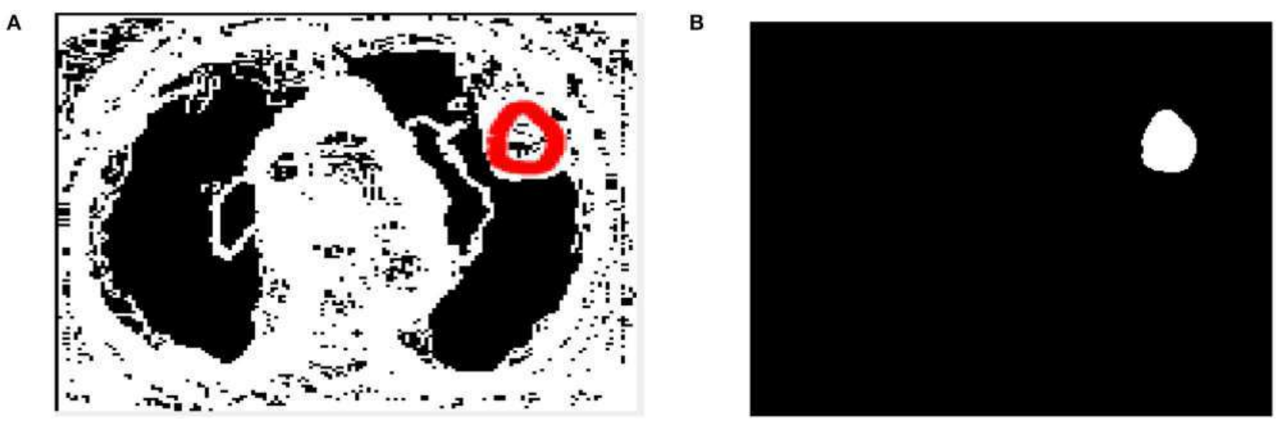


FIGURE 4 | (A) Extracted output. **(B)** Segmented output.

is to be split up into smaller portions in the pooling layer and for each portion, the maximum value is selected and then accomplished in some process for each portion (38). After being portioned, it is placed in the output in the respective position. RELU is a rectified linear unit, as well as a form of hidden layers. The activation function is

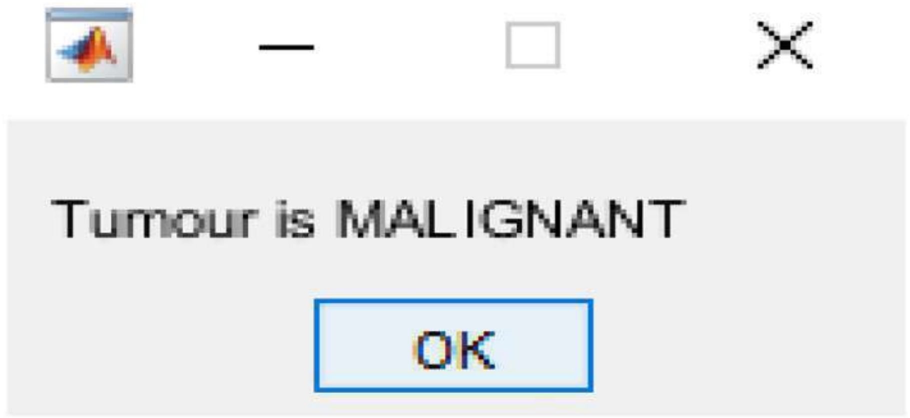


FIGURE 5 | Classification output.

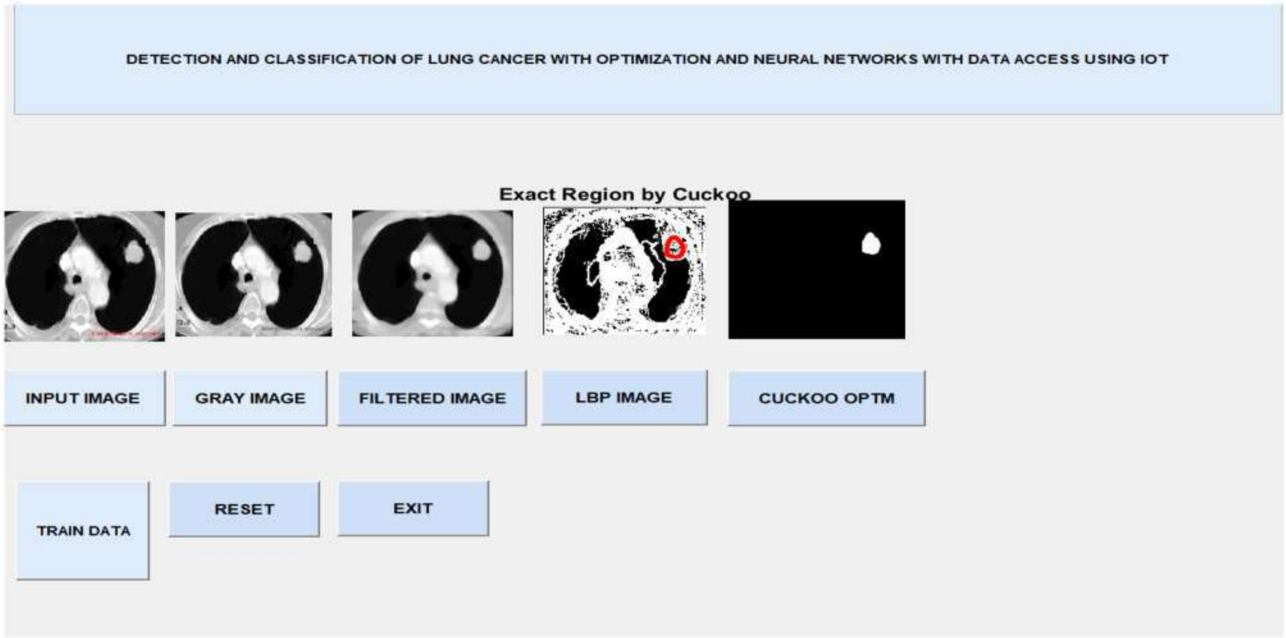


FIGURE 6 | GUI output.

most popularly used in neural networks, predominantly in CNNs (34).

SIMULATION RESULTS

Lung cancer is diagnosed in CT medical images using the novel cuckoo search algorithm and attributes are determined in this study. Lung cancer CT images were collected from a private hospital (Satyam diagnostic center, Anantapur). The adaptive threshold issue in this study is referred to as an optimization problem and it can be resolved using the CSA approach. In this study, the outcomes of the suggested method were compared to those of the PSO and GA algorithms. This work has been carried out by MATLAB software. When compared to open-source tools, MATLAB has a great affinity with deep learning techniques as well as hardware tools. Also, open-source tools have a hard time bringing all of the libraries together in one spot.

This study relies on my prior work (39), in which the outcomes were produced using a Genetic algorithm and particle swarm optimization approaches in addition to LBP and CNN. Cuckoo search optimization is used in this work, together with CNN and LBP, to enhance the accuracy. The input and median filter output of CT lung cancer pictures are depicted in **Figure 3**. Low-frequency noise

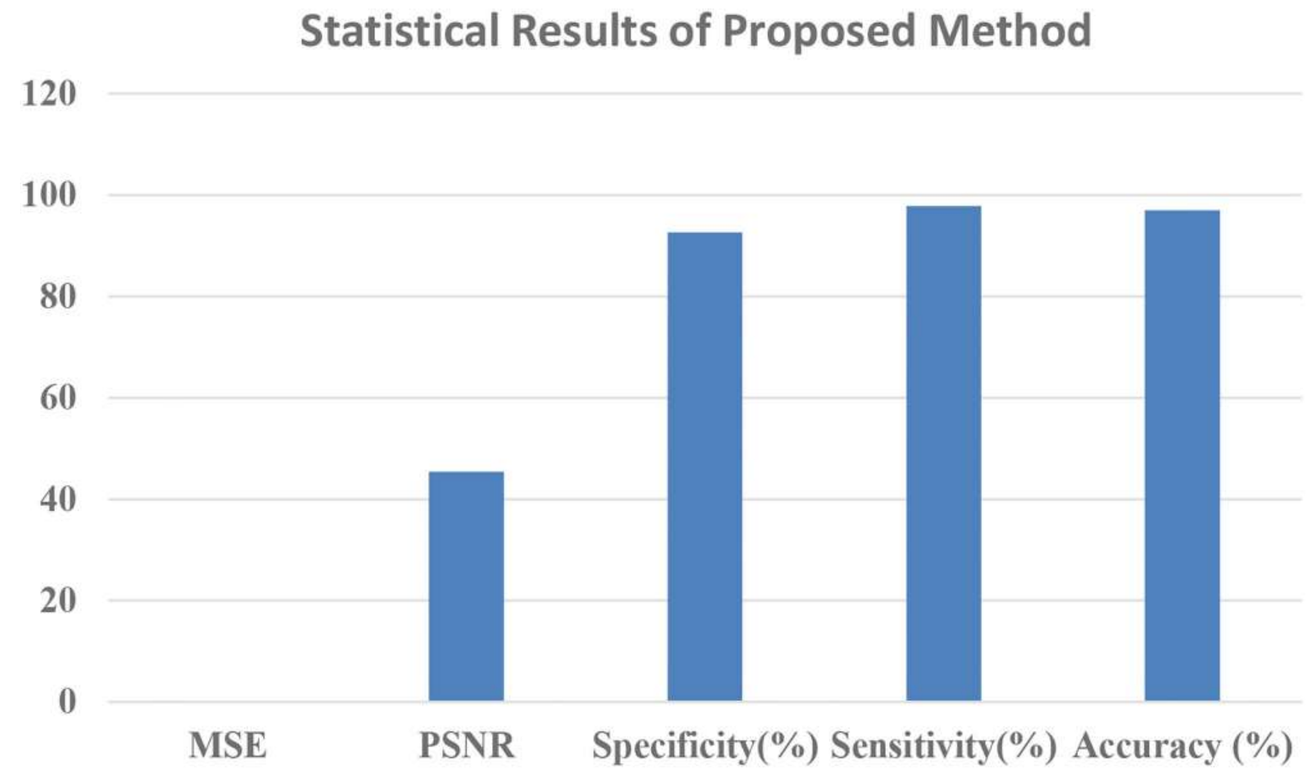


FIGURE 7 | Statistical results graphical representation.

and distortion are common in CT scan images. To reduce noise and distortion, the input image is processed with a median filter.

To identify lung lesions in the CT image, it is split into multiple clusters and then optimized using the Otsu thresholding approach. The CT scan is first split using simple Otsu thresholding, which improves the segmented classes' variation, or "all class variance." The result of the thresholding technique can be enhanced by processing it with cuckoo search optimization. Following partitioning, the image is subjected to LBP feature extraction, which extracts the textural features before extracting the detected output (see **Figure 4**).

After image retrieval, the image is given to CNN classification, which assesses the image as normal or abnormal by showing a message such as "Tumor is MALIGNANT" or "Tumor is BENIGN," as illustrated in **Figure 5**. The system's general function is created in a GUI, as seen in **Figure 6**. The statistical results like performance metrics obtained by 200 iterations for the input image are shown in graphical representation in **Figure 7**.

Table 1 showed that the suggested approach yields higher accuracy of 97%, the sensitivity of 97.8%, specificity of 92.6%, PSNR of 45.38%, and low MSE of 0.013 than conventional systems. These optimum results are obtained for 200 iterations (Our earlier proposed systems).

TABLE 1 | Attributed obtained from the proposed method.

Parameters	Proposed Method (CSO+CNN+LBP)
MSE	0.013
PSNR (%)	45.38
Specificity (%)	92.672
Sensitivity (%)	97.806
Accuracy (%)	96.979

TABLE 2 | Comparative Results with proposed method.

Parameters	Proposed Method (CSO+CNN+LBP)	(PSO+SVM+LBP)	(GA+SVM +LBP)
MSE	0.013	0.0301	0.0651
PSNR	45.38	33.2788	27.5311
Specificity (%)	92.672	60.0000	90.4950
Sensitivity (%)	97.806	96.5783	83.7143
Accuracy (%)	96.979	96.9391	90.4937

Table 2 shows that the suggested approach has higher accuracy (97%) than conventional systems (Our earlier proposed systems). The Comparative Results Graphical Representation is shown in **Figure 8**.

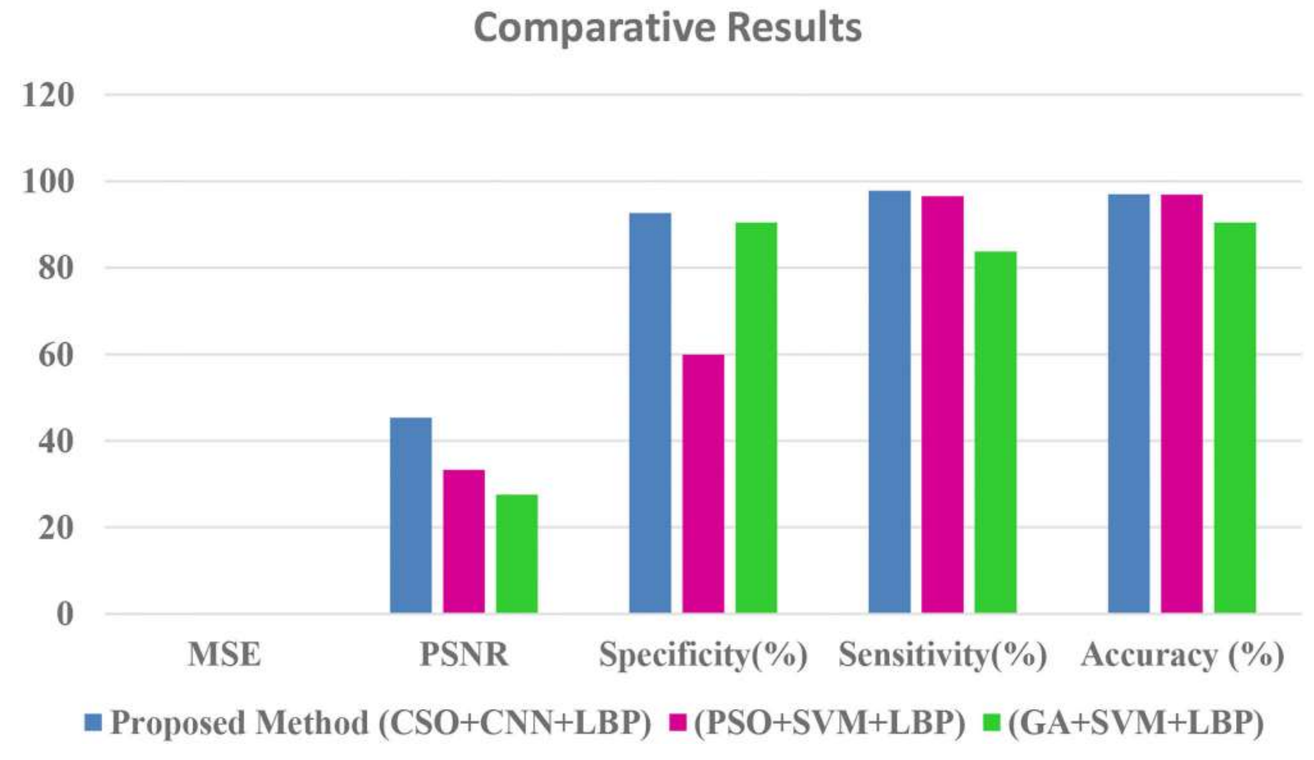


FIGURE 8 | Comparative results graphical representation.

CONCLUSION

In this article, a strong approach for recognizing lung cancer in CT images is developed. For exact cancer diagnosis in CT lung images, the Otsu thresholding-based cuckoo search optimization and CNN classifier approach were used. Based on the simulation findings, it is observed that the suggested method reliably segments CT images and detects lesions of various forms and sizes. Subsequently, the proposed approach comprises successive stages that continuously yield the last detection result. The techniques that are utilized in different stages are basic and simple to actualize.

Based on the simulation findings, the accuracy of the proposed framework' is calculated to be 96.97%, which is greater than any other demonstrative framework found in the literature. As for future work, a powerful strategy could be created by supplanting the CNN Classifier with a profound deep learning method and CAD tools. One may improve the structure for images of lung cancer in different modalities.

DATA AVAILABILITY STATEMENT

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

AUTHOR CONTRIBUTIONS

CV: conceptualization, methodology, software, and visualization. KR: data curation, writing—original draft, data analysis, and investigation. SL: software, validation, and editing. SB: software, validation, and editing. AM: supervision, writing—review & editing. SA: software, validation, and editing. All authors contributed to the article and approved the submitted version.

ACKNOWLEDGMENTS

The authors express sincere thanks to Dr. M. Vijay Kumar, Radiologist from Star Diagnostics Hospital at Ananthapuramu, for providing the lung CT images to carry-out this research work.

REFERENCES

1. Manickavasagam R, Selvan S. Automatic detection and classification of lung nodules in CT image using optimized neuro fuzzy classifier with cuckoo search algorithm. *J Med Syst.* (2019) 43:1–9. doi: 10.1007/s10916-019-1177-9
2. Brown MS, McNitt-Gray MF. Patient-specific models for lung nodule detection and surveillance in CT images. *Proc IEEE Trans Med Imag.* (2001) 20:1242–50. doi: 10.1109/42.974919
3. Cosman PC, Tseng C, Gray RM. Tree-structured vector quantization of CT chest scans: image quality and diagnostic accuracy. *Proc IEEE Trans Med Imag.* (1993) 12:727–39.
4. Dewes P, Frellesen C, Al-Butmeh F, Albrecht MH, Scholtz J-E, Metzger SC, et al. Comparative evaluation of non-contrast CAIPIRINHA-VIBE 3T-MRI and multidetector CT for detection of pulmonary nodules: in vivo evaluation of diagnostic accuracy and image quality. *Eur J Radiol.* (2016) 85:193–8. doi: 10.1016/j.ejrad.2015.11.020

5. Georg Homann MD, Mustaf DF. Improved detection of bone metastases from lung cancer in the thoracic cage using 5- and 1-mm axial images versus a new CT software generating rib unfolding images. *J Acad Radiol*. (2015) 22:505–12. doi: 10.1016/j.acra.2014.12.005
6. De Nunzio G, Massafra A. Approaches to juxta-pleural nodule detection in CT images within the MAGIC-5 collaboration. *J Nucl Instrum Methods Phys Res*. (2011) 648:103–6. doi: 10.1016/j.nima.2010.12.082
7. Dhanamjayulu C, Nizhal UN, Maddikunta PKR, Gadekallu TR, Lwendi C, Wei C, et al. Identification of malnutrition and prediction of BMI from facial images using real-time image processing and machine learning. *IET Image Process*. (2021) 16:647–58. doi: 10.1049/ipr2.12222
8. American Cancer Society. *Key Statistics for Lung Cancer*. Available online at: https://www.cancer.org/cancer/lung-cancer/about/key-statistics.html (accessed January 5, 2022).
9. Prokop M, Galanshi M. *Spiral and Multislice Computed Tomography of the Body*. Stuttgart: Thime Medical Publishers (2003)
10. Abbas S, Jalil Z, Javed AR, Batool I, Khan MZ, Noorwali A, et al. BCD-WERT: a novel approach for breast cancer detection using whale optimization based efficient features and extremely randomized tree algorithm. *Peer J Comput Sci*. (2021) 7:e390. doi: 10.7717/peerj-cs.390
11. Ananya C, Rajamenakshi, Subramanian R, Gaur S. A novel approach for tumor segmentation for lung cancer using multi-objective genetic algorithm and connected component analysis. In: *Proceedings of the 2nd International Conference on Data Engineering and Communication Technology, Advances in Intelligent Systems and Computing Springer Nature*. Singapore (2019). pp. 367–76.
12. Venkatesh C, Polaiah B. A novel approach for lung lesion segmentation using optimization technique. *Helix Scientific Explor*. (2019) 94:4832–7. doi: 10.29042/2019-4832-4837
13. Preeti J, Bajaj SB, Aman J. "Segmentation and Detection of Lung Cancer Using Image Processing and Clustering Techniques" Springer Nature Singapore, Progress in Advanced Computing and Intelligent Engineering. *Adv Intell Syst Comput*. (2019) 1:3–23. doi: 10.1007/978-981-13-1708-8_2
14. Senthil Kumar K, Venkatalakshmi K, Karthikeyan K. Lung cancer detection using image segmentation by means of various evolutionary algorithms. *Hindawi Comput Math Methods Med*. (2019) 2019:1–6. doi: 10.1155/2019/4909846
15. Perumal S, Velmurugan T. Lung cancer detection and classification on CT CT scan images using enhanced artificial bee colony optimization. *Int J Eng Technol*. (2018) 7:74–9. doi: 10.14419/ijet.v7i2.26.12538
16. Ammar O, Ibrahim Al A, Abrahim B. Novel genetic algorithm for early prediction and detection of lung cancer. *J Cancer Treat Res*. (2017) 5:15–8. doi: 10.11648/j.jctr.20170502.13
17. Kamil D, Ali H, Yoney KE. Lung lesion segmentation using gaussian filter and discrete wavelet transform. *ITM Web Conf*. (2017) 11:1–10. doi: 10.1051/itmconf/20171101018
18. Mukesh CA Dr, Bhumika G. Detect mass tissue in lung images using discrete wavelet transformation. In: *Proceedings of the IEEE International Conference on Information Processing (IICIP)* (New Delhi), (2016):1–10.
19. Santos AM, Ode A, Filho C, Silva AC, Nunes RA. Automatic detection of small lung nodules in 3D CT data using gaussian mixture models, Tsallis entropy and SVM. *Eng Appl Artif Intell*. (2014) 36:27–39. doi: 10.1016/j.engappai.2014.07.007
20. Jinsa K, Gunavathi K. Lung cancer classification using neural networks for CT images. *Comput Methods Programs Biomed*. (2014) 113:202–209. doi: 10.1016/j.cmpb.2013.10.011
21. Anita P, Raja SK, Gandharba S. Digital image steganography using LSB substitution, PVD, and EMD. *Hindawi Math Probl Eng*. (2018) 2018:1–11. doi: 10.1155/2018/1804953
22. Venkatesh C, Polaiah B. An investigation of diverse optimization techniques on medical imagery for detection of perilous diseases. *Front J Soc Technol Environ Sci*. (2017) 62:49—255. doi: 10.1108/IJPCC-10-2020-0160
23. Thippa Reddy G, Khare N, Bhattacharya S, Singh S, Maddikunta PKR, In-Ho Ra, et al. Early detection of diabetic retinopathy using PCA-firefly based deep learning model. *Electronics*. (2020) 9:274. doi: 10.3390/electronics9020274
24. Madhav BTP, Pardhasaradhi P,.Manepalli RKNR, Kishore PVV, Pisipat VGKM. Image enhancement using virtual contrast imagefusion on Fe3O4 and ZnO nanodispersed decyloxybenzoic acid. *Liq Cryst*. (2018) 42:1329–36. doi: 10.1080/02678292.2015.1050704
25. Prasanna MN, Polaiah B. Optimization of rotary kiln in cement industry using conventional control systems. *Helix*. (2019) 9:4843–49. doi: 10.29042/2019-4843-4849
26. Venkatesh C, Bhagyalakshmi K, Sivayamini L. Detection of diverse tumefactions in medial images by various cumulation methods. *Int Res J Eng Technol (IRJET)*. (2017) 4:1195–200.
27. Otsu N. A threshold selection method from gray-level histograms. *IEEE Trans Sys Man Cyber*. (1979) 9:62–6.
28. Xin-She Y. *Cuckoo Search and Firefly Algorithm Theory and Applications*. Springer (2014).
29. Yang XS, Deb S. Cuckoo search via levy flights. In: *Proceedings of World Congress on Nature and Biologically Inspired Computing (NaBIC 2009)* (Coimbatore), (2009):210–14.
30. Ojala T, Pietikainen M, Harwood D. A comparative study of texture measures with classification based on feature distributions. *Pattern Recognit*. (1996) 29:51–9. doi: 10.1016/0031-3203(95)00067-4
31. Iwendi C, Bashir AK, Peshkar A, Sujatha R, Chatterjee JM, Pasupuleti S, et al. COVID-19 patient health prediction using boosted random forest algorithm. *Front Public Health*. (2020) 8:357. doi: 10.3389/fpubh.2020.00357
32. Aparna P, Polurie V, Vijay K. An efficient medical image watermarking technique in e-healthcare application using hybridization of compression and cryptography algorithm. *J Intell Syst*. (2017) 27:115–33. doi: 10.1515/jisys-2017-0266
33. Malakonda Reddy B, Rahman ZUr Md. SAR electromagnetic image conditioning using a new adaptive particle swarm optimization. *ACES J*. (2018) 33:1439–46.
34. Kumar EK, Kishore PVV, Sastry ASCS, Teja Kiran Kumar M, Anil Kumar D. Training CNNs for 3-D sign language recognition with color texture coded joint angular displacement maps. *IEEE Signal Process Lett*. (2018) 25:645–9. doi: 10.1109/LSP.2018.2817179
35. Bhubaneswar P, Brintha Therese A. Detection of cancer in lung with K-NN classification using genetic algorithm. *Proc Mater Sci*. (2015) 10:433–440. doi: 10.1016/j.mspro.2015.06.077
36. Othayoth R, Mathi P, Bheemanapally K, Kakarla L, Botlagunta, M. Characterization of vitamin-cisplatin-loaded chitosan nano-particles for chemoprevention and cancer fatigue. *J Microencapsul*. (2015) 32:578–88. doi: 10.3109/02652048.2015.1065921
37. Svitlana K, Chauhdar SH, Iwendi C, Liu L, Yong W, Bashir AK. Socio-Technological factors affecting user's adoption of eHealth functionalities: a case study of China and Ukraine eHealth systems. *IEEE Access*. (2019) 7:90777–88. doi: 10.1109/ACCESS.2019.2924584
38. Ratna Bhargavi V, Rajesh V. Computer aided bright lesion classification in fundus image based on feature extraction. *Intern J Pattern Recognit Artif Intell*. (2018) 32:1850034. doi: 10.1142/S0218001418500349
39. Venkatesh C, Bojja P. Development of qualitative model for detection of lung cancer using optimization. *Int J Innovative Technol Explor Eng*. (2019) 8:3143–47. doi: 10.35940/ijitee.I8619.078919

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Chemical Data Collections 37 (2022) 100820

Contents lists available at ScienceDirect

Chemical Data Collections

journal homepage: www.elsevier.com/locate/cdc

Thermodynamic properties of active pharmaceutical ingredients that are of interest in COVID-19

Harsha Nagar [a], Dhiraj Ingle [b], Chandan Kumar Munagala [b], Aman Kumar Kesari [b], Vineet Aniya [b,*]

[a] *Department of Chemical Engineering, Chaitanya Bharathi Institute of Technology, Hyderabad, 500075, India*
[b] *Department of Process Engineering and Technology Transfer, CSIR-Indian Institute of Chemical Technology, Hyderabad, 500007, India*

ARTICLE INFO

Keywords:
Thermodynamic Estimations
Active Pharmaceutical Ingredient
COVID-19
Pure Component Properties
Group Contribution Method

ABSTRACT

The pure component properties are estimated for active pharmaceutical ingredients that are related or proposed for the treatment of severe acute respiratory syndrome-CoronaVirus-2. These include Baricitinib, Camostat, Chloroquine, Dexamethasone, Hydroxychloroquine, Fingolimod, Favipiravir, Thalidomide, and Umifenovir. The estimations are based on group contribution$^+$ (GC) models that contain combined group contribution and atom connectivity index with uncertainties in the estimated property values. The thermodynamic properties that are reported include boiling point, critical temperature, critical pressure, critical volume, melting point, standard Gibb's energy of formation, standard enthalpy of formation, enthalpy of fusion, enthalpy of vaporization at 298 K, enthalpy of vaporization at boiling point, entropy of vaporization at boiling point, flash point, Hildebrand solubility parameter, octanol/water partition coefficient, acentric factor, and liquid molar volume at 298 K. The reported properties are not available in the literature and thereby is an incremental development for reliable process engineering.

1. Introduction

The Global Pandemic COVID-19 also known as Severe Acute Respiratory Syndrome-CoronaVirus-2 (SARS-CoV-2) has affected the entire world. The pandemic has involved clinicians around the globe to put in an unprecedented effort to develop a better healthcare system. To date, the World health organization (WHO) has issued an emergency use listing for the Pfizer COVID-19 vaccine (BNT162b2), AstraZeneca/Oxford COVID-19 vaccine, and Ad26.COV2. S (Johnson & Johnson). The other includes Sputnik V (Russia), Covaxin (India) Corovac (China), Sinopharm (China), Kexing (China), and Moderna (USA). Ramdesivir has been approved by FDA and has shown clinical evidence for specific treatment against SARS-CoV-2 [1]. Furthermore, the different vaccines are being underdeveloped around and are at various stages of trials. This pandemic as of November 2021, has resulted in 261,926,070 confirmed cases with 5220, 328 deaths, and 236,538,716 recovery cases, while among the active cases, 20,038,269 cases are in mild condition and 83,

Abbreviation: T_b, normal boiling point; T_c, critical temperature; P_c, critical pressure; V_c, critical volume; T_m, normal melting point; ΔG_f, standard Gibbs energy of formation; ΔH_f, standard enthalpy of formation; ΔH_{fus}, normal enthalpy of fusion; H_v, enthalpy of vaporization at 298 K; H_{vb}, enthalpy of vaporization at the normal boiling point; S_{vb}, the entropy of vaporization at the normal boiling point; F_p, flash point; T_{AiT}, auto-ignition temperature; $\delta_D\delta_P\delta_H$, Hansen solubility parameters; δ, Hildebrand solubility parameter; $LogK_{ow}$, octanol/water partition coefficient; ω, acentric factor; V_m, liquid molar volume at 298 K.

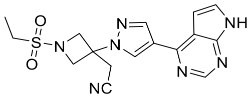
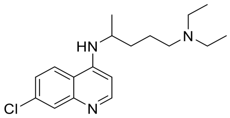
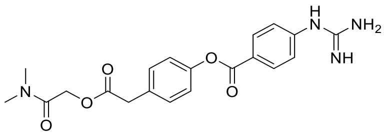
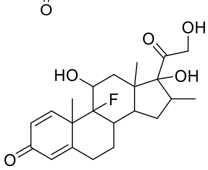
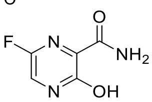
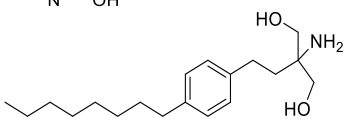
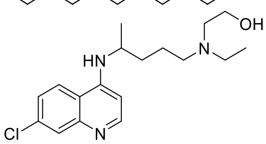
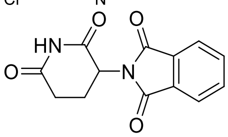
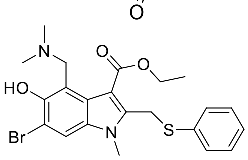
* Corresponding author.
E-mail address: vineet@csiriict.in (V. Aniya).

https://doi.org/10.1016/j.cdc.2021.100820
Received 1 September 2021; Received in revised form 29 November 2021; Accepted 21 December 2021
Available online 23 December 2021

757 cases in a serious or critical condition [2]. Apart from the vaccine, the treatment of the patient by physicians are mainly dependents on the symptom they possess and for the critically ill patients, mainly oxygen therapy or ventilator support is provided. The drugs/ active pharmaceutical ingredient (API) that are presently used or in combination for the treatment of mild symptoms are hydroxychloroquine [3], chloroquine [4] combination of are hydroxychloroquine and azithromycin [5], remdesivir [6–7], lopinavir [8–9] and ritonavir [10]. The candidate drugs in combination are still in clinical trials to combat COVID-19 by various pharmaceutical companies. Likewise, antimalarial drugs such as chloroquine by Sanofi (Aralen) and hydroxychloroquine by CaoSanofi (Plaquenil); Mylan, Teva, Novartis, Bayer, Rising Pharmaceuticals. While the antivirals to combat COVID-19 are Remdesivir by Gilead Sciences; Favipiravir by Fujifilm Toyama Chemical and Umifenovir by Pharm standard. The other drugs such as Baricitinib by Concert Pharmaceuticals, Inc., USA; Dexamethasone by the University of Oxford; Phase-II/III) and fingolimod by Fujian Medical University/Novartis; the clinical stage is presently developed to combat COVID-19.

Once the aforementioned drugs are clinically approved, there will be a requirement of their bulk scale production which in turn requires their physical and chemical thermodynamic properties data set. This data is useful for chemical/process engineers to perform tasks or understand the process design, simulation, and optimization for product development. For the estimation of properties of compounds, the Quantity Structure-Property Relationship method can be used that contains an empirical relationship [11]. This method uses the chemical structure of the compound in which atoms, bonds, groups of atoms in the molecule, topological indices, and molecular descriptors are used for the estimation of properties. Over the year's different empirical relationships based on group

Table 1
Details on the compounds.

Molecule Name	Structure	Formula	Molar Mass (g/mol)	CAS No.
Baricitinib		$C_{16}H_{17}N_7O_2S$	371.42	1,187,594–09–7
Camostat		$C_{20}H_{22}N_4O_5$	398.412	59,721–29–8
Chloroquine		$C_{18}H_{26}ClN_3$	319.872	54–05–7
Dexamethasone		$C_{22}H_{29}FO_5$	392.464	50–02–2
Favipiravir		$C_5H_4FN_3O_2$	157.104	259,793–96–9
Fingolimod		$C_{19}H_{33}NO_2$	307.471	162,359–55–9
Hydroxychloroquine		$C_{18}H_{26}ClN_3O$	335.872	118–42–3
Thalidomide		$C_{13}H_{10}N_2O_4$	258.23	50–35–1
Umifenovir		$C_{22}H_{25}BrN_2O_3S$	477.414	131,707–23–8

contribution (GC) methods such as Joback and Reid, Lydersen, Klincewicz and Reid, Constantino and Gani, and Marrero and Gani has been reported for the estimation of properties of pure organic, inorganic, organometallic, polysaccharides, polymers, and lipid compounds and their mixtures [12–20]. This property includes critical properties [21–23], parameters of state equations [24–25] acentric factor [26–27], activity coefficients [28], vapor pressure [29–30], liquid viscosity [31], gas viscosity [32], heat capacity [33], enthalpy of vaporization [34], entropy of vaporization [34], normal boiling temperature [20–21], liquid thermal conductivity [35], gas thermal conductivity [36], gas permeability and diffusion coefficients [37], liquid density [38–39], surface tension [40] and flash temperatures [41]. The application range and reliability of this method are largely dependent on several factors such as the group definitions used to represent the molecular structure of the pure components; the property model and the quantity and quality of the experimental dataset used in the regression to estimate the model parameters. These GC methods generally do not have all the needed parameters, such as groups and/or their contributions for drugs or larger and complex molecular weight compounds for a specific property. For such special cases, where the molecular structure of a given component is not completely described by any of the available groups, the atom connectivity index (CI) method can be employed together with the GC method to create the missing groups and to predict their contributions. This combined approach leads to the development of a group-contribution+ (GC+) method of a wider application range than before since the missing groups and their contributions can now be easily predicted through the regressed contributions of connectivity indices. The statistical indicators that are used are assessing the parameters for the group contribution method includes standard deviation, average absolute or relative error, and regression coefficient. The inclusion of uncertainty into model parameters are added advantages that are not generally reported. This uncertainty in properties plays an important role in the design and simulations of unit operations such as distillation, liquid-liquid extraction, and others [42]. P.M. Mathias [43] and Hajipour and Satyro [44] have shown the necessity and effect of uncertainties on the optimization calculations using computer-aided software (ASPEN, CAMD, MD). So, in consideration of its importance for reliable and accurate property prediction calculations in engineering design, the present work estimates the properties of important compounds based on GC+ property models. This model is developed by A. S. Hukkerikar et al. [45] that considers a systematic property modeling procedure with an extended CAPEC database that includes new experimental data on various polyfunctional, polycyclic, and complex components with their experimental uncertainty. A total of 3510 compounds that include hydrocarbon, oxygenated, nitrogenated, chlorinated, fluorinated, brominated, iodinated, sulfonated, multifunction compounds are used as data set for the regression and parameter estimation. The model helps to estimate the properties of the compound based on their molecular structure and has shown good accuracy for predicting the properties of the chemical, biochemical, and pharmaceutical compounds.

The present study estimates the pure component properties for 9 APIs that include Baricitinib, Camostat, Chloroquine, Dexamethasone, Hydroxychloroquine, Fingolimod, Favipiravir, Thalidomide, and Umifenovir based on the GC method (**Table 1**). A total of 16 pure component properties are estimated that includes the normal boiling point (T_b), critical temperature (T_c), critical pressure (P_c), critical volume (V_c), normal melting point (T_m), standard Gibbs energy of formation (ΔG_f), standard enthalpy of formation (ΔH_f), normal enthalpy of fusion (ΔH_{fus}), enthalpy of vaporization at 298 K (H_v), enthalpy of vaporization at the normal boiling point (H_{vb}), the entropy of vaporization at the normal boiling point (S_{vb}), flash point (F_p), auto-ignition temperature (T_{AiT}), Hansen solubility parameters (δ_D,δ_P,δ_H), Hildebrand solubility parameter (δ), octanol/water partition coefficient ($LogK_{ow}$), acentric factor (ω), and liquid molar volume at 298 K (V_m). Thereby, the present work is an important source for knowledge about possible drug candidates or active pharma ingredients that are of prime interest shortly

2. Model and methodology

The details about the model development and methodology are reported by A.S. Hukkerikar et al. [45] and Marrero and Gani (MG) [46]. In brief, the estimation of properties is based on a collection of 3 different types of groups viz. 1st order, 2nd order, and 3rd order are present in the compound. In the 1st order, simple molecules are considered that allow estimating the contributions to the property of different classes of organic compounds. The larger group or polycyclic, polyfunctional, and heterocyclic are not be considered here and each group is be kept as small as possible. The entire molecule needs to be covered and no fragments of the given should be left out in 1st order estimations. The overlapping is not allowed and the contributions are independent of the molecule in which the group has occurred. While in the case of 2nd and 3rd order groups, the information/contribution of the molecular fragments or the structural information is considered that otherwise is not provided by the 1st order group. The contributions of the polyfunctional and isomeric compounds are better described by the 2nd order group and the entire molecule need not be covered/described as that in the case of 1st order. Partial overlapping is allowed but one group should not completely overlap the others and in such case the molecule with the complete overlapping need to be considered. 2nd order group fails to provide the information of multi-ring compounds. The information about a multi-ring compound or fused aromatic rings, non-aromatic rings, and non-fused rings joined by chains with the different functional groups are covered in the 3rd order groups. The property prediction model with multilevel successive contribution can be described by the following general Eq. (1) given by MG:

$$f(x) = \sum_i N_i C_i + w \sum_j M_j D_j + z \sum_k O_k E_k \tag{1}$$

In this equation, the function $f(x)$ is dependent on the property X. The C_i is the contribution of the 1st order group of type i that has an occurrence of N_i times, $D_j$is the contribution of the 2nd order group of type j that has an occurrence of M_j times and E_k is the contribution of the 3rd order group of type k that has an occurrence of O_k times in the molecule. In the first step, the value of w and z are set zero for the 1st level of estimation of a given property with C_i contribution. In the second step case of the 2nd level of estimation, the constants w and z are assigned unity and zero values, respectively because only 1st and 2nd order groups are involved while in the 3rd level, both w and z are set to unity values. The property function $f(x)$ is used to define the different properties and is detailed in Table 2. The universal constant/adjustable parameters required for the estimation of 16 thermodynamic properties are reported in Table 3. The MG method reported herein is analyzed through step-wise regression method (STRM) and simultaneous regression method (SIRM) and the results are detailed in the next section.

The deviation in the estimated thermodynamic properties was used using absolute relative deviation (ARD) that is defined as

$$ARD = \frac{x_i - x_j}{x_j} \times 100 \quad (2)$$

Here, x_j is the experimental thermodynamic property and x_i is predicated on the thermodynamic property based on STRM and SIRM.

3. Result and discussion

The thermodynamic properties of the API were estimated based on the group contribution^{+} (GC) model that contains combined group contribution and the atom connectivity index. The model parameters considered have standard uncertainties in the prediction of the thermodynamic property. Each drug/molecule is split into different subgroups at each level for the estimation of the property. Table 4 reports the subgroups, group number, and their occurrence that are considered for predicting the thermodynamic properties of Baricitinib, Camostat, Chloroquine, Dexamethasone, Hydroxychloroquine, Fingolimod, Favipiravir, Thalidomide, and Umifenovir. Based on their contribution at each level i.e., contributions to 1st order group, 2nd order group, and 3rd order groups, the overall contribution to thermodynamic functions $f(x)$ is calculated using Eq. (1). **Table S1** reports the detailed contribution that is considered for the estimation of thermodynamic properties (T_b, T_c, P_c, V_c, T_m, ΔG_f, ΔH_f, ΔH_{fus}, H_v, H_{vb}, S_{vb}, F_p,δ, $LogK_{ow}$, ω, and V_m) based stepwise (STRM) and simultaneous (SIRM) regression methods. While the estimated thermodynamic properties are reported in **Table 5.** Among the 16 thermodynamic properties estimated for the 9 APIs, only the experimental normal melting point (T_m) and for a few octanol/water partition coefficients ($LogK_{ow}$) is reported in the open literature and is mentioned also mentioned in Table 5. The other thermodynamic properties are not found in the literature to the best of our knowledge. The estimated normal melting point (T_m) and octanol/water partition coefficient ($LogK_{ow}$) was therefore compared with that of literature for the performance evaluation of the mentioned GC method. The statistical performance indicator used in the present study is an absolute relative deviation (ARD).

The estimated T_m for the Baricitinib was found to be 492.478 K that showed an ARD of 1.093 (STRM) with that of reported A. S. Alshetaili et al. [47]. While the enthalpy of fusion ΔH_{fus} showed a high ARD of 18.7 with that reported in the literature [47]. The Hansen solubility (δ) for the Baricitinib was found to be 27.197 $MPa^{1/2}$ and was closer with reported literature [47] with a value of 28.90 $MPa^{1/2}$. While the octanol/water partition coefficient ($LogK_{ow}$) is estimated to be 0.252 and consistent with Pengfei Xu et al. (0.24) [48]. The Camostat estimated T_m was found to be 497.05 K (SIRM) with an ARD of 2.85 with that of experimental data reported by J. Yin et al. [49]. While the Chloroquine estimated T_m was found to be 385.54 K against the reported value of 363.15 K by M. Staderini et al. [50]. The estimated T_m for Dexamethasone showed an ARD of 10.87 with that of reported T_m of 524.60 K [51] and was mainly with complex structure and fluorinated compounds present in it. Also, the normal enthalpy of fusion (ΔH_{fus}) was found to be 32.55 kJ/mol and that showed very high ARD (29.27) with reported by X. Cai et al. [50]. The estimated T_m for Favipiravir was found to

Table 2
Property, function and group contributions for the estimation of properties of the compound.

Property (x)	Function (f(x))	Group Contribution terms
Normal boiling point (T_b)	$exp(T_b / T_{b0})$	$\sum_i N_i T_{b1i} + \sum_j M_j T_{b2j} + \sum_k O_k T_{b3k}$
Critical temperature (T_c)	$exp(T_c / T_{c0})$	$\sum_i N_i T_{c1i} + \sum_j M_j T_{c2j} + \sum_k O_k T_{c3k}$
Critical pressure (P_c)	$(P_c - - P_{c1}) - 0.5 - - P_{c2}$	$\sum_i N_i P_{c1i} + \sum_j M_j P_{c2j} + \sum_k O_k P_{c3k}$
Critical volume (V_c)	$V_c - - V_{c0}$	$\sum_i N_i V_{c1i} + \sum_j M_j V_{c2j} + \sum_k O_k V_{c3k}$
Normal melting point (T_m)	$exp(T_m / T_{m0})$	$\sum_i N_i T_{m1i} + \sum_j M_j T_{m2j} + \sum_k O_k T_{m3k}$
Standard Gibbs energy of formation (G_f)	$G_f - - G_{f0}$	$\sum_i N_i G_{f1i} + \sum_j M_j G_{f2j} + \sum_k O_k G_{f3k}$
Standard enthalpy of formation (H_f)	$H_f - - H_{f0}$	$\sum_i N_i H_{f1i} + \sum_j M_j H_{f2j} + \sum_k O_k H_{f3k}$
Standard enthalpy of vaporization at 298 K (H_v)	$H_v - - H_{v0}$	$\sum_i N_i H_{v1i} + \sum_j M_j H_{v2j}$
Normal enthalpy of fusion (H_{fus})	$H_{fus} - - H_{fus0}$	$\sum_i N_i H_{fus1i} + \sum_j M_j H_{fus2j} + \sum_k O_k H_{fus3k}$
Octanol/Water partition coefficient ($LogK_{ow}$)	$LogK_{ow} - K_{ow0}$	$\sum_i N_i LogK_{ow1i} + \sum_j M_j LogK_{ow2j} + \sum_k O_k LogK_{ow3k}$
Flash point (F_p)	$F_p - F_{p0}$	$\sum_i N_i F_{p1i} + \sum_j M_j F_{p2j} + \sum_k O_k F_{p3k}$
Enthalpy of vaporization at normal boiling point (H_{vb})	$H_{vb} - H_{vb0}$	$\sum_i N_i H_{vb1i} + \sum_j M_j H_{vb2j} + \sum_k O_k H_{vb3k}$
Entropy of vaporization at normal boiling point (S_{vb})	$S_{vb} - S_{vb0}$	$\sum_i N_i S_{vb1i} + \sum_j M_j S_{vb2j} + \sum_k O_k S_{vb3k}$
Hildebrand solubility parameter (δ)	$\delta - \delta_0$	$\sum_i N_i \delta_{D1i} + \sum_j M_j \delta_{D2j} + \sum_k O_k \delta_{D3k}$
Acentric factor (ω)	$exp\left(\frac{\omega}{\omega_a}\right)^{\omega_b} - \omega_c$	$\sum_i N_i \omega_{1i} + \sum_j M_j \omega_{2j} + \sum_k O_k \omega_{3k}$
Liquid molar volume (V_m)	$V_m - V_{m0}$	$\sum_i N_i V_{m1i} + \sum_j M_j V_{m2j} + \sum_k O_k V_{m3k}$

Table 3
Value of universal constants for different methods.

Universal Constants	Units	Values Step-wise Method	Simultaneous Method
T_{b0}	$[K]$	244.5165	244.7889
T_{c0}	$[K]$	181.6716	181.6738
P_{c1}	$[bar]$	0.0519	0.0519
P_{c2}	$[bar - 0.5]$	0.1347	0.1155
V_{c0}	$[cc/mol]$	28.0018	14.6182
T_{m0}	$[K]$	143.5706	144.0977
G_{f0}	$[kJ/mol]$	−1.3385	8.5016
H_{f0}	$[kJ/mol]$	35.1778	83.9657
H_{fus0}	$[kJ/mol]$	−1.7795	−1.2993
K_{ow0}	$[--]$	0.4876	0.752
F_{p0}	$[K]$	170.7058	150.0218
H_{v0}	$[kJ/mol]$	9.6127	10.4327
H_{vb0}	$[kJ/mol]$	15.4199	15.0884
ı0	$[MPa1/2]$	21.6654	20.7339
ω_a	$[--]$	0.908	0.9132
ω_b	$[--]$	0.1055	0.0447
ω_c	$[--]$	1.0012	1.0039
V_{mo}	$[cc/kmol]$	0.016	0.0123
Ait_1		0	71.2584
Ait_2	$[K]$	0	525.93

be 465.51 K with a relatively low deviation of 3.41 K with that reported by Q. Guo et al. [52]. Fingolimod showed the lowest ARD with an estimated T_m of 398.69 K and 396.30 K based on STRM and SIRM method. An ARD of 0.36 and 0.95 was found with that reported by S. R. Shaikh et al. [53]. H. Gunaydin reported the octanol/water partition coefficient ($LogK_{ow}$) of 2.8 that has a percentage ARD of 0.29 with STRM (4.5) [54] The estimated T_m for Hydroxychloroquine is 417.16 K (STRM) with an ARD 12.05 [55]. Also, the estimated octanol/water partition coefficient based on STRM ($LogK_{ow}$) (3.22) was found coherent with that reported by M. Nimgampalle et al. with a value of 3.58 [56]. Similar to dexamethasone, Thalidomide estimated $T_m(441.49\ K)$ should a high ARD of 18.71 with that of reported T_m of 543.15 K by et al. B.D. Vu et al. [57]. The presence of carboxyl carbonyl groups and fused aromatic rings are the possible reason for larger deviations. The Umifenovir estimated T_m was found to be 447.21 K (STRM) with an ARD of 7.72 58. A. Kons et al. [58].

Based on the above statistical analysis of T_m the overall average relative deviation for all the APIs was found to be 7.27 and 8.39 for STRM and SIRM methods, respectively. This relative deviation in the predicted properties is related to the experimental data set that was used for regression and the estimation of universal constants for empirical correlations. The group-contribution+ (GC+) method used in the present work is developed from the DIPPR 801® databank that has used experimental data with reported uncertainties. The experimental data itself has standard uncertainties in the measurements. For example, the normal boiling point [K] with data points of 1306 has an experimental average measurement error of 6.32% while that of predication based on the GC + method has 6.17%. Similarly, the deviations are seen for Normal melting point [K] with data points of 1385 has an experimental average measurement error of 5.10 while that of predication has 15.99. It has been found that for the 16 estimated properties the prediction error is lower than (or at least comparable to) the average measurement error, except for the case of normal melting point (T_m) and standard enthalpy of fusion (ΔH_{fus}). For these two properties, group contribution methods, in general, have difficulties in providing a reliable estimation. This is mainly due to the strong dependency of the melting point on intermolecular interaction and molecular symmetry. With low deviation in STRM and SIRM method, the present GC+ method showed its capability to accurately predict the thermodynamic properties. The predicted thermodynamic properties not perfect/exact (lack of experimental value) still provide the basis for engineering design.

4. Conclusion

The group contribution (GC) method was used to estimate thermodynamic properties for the drugs/compounds/API that are related or proposed for the treatment of severe acute respiratory syndrome-CoronaVirus-2. The GC method based on stepwise regression parameters showed a low average deviation for the melting point. A total of 16 thermodynamic properties are reported for 9 API which is helpful in the product-process design, simulation, and optimization calculations. The properties contribute to reliable and robust engineering solutions for pharmaceutical product development.

Table 4
Group orders, group number and their occurrence in each compound for the estimation of thermodynamic properties.

Baricitinib Groups	Gr. No.	FN	Camostat Groups	Gr. No.	FN	Chloroquine Groups	Gr. No.	FN	Dexamethasone Groups	Gr. No	FN
First-order			First-order			First-order			First-order		
CH_3	1	1	CH_3	1	1	CH_3	1	1	CH_3	1	3
CH_2	2	1	CH_2	2	9	CH_2	2	9	OH	29	3
aCH	15	5	aCH	15	4	CH	3	1	CH_2CO	34	1
aC fused with aromatic ring	16	2	aC except as above	18	1	aCH	15	4	CF	116	1
aC except as above	18	2	CH_2CO	34	1	aC fused with aromatic ring	16	1	CH_2 (cyclic)	168	4
aN in-aromatic ring	19	5	aC—CO	37	1	aN in-aromatic ring	19	2	CH(cyclic)	169	4
CH_2CN	68	1	CH_2–COO	41	1	CH_2N	61	2	C (cyclic)	170	3
SO_2	149	1	aC—O	53	1	aC—NH	63	1	CH=CH (cyclic)	171	1
CH_2 (cyclic)	168	2	CH_3N	60	1	aC—Cl	123	1	CH=C (cyclic)	172	1
C (cyclic)	170	1	aC—NH	63	1				CO (cyclic)	180	1
N (cyclic)	176	1	NH_2 expect as above	65	1						
			C = N	67	1						
Second-order			**Second-order**			**Second-order**			**Second-order**		
Ccyc-CH_2	100	1	aC—CH_2–COO	64	1	No Occurrences			CHcyc-OH	84	1
AROMRING s^1s^4	106	1	AROMRING s^1s^4	106	2				Ccyc-CH_3	99	2
									Ccyc-OH	101	1
									CHcyc-CH_3	76	1
Third orders			**Third-order**			**Third-order**			**Third-order**		
aC- aC (different ring)	15	1	aC—O-C-aC (different rings)	47	1	ARMOFUSED [2]S^1S^4	55	1	CH multiring	22	2
AROMFUSED S^1	52	1									

Favipiravir Groups	Gr. No.	FN	**Fingolimod** Groups	Gr. No.	FN	**Hydroxychloroquine** Groups	Gr. No.	FN	**Thalidomide** Groups	Gr. No	FN	**Umifenovir** Groups	Gr. No.	FN
First-order			First-order			First-order			First-order			**First-order**		
aCH	15	1	CH_3	1	1	CH_3	1	2	aCH	15	4	CH_3	1	1
aN in- AR	19	2	CH_2	2	9	CH_2	2	5	aC fused with non-AR	17	2	CH_2	2	9
aC—OH	30	1	aCH	15	4	CH	3	1	CH(cyclic)	169	1	aCH	15	4
aC—$CONH_2$	94	1	C	4	1	aCH	15	1	CH_2 (cyclic)	168	2	aC fused with AR	16	1
NH_2 expect as above	65	1	NH_2 expect as above	65	1	aC fused with AR	16	2	NHCO except as above	107	1	aN in- AR	19	2
aC—CO	37	1	OH	29	2	aN in- AR	19	2	N(cyclic)	176	1	aCH_2	21	2
aC-F	124	1	aC—CH_2	21	2	OH	29	1	CO(cyclic)	180	3	aC—OH	30	1
						CH_2N	61	1				aC—COO	45	1
						aC—Cl	123	1				CH_3N	60	1
												aC-Br	126	1
												aC-S-	147	1
Second-order			**Second-order**			**Second-order**			**Second Order**			**Second Order**		
AROMRING s^1 s^2s^4	108	1				No Occurrences			No Occurrences			AROMFUSED [2]	51	1
												aC—CH_2-S-	59	1
Third-order						**Third-order**			**Third-order**			**Third-order**		
No Occurrences						ARMOFUSED [2]S^1S^4	55	1	aC- CO cyc (Fused rings)	32	2	PYRIDINE FUSED [2]	64	1
									AROMFUSED [2]	51	1			

FN: Occurrences.
Gr.No: Group Number.
AR: Aromatic Ring.

Table 5
Estimated properties of compounds based on stepwise regression method (STRM) and simultaneous regression method (SIRM).

Property	Units	Baricitinib			Camostat			Chloroquine			Dexamethasone			Favipiravir		
		STRM	SIRM	[47]	STRM	SIRM	[49]	STRM	SIRM	[50]	STRM	SIRM	[51]	STRM	SIRM	[52]
T_b	[K]	794.469	760.077		757.482	771.300		688.550	682.652		722.347	717.264		618.948	688.871	
T_c	[K]	997.035	973.995		934.420	949.155		907.967	907.964		905.005	905.004		815.160	815.158	
P_c	[bar]	0.076	0.091		0.140	0.145		0.152	0.147		0.100	0.109		0.063	0.064	
V_c	[cc/mol]	58.885	903.061		1381.063	1410.628		1196.016	1175.118		−1061.953	1080.050		299.645	326.270	
T_m	[K]	492.478	458.127	487.15	453.792	468.250	467.15	393.375	385.545	363.15	467.128	462.022	524.15	465.514	497.056	450.15
G_f	[kJ/mol]	632.951	395.881		−122.228	−121.518		737.893	757.660		−536.591	−574.299		−187.165	−173.038	
H_f	[kJ/mol]	223.416	107.378		−706.079	−634.774		145.400	259.584		−1054.777	−994.887		−385.972	−417.504	
H_{fus}	[kJ/mol]	100.855	49.657		66.271	66.260		70.036	51.288		32.559	33.055		53.974	46.568	
$\log Kw_0$		0.5811	2.32		5.4578	3.6913		5.463	6.0219		1.743	1.844		−0.867	−2.097	
F_P	[K]	–	607.007		–	150.022		570.936	552.786		723.751	736.511		–	–	
H_V	[kJ/mol]	–	118.800		–	10.433		149.382	154.790		184.173	182.672		118.959	124.073	
H_{Vb}	[kJ/mol]	–	–		–	–		–	–		143.507	144.740		–	–	
S_{Vb}	[kJ/mol]	–	–		–	–		–	–		204.871	207.947		–	–	
δ	[MPa$^{1/2}$]	27.197	27.445		23.924	26.165		14.350	16.843		27.291	26.661		27.886	27.635	
ω		0.013	−0.002		0.023	–		0.013	−0.001		0.024	0.018		0.015	−0.001	
V_m	[cc/kmol]	0.251	0.301		0.507	–		0.333	0.333		0.278	0.462		0.134	0.129	

Property	**Units**	**Fingolimod**			**Hydroxychloroquine**			**Thalidomide**			**Umifenovir**		
		STRM	SIRM	[53]	STRM	SIRM	[55]	STRM	SIRM	[57]	STRM	SIRM	[58]
T_b	[K]	687.396	689.003		681.390	714.345		648.734	650.895		776.060	772.496	
T_c	[K]	848.840	848.839		909.231	920.028		936.906	931.518		973.718	973.718	
P_c	[bar]	0.106	0.115		1.335	0.142		2.419	0.069		0.142	0.146	
V_c	[cc/mol]	1058.116	1069.218		865.489	1199.529		566.423	575.560		1292.282	–	
T_m	[K]	398.693	396.310	400.15	417.170	414.588	367.1	441.490	440.390	543.15	447.213	448.603	415
G_f	[kJ/mol]	−21.235	−2.948		346.872	595.810		−229.399	−255.484		171.261	–	
H_f	[kJ/mol]	−478.253	−448.244		−63.171	45.765		−456.385	−446.716		−397.082	–	
H_{fus}	[kJ/mol]	48.469	46.357		58.150	55.237		23.906	32.053		70.472	–	
$\log Kw_0$		4.532	4.064		3.2266	4.6854		0.0915	0.3946		6.251	6.1162	
F_P	[K]	615.936	616.728		594.016	640.443		567.197	570.199		–	–	
H_V	[kJ/mol]	162.707	162.094		144.748	178.760		147.419	152.094		–	–	
H_{Vb}	[kJ/mol]	98.831	–		–	–		99.785	159.024		–	–	
S_{Vb}	[kJ/mol]	134.617	–		–	–		168.272	181.299		–	–	
δ	[MPa$^{1/2}$]	28.506	29.829		18.645	19.895		23.247	27.758		21.665	–	
ω		0.022	–		0.013	–		0.011	−0.002		0.019	−0.013	
V_m	[cc/kmol]	0.360	0.294		14.941	0.337		0.168	0.160		0.419	−1.453	

Disclosure statement

No potential conflict of interest was reported by the authors.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

We thank Director, CSIR-IICT (Ms. No. IICT/Pubs./2021/113) for providing all the required facilities to carry out the work. The authors also acknowledge financial support sponsored by CSIR under IICT-FBR Project (MLP0073).

References

[1] https://www.who.int/news–room/q–a–detail/coronavirus–disease–(covid–19)–vaccines.
[2] https://www.worldometers.info/coronavirus/?fbclid=IwAR35ZFiRZJ8tyBCwazX2N–k7yJjZOLDQiZSA_MsJAfdK74s8f2a_Dgx4iVk.
[3] X. Yao, F. Ye, M. Zhang, C. Cui, B. Huang, P. Niu, X. Liu, L. Zhao, E. Dong, C. Song, S. Zhan, R. Lu, H. Li, W. Tan, D. Liu, In vitro antiviral activity and projection of optimized dosing design of hydroxychloroquine for the treatment of severe acute respiratory syndrome coronavirus 2 (SARS–CoV–2), Clin. Infect. Dis. 71 (2020) 732–739.
[4] J. Fantini, C.D. Scala, H. Chahinian, N. Yahi, Structural and molecular modelling studies reveal a new mechanism of action of chloroquine and hydroxychloroquine against SARSCoV–2 infection, Int. J. Antimicrob. Agents. 55 (2020), 105960.
[5] P. Gautret, J.C. Lagier, P. Parolaa, V.T. Hoang, L. Meddeba, M. Mailhea, B. Doudier, J. Courjone, V. Giordanengo, V.E. Vieira, H.T. Dupont, S. Honoréi, P. Colsona, E. Chabrièrea, B.L. Scolaa, J.M. Rolaina, P. Brouqui, D. Raoult, Hydroxychloroquine and azithromycin as a treatment of COVID–19: results of an open–label non–randomized clinical trial, Int. J. Antimicrob. Agents. 56 (2020), 105949.
[6] M. Wang, R. Cao, L. Zhanget, Remdesivir and chloroquine effectively inhibit the recently emerged novel coronavirus (2019–nCoV) in vitro, Cell Res 30 (2020) 269–271.
[7] B. Cao, Y. Wang, D. Wen, W. Liu, J. Wang, G. Fan, L. Ruan, B. Song, Y. Cai, M. Wei, X. Li, J. Xia, A Trial of Lopinavir–Ritonavir in Adults Hospitalized with Severe Covid–19, N Engl J Med. 382 (2020) 1787–1799.
[8] Clinical Trials. Gov website: https://clinicaltrials.gov/ct2/show/NCT04470427, 2020, A Study to Evaluate Efficacy, Safety, and Immunogenicity of mRNA–1273 Vaccine in Adults Aged 18 Years and Older to Prevent COVID–19.
[9] J.D. Williams, Prediction of melting and heat capacity of inorganic liquids by the method of group contributions, Thesis, New Mexico State Univ., Las Cruces, NM, USA, 1997.
[10] A.J. Briard, M. Bouroukba, D. Petitjean, M. Dirand, Models for Estimation of Pure n–Alkanes' Thermodynamic Properties as a Function of Carbon Chain Length, J. Chem. Eng. 48 (2003) 1508–1516.
[11] V. Majer, V. Svoboda, J. Pick, 1989 Heats of Vaporization of Fluids, Elsevier, 1989, 044498920.
[12] A.J.L. Costa, J.M.S.S. Esperança, I.M. Marrucho, L.P.N. Rebelo, Densities and Viscosities of 1–Ethyl–3–methylimidazolium n–Alkyl Sulfates, J. Chem. Eng. Data. 56 (2011) 3433–3441.
[13] E.D. Nikitin, A.P. Popov, Y.G. Yatluk, V.A. Simakina, Critical Temperatures and Pressures of Some Tetraalkoxytitaniums, J. Chem. Eng. Data 55 (2010) 178–183.
[14] A. Papaioannou, S. Morin, A.M. Cheung, S. Atkinson, J.P. Brown, S. Feldman, D.A. Hanley, A. Hodsman, S.A. Jamal, S.M. Kaiser, B. Kvern, K. Siminoski, W. D. Leslie, Clinical practice guidelines for the diagnosis and management of osteoporosis in Canada: summary, Can. Med. Assoc. J. 182 (2010) 1864–1873.
[15] O. Lobanova, K. Mueller, L. Mokrushina, W. Arlt, Estimation of Thermodynamic Properties of Polysaccharides, Chem. Eng. Technol. 34 (2011) 867–876.
[16] K.C. Satyanarayana, R. Gani, J. Abildskov, Polymer property modeling using grid technology for design of structured products, Fluid Ph. Equilibria. 261 (2007) 58–63.
[17] S.Y. Oh, Y.C. Bae, Group contribution method for group contribution method for estimation of vapor liquid equilibria in polymer solutions, Macromol. Res. 17 (2009) 829–841.
[18] C.A.D. Tovar, R. Gani, B. Sarup, Lipid technology: property prediction and process design/analysis in the edible oil and biodiesel industries, Fluid Ph. Equilibria. 302 (2011) 284–293.
[19] J.S. Browna, C. Ziliob, A. Cavallini, Thermodynamic properties of eight fluorinated olefins, Int J Refrig 33 (2010) 235–241.
[20] R. Gani, L. Constantinou, Molecular structure–based estimation of properties for process design, Fluid Ph, Equilibria 116 (1996) 75–86.
[21] K.G. Joback, R.C. Reid, Estimation of pure–component properties from group–contributions, Chem. Eng. Commun. 57 (1987) 233–243.
[22] D. Ambrose, Correlation and Estimation of Vapour–Liquid Critical Properties. II. Critical Pressures and Volumes of Organic Compounds. National Physical Laboratory, Teddington: NPL Rep, Chem. 98 (1979).
[23] D. Ambrose, Correlation and Estimation of Vapour–Liquid Critical Properties. I. Critical Temperatures of Organic Compounds. National Physical Laboratory, Teddington: NPL Rep, Chem 92 (1978).
[24] S. Pereda, E. Brignole, S. Bottini, Equations of state in chemical reacting systems. In: applied Thermodynamics of Fluids, Royal Society of Chemistry; 1st Ed. (2011) 433–459.
[25] B. Schmid, J. Gmehling, Revised parameters and typical results of the VTPR group contribution equation of state, Fluid Ph. Equilibria. 317 (1012) 110–126.
[26] L. Constantinou, R. Gani, New group contribution method for estimating properties of pure compounds, AIChE J. 40 (1994) 1697–1710.
[27] K. Manago, C. Otobrise, Estimation of pure–component properties of fatty acids and esters from group contributions, J. Chem. Soc. Nigeria. (2010), 352142148.
[28] K. Tochigi, S. Kurita, Y. Okitsu, K. Kurihara, Measurement and Prediction of Activity Coefficients of Solvents in Polymer Solutions Using Gas Chromatography and a Cubic–Perturbed Equation of State with Group Contribution, Fluid Ph. Equilibria. 228 (2005) 527–533.
[29] B.E. Poling, J.M. Prausnitz, J.P. O' Connell, The properties of gases and liquids, Fifth Edition, ISBN: 9780070116825, McGraw-Hill Education.
[30] D.G. Miller, Estimating Vapor Pressures–Comparison of Equations, Ind. Eng. Chem. 56 (1964) 46–57.
[31] E. Conte, A. Martinho, H.A. Matos, R. Gani, Combined Group–Contribution and Atom Connectivity Index–Based Methods for Estimation of Surface Tension and Viscosity, nd. Eng, Chem. Res. 47 (2008) 7940–7954.
[32] D. Reichenberg, New methods for the estimation of the viscosity coefficients of pure gases at moderate pressures (with particular reference to organic vapors), AIChE Journal 21 (1975) 181–183.
[33] V.R. Jr, E.S. Domalski, Estimation of the Heat–Capacities of Organic Liquids as a Function of Temperature Using Group Additivity. 1. Hydrocarbon Compounds, J Phys Chem Ref Data 22 (1993) 597.
[34] Z. Kolská, V. Růžička, R. Gani, Estimation of the Enthalpy of Vaporization and the Entropy of Vaporization for Pure Organic Compounds at 298.15 K and at Normal Boiling Temperature by a Group Contribution Method, Ind. Eng. Chem. Res. 44 (2005) 8436–8454.
[35] M. Nagvekar, T.E. Daubert, A Group Contribution Method for Liquid Thermal Conductivity, Ind. Eng. Chem. Res. 26 (1987) 1362–1365.

[36] T.H. Chung, L.L. Lee, K.E. Starling, Applications of Kinetic Gas Theories and Multiparameter Correlation for Prediction of Dilute Gas Viscosity and Thermal Conductivity, Ind. Eng. Chem. Fundam. 23 (1984) 8–13.
[37] Y. Yampolskii, S. Shishatskii, A. Alentiev, K. Loza, Group Contribution Method for Transport Property Predictions of Glassy Polymers: focus on Polyimides and Polynorbornenes, J. Membr. Sci. 149 (1998) 203–220.
[38] M.S. Cruz, G.A. Aca, O.S. Daza, T.L. Arenas, Predicting critical properties, density and viscosity of fatty acids, triacylglycerols and methyl esters by group contribution methods, Computer–Aided Chemical Engineering (2010), 2817631768.
[39] K. Shahbaz, S. Baroutian, F.S. Mjalli, M.A. Hashim, Densities of ammonium and phosphonium based deep eutectic solvents: prediction using artificial intelligence and group contribution techniques, Thermochim Acta 527 (2012) 59–66.
[40] A. Awasthi, B.S. Tripathi, A. Awasthi, Applicability of corresponding–states group–contribution methods for the estimation of surface tension of multicomponent liquid mixtures at 298.15 K, Fluid Ph, Equilibria. 287 (2) (2010) 151–154.
[41] H. Liaw, V. Gerbaud, Y. Li, Prediction of miscible mixtures flash–point from UNIFAC group contribution methods, Fluid Ph, Equilibria. 300 (2011) 70–82.
[42] V. Aniya, A. Singh, D. De, B. Satyavathi, An energy efficient route for the dehydration of 2–Methylpropan–2–ol: experimental investigation, modeling and process optimization, Sep, Purif. Technol. 156 (2015) 738–753.
[43] P.M. Mathias, A. Soto, L. Fele–Zilnik, J.C. de Hemptinne, A. Bazyleva, J. Abildskov, Data Quality and Assessment, Validation Methods and Error Propagation through the Simulation Software: report from the Round–Table Discussion at the 10th World Congress of Chemical Engineering in Barcelona (October 1–5, 2017), Chem Eng Res Des. 137 (2018) A1–A8.
[44] P.M. Mathias, A. Soto, L. Fele–Zilnik, J.C. de Hemptinne, A. Bazyleva, J. Abildskov, S. Hajipour, M.A. Satyro, M.W. Foley, Uncertainty Analysis Applied to Thermodynamic Models and Fuel Properties – Natural Gas Dew Points and Gasoline Reid Vapor Pressures, Energy Fuels 28 (2014) 1569–1578.
[45] A. Shivajirao, H. Bent, S. Antoon, T. Katec, J. Abildskova, G. Sina, R. Gania, Group–contribution+ (GC+) based estimation of properties of pure components: improved property estimation and uncertainty analysis, Fluid Ph. Equilibria. 321 (2012) 25–43.
[46] J. Marrero, R. Gani, Group-contribution based estimation of pure component properties, Fluid Ph. Equilibria. 183–184 (2001) 183–208.
[47] A.S. Alshetaili, Solubility and Solution Thermodynamics of Baricitinib in Six Different Pharmaceutically Used Solvents at Different Temperatures, Z. Phys. Chem. 233 (2019) 1129–1144.
[48] P.Shen P.Xu, H. Wang, L. Qin, J. Ren, Q. Sun, J.Bian R.Ge, Y. Zhong, Z. Li, J. Wang, Z. Qiu, Discovery of imidazopyrrolopyridines derivatives as novel and selective inhibitors of JAK2, Eur. J. Med. Chem. 218 (2021), 113394.
[49] J. Yin, Y. Noda, N. Hazemoto, Toshihisa Yotsuyanagi, Distribution of Protease Inhibitors in Lipid Emulsions: gabexate Mesilate and Camostat Mesilate, Chem. Pharm. Bull. 53 (8) (2005) 893–898.
[50] M. Staderini, M.L. Bolognesi, J.C. Menndez, Lewis Acid-Catalyzed Generation of C–C and C–N Bonds on π-Deficient Heterocyclic Substrates, Adv. Synth. Catal. 357 (1) (2015) 185–195.
[51] X. Cai, D.J.W. Grant, T.S. Wiedmann, Analysis of the Solubilization of Steroids by Bile Salt Micelles, J. Pharm. Sci. 86 (1997) 372–377.
[52] Q. Guo, M. Xu, S. Guo, F. Zhu, Y. Xie, J. Shen, The complete synthesis of favipiravir from 2–aminopyrazine, Chem. pap. 73 (2019) 1043–1051.
[53] R.S. Shaikh, S.S. Schilson, S. Wagner, S. Hermann, P. Keul, B. Levkau, M. Schäfers, G. Haufe, Synthesis and Evaluation of Fluorinated Fingolimod (FTY720) Analogues for Sphingosine–1–Phosphate Receptor Molecular Imaging by Positron Emission Tomography, J. Med. Chem. 58 (2015) 3471–3484.
[54] H. Gunaydin, Probabilistic Approach to Generating MPOs and Its Application as a Scoring Function for CNS Drugs, ACS Med. Chem. Lett. 7 (2016) 89–93.
[55] P. Joshi, S. Dhaneshwar, Novel drug delivery of dual acting prodrugs of hydroxychloroquine with aryl acetic acid NSAIDs: design, kinetics and pharmacological study, Drug Deliv. and Transl. Res. 7 (2017) 709–730.
[56] M. Nimgampalle, V. Devanathan, A. Saxena, Screening of Chloroquine, Hydroxychloroquine and its derivatives for their binding affinity to multiple SARS-CoV-2 protein drug targets, J. Biomol. Struct. Dyn. (2020), https://doi.org/10.1080/07391102.2020.1782265.
[57] B.D. Vu, N.M.H. Ba, D.C. Phan, Facile Synthesis of Thalidomide,Org, Process Res. Dev. 23 (2019) 1374–1377.
[58] A. Kons, A. Berzins, K. Krukle-Berzina, A. Actins, Characterization and physicochemical evaluation of molecular complexes formed between umifenovir and dicarboxylic acids, Latv. J. Chem. 52 (2014) 28–40.

Certificate ID: IICHE/2022/AD12242
Date of Issue: 30th May, 2022

Online Internship Program (OIP)

25th February to 30th May, 2022 (40 hrs)

Indian Institute of Chemical Engineers

Dr. H. L. Roy Building, Jadavpur University Campus, Kolkata- 700 032

CERTIFICATE OF COMPLETION

This certificate is hereby awarded to

ABHITHA SUGGALA

from CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY

who has successfully completed the INTERNSHIP PROGRAMME *on the subject* PETROCHEMICAL ENGINEERING *following all necessary criteria of the Institute with " A " Grade.*

Grading System :

A+: 90-100%
A: 75-89%
B+: 65-74%
B: 55-64%

D M Butala
President, IIChE

P K Saxena
Honorary Registrar, IIChE

Dr. Avijit Ghosh
Honorary Secretary, IIChE
Convener, OIP

Certificate ID: IICHE/2022/AD12242
Date of Issue: 30th May, 2022

Online Internship Program (OIP)

25th February to 30th May, 2022 (40 hrs)

Indian Institute of Chemical Engineers

Dr. H. L. Roy Building, Jadavpur University Campus, Kolkata- 700 032

CERTIFICATE OF COMPLETION

This certificate is hereby awarded to

ABHITHA SUGGALA

from **CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY**

who has successfully completed the INTERNSHIP PROGRAMME *on the subject* **PETROCHEMICAL ENGINEERING** *following all necessary criteria of the Institute with "* **A** *" Grade.*

Grading System :

A+: 90-100%
A: 75-89%
B+: 65-74%
B: 55-64%

D M Butala
President, IIChE

P K Saxena
Honorary Registrar, IIChE

Dr. Avijit Ghosh
Honorary Secretary, IIChE
Convener, OIP

SEUTIC PHARMA PVT.LTD.

Date: 08.10.2022

CERTIFICATE

This is to certify that Mr.**Aveti Dhanush Reddy** S/o. Mr. **A.Bhargava, UID No: 160119802024a** bonafied student of B.Tech Chemical, Chaitanya Bharathi Institute of Technology, Gandipet,Near Financial District, Hyderabad. has undergone his industrial training program in our organization.

The training program carried out from 01^{st} July 2022 to 31^{st} July 2022, as a part of partial fulfillment of her B.Tech (Chemical) Course.

During the afore-said period, we found him hardworking, sincere and learning attitude.

For SEUTIC PHARMA PVT.LTD

P. sudhakar 08/10/22

Authorized Signatory

SEUTIC PHARMA PVT. LTD. HYD

Office & Works :
Plot No. 42/A, Sy. No. 308, S.V. Co-op. Indl. Estate, I.D.A., Jeedimetla, Hyderabad - 500 055, India.
Phone : 040-23095711, Email : seuticpharma@gmail.com
CIN No : U2429TG2014PTC095025

VARUN BEVERAGES LIMITED

MUMBAI HIGHWAY, POTHIREDDYPALLY (V), SANGAREDDY (M). SANGAREDDY (DIST) - 502 295,
TELANGANA, PH : 7331166777, FAX : 08455 - 277405.
CIN : L74899DL1995PLC069839 Web : www.rjcorp.in E-mail ID : Info@rjcorp.in

29th July 2022

TO WHOM SO EVER IT MAY CONCERN

This is to certify that **"Mr. Sai Mani Yogesh Kosuru"** B. Tech (Chemical Engineering) Third year Student of Chaitanya Bharathi Institute of Technology (A), Rangareddy has Completed Internship on **"WATER TREATMENT MANAGEMENT"** from **01st July 2022 to 29th July 2022** in our organisation.

During the tenure of project his behaviour and conduct was satisfactory.

We wish all success in his future endeavours.

For **Varun Beverages Limited**

Authorized Signatory

VARUN BEVERAGES LIMITED

NH-65, MUMBAI HIGHWAY, POTHIREDDYPALLY (V), SANGAREDDY (M),
SANGAREDDY (DIST) - 502 295, TELANGANA.
CIN : L74899DL1995PLC069839 Web : www.varunpepsi.com E-mail ID : info@rjcorp.in

29th July 2022

TO WHOM SO EVER IT MAY CONCERN

This is to certify that "**Mr. Yashraj Delhiwala**" B. Tech (Chemical Engineering) Third year Student of Chaitanya Bharathi Institute of Technology (A), Rangareddy has Completed Internship on "**WATER TREATMENT MANAGEMENT**" from **11th July 2022 to 29th July 2022** in our organisation.

During the tenure of project his behaviour and conduct was satisfactory.

We wish all success in his future endeavours.

For **Varun Beverages Limited**

Authorized Signatory

SEUTIC PHARMA PVT.LTD.

Date: 08.10.2022

CERTIFICATE

This is to certify that Ms.**Devarabhatla Krishna Priya** D/o. Mr.D.Venkata Ravi **Prasad, UID No: 160119802007** a bonafied student of B.Tech Chemical, Chaitanya Bharathi Institute of Technology, Gandipet,Near Financial District, Hyderabad. has undergone her industrial training program in our organization.

The training program carried out from 01st July 2022 to 31st July 2022, as a part of partial fulfillment of her B.Tech (Chemical) Course.

During the afore-said period, we found her hardworking, sincere and learning attitude.

For SEUTIC PHARMA PVT.LTD

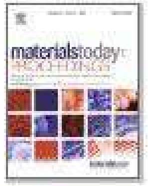
P. Sudhakar 08/10/22

Authorized Signatory

SEUTIC PHARMA PVT. LTD. HYD

Office & Works :
Plot No. 42/A, Sy. No. 308, S.V. Co-op. Indl. Estate, I.D.A., Jeedimetla, Hyderabad - 500 055, India.
Phone : 040-23095711, Email : seuticpharma@gmail.com
CIN No : U2429TG2014PTC095025

materialstoday: PROCEEDINGS



Volume 52, Part 3, 2022, Pages 873-881

Towards sustainable construction through the application of low carbon footprint products

K. Ranjetha [a], U. Johnson Alengaram [a], Ahmed Mahmoud Alnahhal [a], S. Karthick [a], W.J. Wan Zurina [a], K.J. Rao [b]

Show more

+ Add to Mendeley Share Cite

https://doi.org/10.1016/j.matpr.2021.10.275

Abstract

The production of cement generates a large amount of greenhouse gas emissions; in addition, the scarcity of natural resources used in the development of building materials and products has propelled many governments, non-government

materials today: PROCEEDINGS

Volume 62, Part 4, 2022, Pages 1785-1789

Experimental assessment of coir geotextile to improve the strength of weak subgrade at different load conditions

D. Harinder [a], P. Yugendar [b], S. Shankar [c]

Show more

+ Add to Mendeley Share Cite

https://doi.org/10.1016/j.matpr.2021.12.351

Abstract

The stabilization of weak subgrade soil with natural geotextile fiber is a cost-effective to improve the stability of the low-volume roads (LVRs). The coir geotextile is naturally available eco-friendly and biodegradable material having the high strength and durability compared to the other natural material. The present study, the test were conducted to determine the effectiveness of coir fiber and coir geotextile mats under the static and dynamic loading condition by using the CBR and WTT respectively. The inclusion of the coir geotextile fiber to the BC soil

materials today: PROCEEDINGS

Volume 52, Part 3, 2022, Pages 1201-1210

A study on mechanical properties of high strength concrete with alccofine as partial replacement of cement

Bhotla Harish [a], N.R. Dakshinamurthy [b], Malegam Sridhar [c], K. Jagannadha Rao [d]

Show more

+ Add to Mendeley   Share   Cite

https://doi.org/10.1016/j.matpr.2021.11.037

Abstract

Concrete is considered as a strong and durable construction material lasting over a century. Reinforced concrete is one among the most popular materials used for construction around the world. Reinforced concrete is subjected to deterioration in harsh environments like in coastal regions. Therefore, several researchers are striving hard with their efforts towards developing a new material to overcome this problem. Invention of large construction plants and equipment around the world added to the enlarged use of construction materials which has raised the curtain to

RESEARCH ARTICLE

Dynamic convolutional neural network based e-waste management and optimized collection planning

C. Jenifa Latha, K. Kalaiselvi, S. Ramanarayan, R. Srivel, S. Vani, T. V. M. Sairam

First published: 13 May 2022 | https://doi.org/10.1002/cpe.6941

Read the full text >   PDF   TOOLS   SHARE

Abstract

Electronic waste, also known as e-waste, refers to electrical or electronic devices that are discarded from households and workplaces. These used e-wastes are meant to be renovated, reused, recycled, or disposed of, and the processing of these wastes often causes disease and harms the environment. As a result, it is important to handle waste and collect it from the disposal site on a regular basis. Besides, in order to separate precious metals from discarded waste, it is important to identify them by category. Therefore, this article proposes a novel method known as e-waste management by exploiting the dynamic convolutional neural network (DCNN). This enhances the classification accuracy with the aid of exactly mapping the features of the images. Meanwhile, the collection of waste can be optimized in order to reduce the distance and time. The e-wastes in the smart garbage bin are frequently monitored by smartphone applications to collect the waste on time. Moreover, it also significantly reduces the training error, classification error, localization error, and validation error on the test images. The experimental depicts that the proposed method hones up the classification accuracy to the great extent.

Correlation of strength development of RCA in quaternai cementitious system

Sastri, M.V.S.S. (Department of Civil Engineering, Vasavi College of Engineering (A)) ;
Rao, K. Jagannadha (Department of Civil Engineering, Chaitanya Bharathi Institute of Technology (A)) ;
Bhikshma, V. (Department of Civil Engineering, University College of Engineering (A), Osmania University)

Received : 2020.02.14 Accepted : 2021.06.29 Published : 2021.09.25

https://doi.org/10.12989/amr.2021.10.3.211 Copy Citation

Abstract

Recycled concrete aggregate (RCA) obtained from demolished structures can be used for concrete making, and is established as a promising material in the field of construction. In the present study, the effect of RCA on the mechanical properties of different strength concretes admixed with Micro silica, fly ash and nano-silica as a part replacement to cement was considered. The quantity of cement varied from 350-690 kg/m^3 with the additions of Fly ash at 0, 20 and 30%, micro silica at 0, 5, 10 and 15%, and Nano silica at 0, 1, 2, 3 and 4%. The samples were cured for 7, 28, 56 and 90 days and tested for Compressive strength. Split tensile and flexural strength evaluation was carried out on samples which have been cured for 28 days. The workability of fresh concrete was determined. With the help of the tested database, equations for prediction of compressive strength using modified Bolomey's equation were generated. Equations for the flexural strength and split tensile strengths based on compressive strength were developed and compared with equations available in the literature.

Int J Syst Assur Eng Manag
https://doi.org/10.1007/s13198-021-01454-8

ORIGINAL ARTICLE

Detection of normal and epileptic EEG signals using by lifting based HAAR wavelet transform and artificial neural network

Vani S[1] · ChandraSekhar P[2] · Ramanarayan Sankriti[3] · Aparna G[4]

Received: 28 June 2021 / Revised: 10 September 2021 / Accepted: 1 October 2021

Abstract Electroencephalograms (EEGs), having a substantial potential to diagnose and cure mental and brain diseases and abnormalities, are becoming vital assessment of brain activity. The unit may contain indicators of existing disease or warnings about upcoming disturbances. EEG data can be used to improve the understanding of the mechanics behind epileptic convulsions. This study aims at the analysis of EEG signals and the detection of epilepsy levels. The algorithms for the lifting haar and the neural network are used. Lifting algorithms for non-stationary data such as EEG are useful. The EEG signals are divided into the subbands using LBHWT. Since LBHWT minimizes the cost of processing, the haar wavelet transform is faster than traditional. The coefficients of the wavelet are included in the classifier to find out if there is a seizer or not. The following steps are proposed: 1.Collection of data 2.Functional extraction, 3.Classification. This research is designed to provide an automated diagnostic tool to support a doctor.

Keywords EEG · Feature extraction · Lifting based haar wavelet transformation · Epilepsy · Epilepsy seizures · Neural network

✉ Vani S
krivani81@gmail.com

ChandraSekhar P
chandra4research@gmail.com

Ramanarayan Sankriti
ramanarayan_civil@cbit.ac.in

Aparna G
aparnaece27@gmail.com

1 Department of Mechatronics Engineering, S.R.M Institute of Science and Technology, Chennai, India

2 Department of ECE, Chaitanya Bharathi Institute of Technology (A), Hyderabad, India

3 Department of Civil, Chaitanya Bharathi Institute of Technology (A), Hyderabad, India

4 Research Scholar, ECE Department, University College of Engineering, Osmania University, Hyderabad, Telangana, India

1 Indroduction

Epilepsy is a progressive nervous brain condition affecting persons of all ages. The second most common brain disease is epilepsy. The sign of epilepsy is frequent convulsions. Epileptic seizures occur when a huge number of neurons begin to fire in a highly co-ordinated rhythmical way in the brain's cortex. Epilepsy affects 70 million people worldwide (about 1 percent of the population).

In general, electrodes into the scalp are used to record electrical activity to measure the voltage variations of the brain. Electrical impulses cause brief changes in motion, actions, feeling or perception of the brain's electrical system. Neurological condition that cannot be controlled by multiple medical treatments. The origin of the majority of unexplained cases. Only trained physicians can manually diagnose EEG signal epilepsy seizers.

The EEG provides a lot of interesting details about the different physiological stages of the brain and is also a very effective method for studying brain disorders, such as epilepsy. Normally, the EEG signal is used to record the neural activity of a brain wave that normally happens in the human brain.EEG is also used for diagnosis of sleep disorders, and estimation of anesthesia or coma depth. The detection of seizure is done by calculating the spikes in the EEG signals. The presence of epileptic component behavior in the EEG validates the diagnosis of epilepsy, which is sometimes correlated with other conditions, resulting in related seizure activity. Section 1.1 Discusses different feature extraction and classification methods that are intended to identify the epilepsy seizures. The review is carried out by dividing the problem of automated classification system into different Classes. Section 2 provides the details of the proposed feature extraction method using Lifting based Haar wavelet transform and presents the feature extracted using this method. Section 2.3 further provides the details of the classifying the epilepsy seizures using ANN. An efficient statistical analysis is presented to extract informative parameters set for classification of EEG signals that can be further used for deciding and selecting processes to achieve the desired objective.

Chapter 4 provides performance of Modified Haar Wavelet and ANN. This chapter investigates the possibility designing MHW and NN for prognosis of epileptic seizure with less computational complexity and higher accuracy.

Chapter 5 concludes the entire work with possible directions for future scope.

1.1 Literature review

Osorio et al (2002) revealed that the occurrence of seizures could be classified by observing epilepsy and that a delay of 2.1 s was identified. Gotman (1999) has estimated that about fifty million individuals globally are suffering from epilepsy. The tremendous number of diagnosed patients shows the crucial value of a better diagnostic tool to improve patients' standard of health. Cetin et al. (2015) used an autoregressive data reduction model and various AR model techniques to measure coefficients and designed a Matlab-based Interface that offers versatile and visual use for regular or epileptic monitoring.

Rajendran and Sridhar (2019) reported a PNN (Probabilistic Neural Network) dependent Epileptic Lobe Seizure classifier that attained a maximum accuracy of 96.30 percent. This analysis was performed utilizing the EEG dataset from Karunya University, based on parametric features such as AR Burg.

Güler et al. (2005) studied the application of the Recurrent Neural Network (RNN) as an epilepsy classification model. This approach tests the Lyapunov exponents developed with the Levenburg-Marquardt EEG signal algorithm for diagnostic accuracy of the RNN. In the development of a stable classification method for electroencephalographic transitions, a methodology based on the assumption that EEG are chaotic signals. This study was successfully tested with non-linear dynamics tools, such as Lyapunov exponent computation. A satisfactory rating accuracy of 96.79% has been achieved. The automatic seizure detection system was proposed by Aarabi et al. (2006) used typical EEG features from the frequency, time, and wavelet domains, as well as self-regressing coefficients and cepstral features. EEG with parameters the Relief procedure and two linear correlation-based data collection methods were used to evaluate data from six neonates aged 39–42 weeks. The characteristics obtained from seizures and non-seizure fragments were analyzed using these procedures in order to assess the efficacy of these procedures. To be categorized, the optimized classified subsets of functions were fed into a neural context network. The output has been used to assess how successful the feature selection was. All of these features were fed into two hidden layers in the Back Propagation Neural Network classification (BPNN) classification and resulted in 93% on average.

The technical claims to be analyzed on a Multi-channel EEG dataset in HAAR-based wavelet decomposition with a wave- and Artificial Network (ANN) automatic evaluation technique are proposed (Vani et al. 2016).

Fu et al. (2015) suggested a new methodology for automatic detection of EEG signals based on Hilbert Marginal Spectrum analysis (HMS).The traditional Fourier analysis that supports the sinusoidal signal does not accurately reflect the amplitude contribution of each frequency spectrum because the EEG signal is extremely non-linear and non-static. The Empirical Mode Decomposition (EMD) divides the signal into an Intrinsic Mode Function package, which is the basis for the HMS (IMF).This breakdown works well for non-linear and non-stop procedures since it is dependent on the signal's local time scale. The frequency entropies and energy properties of the rhythms are extracted and transmitted using the SVM HMS analysis for the detection of EEG signal seizures in this study. With SVM, the accuracy of classification has been improved to 98.8%. Wang et al. (2017) obtained 98.3 percent classification accuracy using lifting-based DWT and sparse extreme learning machine for epileptic seizure detection. Despite various techniques used, polynomial transformations for EEG signal classifications are still not extremely linked with feature extraction methods even though it has already been shown that the spectral coefficient physical interpretation leads to a new issue for automatic diagnosis of seizures and detection of eye conditions Nkengfack et al. (2021). Evolutionary algorithms can improve DWT performance by selecting more dependable features (Al-Qerem et al. 2020).

Feature extraction and selection technology are critical for finding certain properties that can be utilized to properly classify EEG signals. Several methods are employed in feature extraction, including time domain analysis, frequency domain analysis, and time–frequency domain analysis. Features such as minimum, maximum, mean, standard deviation, and energy are often utilized in time-domain analysis. The disadvantages of the time-domain technique include the high sensitivity of the selected features and the need for more storage space (Vani et al. 2019).

Several classification methods have been used to detect epilepsy from EEG signals:

Parametric method, Mimetic method, Syntactic method, ANN, Expert systems, Clustering Techniques.

- *Parametric method*—These approaches are based on conventional signal processing, which considers that a seizure occurs when the EEG signal deviates from the expected value, and that the time series measurement at a given time is dependent on the prior values and the random disturbance used to detect epilepsy.
- *Mimetic method*—Where the value of the parameters measured for the detection of epilepsy from each wave and the thresholds
- *Synaptic method*—detection of epilepsy is based on structural features.
- *Expert systems-* This method defines seizures by imitating the information and thought process of an expert.
- *Clustering Techniques*—Detections are predicated on systemic agglomeration processes and self-organization maps.
- *Neural Networks*—This approach takes into account the machine learning of epileptic seizure-related transients. NN has more advantages than other methods.

2 Materials and methods

Two publicly available datasets are used in this study (Karunya and Bonn University). The HAAR (DB2) wavelet is used to decompose the dataset. The lifting technique was discovered to be an excellent and faster wavelet for evaluating spikes in EEG signals. DB2 WT breaks down EEG signals into sub-bands. The WT coefficients are fed into a qualified NN using the BP algorithm. The Lifting Wavelet Transform is used to extract characteristics (Energy, Standard Deviation, and Entropy) and reduce the complexity of the computation.. The following stages are proposed 1. Dataset collection, 2.Feature extraction, 3.Classification. The purpose of the paper is to include an automated device to assist a doctor in the process of diagnosis (Fig. 1).

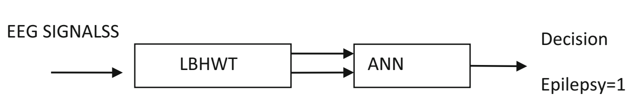


Fig. 1 Block diagram of proposed method

2.1 Dataset

2.1.1 Bonn data set

There are 5 sets at Bonn University (http://epileptologie-bonn.de/cms/upload/workgroup/lehnertz/eegdata.html), and each set contains one hundred single EEG channel segments. These segments are acquired after visual inspection for ocular artefacts from multichannel EEG recordings. Set A was reported from healthy eye -opened volunteers, and Set B was also taken from healthy eye-closed volunteers. From epileptic patients, sets C, D, and E were registered. Set C and D consist of intervals free of seizures, and Set E only includes seizure activity.

2.1.2 Karunya data set

A dataset of 10 s EEG seized cases was also used in the Karunya University EEG database (https://karunya.edu/research/EEGdatabase/public/index.php). The dataset is sampled at 256 Hz using an analog band pass filter with cut-off of 0.01–100 Hz. The total of 175 EEG seizures from patients is presented in this dataset.

2.2 Wavelet transform

Wavelets or Wavelet investigation is also known as wavelet theory. It gained enormous Popularity in the field of signal processing and much consideration. It was efficient. Implemented in various applications, such as image investigation, transient signal investigation, communications frameworks and other applications for digital signal processing. Wavelets are a particular kind of functions that for a short amount of time exhibit oscillatory activity and then die production. A single function and its dilations and translations are used in wavelets to produce a series of orthonormal basis functions to describe a signal.

2.2.1 Discrete wavelet transform

The signal is translated into a set of basic functions by Wavelet. Such basic functions are referred to as wavelets. Wavelets are obtained by dilation and shifting from a single wavelet template called the mother wavelet:

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}}\psi\left(\frac{t-b}{a}\right)$$

where a is the scaling parameter and b is the shifting parameter.

2.2.2 *Lifting scheme*

Based on the domain characteristics, a new wavelet is generated from the mother wavelets in the lifting scheme. Odd samples are extracted from samples, and basic estimation and update blocks are replaced with filter. Based on their shapes and their ability to interpret the signal for the specified function, the wavelets are chosen.

The lifting scheme has three steps for creating wavelets: the first is a split method that divides data into odd and even sets. The second step is to guess which step will result in the odd set being even. High-pass polynomial cancellation is ensured by the predictive method. The third step is the update stage, which can also be modified by calculating the scaling function using the wavelet coefficient. Low-pass moments are preserved during the upgrade process.

Split step: The signal is separated into even and odd points, with the pixels with the highest similarity being used in the following forecast step. The first step is to predict: even samples are multiplied by the predictor factor, then the results are added to the odd samples to get detailed coefficients. (High Pass Filter Coefficients) The detailed coefficients generated by the prediction step are multiplied by the update factor to create the rough coefficients, which are then applied to even the samples. (Low Pass Filter Coefficients) (Fig. 2).

X is the input here. Even and odd samples are separated from X. Using X_E, forecast Xo and replace Xo with prediction error. The main idea is to factor the polyphase matrix of the wavelet filter into a collection of alternating upper and lower triangular matrices, as well as a diagonal matrix. Wavelet implementations based on banded-matrix multiplication are thus possible. The dual polyphase factorization, which also includes predicting and updating processes, can be written like this:

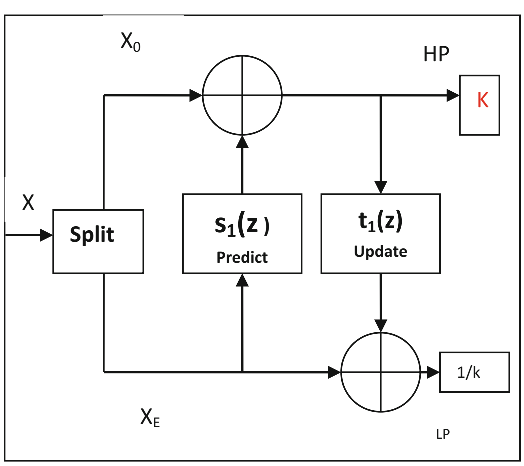


Fig. 2 Lifting based forward DWT

$$\widetilde{P1}(Z) = \begin{bmatrix} k & 0 \\ 0 & \frac{1}{k} \end{bmatrix} \prod_{i=1}^{m} \begin{vmatrix} 1 & s_1(Z) \\ 0 & 1 \end{vmatrix} \begin{vmatrix} \begin{bmatrix} 1 & 0 \\ t_1(Z) & 1 \end{bmatrix} \end{vmatrix}$$

$$\widetilde{P2}(Z) = \begin{bmatrix} k & 0 \\ 0 & 1/k \end{bmatrix} \prod_{i=1}^{m} \begin{vmatrix} 1 & 0 \\ t_1(Z) & 1 \end{vmatrix} \begin{bmatrix} 1 & S_1(Z) \\ 0 & 1 \end{bmatrix}$$

where $\boldsymbol{s_I}(z)$ (primary lifting steps) and $\boldsymbol{t_I}(z)$ (dual lifting steps) are filters and $\boldsymbol{K}$ is a constant.

As this factorization is not unique, several $\{\boldsymbol{s_I}(z)\}$, $\{\boldsymbol{t_I}(z)\}$ and $\boldsymbol{K}$ are admissible.

2.3 Neural network

Neural networks are devices of nonlinear signal processing interconnected and organisedintolayersy neurons. Neurons are responsible for information processing and the signals are transmitted through links. The 'nodes' interconnected contain an' activation mechanism '. Patterns are provided via the 'input layer' to the network, which communicates to one or more 'hidden layers' where the real processing is carried out via a weighted 'connections' method.

Training algorithm for the network is required to achieve the correct output. A good training algorithm will reduce the time taken for training. There are many number of training algorithms are available to train the neural network. Back propagation is used as training algorithm in this research.

Back propagation is a supervised process that aims to train the network to achieve a balance between the input patterns that are used for training and to provide better responses to the inputs that are similar. Each time the network is presented with an input pattern an output is computed from the net, and the total squared error of output is minimized by the gradient descent or generalized delta rule.

EEG data is categorized as 'epileptic' or 'normal.' The data used for training is quite significant. The components of the NN are activated by biological nervous systems. Commonly, NN are trained to perform complex functions. The functions are determined from the connections of the elements. NN shows poor false detection rate (Fig. 3).

The Neural Network architecture has the following steps,

1. *Initialized*–all neurons are subject to initial weights(w_{ji}).
2. *Forward propagation*–the neural network is fed inputs from a training set, and an output is generated. A sigmoid's output is a continuous real-value between 0 and 1.
3. *Error function*–Because the NN is working with a training set, the correct output is known. Given the

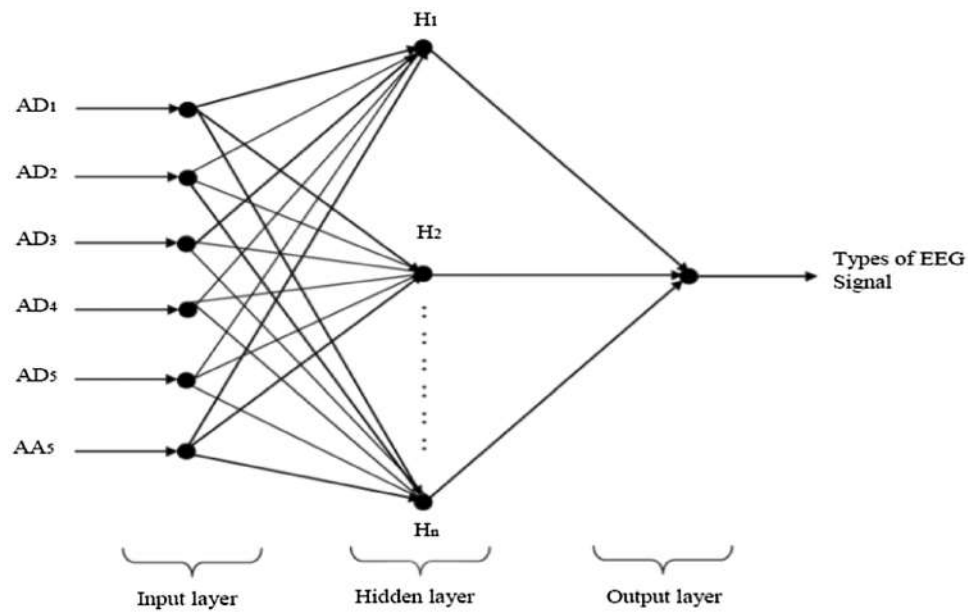


Fig. 3 Proposed neural network training structure(AD1–AD5– detailed coefficients of MHWT,AA1- Approximate coefficients of MHWT)

present model weights, an error function is built that captures the delta between the correct output and the actual output of the model.

4. *Backpropagation*–The goal of back propagation is to modify the weights of the neurons in order to minimise the error function. A back propagation network learns in two stages. Each pattern Ip is first provided to the network and then transmitted to the output. Second, to reduce the overall error on the patterns in the training set, a method known as gradient descent is applied.
5. *Weight update*-weights are adjusted to the best possible values based on the outcomes of the backpropagation method. As a result, the relevant change in the weight wij in relation to the error Ep can be represented as as $\Delta_p w_{ji} = \eta\ d_{pj} o_{pj}$. The parameter ŋ in this equation is known as the learning rate. The learning rate is 0.5, and the learning rule that was used is as follows:

$$w_{ij}(n+1) = w_{ij}(n) + \eta(n)\left[x_i - w_{ij}(n+1)\right]$$

where the weights are updated with η as the learning rate.

Iterate until convergence—because the weights are changed a little delta step at a time, the network requires numerous iterations to learn. The down gradient force changes weights to less and less world-wide loss-function after each iteration.

Pattern-net, the default network for function fitting issues, is a feedforward network with tan-sigmoid transfer functions in both the hidden and output layers. Ten neurons are assigned to a single hidden layer. Because there are two target values associated with each input vector, each output neuron represents a category in the network. When an acceptable category input vector is presented to the network, the corresponding neuron should produce a 1, while the remaining neurons should create a 0.

Training is halted if any of the following conditions are met:

- The number of epochs (repetitions) has been reached.
- The time limit has been exceeded.
- Performance is reduced to achieve the aim.
- The performance gradient is less than min grad.
- Since the last drop, validation performance has improved more than max fail times.

The validation and test data sets are each limited to 15% of the original data. The input and target vectors will be randomly separated into three sets with these settings: 70% of the time is spent on training. 15% are used to check that the network is generalising and to halt training before it becomes overfit. The remaining 15% is employed as a completely independent test of network generalisation. The dataset details is given in Table 1.

Table 1 Dataset used for classification

Data set	Training set	Testing set	Validation	Total
Set A,E	140	30	30	200
(Set A,D,E)	210	45	45	300
Karunya data set	21	2	2	25

Table 2 The values of statistical Parameters

EEG dataset	Classification accuracy (%)	Specifity (%)	Sensitivity (%)	Over all accuracy(%)
Set A,E	100	100	100	100
(Set A,D,E)	99	99	99	99
Karunya data set	99	99	99	99

3 Performance analysis and results

In the proposed wavelet, computational complexity is reduced by a factor of 2 compared to the traditional analysis filter bank. Daubechies structure consisting of advantageous polyphase decompositions dealing with multiplication instead of convolution is presented. As we know, choosing appropriate features can improve the accuracy of classifier. It has been found that the proposed method can gain high performance among the Automatic epileptic seizure detection on condition that associate acceptable set of uncorrelated features is extracted. For the evaluation of the Neural Network, classification results and statistical parameters are used. Energy, variance, Mean, standard deviation and entropy are calculated for the extracted sub-bands. The classification set used in this study, a)S ets A and E. Only the A, E segments are used and they are classified as normal/seizure. b) Karunya data sets are used and classified as normal\epilepsy. The sensitivity, selectivity, and classification accuracy parameters are used to evaluate the classifier's efficiency. Table 2 lists the statistical parameters of the proposed work's statistical parameters.

Sensitivity is defined as the ratio of the number of correctly observed patterns to the total number of true positive patterns. It is also known as the true positive ratio. The True Positive pattern indicates that seizures have been observed.

The number of correctly identified negative patterns multiplied by the total number of true negative patterns equals specificity. True Negative Ratio is another name for it.

The relationship between the number of patterns successfully identified and the total number of patterns is known as Accuracy.

Table 3 Comparison with previous work

Authors	Dataset	Methods	Accuracy
Riaz et al. (2016)	A,D,E	SVM	85%
Wang (2017)	A,D,E	LDWT, and SELM	98.4%
Proposed work	A,D,E	LBDWT and ANN	99%

4 Conclusion

Only qualified professionals from EEG recordings can diagnose the epilepsis seizers manually. An epileptic seizure is not predictable. The purpose of the project is to equip a medical doctor with an automated diagnostic gadget. In this investigation, the DB2 wavelet decomposes the encephalogram signal. Modified HWT is a very appropriate and quicker wavelet in order to investigate the spikes of EEG signals. The DB2 WT is used in sub-bands to segregate EEG signals. The WT coefficients are provided with the NN trained BP algorithm. The Transforming Lifting Wavelet is used to remove features (energy, standard deviation, entropy and other devices) and reduce calculating complexity. A variety of multiplications and adds is decreased as the level of decomposition is increased. With the continuous EEG recording from Bonn University database (A, E) our suggested approach obtained 100% accuracy while the multichannel EEG dataset of Karunya University reached 97.5 percent. The system has an excellent degree of precision in accordance with the proposed method, which makes it suited for detection of EEG signal epileptic seizure activities. It is clear that our framework can get better categorization results than the rest of the current findings given (Table 3).

With the help of proposed method, it has been proved that, 99% classification accuracy rate has been achieved and consequently the computational complexity of the wavelet is reduced.

Funding No founding is provided for this work.

Declarations

Conflict of interest No conflicts of interest associated with this publication and there has been no significant financial support for this work that could have influenced its outcome.

References

Aarabi A, Wallois F, Grebe R (2006) Automated neonatal seizure detection: a multistage classification system through feature selection based on relevance and redundancy analysis. Clin Neurophysiol 117(2):328–340

Al-Qerem A, Kharbat F, Nashwan S, Ashraf S, Blaou K (2020) General model for best feature extraction of EEG using discrete wavelet transform wavelet family and diferential evolution 2020.

Int J Distrib Sens Netw. https://doi.org/10.1177/1550147720911009

Bonn dataset: http://epileptologie-bonn.de/cms/upload/workgroup/lehnertz/eegdata.html

Cetin GD, Cetin O, Bozkurt MR (2015) The detection of normal and epileptic EEG signals using ANN methods with matlab-based GUI. Int J Comput Appl 114(12)

Fu K, Qu J, Chai Y, Zou T (2015) Hilbert marginal spectrum analysis for automatic seizure detection in EEG signals. Biomed Signal Process Control 18:179–185

Gotman J (1999) Automatic detection of seizures and spikes. J Clin, Neurophysiol 16:130–140

Güler NF, Übeyli ED, Güler I (2005) Recurrent neural networks employing Lyapunov exponents for EEG signals classification. Expert Syst Appl 29(3):506–514

Nkengfack LCD, Tchiotsop D, Atangana R, Louis-Door V, Wolf D (2021) Classification of EEG signals for epileptic seizures detection and eye states identification using Jacobi polynomial transforms-based measures of complexity and least-square support vector machine. Inf Med Unlocked (2021)

Karunya dataset: https://karunya.edu/research/EEGdatabase/public/index.php.

Osorio I, Frei MG, Giftakis J, Peters T, Ingram J, Turnbull M, Herzog M, Rise MT, Schaffner S, Wennberg RA, Walczak TS, Risinger MW, Ajmone-Marsan C (2002) Performance reassessment of a real-time seizure-detection algorithm on long ECoG series. Epilepsia 43(12):1522–1535

Rajendran and K.P. Sridhar (2019) Epileptic seizure-classification using probabilistic neural network based on parametric features. J Int Pharma Res 46(1):209–216

Riaz A, Hassan S, Rehman IK, Niazi, K. Drem- strup (2016) EMD-based temporal and spectral features for the classification of EEG signals using supervised learning. IEEE Trans Neural Syst Rehabil Eng 24(1):28–35

Vani S, Suresh GR, Balakumaran T (2016) Automatic recognition of epilepsy seizure using lifting based wavelet transform and artificial neural network. Int J Print, Packag Allied Sci 4(6):4420–4428

Vani S, Suresh GR, Balakumaran T (2019) EEG signal analysis for automated epilepsy seizure detection using wavelet transform and artificial neural network. J Med Imaging Health Inform 9:1301–1306

Wang Y, Li Z, Feng L, Zheng C, Zhang W (2017) Automatic detection of epilepsy and seizure using multiclass sparse extreme learning machine classification. Comput Math Methods Med. https://doi.org/10.1155/2017/6849360

Neural Network Based Prolong Structural Building Monitoring Management System Using Hybrid Wireless Sensor Network

[1]Dr. C. Jenifa Latha, [2]Ramanarayan Sankriti, [3]Dr.Sudha Arvind, [4]Ali Baig Mohammad,

[5] Dr.MS.Mariam Jenabhar

[1]Professor, Department of Civil Engineering , Saveetha Engineering College (SEC),Saveetha Nagar, Thandalam, Chennai - 602 105, Tamil Nadu, India

[2]Assistant Professor/Dept. of Civil, Chaitanya Bharathi institute of Technology, Hyderabad, India

[3]Professor, ECE Department, CMR Technical Campus, Hyderabad

[4]Assistant Professor, ECE Dept., Koneru Lakshmaiah Education Foundation, Vaddeswaram, Andhra Pradesh, India

[5]Assistant Professor, Department of Commerce, Sree Sevugan Annamalai College, Devakottai, Tamil Nadu

Abstract

The wireless sensor network is well suited for smart building monitoring system as it is cost effective compared with wired network. A hybrid protocol that uses concept of LEACH to increase the life time of sensor network and HRP protocol for effective communication to base station is use in building monitoring system. The sensors such as humidity sensor, strain sensor, temperature sensor, and electrical current monitoring sensor are deployed in huge amount around the building infrastructure and these sensor are organized as different clusters. At initial stage cluster head sensor activates few member sensors for gathering information and rest of sensors are inactive state. This increases the life time of sensor network. If anything happens abnormal then few more sensors are activated to gather more information by cluster head. A neural network is implemented to isolate information from the cluster member nodes and communicate effectively to sink node. In order to reduce the work load of cluster head node, Information Collection Node (ICN) and Energy Monitoring Node (EMN) are added in each cluster to isolate information and monitor energy level of cluster member nodes. Thus hybrid protocol for structural building monitoring system is implemented that increases life time and effectively convey sensory information to sink node in wireless sensor network.

Keywords: - Structural building monitoring system, Hierarchical clustering, LEACH, HRP.

1. Introduction

Now a day's wireless sensor networks play a major role in collecting information from sensors and sending to the sink node. Sensor networks are deployed in various areas such as healthcare monitoring, civil building application, and military defenses. Wireless sensor networks can sense information humidity, pressure, temperature etc .These sensors collects information about the surroundings then process the information and send to base station (BS) [1].

Internet of Things (IoT) is the emerging technology that are used in various application such as health care smart cites, vehicular networks, machine to machine communication and other areas . In IOT uses RFID to uniquely identify things and controlling them with various mechanisms. IOT is used in pervasive computing i.e. present everywhere that gather information for all IOT device component and processes these information. Networking Scientists, Research & Development industries, and many other business organization are developing an achievable and robust architecture to make it commercial and it is used by normal people [2, 3].

Wireless sensor Networks comprises of three main parts such as communicating unit sensing unit and processing unit. The sensing unit has analogue to digital converter that converts analog signal to digital signal and sensors for sensing information they are powered by battery and replacement of these battery is difficult as most sensors are deployed in typical areas where human cannot enter such as volcanic environment, military area surveillance, etc. As energy drains sensors fails since replacement of battery is not possible then we need to deploy new sensory to make it WSN active. Next the communication unit is responsible for sending data over a sensory channel. At last, the processing unit has some microcontroller and a microprocessor, which controls the sensor nodes by sending control signals to microcontroller and a microprocessor.

Recent advances in materials and sensor technologies have brought significant new tools for upgrading building systems. Many of the structures date back to the decades in their fundamental form, the intelligence added approaches tackles the issues that hinder efficient implementation. active structural damage control SHM has become a main stay for evaluating entire behaviour, especially from manufacture through its usefulness

Buildings are subject to dynamic loading, hence dynamic analysis is required.

Using SHM techniques, the perpetually berated transfer function has been used to predict seismic wave effects on buildings . Analyze the buildings' state A neural network (NN) can analyse both damaged and undamaged data. Artificial free vibration is used to inspect places and forecast vibration range and level. Dynamic loading causes harm. Most SHM buildings are static loaded. The main parameters for static loading are displacement, strain, temperature, and time. Warm and cold seasons Buildings under static load typically use a 1D or 2D scheme. Modern technologies and a global positioning system can trace the structure in 3D.

Damage detection is critical in SHM, hence methods like ambient vibrations have been employed to foresee damage. For example, the SHM technique was used to monitor the San Pedro Apostol church both short and long term. long-term temperature and humidity monitoring [1]. SHM can be used to protect historic structures as well as modern structures. Non-destructive approaches can be employed to monitor Ottoman constructions. strategies for forecasting damage and monitoring building health

In order to minimize battery energy conception sensor nodes has limited coverage and the sensors that are not in coverage area of base station can communicate through intermediate sensor nodes. However, some case the wireless sensors nodes are organized as different clusters and the cluster head is responsible to collect information from cluster area and send to base station. It indicates the separation of data into groups of cluster head data. To maximize life time all sensors sends their gather information to cluster head and these cluster head aggregate these information and send to the base station

Only cluster head is responsible for gathering information process them and sending to the sink node this saves the energy of other sensor nodes as they are going to send information to cluster head and they are not going to do any processing. The cluster head also plays a vital role in wireless sensor network. It reduces the redundant information while aggregating data from different sensor nodes. In general if anything happens it is detected by multiple sensors deployed around the environment. As this information is Send to cluster head, it eliminates the redundant information and forward to sink Node. Thus it reduces the bandwidth requirement of wireless sensor network as sending information is reduced and it also saves the energy of Sensors node [5].

The sensor near to base station can directly transmit its information to it. If it is On hop transmission. Generally much of Sensor nodes are deployed far away from base station and they com communicate only through neighbor Sensor nodes. This is multi hop transmission and it requires dynamic topology creation in Wireless sensor network [6}

The cluster formation and grouping member Sensor nodes with the cluster head is a major problem in wireless Sensor Network. As much of work such as collecting sensory data, aggregating it, removing redundancy and sending to the base station are done by cluster head. This lead to drain in energy level of cluster head and soon it become unavailable. Then a new cluster head is detected [7].

SHM systems notify users when there are significant structural changes or deterioration. The main purpose of structural damage detection is to figure out what caused the damage, where it happened, and what kind of damage it is. the remaining life expectancy of the structure Internal and external factors like as corrosion, fatigue, and ageing, as well as earthquakes, wind, and collision, cause structural defects that cause collapses. The effects of the above mentioned causes may be gradual and not obvious until the structure is compromised. considerable, and only seldom repairable at a high cost. The ability to detect structural damage is critical to a structure's long-term safety. Infrastructure It assesses the structural system's measurable and qualitative degradation under severe loads. It is vital to keep track of things.the occurrence, location, and magnitude of safety and performance degradation

Design and Implementation of Hospital Management System

Hospitals are information-intensive organisations that require adequate information systems to handle and interpret data. Hospital information systems (HIS) and clinical information systems (CIS) are computer-based systems used in hospitals to help manage patient records, accounting, human resource management, asset management, stock management, and knowledge management. The monitoring system's sensing component contains sensors, recorders, servers, and other firmware to measure the building's response, collect data, and send it to an onsite server in real-time.

Figure 1a Represents the 3D view of Structural Health Monitoring System in Hospitals

2. Proposed System

Building is categorized into different areas and different sensors such as strain sensor, temperature Sensor, humidity sensor etc around the area are grouped into clusters. Let CL1, CL2, and CL3 ... CLN) be different clusters formed around the building.

Select the cluster head (Ch) such as Ch1 ∑ CL1, Ch2 ∑ CL2, ... ChN ∑ CLNInformation collection Node (ICN) Such as ICN1∑ CL1, ICN2∑ CL2... ICNN∑ CLN and Energy Monitoring Node EMN1∑ CL1, EMN2∑ CL2, …EMNn∑ CLn in every cluster area.

Cluster head node activates some of cluster member nodes and deactivates other robes based on their requirement. The cluster head also gather information from Information collection Node and process them and send it to sink station either directly or through other cluster head nodes. Information Collection Node effectively collects and isolates information from cluster member node and eliminating duplicate information through neural network and sends to cluster head Node. Energy Monitoring Mode gathers information about the energy level of member node and send to cluster head Node. Cluster head nodes select few sensors for gathering environment data ,based on the energy level sent by based energy monitoring node.

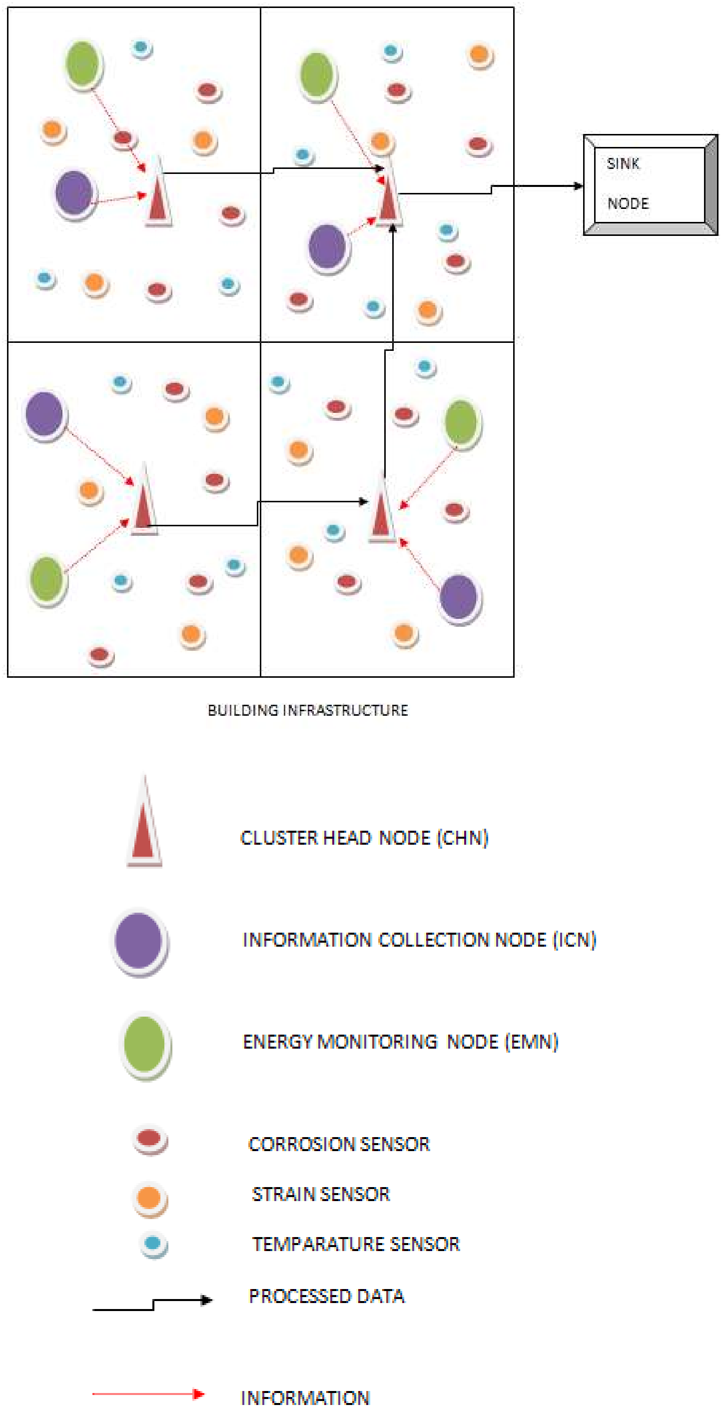


Figure 1brepresents the Building Monitoring Diagram

2 a. Information Collection Node

Information collection node in each cluster collects information from each cluster member sensor nodes through neural network and eliminates duplicate information then it sends to cluster head node in their respective TDMA time slot.

routine Information_Collection_Node (Member_Node m)

{

For Each (m)

m gathers environment data around building

m transfer sensed data to Information Collection Node (ICN) in corresponding cluster in their respective slot of TDMA.

End for

send consolidated information to cluster head

}

2. b. Energy Monitoring Node

Energy monitoring node in each cluster collects residual energy level of each cluster member nodes and send to cluster head node in their respective TDMA time slot.

routine Energy_Monitoring_Nod*e* (Member_Node m)

{

For Each (n)

m sends their residual energy level to Energy Monitoring Nodes in their respective cluster in their allocated time slot.

End for

Send Energy Level status information of all member nodes to cluster head Node

}

2. c Cluster Head Node

The cluster head node collects information from the information isolation node and sent to sink node directly if it is in its coverage area or else it may forward information through intermediate cluster head node. It also receives information from energy monitoring node and takes decision about whether the sensors around building are kept active or not based on their energy level.

routine Cluster_Head_Node (Information_Collection_Node i,Energy_Monitoring_Node e)

{

Process Information from i node

Send processed Information to base station either directly or through other cluster head.

Collects energy information from e and make necessary sensor active

}

Initialize each sensor node has some residual energy as assign the Cluster Head and record this information shared with one hop neighbour node using ADV packets. To make the decision based on their residual energy in a node receiving ADV packets. If compare advertise or new receiving ADV packets of residual energy are greater than node's own ADV packets of residual energy then new receiving node as assign the CH and update every level and also shared by all other one hop neighbour node except source node. Otherwise, no there is updating ADV packets and forwarding to the source node. After a particular time interval all nodes have a same ADV packet with same id and energy level is chosen as CH. This process is Repeat the cluster head energy level becomes low

3. Cluster Head Selection Algorithm steps

Step1: Each sensor node around the building initially consider them as cluster head and record their residual energy level in ENERGY_ADV advertise packets

Step2: ThenENERGY_ADV advertise packet is shared among their neighbour node sensors.

Step3: When other sensor nodes receive ENERGY_ADV advertise packets then the decision are taken based on their residual energy level and received advertise energy level.

Step 3.1 if advertise ENERGY_ADV packet residual energy is higher than sensor node's own residual energy then assign advertise node as Cluster Head and update energy level and broadcast to all other neighbor node except sending sensor node.

Step 3.2if advertise ENERGY_ADV packet residual energy is smaller than the sensor node's own residual energy then there is no updating and just forwards to the sending sensor node

Step4: After a particular time period, all sensor nodes have a sameENERGY_ ADV packet with same id and energy level and it is chosen as Cluster Head.

Step5: Repeat step2 cluster head energy level becomes low.

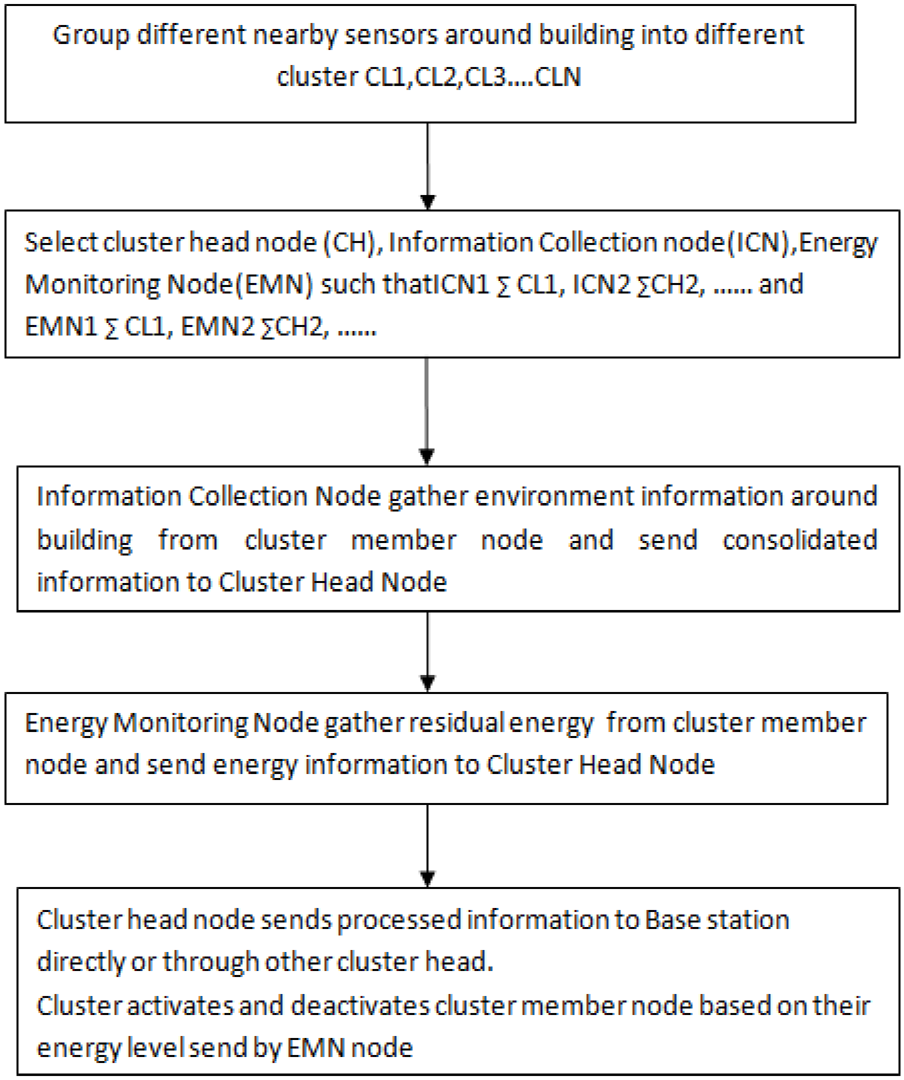


Figure 2: Represents the Flow Diagram of Cluster Head Selection Algorithm

3. a. Cluster Maintenance

If energy level of cluster head node or Information collection node or Energy monitoring node is drained then alternate node with sufficient energy level is selected and all responsibilities is delegated to the selected node.

routine Cluster_Maintenance ( )

{

If (Energy_level_Cluster_Head_ Node <Threshold_ Value)

Initiate election to find another member node with sufficientenergy level in cluster as Cluster Head Node

If (Energy _level_Information _collection_ Node <Threshold_Value)

Initiate election to find another member node with sufficient energy in cluster as Information collection Node

If (Energy _level_energy _monitoring_ Node <Threshold_Value)

Initiate election to find another member node with sufficient energy in cluster as energy monitoring Node

}

4. Simulation Result and Analysis:

A. Power Consumption

At initial stage cluster head makes only few sensor in active state and rest are inactive .When there is a need i.e. when something went wrong,based on our requirement more sensors on particular area or time are activated to gather more amount of information .This will drastically increase efficiency and reduces the power consumption. The combination of LEACH and hierarchical cluster formation will produces better result in structural building monitoring sensor. Now we compare the result of normal LEACH and our proposed system based on power consumption.

Table 1. Represents the Comparison between the Existing leach Algorithm and Proposed Method

No of nodes	Power consumption	
	LEACH	Proposed
40	22	20
45	26	23
50	29	25
55	33	29
60	36	32
65	39	36
70	43	38
75	44	40
80	46	41
85	49	43

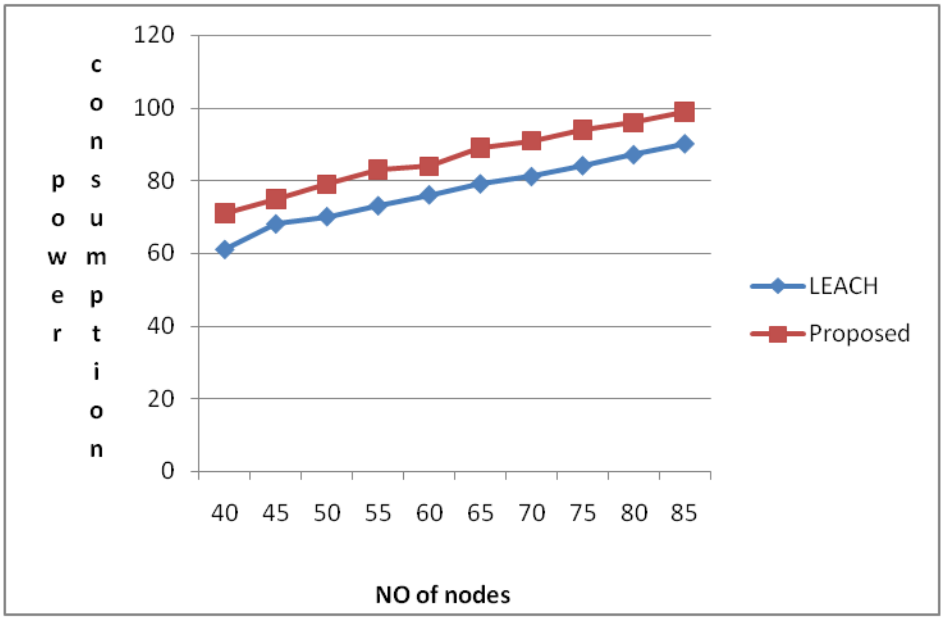


Figure3 Represents the Plot between Number of Nodes & Power Consumption

B. Information Gathering

The amount of information collection is also increased since we are isolating all information collected by different sensors by information collection node and send to cluster head node in turn it forward to sink node either directly are through intermediate cluster head sensor nodes. The neural network is used to gather environment information effectively from cluster member nodes and duplicate information are avoided before sending to cluster head nodes.

Table2. Represents the Comparison between the Existing Leach Algorithm and Proposed Method

No of nodes	Information Gathering (in %)	
	LEACH	Proposed
40	61	71
45	68	75
50	70	79
55	73	83
60	76	84
65	79	89
70	81	91

75	84	94
80	87	96
85	90	99

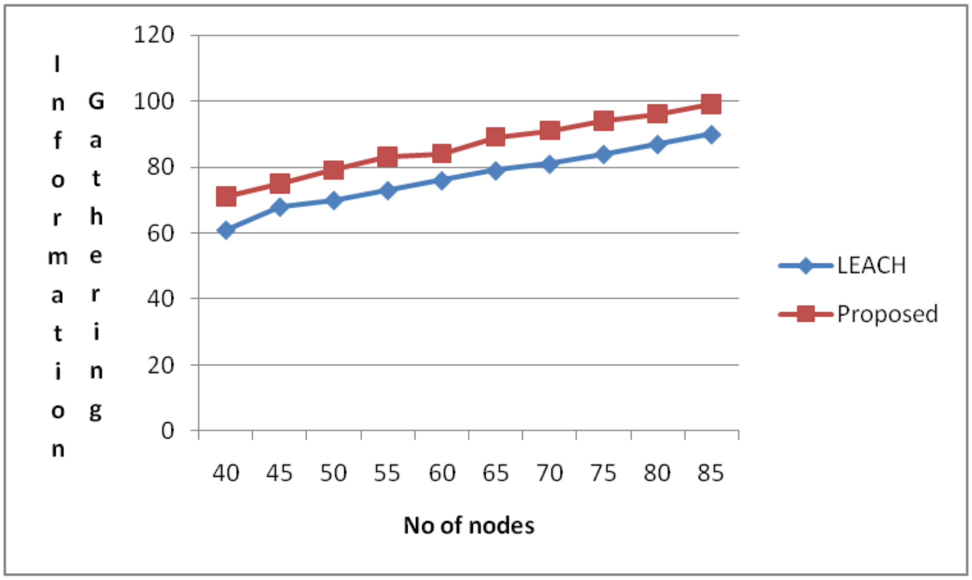


Figure-4 Represents the Plot between Number of Nodes & Information Gathering

Simulation parameters

Parameter	Default value
Simulation area	130 m × 130 m
Number of nodes	60–120
Packet length (from cluster head to BS)	6500
ctrPacket length (default packet length from normal node to cluster head)	195
Initial energy	0.45
Base station coordinates	(50, 50)
Probability to the node to become a CH	0.15
Energy for transferring of each bit	49*0.000000001
Energy for receiving	49*0.000000001

Energy for free space model	11*0.000000000001
Energy for multipath model	11*0.000000000001
Energy for data aggregation	5*0.000000001

In this cluster Head node is select based on the energy level analysis by cluster selection algorithm. All Sensor nodes collect the environmental data and do some initial processing and send the Information Isolation Node in cluster. Based on the processed data collected from each Sensor node, the cluster Head node takes decision and forward to Base Station using time division multiplexing or code division multiplexing.

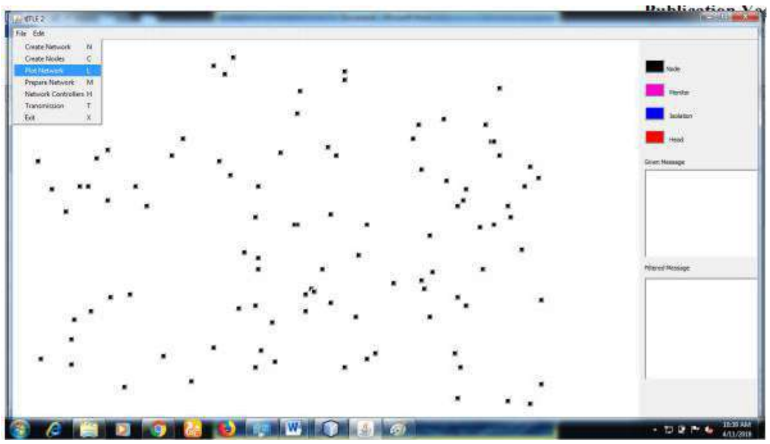


Figure-5 represents the node creation

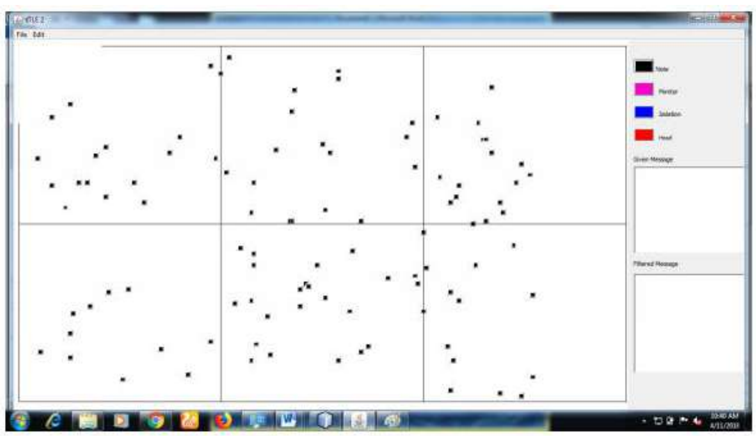


Figure 6represents the cluster node formation

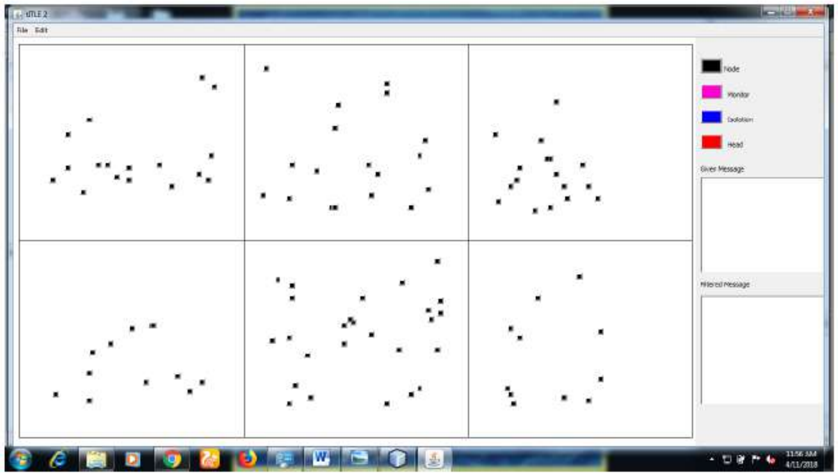


Figure7 represents partially active nodes in cluster

5. Conclusion

The combined effort of LEACH Protocol and hierarchical routing protocol increase the life time of wireless Sensor Network and effectively communicate the sensor information to the base station for building monitoring System. The LEACH protocol initially makes some of sensor node in active State and makes some sensor to be switched off. This will reduce the power consumption of wireless sensor network. When a unusual circumstance arises then it will activate some more sensors in that area to collect more information and send to Sink node for further effective measures. The hierarchical routing protocol makes some cluster information for effectively collecting and communicating to the sink node. The cluster head performs data aggregation and process them to make decision and convey to sink node. The information isolation node and energy monitoring node in cluster reduces the work load of cluster head and increases the life period of cluster head node.Thus various types of building sensors have been placed and the current situation of the building is monitored. If there are any variations in normal conditions then the information is send to remote system through cloud and corresponding activators are activated from the remote system in order to bring the situation to normal condition. This ensures a safety measures for hosipital building management system.

References

Akyildiz IF, Melodia T, Chowdhury KR (2007) A survey on wire- less multimedia sensor networks. Computer Networks (Elsevier) J 51:921–960

Al-Turjman F, Alfagih A, and Hassanein H (2012) A novel cost- effective architecture and deployment strategy for integrated RFID and WSN systems. In Proc. of the IEEE Int.Conf. On computing, networking and communications (ICNC), Maui, Hawaii, pp 835– 839

Al-Turjman F (2016) Impact of user's habits on smartphones' sen- sors: an overview. HONET-ICT International IEEE Symposium

Al-Turjman F (2016) Hybrid approach for mobile couriers election in smart-cities. In Proc. of the IEEE Local Computer Networks (LCN), Dubai, UAE, pp 507–510

Soro S, Heinzelman W (July 2009) Cluster head election tech- niques for coverage preservation in wireless sensor networks. Ad Hoc Netw 7(5):955–972

Abdulqader M, Walid A, Khaled A, Abdulrahman A (2016) A robust harmony search algorithm based Markov model for node deployment in hybrid wireless sensor network. Int J Geomate 11(27):2747–2754

Yick J, Mukherjee B, Ghosa D (2008) Wireless sensor network survey. Comput Netw 52:2292–2330

Dhawan H, Waraich S (2014) A comparative study on LEACH routing protocol and its variants in wireless sensor networks: a survey. Int J Comput Appl 95(8):975–8887

Barai LY, Gaikwad MA (2014) Performance evaluation of LEACH protocol for wireless sensor network. International Journal of Innovative Research in Advanced Engineering (IJIRAE) 1(6)

Mehndiratta N, Manju, Bedi H (2013) Wireless sensor network LEACH protocol: a survey. International Journal of Emerging Research in Management &Technology ISSN: 2278– 9359(Volume-2, Issue-3)

Malik M, Singh Y, Arora A (2013) Analysis of LEACH protocol in wireless sensor network. Volume 3, Issue 2

Heinzelman WR, Chandrakasan A, Balakrishnan H (2000) Energy- efficient communication protocol for wireless microsensor net- works. Proceedings of the 33rd Hawaii International Conference on System Sciences

Sonone, Abhay, Ajitkumar R. Khachane, and Deepali Vora. "A Dynamic Cluster formation for Query-Based Data Aggregation in Wireless Sensor Network: A Survey."International Journal of Computer Networking, Wireless and Mobile Communications (IJCNWMC) 4.2, Apr 2014, 137-142

Rashmi, M. R. "Analysis Of Data Aggregation Techniques In Wireless Sensor Networks." *International Journal of Applied Engineering Research and Development (IJAERD) ISSN (P): 2250* 1584.

Pavani, Movva, and P. Trinatha Rao. "Routing protocols in wireless sensor networks: A survey." *International Journal of Computer Networking, Wireless and Mobile Communications (IJCNWMC)* 6.2 (2016): 1-12.

Rathore, Nemi Chandra, et al. "Improved clustering protocol for delay minimization." *International Journal of Computer Networking, Wireless and Mobile Communications* 3.5 (2013): 41-48.

Pandey, Bhoomika, and Hardwari Lal Mandoria. "Analyzing Routing Protocol Based on Bat Algorithm for Sparely and Densely Deployment of Sensors in Wireless Sensor Network."International Journal of Mathematics and Computer Applications Research (IJMCAR) 6.4, Aug 2016, 1-8

Register of Patents

Patents Act 1990

Innovation Patent Patent no: 2021104204

Patentee(s):	Dakshina Murthy, N R DR of Associate Professor Civil Engineering CBIT Gandipet Telangana 500075 India
Inventor(s):	Srikanth, Nune Seshu, D .Rama
Title:	Self-Compacting Concrete With Processed Recycled Coarse Aggregate
Term:	Eight years from 15 July 2021
Date Granted:	25 August 2021
Date Certified:	
Date of Patent:	15 July 2021
Status:	GRANTED
Expiry Date:	15 July 2029
Date Ceased:	
Date Revoked:	

(12) PATENT APPLICATION PUBLICATION

(19) INDIA

(22) Date of filing of Application :22/06/2021

(21) Application No.202141028030 A

(43) Publication Date : 02/07/2021

(54) Title of the invention : HIGH STRENGTH CONCRETE WITH ALCCOFINE AS REPLACEMENT OF CEMENT

(51) International classification	:C04B0028020000, C04B0028040000, C04B0040000000, C04B0018160000, C04B0024260000
(31) Priority Document No	:NA
(32) Priority Date	:NA
(33) Name of priority country	:NA
(86) International Application No	:PCT//
Filing Date	:01/01/1900
(87) International Publication No	: NA
(61) Patent of Addition to Application Number	:NA
Filing Date	:NA
(62) Divisional to Application Number	:NA
Filing Date	:NA

(71)**Name of Applicant :**
1)Dr N R DAKSHINA MURTHY
Address of Applicant :Associate Professor, Civil Engineering, CBIT, Gandipet, Telangana 500075, India Telangana India
(72)**Name of Inventor :**
1)Dr N R DAKSHINA MURTHY
2)Dr K. JAGANNADHA RAO
3)Dr D. RAMA SESHU

(57) Abstract :
ABSTRACT HIGH STRENGTH CONCRETE WITH ALCCOFINE AS REPLACEMENT OF CEMENT A high performance concrete (102), the HPC (102) comprising: cement (104) in a range of 325 kilograms to 515 kilograms having a strength of about 70 N/mm2 at 28 days; fine aggregates (106) having a silt content less than 4% by weight, wherein the fine aggregates is of zone 1 or zone 2; coarse aggregates (108) having a specific gravity of 2.83 and water absorption of 0.2% by weight; and alccofine (110) as an admixture, wherein the alccofine is added in fixed a ratio of 10% by weight.

No. of Pages : 28 No. of Claims : 10

Workers Safety at Indian Construction Sites—A Survey

M. V. Krishna Rao, G. Tarun & V. Hari Leela

Conference paper | First Online: 14 May 2022

Part of the Lecture Notes in Civil Engineering book series (LNCE,volume 233)

Abstract

The construction industry is known as one of the unpredictable and hazardous sectors in which the workers are more susceptible to construction accidents. Despite many efforts put in to enhance construction site safety, construction accounts for quite disproportionate number of occupational injuries and fatalities. Urbanized nations attempt to guarantee stringent lawful enforcement of construction site safety in the industry by implementing various safety management systems. Conversely, occupational safety in the industry is found to be quite poor in a developing country like India due to lack of safety regulations and standards, data on safety at construction sites, safety training, safety awareness, and organized safety management systems. Although much improvement is achieved in construction safety, India is still lagging behind many other countries in this regard. This study focuses on the prevalent safety management practices in the Indian industry and the construction sites in the twin cities of Hyderabad and Secunderabad (TS, India), a hub of Infrastructure development in India, has been chosen as the area of study. The data collection was done through an industry survey by means of questionnaire and interviews with construction workers. The results revealed that the

Search results - chaitanya | Project Part-I CIE AY 2022- | Experimental assessment | ✓ Merge PDF - quickly, on | Sustainable Engineering - | Sci-Hub: article not found | Dr.M.V.Krishna Rao Profes

https://skkatariaandsons.com/view_book.aspx?productid=18587&Book=Sustainable Engineering&buyEBook=1

Other | KATSON BOOKS | Account | Cart

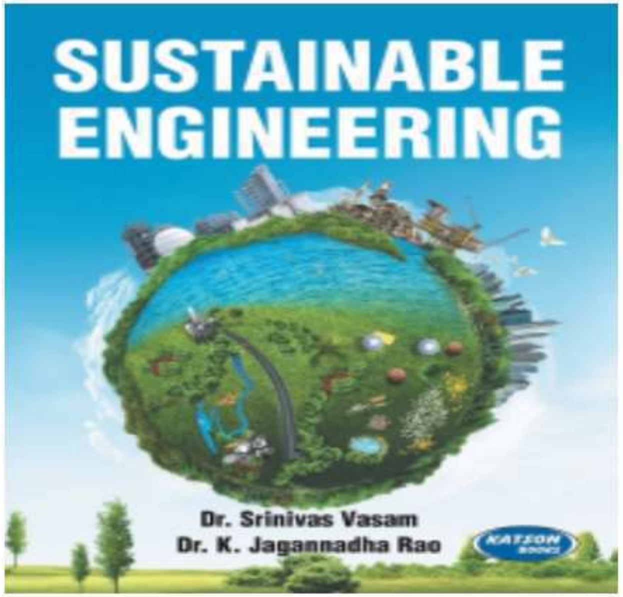
S.K. KATARIA & SONS

Educational Publisher

Search ... GO | 0 MY CART

HOME | AICTE BOOKS | NEW ARRIVALS | BEST SELLERS | OFFERS | CONTACT US | AUTHORS | CATALOGUE | SUBSCRIPTION | E-BOOKS | BLOG

Home | Book - Details



View E-Book Sample

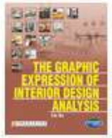
Sustainable Engineering

IN STOCK

ISBN	978-93-5014-714-6
Author	Dr. Srinivas Vasam & Dr. K. Jagannadha Rao
Publisher	S.K. Kataria & Sons
Edition	1st 2021
Publish Year	2021
Total Pages	300
BookType	E-Book
Availablity	In Stock

~~Rs 265.00~~Rs 199.00

RELATED PRODUCTS



The Graphic Expression Of Interior Design Analysis

~~Rs 595.00~~Rs 536.00

BUY / INSTITUTION/ BUY

MY WISH LIST

Please login to access Wishltist

Economy Delivery(Print Book Only): in 7 - 10 Working Days.

Express Delivery(Print Book Only) : in 5 -7 Working Days

Type here to search | 26°C Cloudy | ENG | 14:49 26-11-2022

Macroscopic Analysis of Traffic Flow Behaviour on Multilane Highways Under Heterogeneous Traffic Conditions

Kanchumurty Anusha, Poojari Yugendar & S. Moses Santhakumar

Conference paper | First Online: 15 December 2021

Part of the Lecture Notes in Civil Engineering book series (LNCE,volume 172)

Abstract

Traffic flow behaviour is a complex phenomenon and needs better understanding and concepts for its analysis. The highways in India normally operate under mixed traffic conditions, and the driving behaviour varies from one place to another. Macroscopic models which are quite suitable for describing the behaviour of entire stream and further accepted worldwide for estimation of capacity. The present study demonstrates the dynamic nature of PCU factors on two-way two-lane highways under highly heterogeneous traffic composition. Dynamic PCU's were estimated based on speed and size of vehicle type in the traffic stream with respect to a standard passenger car. The PCU values obtained in this study were compared with the existing static PCU's to get an overview of how the PCU varies when dynamics is involved. The present study also analyses the macroscopic traffic flow behaviour such as capacity and speed flow modelling on multilane highways. The VISSIM model parameters those were sensitive to capacity are calibrated based on the traffic composition