Wireless Communications and Mobile Computing

Smart Antennas and Intelligent Sensors Based Systems: Enabling Technologies and Applications 2021

Lead Guest Editor: Fawad Zaman
Guest Editors: Hing Cheung So, Daehan Kwak, Farman Ullah, and Sungchang Lee

Smart Antennas and Intelligent Sensors Based Systems: Enabling Technologies and Applications 2021

Wireless Communications and Mobile Computing

Smart Antennas and Intelligent Sensors Based Systems: Enabling Technologies and Applications 2021

Lead Guest Editor: Fawad Zaman
Guest Editors: Hing Cheung So, Daehan Kwak, Farman Ullah, and Sungchang Lee

Contents

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 7386049, 16 pages
https://doi.org/10.1155/2022/7386049

Research Article

Exploiting Machine Learning to Detect Malicious Nodes in Intelligent Sensor-Based Systems Using Blockchain

Maimoona Bint E. Sajid,[1] Sameeh Ullah,[2] Nadeem Javaid,[1,3] Ibrar Ullah,[4] Ali Mustafa Qamar,[5] and Fawad Zaman[6]

[1]*Department of Computer Science, COMSATS University Islamabad, Islamabad 44000, Pakistan*
[2]*School of Information Technology, Illinois State University USA, Normal, IL 61761, USA*
[3]*School of Computer Science, University of Technology Sydney, Ultimo, NSW 2007, Australia*
[4]*Department of EE, University of Engineering and Technology Peshawar, Bannu 28100, Pakistan*
[5]*Department of Computer Science, College of Computer, Qassim University, Buraydah 52571, Saudi Arabia*
[6]*Department of ECE, COMSATS University Islamabad, Islamabad 44000, Pakistan*

Correspondence should be addressed to Nadeem Javaid; nadeemjavaidqau@gmail.com

Received 29 July 2021; Revised 30 November 2021; Accepted 13 December 2021; Published 18 January 2022

Academic Editor: Abdul Basit

In this paper, a blockchain-based secure routing model is proposed for the Internet of Sensor Things (IoST). The blockchain is used to register the nodes and store the data packets' transactions. Moreover, the Proof of Authority (PoA) consensus mechanism is used in the model to avoid the extra overhead incurred due to the use of Proof of Work (PoW) consensus mechanism. Furthermore, during routing of data packets, malicious nodes can exist in the IoST network, which eavesdrop the communication. Therefore, the Genetic Algorithm-based Support Vector Machine (GA-SVM) and Genetic Algorithm-based Decision Tree (GA-DT) models are proposed for malicious node detection. After the malicious node detection, the Dijkstra algorithm is used to find the optimal routing path in the network. The simulation results show the effectiveness of the proposed model. PoA is compared with PoW in terms of the transaction cost in which PoA has consumed 30% less cost than PoW. Furthermore, without Man In The Middle (MITM) attack, GA-SVM consumes 10% less energy than with MITM attack. Moreover, without any attack, GA-SVM consumes 30% less than grayhole attack and 60% less energy than mistreatment. The results of Decision Tree (DT), Support Vector Machine (SVM), GA-DT, and GA-SVM are compared in terms of accuracy and precision. The accuracy of DT, SVM, GA-DT, and GA-SVM is 88%, 93%, 96%, and 98%, respectively. The precision of DT, SVM, GA-DT, and GA-SVM is 100%, 92%, 94%, and 96%, respectively. In addition, the Dijkstra algorithm is compared with Bellman Ford algorithm. The shortest distances calculated by Dijkstra and Bellman are 8 and 11 hops long, respectively. Also, security analysis is performed to check the smart contract's effectiveness against attacks. Moreover, we induced three attacks: grayhole attack, mistreatment attack, and MITM attack to check the resilience of our proposed system model.

1. Introduction

The wireless sensor networks (WSNs) play a vital role in the Internet of Sensor Things (IoST). The IoST networks consist of sensor nodes that are deployed for the environmental monitoring [1]. These sensor nodes are used to sense ambient information like temperature, humidity, and pressure [2]. IoST is an emerging domain, which supports different applications like industrial Internet of Things (IoT), smart cities, air pollution detection, and underwater monitoring.

These networks face well-known issues due to deployment in harsh and unattended environments like attackers can attack the network by compromising the sensor nodes, which have very sensitive information like identification (ID) and location [3, 4]. The credentials are used in different cryptographic functions like encryption and decryption for

generating cipher text. However, these credentials are misused by physically accessing the nodes. As a result, the authors propose different mechanisms to address this issue. The authors store their data at a centralized server, which can securely and privately keep the records using different cryptographic techniques [5, 6]. Moreover, as the systems are controlled by a centralized authority, therefore, it is easy to be manipulated, which can lead to trust issues.

1.1. Problem Statement. The IoST nodes are threatened by the external malicious nodes (MNs), which eavesdrop the communication channels and extract personal information for their interest. Also, the nodes perform malicious activities during routing operations in the network and cause the grayhole attack [7]. To address the issue, the authors in [8] propose a model that consists of two blockchains, encryption schemes, and digital signatures to achieve authentication in the IoST. However, the usage of two blockchains causes extra communication and computational overheads. Moreover, the authors in [9, 10] use blockchain to resolve a single point of failure issue and trace the malicious activities using Merkle tree. The author in [11] uses blockchain-based secure hashing algorithm for authentication of nodes. The algorithm helps to detect MNs in the network. In [12, 13], the authors propose a blockchain-based secure data storage. However, data storage over the blockchain is costly as compared to other centralized storage platforms. In [14], trust aware localized routing is performed to find the secure route. Also, a blockchain-based authentication mechanism is proposed. However, high energy is consumed by the forwarding nodes when delivering the data packets without considering the optimal route. In [15], a blockchain-based nonrepudiation service provisioning scheme is proposed where denial of services by both the provider and client is not possible. The reason is that the proposed scheme uses homomorphic hashing for service verification, which consumes high computational cost.

1.2. Research Objectives. The research objectives (ROs) of this research work are as follows.

RO 1. The CHs are used to verify and aggregate the nodes' data and perform extensive routing. Therefore, CHs die early because of large computational overhead. One of the objectives of this research is to reduce the computational overhead (related research questions (RQs) are 1, 2, and 3).

RO 2. In multihop routing, a source node has multiple paths to forward the data packets towards the destination. However, selecting the longest path increases the delay. Therefore, it is one of our objectives to find the shortest path from source to destination node (related RQ is 4).

RO 3. In multihop routing, the malicious nodes drop the data packets. Therefore, it is our objective to detect the malicious nodes in order to increase the network throughput and Packet Delivery Ratio (PDR) (related RQ is 5).

1.3. Research Questions. RQs are given on the basis of ROs.

RQ 1. How to register the nodes that participate in routing?

RQ 2. Where the routing data will be stored?

RQ 3. How routing data is verified?

RQ 4. How to find the shortest path for forwarding the data?

RQ 5. How to detect the malicious nodes during routing?

1.4. Research Answers. Research answers (RAs) to the RQs are given as follows.

RA 1. The blockchain is used to register nodes in the network (for more details, see Section 3).

RA 2. Routing data, which includes the number of nodes, sent from source to destination and stored in the blockchain (for more details, see Section 3).

RA 3. The PoA consensus mechanism is used to verify the routing data (for more details, see Section 3).

RA 4. The Dijkstra algorithm is used to find the shortest path from source to destination (for more details, see Section 3).

RA 5. The GA-SVM and GA-DT are used to classify the nodes as malicious and legitimate (for more details, see Section 3).

1.5. Contributions. The abovementioned problems are resolved in this paper. The research contributions of the work are given as follows.

(i) A lightweight blockchain-based registration and authentication mechanism is proposed to make the network secure

(ii) The blockchain is used to store routing information in a decentralized fashion so that nobody can alter it

(iii) PoA consensus mechanism is used to reduce the computational cost consumed by PoW because in our scenario, nodes are resource constrained and are not capable of solving the puzzle

(iv) MNs are detected using the Support Vector Machine (SVM) and Decision Tree (DT) algorithms to make the system secure from internal malicious nodes

(v) We have performed security analysis for evaluating the smart contract using Oyente tool. Moreover, the proposed model's robustness against attacks is evaluated by inducing three attacks in the attacker model: grayhole attack, mistreatment attack, and MITM attack

1.6. Blockchain. The blockchain is explored by different authors to make the network free from concept of centralization because blockchain consists of a decentralized and distributed ledger. For decision making, a third party is replaced with a smart contract, which is used to write all the agreements. The data saved in blockchain is immutable and does not face a single point of failure issue [16]. The blockchain has different applications like healthcare [17], banking, energy trading [18], and smart cities [19]. The blockchain has three types: public, private, and consortium. In the public blockchain, any node can join the network and each node has the authority to add and access the data within the blockchain. In private blockchain, the data is not publicly available and only the authorized nodes have the authority to add and access the data, whereas the

consortium blockchain is a semiprivate blockchain, where multiple organizations become part of the network. In this type of blockchain, different nodes have different rights in the network. For example, some nodes have the authority to perform and validate transactions while others are authorized to access the stored data only. Furthermore, different consensus mechanisms like Proof of Work (PoW), Proof of Authority (PoA), and proof of stake are used to make consensus between nodes to add new data blocks in the blockchain. Moreover, in PoW, the miners solve the mathematical puzzle. A miner who solves the puzzle first has the authority to add a new block in the blockchain. In the PoA, preselected nodes perform mining for adding new data blocks in the blockchain in which miners are selected based on the reputation values [7].

The rest of the paper is organized as follows. Section 2 discusses the related work. The proposed model is explained in Section 3. In Section 4, simulation results are presented. Section 5 provides the conclusion.

2. Related Work

In this section, categorization of related work is done on the bases of addressed limitations.

2.1. Authentication and Privacy Leakage. In [1], the authors motivate the users to use mobile devices by providing incentives. However, the authors ignore the privacy leakage issue. The customers' trust is not developed because of privacy leakage. The privacy of data is the main issue in crowdsensing. In [8], traditional authentication of IoST depends on a third party, which may act maliciously. In [20], IoST is useful for sending data from source to destination. However, data is not secure because of malicious attacks. Sensor nodes are not authentic. Therefore, they act maliciously and drop the data packets. In [15, 21], records are not secured and data privacy is compromised, which leads to trust issues in data trading. Furthermore, IoST generates excessive data, which causes privacy issue. In [22], the authors propose an authentication-based signature mechanism for securing the communication. Moreover, in [23], data trading is performed and high computational power is required for storing and trading data. However, the IoST has limited storage capacity and battery. The authors in [24] mention that single side authentication is not efficient and can be attacked by the outsider nodes. Also, a single centralized site could break down due to any external or internal attack. In contrast to the mentioned problems, the authors propose a blockchain-based decentralized authentication model, which provides the best fault tolerance to the network. The authors in [25] state that in smart cities, network can extract data from outdated devices, which may be operated in unattended environments.

2.2. Malicious Nodes' Detection and Secure Routing. Uddin et al. propose the hierarchical transmission of data in underwater sensor network, where sensors are mobile and data transmission is not secure because of unattended environment [26]. The authors in [27] propose an intrusion detection system on the basis of data generated by the intruders. The authors use the deep learning techniques to train the model for intrusion detection. For hierarchy distribution, fog and cloud are used with secure data transmission. This model ensures the legitimacy of data. Feng et al. state in [28] that traditional models have a centralized database in which temperature, humidity, and gas information of the seafood like shellfish is stored. However, due to the usage of a centralized database, there are high chances of malicious attacks that tamper the data to affect the quality of shellfish. In [10], the WSN faces two types of attacks: internal and external. In the internal attack, internal and legitimate nodes (LNs) become malicious, which send and access the data as legitimate nodes, which affect the overall performance of the network, while in the external attack, the attacker can attack the entire network from outside. In [12], the WSN nodes work as mobile nodes. When any node sends the data from the source to the destination, intermediate mobile routing nodes act as MNs, which tamper and drop the data packets. The authors state in [16, 29] that security and privacy risks increase through modification of the data packets while sending them from the source to destination. In traditional architectures [30], WSNs are centralized and are controlled by the third party, which stores the data. However, a single point of failure issue occurs, and MNs tamper the data. In [31–33], IoST faces the issue of internal attacks that affect the network's lifetime, scalability, and throughput. In [7], MNs broadcast false destination address, which affects the network's performance because data packets are continuously moved in the loop. As a result, the data packets are timed out and are dropped. The authors in [34] assume that BSs could also behave maliciously in the network. Therefore, a blockchain-based secure verification mechanism is proposed to authenticate the BSs. In [11], the author proposes a lightweight authentication and security protocol because sensor and IoT nodes are small in physical size. Also, the author proposes a model for interoperability between different vendors' devices. In [35], MNs are deployed in the network and where they act as beacon nodes. MNs send tampered data to the base station and broadcast the wrong location of the beacon nodes. Therefore, localization errors occur during the localization process. Moreover, wrong localization affects the lifetime of WSN and consumes high energy. In [36], the authors state that the localization of unknown nodes is the main issue in the range free WSNs. However, some beacon nodes act maliciously and broadcast their wrong location. Resultantly, unknown nodes' positions are not accurately calculated. In [37], the authors propose a lightweight routing mechanism based on swarm intelligence for fast and efficient communication in the network.

2.3. Single Point of Failure. The traditional routing protocols are centralized and use a central authority for authentication of IoST [9]. In [29], IoST is widely used in smart city and the number of IoST is increasing on daily basis. However, the model is centralized in which only cloud servers are used for storing and accessing the data. In [21], sensors are used for sensing the data and saving the records on a centralized database. There are high chances of single point of failure. In

[38, 39], the software defined network (SDN) is commonly used for the communication of nodes. The SDN separates the data plane from the control plane. However, the SDN is centralized and faces the issue of single point of failure. Security issue also arises in the centralized system. In [40], IoST also faces the main challenge of security. Moreover, devices are centralized, and issue of single point of failure occurs.

2.4. Resource Constrained Nodes. In traditional models [41–43], blockchain is only used for maintaining the data in WSNs. However, the PoW consensus mechanism consumes high computational power. In [44, 45], mobile sensors cover large areas of smart cities. However, sensor nodes are resource constrained, and their deployment and maintenance cost is very high. In [46], the sensor nodes have to send data to the base station. The sensor nodes are resource constrained and have no computational power to send data over a large area. In [47, 48], IoST uses WSNs for storing the data. However, WSNs are resource constrained.

3. Proposed System Model

In this section, the details of the proposed system model are presented. Firstly, the blockchain is deployed. Secondly, registrations and authentication mechanism is proposed. Thirdly, MNs are detected using two machine learning algorithms: SVM and DT. Fourthly, the details of the shortest path for routing are given, and secure routing is performed, as shown in Figure 1.

3.1. Network Deployment. In the proposed system model, 200 nodes are deployed in the network in which one node is appointed as a base station, some nodes are selected as cluster heads (CHs) based on their resources while remaining nodes are regarded as ordinary nodes. Moreover, blockchain is deployed on the CHs and base station because they have sufficient storage and computational resources. The private blockchain is integrated with IoST to ensure the security of the network and data by keeping record of every transaction in its ledger. Also, the nodes are registered in blockchain where a unique number is assigned to each node. Moreover, the PoA consensus mechanism is suitable for the private blockchain; therefore, it is implemented for verifying the transactions.

3.2. Registration and Authentication. Authentication of nodes is necessary because the nodes are deployed in an unattended and harsh environment. In this environment, nodes are more likely to be attacked by the attackers. These attackers physically access and misuse the connection of particular nodes with the network. Also, attackers can reuse the IDs of the nodes to reenter the network and for doing other malicious activities. Therefore, many authors propose different registration and authentication mechanisms using blockchain where the chances of nodes acting maliciously are minimum because of blockchain's unique characteristics, discussed in Section 1. However, there are still chances of different types of attacks that can be done by manipulating the credentials of the network nodes, although, in most of the papers, hashes are stored on the blockchain, which can be guessed by different methods such as brute force attack and are involved in the malicious activities. Therefore, in this section, a secure and lightweight registration and authentication mechanism is proposed. However, the storage over blockchain is very costly as compared to traditional storage mechanisms. Therefore, the process of registration and authentication is made lightweight to minimize the burden of storage from blockchain. According to our network's requirement, the system will be less benefited because nodes have limited time period for communication. Therefore, nodes' lifetime will finish before accessing and guessing the hashed value of the credentials by the attackers. The detailed and step-wise workflow of the registration and authentication mechanism is as follows. Moreover, Algorithm 1 also depicts the pseudocode of the proposed registration and authentication mechanism.

Step 1. The first step of the registration is to send the credentials to the blockchain B. The package of registration request req $\longrightarrow B$ contains identity of the sensor node ID_S and location of the node L_S.

$$\mathrm{Reg}_{\mathrm{req}\longrightarrow B} = (\mathrm{ID}_S, L_S). \tag{1}$$

Step 2. Blockchain will generate a unique three digit number $\mathrm{No.}_{\mathrm{unique}}$. These digits will be added to the above package to get rid from the collision of hashes. The unique number will be assigned to the node as its password for future correspondence.

$$B \longrightarrow \mathrm{hashing} = \left(\mathrm{ID}_S, L_S, \mathrm{No.}_{\mathrm{unique}}\right)_{\mathrm{hash}}. \tag{2}$$

Step 3. The hash and the unique number are stored in the blockchain for future use like authentication. The hashes will be converted to the cipher text, which would not be guessed by the attacker even if ID and location of the node are obtained.

Step 4. After successful registration of the nodes, if a node wants to access the network, it will send authentication request to blockchain. The authentication request contains the following data in its package.

$$\mathrm{Auth}_{\mathrm{req}\longrightarrow B} = \left(\mathrm{ID}_S, L_S, \mathrm{No.}_{\mathrm{unique}}\right). \tag{3}$$

Step 5. The blockchain will generate the hash of the above given credentials and will compare it with the already stored hash. If the match is successful, blockchain will allow the sensor nodes to communicate successfully. Otherwise, the authentication error message will be popped up. All the steps involved in authentication and registration process are given in Algorithm 1.

Note. It is our assumption that the unique number is securely shared with the node. However, we did not consider the method of sharing the number. Sharing the unique number is necessary because if some other entity knows about this number, we will lose our objective to use this number.

Step 6. After the authentication process, the source node sends the data packets to the destination node through

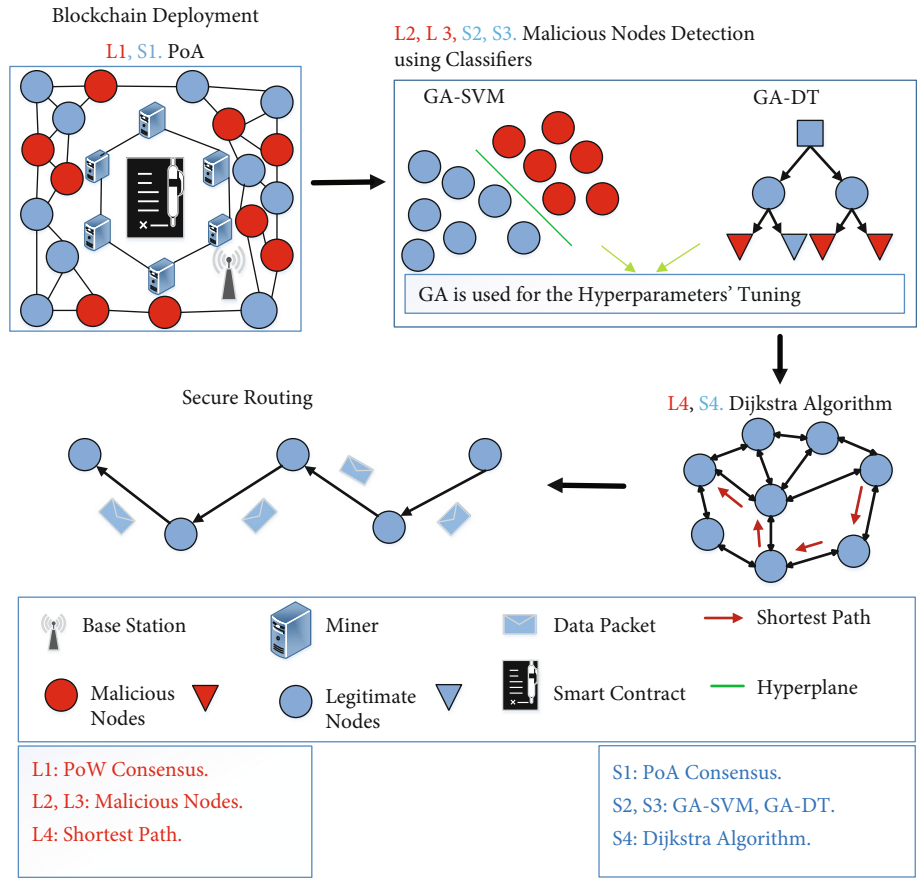


Figure 1: GA-SVM and GA-DT-based MN detection during routing using blockchain.

1 Initialization
▷ Registration
2 Send to Blockchain (ID_S, L_S);
3 Blockchain Generate $No._{unique}$;
4 Blockchain Calculates and Stores the Hash $(ID_S, L_S, No._{unique})$;
▷ Authentication
5 Send to Blockchain $(ID_S, L_S, No._{unique})$;
6 Blockchain Calculates the Hash $(ID_S, L_S, No._{unique})$;
7 if Hash == Already Stored Hash then
8 Node is Authenticated and Allowed to Communicate;
9 else
10 Node is Deauthenticated and Revoked by the Blockchain;
11 end

Algorithm 1: Registration and authentication process.

intermediate nodes like CHs, which perform secure routing by storing the routing data on the blockchain.

3.3. Malicious Node Detection. After the deployment of the network, the presence of MNs in the network affects the network performance. In our proposed model, these nodes are detected by GA-DT and GA-SVM. Before detecting MNs, we first synthesize the data for the algorithms.

3.3.1. Synthetic Data Generation. In the IoST, each node sends the data packets to the destination. After forwarding the data packets to the destination, we calculate how many

data packets are dropped, misrouted, modified by the intermediary MNs, and successfully received by the destination. The following features are used to generate the synthetic data [49].

(a) *Packet drop Ratio (PdR)*. PdR is the ratio of total number of data dropped *Pd* to the total number of data packets coming from the node *Pi*, which is calculated using Equation (4) [49].

$$\text{PdR} = . \tag{4}$$

(b) *Packet Modification Ratio (PMR)*. The PMR is the ratio of the total number of data packets modified Pm to the total number of data packets coming from the source node *Pi*, which is calculated using Equation (5) [49].

$$\text{PMR} = . \tag{5}$$

(c) *Packet Misroute Rate (PMiR)*. The PMiR is the ratio of the total number of data packets misrouted Pmi to the total number of data packets transmitted by the source node *Pf*, which is calculated using Equation (6) [49].

$$\text{PMiR} = \frac{\text{Pmi}}{Pf}. \tag{6}$$

(d) *Packet Delivery Ratio*. The PDR is the ratio of the total number of data packets successfully received at the destination *pt* to the total number of data packets coming from the source node *Pt*, which is calculated using Equation (7) [49].

$$\text{PDR} = . \tag{7}$$

The thresholds are defined for PdR, PMR, PMiR, and PDR in Algorithm 2, and nodes are labeled as malicious and legitimate. The synthetic dataset is used for GA-SVM and GA-DT. Moreover, threshold used in Algorithm 2 is taken from [49].

3.3.2. Classification Using Genetic Algorithm-Decision Tree. DT splits the data into the root, internal, and leaf nodes that form the tree and classifies a particular node as malicious or legitimate. In DT, the dataset is given to root node, which performs testing and splits the dataset into subclasses. The internal nodes represent the outcome of the root node. The leaf nodes that do not split further are the results of classification and represent the class labels. The hyperparameters used in GA-DT are as follows.

```
1 Initialization (nodes =200, data packets =200)
2 for i =1 : nodes do
3 for j =1 : data packets do
4 if receivepackets(i) ≠ inf then
▷S(i) is a structure
5 PDR(i) = (S(i).transmitted data)/j;
6 PdR(i) = S(i).drop packets/j;
7 PMiR(i) = (S(i).misroute packets)/forwarded packets;
8 PMR(i) = (S(i).modified packets)/(S(i).transmitted data);
9 end
10 if PdR(i) ≥2 && PMiR(i) ≥3.9800 &&
PMR(i) ≥0.9975 && PDR(i) ≥98.2500 then
11 MN =0;
12 else
13 LN =1;
14 end
15 end
16 end
```

Algorithm 2: Synthetic data generation.

(a) *Maximum Features*. It shows that to what extent a DT should be splitted. When a tree is splitted deeply, better classification is performed. The maximum depth values are auto, square root (sqrt), log, and none.

(b) *Minimum Sample Leaf*. It is the minimum number of samples present at the leaf node. The parameters are int, float, and none optional.

(c) *Minimum Sample Split*. It is the splitting of the internal node in the minimum range. The parameters are int, float, and optional.

(d) *Criterion*. It is used to measure that how perfectly a DT is splitted. Gini index and entropy are features.

Gini index and entropy criteria are used to measure that either DT is splitted perfectly or not. The data is labeled randomly, and Gini is used to measure the amount of data falsely labeled. If the Gini's value is low, it means classes are perfectly labeled. Moreover, entropy is used for the best splitting of DT. The best classification is performed by tuning the hyperparameters.

The aforementioned parameters are tuned using GA, which evaluates the input values in such a manner that it generates the best output. In GA [50, 51], population is generated randomly, as shown in Algorithm 3. The fittest individual is selected as a parent on the basis of the objective function. After the parent's selection, crossover is performed and offsprings are generated. In the mutation, offspring genes are altered and the optimal solution is generated. As a result, the best hyperparameters are selected from the given features. After tuning the parameters, training and testing are performed. Then, DT classifies LNs and MNs. The detected MNs do not participate in the routing.

```
1 load dataset;
2 split the data 70% = testing and 30% = training;
3 initialize hyperparameters = (Maximum Depth, Minimum Sample Leaf, Minimum Sample Split, Criterion);
4 GA:Maximum Depth, Minimum Sample Leaf, Minimum Sample Split, Criterion;
5 initialize population
6 (generation,populationsize,offspringsize);
7 perform tuning of hyperparameters;
8 perform selection;
9 perform crossover;
10 perform mutation;
▷Decision Tree is DT
11 train DT on selected parameters;
12 DT saves the model;
13 DT tests the model;
14 DT loads the model;
```

Algorithm 3: Detection of malicious nodes with classification by GA-DT.

3.3.3. Classification Using Genetic Algorithm and Support Vector Machine. SVM is a supervised learning technique, and it is the foremost classification model. It performs linear and nonlinear classification. In the proposed model, a linear classification is performed and the hyperplane is created by the line formula [52], as given in Equation (8).

$$y = mx + b, \tag{8}$$

where m is the gradient that shows the height of the y-axis and divided by the distance of the x-axis. The variable b shows the interception of x and y. A hyperplane is created through line equation by defining $x = (x1, x2)$ and $w = (m, -1)$. By putting the values of x and w in Equation (8), Equation (9) is derived [52].

$$w.x + b = 0. \tag{9}$$

The hyperplane is a decision boundary that is used to separate two different classes. Each class margin shows the distance of support vector to the hyperplane. When margin is maximum, the error rate is minimum. In the proposed model, the synthetic dataset is loaded and classification is performed using SVM. The hyperplane classifies both MNs and LNs. SVM hyperparameters are tuned for the best classification.

SVM hyperparameters are as follows.

(a) *C Classification*. Minimum accuracy of classification is high, and perfect decision boundary is created. Its values are 0.1, 1, 2, 5, 7, and 10.

(b) *Kernel*. It tells that how hyperplane is created for one dimension or multiple dimensions. Radial basis function, linear, and polynomial are features of kernel.

The aforementioned parameters are tuned using GA. As a result, the best values are selected from the given hyperparameters. After tuning the hyperparameters, the model is trained, as given in Algorithm 4. Then, SVM excellently classifies both legitimate and malicious classes. After the classification, GA-SVM and GA-DT results are compared in which GA-DT has 96% accuracy and GA-SVM has 98% accuracy. Therefore, MNs are detected by GA-SVM. On the basis of GA-SVM's accuracy, MNs are revoked and only the LNs are allowed in the network.

3.4. Shortest Path for Routing. After the detection of MNs, in the third step, the source node selects the shortest path to the destination and performs secure routing in the absence of MNs. In the proposed model, the Dijkstra algorithm [53] is used to find the shortest path and it considers all paths from the source to the destination node. The energy consumption of the nodes is reduced after finding the shortest path on the basis of weights assigned to each edge of the node [54, 55]. In Figure 2, the source node is CH1 and the destination node is CH7. The source node has six possible paths going towards the destination. The Dijkstra algorithm finds the shortest path from the given six paths on the basis of the distance. As a result, the shortest path distance is eight hop and CH2, CH3, and CH6 are intermediary nodes used to forward the data packets to the destination, as given in Algorithm 5. After shortest path selection, secure routing is performed.

Table 1 shows the mapping between limitations identified, solutions proposed, and validations done. The first limitation (L1) is that PoW consumes high computational power.

Therefore, in the proposed model, PoA is used for verifying the data because the transaction cost of PoA is less than PoW. The second and third limitations (L2 and L3) are MN detection and grayhole attack, respectively, which are detected using GA-SVM and GA-DT. The features that are used to train the model are PDR, PMiR, PMR, and PdR. GA is used for tuning the hyperparameters of SVM and DT. After tuning the hyperparameters, optimal classification is performed. In the fourth limitation (L4), high energy is consumed using the longest path for routing. The source node sends the data packets to the intermediary nodes who forward the packets to the destination. If the

```
1 load dataset;
2 split the data 70% = testing and 30% = training;
3 initialize hyperparameters = (kernel, c);
4 GA:kernel, c;
5 initialize population
6 (generation,populationsize,offspringsize);
7 performs tuning;
8 Perform selection (c = [0.1, 1, 2, 5, 7, 10]);
9 Perform crossover ([0.1, 1, 2, 5, 7, 10]);
10 Perform mutation ([0.1, 1, 2, 5, 7, 10]);
11 selection (kernel = 'radialbasisfunction','linear','polynomial');
12 crossover ('radialbasisfunction','linear','polynomial');
13 mutation ('radialbasisfunction','linear','polynomial');
14 SVM trains the model on the selected parameters [c=2, kernel=linear, degree=1];
15 SVM saves the model;
16 SVM tests the model;
17 SVM loads the model;
```

Algorithm 4: Detection of malicious nodes with classification by GA-SVM.

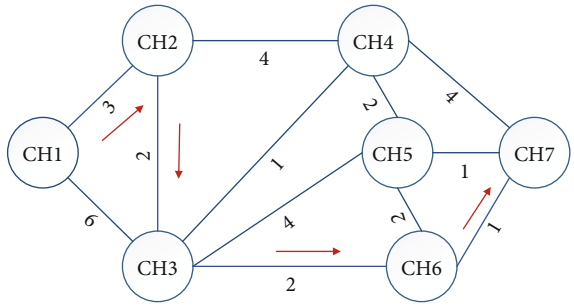


Figure 2: Shortest path selection using the Dijkstra algorithm.

intermediary node is not in the range of destination node, it forwards the data packets to the next intermediary node. In this way, a long routing path is created and energy consumption is increased. In the proposed model, the Dijkstra algorithm is used to find the shortest path. The Dijkstra algorithm computes the distance from all nodes and selects the shortest path. After the selection of the shortest path, secure routing is performed by sending the data packets to the destination.

4. Performance Evaluation

The simulations are performed to validate the performance of the proposed system model. The ML techniques are exploited parallelly working with blockchain. GA-SVM and GA-DT techniques are used to classify LNs and MNs on the basis of their behaviour. The features on which the model is trained are PDR, PdR, PMR, and PMiR. The class labels are malicious and LNs. The dataset is splitted into 70% training and 30% testing sets. Moreover, novelty of system model is depicted through Table 2. The simulation environment used in the proposed work comprises Metamask, Ganache, and Remix-IDE for the implementation of blockchain. Moreover, Python-based Web3.api is used for the interaction of blockchain with networks. ML techniques

```
1 if Source == Destination then
2 Cost = 0;
3 Route=Source;
4 end
5 if Destination==1 then
6 Destination = Source;
7 end
8 Adjacency=exchange the node (Adjacency,1,Source);
9 LengthA=size(Adjacency);
10 */Weight is W*/
11 for i=2 : LengthA do
12 W(1,i)=i;
13 W(2,i)=Adjacency(1,i);
14 end
15 for i=1:Length Adjacency do
16 Destination(i,1)=Graph(1,i);
17 Destination(i,2)=i;
18 end
19 while Cost < = (size(W,1)-1) do
20 Cost=Cost+1;
21 for i=1:size(Distance,1) do
22 Calculate the distance;
23 end
24 Destination=Sort the rows and find Destination;
25 Add the Distance of each Route to the Destination;
26 end
```

Algorithm 5: Dijkstra algorithm.

are implemented in Python to classify LNs and MNs. Moreover, the network parameters are depicted in Table 3.

4.1. Simulation Results and Discussion of Proposed Model. In Figure 3(a), GA-SVM is compared with DT, SVM, and GA-DT to show the effectiveness of the proposed model. DT is used for the classification of LNs and MNs. However, DT consumes large time for training the model because it is complex in nature. Therefore, more computational resources are required to train the model. DT has 88% accuracy, and the hyperparameters used for DT are criterion, minimum

sample leaf, and maximum features. The accuracy is low as compared to other models because DT itself chooses the hyperparameters for tuning. On the other hand, SVM's accuracy is 93%, which is more than DT because SVM performs better linear classification than DT. It also works better with few features and data instances as compared to DT. Furthermore, GA is used to tune the hyperparameters of DT and SVM to improve their accuracy. GA performs tuning and selects the best parameters from the given parameters. As compared to simple DT and SVM, GA does not select the parameters itself while it takes the best hyperparameters using crossover and mutation. In crossover, the best parents are taken from the population and crossover is performed between them. If the resultant offsprings are better than the existing ones, then they are replaced with previous ones. Moreover, mutation is used for making the solutions more diverse. Resultantly, the required results are obtained. The accuracy of GA-DT is 96%, which is higher than SVM and DT. The reason is that SVM has higher accuracy as compared to DT, and when parameters of SVM are tuned using GA, its accuracy further increases. Moreover, the proposed model uses GA-SVM for classification, and it has 98% accuracy, which is higher than DT, SVM, and GA-DT because SVM's accuracy is greater than DT's accuracy. SVM's input parameters are tuned using GA, which helps SVM to achieve better results than simple SVM, DT, and GA-DT. GA-SVM's accuracy is high because GA efficiently tunes C, kernel, and degree hyperparameters. GA evaluates the best value for the given parameters and tunes the parameter according to the selected value. Simple SVM is not good for predicting the range of hyperparameters. It is because SVM selects the hyperparameters on its own, and its accuracy is less than GA-SVM.

The precision is the ratio of true positive to the sum of false positive and true positive. The precision of DT is 100% while SVM has 92% precision. The reason is that SVM has less number of true positives as compared to DT. GA-DT and GA-SVM have 94% and 96% precision, respectively, as shown in Figure 3(b).

Figure 4(a) illustrates that the PdR is decreased when only the LNs are present in the network. In the presence of MNs, the PdR increases to 53% because MNs receive the data packets and do not forward them to the destination. GA-SVM detects 21.75% MNs that are present in the network.

In Figure 4(b), the red bar shows that PMR is 85% in the presence of MNs. MNs tamper the data packets and forward them to the destination. The blue bar shows the number of nodes that do not modify the data packets.

MNs receive the data packets and misroute them to the wrong destination. In the presence of MNs, PMiR increases, as compared to the LNs, as shown in Figure 5(a). When the data packets are misrouted, it causes network congestion.

PDR is higher in the absence of MNs because maximum number of data packets sent from the source are successfully received at the destination, as shown in Figure 5(b). When the PDR is maximum, it increases the performance of the IoST. GA-SVM classifies MNs and LNs. When MNs are revoked from the network, PDR becomes 99.72%.

The source node sends the data packets to the destination node. In Figure 2, seven nodes are selected for the routing. One and seven are source node and destination node, respectively. The Dijkstra algorithm finds the shortest path in the network from the source to the destination. As shown in Table 4, distances are calculated for possible paths from the source to the destination. Distance is calculated on the basis of weights. The selected path's distance is eight, which is smaller as compared to other paths. The source node CH1 sends the data packets using the shortest path, which consumes less energy. The Dijkstra algorithm is compared with the Bellman Ford algorithm. The Bellman Ford's calculated distance is 11 hop, which is greater from Dijkstra for the same source and destination. The Bellman Ford is also more time-consuming than Dijkstra. Bellman Ford follows dynamic programming in which current problem is solved using previous solutions. Moreover, Dijkstra is a greedy algorithm, which solves the problem by finding optimal solution.

We use PoA consensus mechanism in the blockchain for nodes' registration and data storage. In Figure 6(b), PoA is compared with the PoW. The PoW uses the Ropsten network and PoA uses the Rinkeby network. The PoA consensus is used because private blockchain is deployed in the proposed model. The PoW consensus works efficiently in the public blockchain. As a result, PoW is expensive to deploy the smart contract and consumes higher transaction cost as compared to PoA. Transaction cost occurs when a smart contract is deployed while storing the data of node. When the number of nodes increases, transaction cost also increases because the number of records increases.

4.2. Attacker Model. The security analysis of the proposed model is performed after inducing a grayhole attack [56], a mistreatment attack [57, 58], and MITM attack [59]. Their analysis is shown in Table 5. In the proposed model, we use GA-SVM for detecting malicious nodes and revoked them from the network.

4.2.1. Energy Consumption. Figure 7 shows the effect of grayhole, mistreatment and MITM attacks, and the proposed solution based on energy consumption. During routing, the nodes' energy is consumed when data packets are sent toward the destination. From the figure, it is shown that the grayhole attack has less impact on the energy consumption as compared to mistreatment attack. For the grayhole attack, the malicious nodes do not forward all of the data packets to the destination, and they drop some of the data packets before reaching the destination. Therefore, some of the data packets do not reach CHs, which make it to consume less energy. Furthermore, the mistreatment attack has a high energy consumption as compared to the grayhole and MITM attacks. In this attack, the malicious nodes tampered with the routing table and changed their destination. Therefore, the data packets are forwarded to the wrong destination, and the nodes consume high energy because malicious nodes give long paths and false destinations. Moreover, low energy is consumed in the MITM attack as compared to grayhole attack when sending the tampered

Table 1: Mapping of identified limitations with proposed solutions and validations.

Limitations identified	Solutions proposed	Validations done
L1: PoW utilizes high computational power [10].	S.1: PoA is used that utilizes low computational power.	V.1: transaction cost, as shown in Figure 6
L2: presence of MN in the network [29] L3: grayhole attack is possible on routing nodes [7].	S.2, S.3: GA-SVM and GA-DT are used for the detection of MNs.	V.2, V.3: accuracy, precision, PDR, PMR, PdR, and PMiR, as shown in Figures 3(a) and 3(b), 4(a) and 4(b), and 5(a) and 5(b)
L4: long paths deplete nodes' energy [14].	S.4: the Dijkstra algorithm is used to find the shortest path.	V.4: distance from source to destination is calculated, as shown in Figure 5.
L5: registration consumes more gas due to hybrid blockchain [14].	S.5: lightweight registration and authentication mechanisms	V.5: transaction cost, as shown Figure 6

Table 2: Feature comparison with respect to proposed model.

Contributions	Ref. No.	Similarities	Differences	Limitations
Registration and authentication	[8]	Distributed network authentication	Hybrid blockchain different levels of nodes registered at different platforms	Complex to manage two blockchains
	[30]	Authentication	Use of multiblockchains and cluster manager	Difficult to manage
Routing	[7]	Routing, MN detection	Calculate through Qlearning	Consumes more energy while discovering the route
	[14]	Distributed network, routing	Localization-based routing	Select relatively longer path, which causes rapid energy
MN detection	[10]	Distributed network, registration, MN detection	Detection through network parameters, calculation of node's reputation	Uses computationally complex consensus algorithm
	[29]	Detection through ML, distributed network	SDN-based network, multilevel detection	Considered very few attacks

Table 3: Simulation parameters.

Parameters	Values
Sensing area	$500 \times 500\ m^2$
Deployment	Random
Total nodes	200
Total data packets of each node	200
Consensus	PoA
Network interface	Wireless

data packets toward the destination. The reason is that when the node does not send the correct data packets to the destination, the node again requests for the data from the source nodes. Moreover, less energy is consumed after the detection of malicious nodes using GA-SVM in the proposed model. In GA-SVM, the legitimate nodes perform routing without tampering the data packets and changing routes; therefore, the energy consumed is less as shown in Figure 7.

4.2.2. Dead Nodes. Table 5 shows that the mistreatment attack highly affects the proposed model because it has a low network lifetime. In the mistreatment attack, the nodes consume high energy; therefore, all nodes are dead at 2100 rounds. Furthermore, in the MITM attack, all of the nodes are dead at 2300 rounds because the malicious nodes tampered with the data packets and forwarded the malicious data packets to CHs. Moreover, in the grayhole attack, the nodes do not consume high energy; therefore, their networks' lifetime is higher than both the other attacks. In the proposed model, GA-SVM is used to detect the malicious nodes and revoked them from the network. Using the proposed GA-SVM, the model is free from malicious nodes. The legitimate nodes consume less energy, and as a result, their networks' lifetime became high. Therefore, all of the legitimate nodes are dead at 4000 rounds as shown in Figure 8.

4.2.3. Grayhole Attack. In the grayhole attack, the value of PdR increases by 53% because the malicious nodes received the data packets and do not forward it to the destination. The GA-SVM model detects 21.75% malicious nodes that are present in the network and drops the data packets. Therefore, after the detection of malicious nodes, the value of PdR decreases as shown in Figure 9(a).

The value of PdR for the GA-SVM model is higher than the grayhole attack in the system. In the absence of the malicious nodes, the total number of data packets sent from the source is successfully received at the destination. The value of PdR is minimum in the presence of the malicious nodes because the malicious nodes do not forward the data packets toward the destination as shown in Figure 9(b). However,

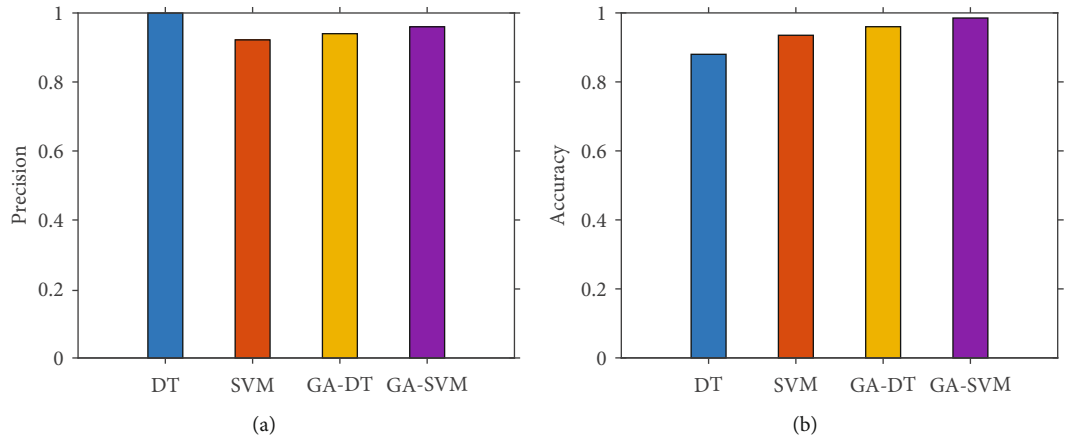


Figure 3: (a) Comparison of accuracy. (b) Comparison of precision.

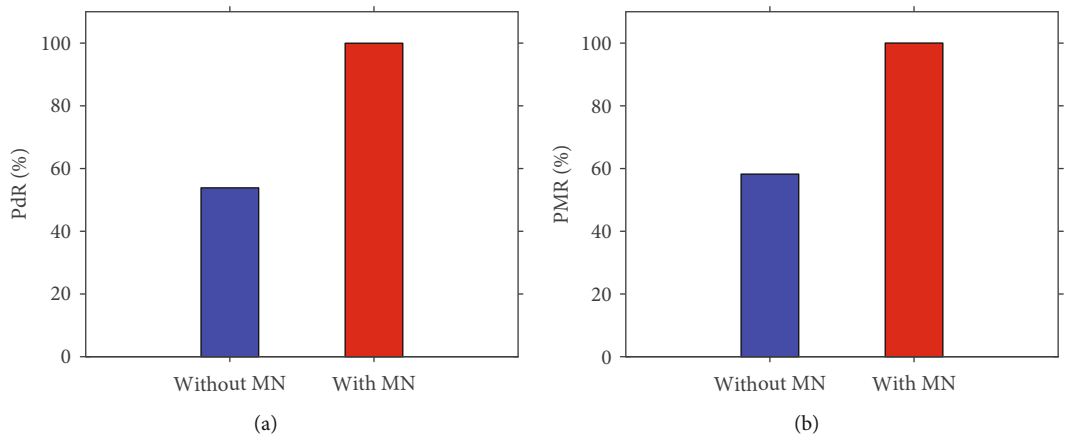


Figure 4: Comparison of (a) PdR between LNs and MNs and (b) PMR between LNs and MNs.

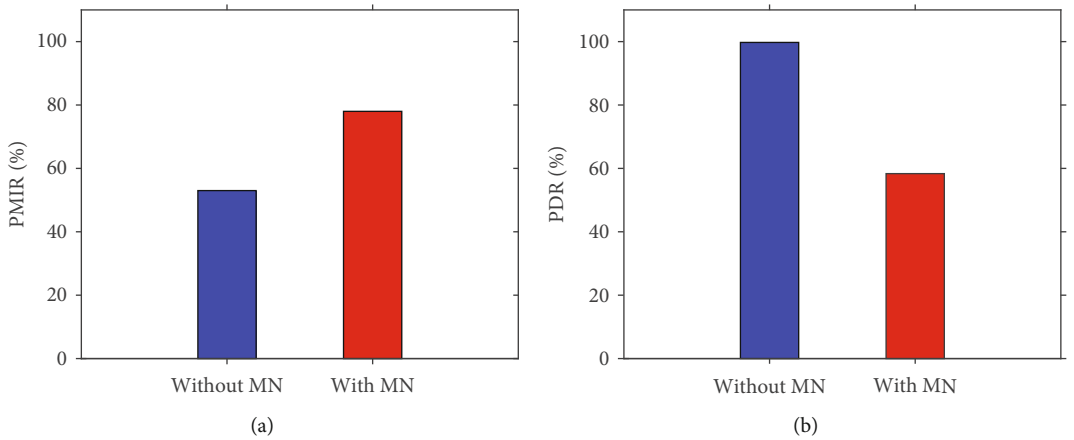


Figure 5: (a) Comparison of PMiR between LNs and MNs. (b) Comparison of PDR between LNs and MNs.

when the value of PdR is maximum, the performance of the IoST network is increased. Moreover, the GA-SVM model classifies the malicious and legitimate nodes in order to remove the malicious node from the network. After the malicious nodes are revoked from the network, the value of PdR increases up to 99.72%.

4.2.4. Mistreatment. In the misrouted attack, the malicious nodes received the data packets from source nodes; however, they are misrouted. In the presence of malicious nodes, the value of mistreatment is high as shown in Figure 10(a). In the network, if the data packets are misrouted, the network becomes congested. Otherwise, it will not be congested.

Table 4: Shortest path using the Dijkstra algorithm.

Paths	Distances (hop)
[1, 3, 6, 7]	9
[1, 3, 4, 7]	11
[1, 2, 4, 5, 7]	10
[1, 3, 5–7]	13
[1–3, 6, 7]	8
[1, 2, 4–7]	12

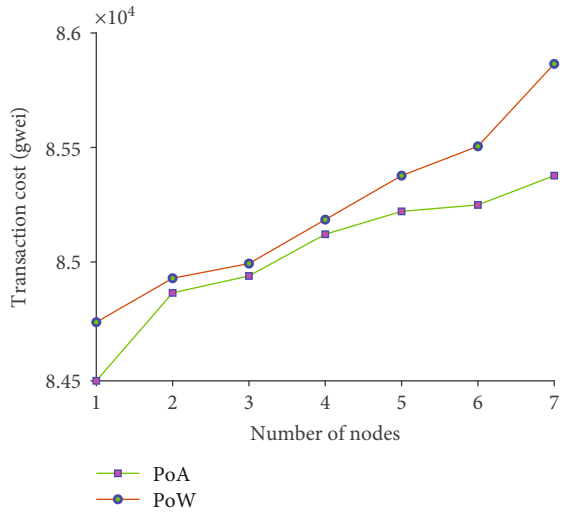


Figure 6: Transaction cost (TC) of PoA and PoW.

Table 5: Performance analysis of dead nodes in grayhole attack, mistreatment attack, MITM attack, and GA-SVM.

		First node dead	Last node dead
Attacks	Grayhole attack	400 rounds	3800 rounds
	Mistreatment attack	300 rounds	2100 rounds
	MITM attack	200 rounds	2300 rounds
Without attack	GA-SVM	700 rounds	4000 rounds

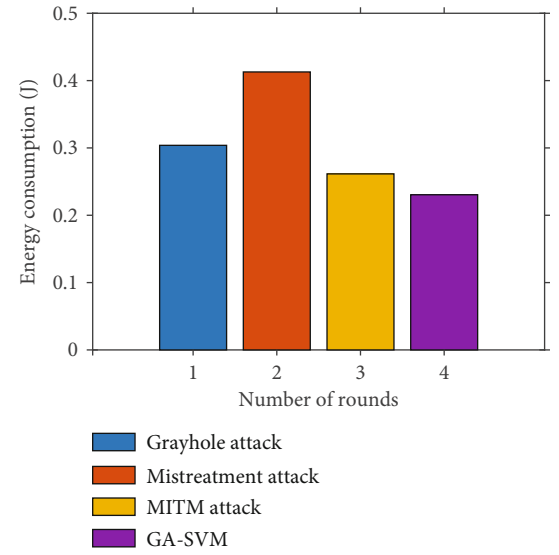


Figure 7: Performance analysis of energy consumption under grayhole, mistreatment, MITM attacks, and proposed GA-SVM solution.

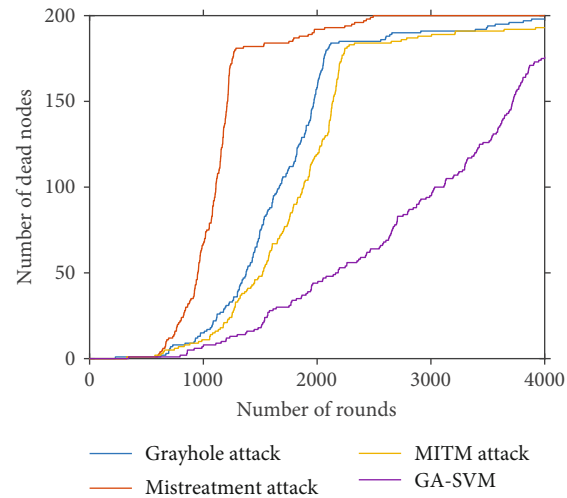


Figure 8: Performance analysis of dead nodes under grayhole, mistreatment, MITM attacks, and proposed solution GA-SVM.

4.2.5. MITM. In Figure 10(b), MITM attack is induced in the proposed model. Therefore, the value of PMR becomes high in the presence of malicious nodes, i.e., 89%. It is high because the malicious nodes tampered with the data packets and forwarded them toward the destination. After the detection of malicious nodes using GA-SVM, the value of PMR becomes low. The purple bar of GA-SVM shows the number of nodes that do not modify the data packets. The proposed system model is robust against the following two attacks.

(1) Spoofing Attack. In the spoofing attack, the malicious node takes the identity (ID) of the legitimate node and acts as a trustworthy node. This attack is not possible because all of the nodes are registered in the blockchain, which makes it difficult to compromise the system.

(2) Bad Mouthing Attack. In this attack, the malicious nodes change the reputation value of the forwarder nodes; as a result, the nodes become untrusted in the system. Therefore, this attack is not possible in the proposed model because the forwarder nodes are not selected based on their reputation values

4.3. Security Analysis of Smart Contract. The proposed smart contract is analyzed using an Oyente tool [4]. The tool is used to analyze the smart contract using symbolic

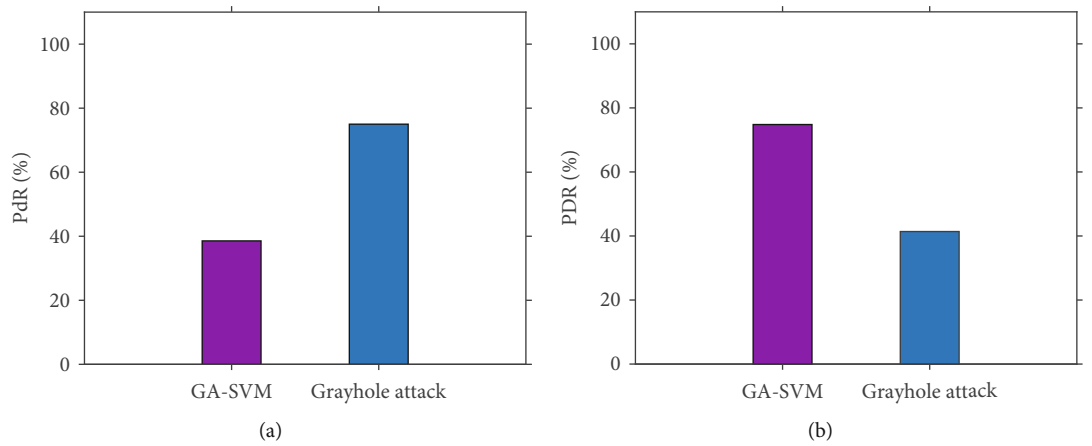


Figure 9: Comparison of (a) PdR between grayhole attack and GA-SVM and (b) PDR between grayhole attack and GA-SVM.

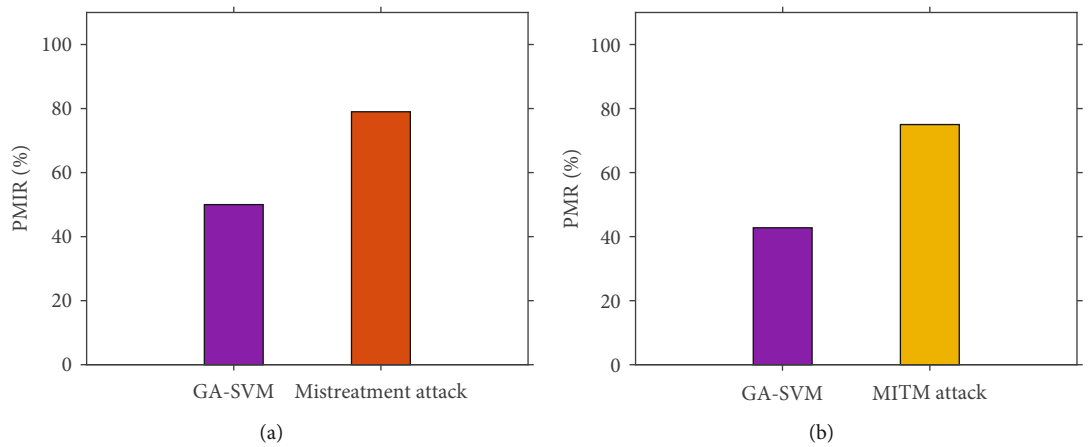


Figure 10: Comparison of (a) PMIR between mistreatment attack and GA-SVM and (b) PMR between MITM attack and GA-SVM.

```
root@68c239632c96: /oyente/oyente#                    Phython oyente.py -s LRAB-SC.sol
WARNING: root: You are using evm version 1.8.2. The supported version is 1.7.3
WARNING: root: You are using solc version 0.4.21, The latest supported version is
0.4.19
INFO:root:contract LRAB-SC.sol:LAB-SC:
INFO:symExec:        ============ Results ===========
 INFO:symExec:          EVM code coverage:                          8.0%
 INFO:symExec:          Integer Underflow:                          False
 INFO:symExec:          Integer Underflow:                          False
INFO:symExec:          Parity multisig Bug 2:                      False
INFO:symExec:          Callstack depth attack vulnerability:       False
INFO:symExec:          Transaction-ordering dependence (TOD):      False
INFO:symExec:          Timestamp Dependency:                       False
INFO:symExec:          Re-Entrancy vulnerability:                  False
INFO:symExec:        ====== Analysis completed ======
root@68c239632c96: /oyente/oyente#
```

Figure 11: Security analysis of smart contract using Oyente tool.

execution techniques. From Figure 11, it is shown that the outputs of all the analysis report are "false," which means that the smart contract is robust against the vulnerabilities. The following vulnerabilities and attacks are analyzed in the proposed model.

4.3.1. Integer Underflow and Overflow. The proposed smart contract is robust against this attack. This attack arises when the integer value is less than 1 bit or more than 256 bits. Therefore, the proposed smart contract gives an error and failed to deploy. Therefore, the proposed smart contract is robust against this attack.

4.3.2. Parity Multisig Bug 2. In this attack, the malicious node creates multiple accounts and generates fake signatures in the smart contract. When a malicious node performs

transactions using the fake accounts, then the smart contract will execute and deploy in the proposed system. Therefore, the proposed smart contract is robust against this attack as shown in Figure 11.

4.3.3. Call Stack Attack. In a smart contract, when a function is called by other functions, its depth is 1023 frames. In this attack, the malicious node exceeds the frame size to 1024 frames. Therefore, the function is failed to call, and the smart contract stopped working. However, the smart contract analysis shows that this attack is not possible in the proposed system model.

4.3.4. Timestamp Dependency. This attack is performed on miners. The miner who mines the block in minimum time is allowed to mine the next block. In the attack, the malicious node changes the mining time from the timestamp table. This attack is not possible on the proposed smart contract because PoA is used, and each miner has a copy of ledger. Therefore, if any malicious node changes the time, it can be traced and detected by other miners.

4.3.5. Reentrancy Vulnerability. In this attack, the malicious node calls the same function repeatedly and does not return any value. When a function is executed, then other functions cannot be executed. However, the proposed smart contract is robust against this attack as shown in Figure 11.

5. Conclusion

In this paper, ML techniques are exploited using blockchain for enhancing the security of the network. Firstly, registration is performed for the unauthenticated nodes, which can prove harmful for the network. Blockchain is used to register the nodes that are taking part in the routing process. Secondly, it stores the routing information that is generated during routing process. Moreover, a PoA consensus mechanism is used for validating the transactions because PoW requires more computational resources. Thirdly, ML techniques, GA-SVM and GA-DT, are exploited for MN detection. GA-SVM is used for the detection of MNs because of its higher accuracy as compared to GA-DT. MNs that are detected using GA-SVM are revoked from the network by deleting their registration from the blockchain, which allows the LNs to take part in the routing process. The model is trained using the following: PDR, PdR, PMR, and PMiR. Moreover, routing is performed by finding the shortest path using the Dijkstra algorithm. Once the route is calculated, source is able to securely and efficiently send the data packets to the destination. The proposed model is validated through extensive simulations. Moreover, security analysis using Oyente tool and three attacker models' induction is performed to evaluate the smart contract and the proposed system model, respectively. We compare the PoA and PoW consensus mechanisms using transaction cost in which PoA performs well and consumes up to 30% less cost. Furthermore, without MITM attack, GA-SVM consumes 10% less energy than with MITM attack. Moreover, without any attack, GA-SVM consumes 30% less energy than with grayhole attack and 60% less energy than mistreatment. The Dijkstra algorithm is compared with Bellman Ford in which the Dijkstra algorithm consumes less time to find the shortest path for routing. Dijkstra's and Bellman Ford's shortest paths are 8 and 11 hops, respectively. DT, SVM, GA-DT, and GA-SVM results are compared on the bases of accuracy and precision. The accuracy of DT, SVM, GA-DT, and GA-SVM is 88%, 93%, 96%, and 98%, respectively, while the precision of DT, SVM, GA-DT, and GA-SVM is 100%, 92%, 94%, and 96%, respectively.

Acronyms

CHs: Cluster heads
DT: Decision Tree
GA: Genetic Algorithm
GA-DT: Genetic Algorithm-Decision Tree
GA-SVM: Genetic Algorithm-Support Vector Machine
MITM: Man In The Middle
IoST: Internet of Sensor Things
PDR: Packet Delivery Ratio
PdR: Packet drop Ratio
PMiR: Packet Misroute Rate
PMR: Packet Modification Ratio
PoA: Proof of Authority
PoW: Proof of Work
RAs: Research answers
ROs: Research objectives
RQs: Research questions
SDN: Software defined network
SVM: Support Vector Machine
WSNs: Wireless sensor networks.

Data Availability

No supporting data exists separately.

Conflicts of Interest

The authors declare that there is no known conflict of interest.

References

[1] B. Jia, T. Zhou, W. Li, Z. Liu, and J. Zhang, "A blockchain-based location privacy protection incentive mechanism in crowd sensing networks," *Sensors*, vol. 18, no. 11, p. 3894, 2018.

[2] Y. Tian, Z. Wang, J. Xiong, and J. Ma, "A blockchain-based secure key management scheme with trustworthiness in DWSNs," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 9, pp. 6193–6202, 2020.

[3] A. Shankar, P. Pandiaraja, K. Sumathi, T. Stephan, and P. Sharma, "Privacy preserving E-voting cloud system based on ID based encryption," *Peer-to-Peer Networking and Applications*, vol. 14, no. 4, pp. 2399–2409, 2021.

[4] S. Abbas, N. Javaid, A. Almogren, S. M. Gulfam, A. Ahmed, and A. Radwan, "Securing genetic algorithm enabled SDN routing for blockchain based Internet of Things," *IEEE Access*, vol. 9, pp. 139739–139754, 2021.

[5] A. P. Atmaja, A. El Hakim, A. P. A. Wibowo, and L. A. Pratama, "Communication systems of smart agriculture based on wireless sensor networks in IoT," *Journal of Robotics and Control (JRC)*, vol. 2, no. 4, pp. 297–301, 2021.

[6] A. S. Yahaya, N. Javaid, A. Almogren, A. Ahmed, S. M. Gulfam, and A. Radwan, "A two-stage privacy preservation and secure peer-to-peer energy trading model using blockchain and cloud-based aggregator," *IEEE Access*, vol. 9, pp. 143121–143137, 2021.

[7] J. Yang, S. He, Y. Xu, L. Chen, and J. Ren, "A trusted routing scheme using blockchain and reinforcement learning for wireless sensor networks," *Sensors*, vol. 19, no. 4, p. 970, 2019.

[8] Z. Cui, X. U. E. Fei, S. Zhang et al., "A hybrid blockchain-based identity authentication scheme for multi-WSN," *IEEE Transactions on Services Computing*, vol. 13, no. 2, pp. 241–251, 2020.

[9] G. Ramezan and C. Leung, *Wireless Communications and Mobile Computing*, vol. 2018, Article ID 4029591, Wireless Communications and Mobile Computing, 2018.

[10] W. She, Q. Liu, Z. Tian, J. S. Chen, B. Wang, and W. Liu, "Blockchain trust model for malicious node detection in wireless sensor networks," *IEEE Access*, vol. 7, pp. 38947–38956, 2019.

[11] S. Hong, "P2P networking based Internet of Things (IoT) sensor node authentication by blockchain," *Peer-to-Peer Networking and Applications*, vol. 13, no. 2, pp. 579–589, 2020.

[12] K. Haseeb, N. Islam, A. Almogren, and I. U. Din, "Intrusion prevention framework for secure routing in WSN-based mobile Internet of Things," *IEEE Access*, vol. 7, pp. 185496–185505, 2019.

[13] Y. Liu, K. Wang, Y. Lin, and W. Xu, "LightChain: a lightweight blockchain system for industrial Internet of Things," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 6, pp. 3571–3581, 2019.

[14] M. H. Kumar, V. Mohanraj, Y. Suresh, J. Senthilkumar, and G. Nagalalli, "Trust aware localized routing and class based dynamic block chain encryption scheme for improved security in WSN," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 5, pp. 5287–5295, 2021.

[15] Y. Xu, J. Ren, G. Wang, C. Zhang, J. Yang, and Y. Zhang, "A blockchain-based nonrepudiation network computing service scheme for industrial IoT," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 6, pp. 3632–3641, 2019.

[16] P. K. Sharma and J. H. Park, "Blockchain based hybrid network architecture for the smart city," *Future Generation Computer System s*, vol. 86, pp. 650–655, 2018.

[17] A. U. Khan, A. Shahid, F. Tariq et al., "Enhanced decentralized management of patient-driven interoperability based on blockchain," in *Lecture Notes in Networks and Systems*, pp. 815–827, Springer, Cham, 2019.

[18] M. U. Javed, N. Javaid, M. W. Malik et al., "Blockchain based secure, efficient and coordinated energy trading and data sharing between electric vehicles," in *Cluster Computing*, pp. 1–29, Springer, 2021.

[19] R. Khalid, M. W. Malik, T. A. Alghamdi, and N. Javaid, "A consortium blockchain based energy trading scheme for electric vehicles in smart cities," *Journal of Information Security and Applications*, vol. 63, article 102998, 2021.

[20] C. Wang, D. Wang, Y. Tu, G. Xu, and H. Wang, "Understanding node capture attacks in user authentication schemes for wireless sensor networks," *IEEE Transactions on Dependable and Secure Computing.*, p. 1, 2020.

[21] G. Rathee, M. Balasaraswathi, K. P. Chandran, S. D. Gupta, and C. S. Boopathi, "A secure IoT sensors communication in industry 4.0 using blockchain technology," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 1, pp. 1–13, 2020.

[22] R. Fotohi and F. S. Aliee, "Securing communication between things using blockchain technology based on authentication and SHA-256 to improving scalability in large-scale IoT," *Computer Networks*, vol. 197, article 108331, 2021.

[23] R. Khalid, N. Javaid, A. Almogren, M. U. Javed, S. Javaid, and M. Zuair, "A blockchain-based load balancing in decentralized hybrid P2P energy trading market in smart grid," *IEEE Access*, vol. 8, pp. 47047–47062, 2020.

[24] M. Zhaofeng, M. Jialin, W. Jihui, and S. Zhiguang, "Blockchain-based decentralized authentication modeling scheme in edge and IoT environment," *IEEE Internet of Things Journal*, vol. 8, no. 4, pp. 2116–2123, 2021.

[25] C. M. Ferreira, C. T. Garrocho, R. A. Oliveira, J. S. Silva, and C. F. Cavalcanti, "IoT registration and authentication in smart city applications with blockchain," *Sensors*, vol. 21, no. 4, p. 1323, 2021.

[26] M. A. Uddin, A. Stranieri, I. Gondal, and V. Balasurbramanian, "A lightweight blockchain based framework for underwater iot," *Electronics*, vol. 8, no. 12, p. 1552, 2019.

[27] M. Mahdavisharif, S. Jamali, and R. Fotohi, "Big data-aware intrusion detection system in communication networks: a deep learning approach," *Journal of Grid Computing*, vol. 19, no. 4, pp. 1–28, 2021.

[28] H. Feng, W. Wang, B. Chen, and X. Zhang, "Evaluation on frozen shellfish quality by blockchain based multi-sensors monitoring and SVM algorithm during cold storage," *IEEE Access*, vol. 8, pp. 54361–54370, 2020.

[29] S. Rathore, B. W. Kwon, and J. H. Park, "BlockSecIoTNet: blockchain-based decentralized security architecture for IoT network," *Journal of Network and Computer Applications*, vol. 143, pp. 167–177, 2019.

[30] A. Mubarakali, "An efficient authentication scheme using blockchain technology for wireless sensor networks," in *Wireless Personal Communications*, pp. 1–15, Springer, 2021.

[31] N. Tariq, M. Asim, F. A. Khan, T. Baker, U. Khalid, and A. Derhab, "A blockchain-based multi-mobile code-driven trust mechanism for detecting internal attacks in Internet of Things," *Sensors*, vol. 21, no. 1, p. 23, 2021.

[32] H. Honar Pajooh, M. Rashid, F. Alam, and S. Demidenko, "Multi-layer blockchain-based security architecture for Internet of Things," *Sensors*, vol. 21, no. 3, p. 772, 2021.

[33] X. Wu and J. Liang, "A blockchain-based trust management method for Internet of Things," *Pervasive and Mobile Computing*, vol. 72, article 101330, 2021.

[34] J. Wang, Y. Liu, S. Niu, H. Song, W. Jing, and J. Yuan, "Blockchain enabled verification for cellular-connected unmanned aircraft system networking," *Future Generation Computer Systems*, vol. 123, pp. 233–244, 2021.

[35] T. H. Kim, R. Goyat, M. K. Rai et al., "A novel trust evaluation process for secure localization using a decentralized blockchain in wireless sensor networks," *IEEE Access*, vol. 7, pp. 184133–184144, 2019.

[36] R. Goyat, G. Kumar, M. K. Rai, R. Saha, R. Thomas, and T. H. Kim, "Blockchain powered secure range-free localization in

wireless sensor networks," *Arabian Journal for Science and Engineering*, vol. 45, no. 8, pp. 6139–6155, 2020.

[37] J. Wang, Y. Liu, S. Niu, and H. Song, "Lightweight blockchain assisted secure routing of swarm UAS networking," *Computer Communications*, vol. 165, pp. 131–140, 2021.

[38] A. Rahman, M. J. Islam, A. Montieri et al., "SmartBlock-SDN: an optimized blockchain-SDN framework for resource management in IoT," *IEEE Access*, vol. 9, pp. 28361–28376, 2021.

[39] W. Meng, W. Li, and J. Zhou, "Enhancing the security of blockchain-based software defined networking through trust-based traffic fusion and filtration," *Information Fusion*, vol. 70, pp. 60–71, 2021.

[40] C. M. Chen, X. Deng, W. Gan, J. Chen, and S. H. Islam, "A secure blockchain-based group key agreement protocol for IoT," *The Journal of Supercomputing*, vol. 77, no. 8, pp. 9046–9068, 2021.

[41] K. Sergii and F. Prieto-Castrillo, "A rolling blockchain for a dynamic WSNs in a smart city," 2018, http://arxiv.org/abs/1806.11399.

[42] M. Liu, F. R. Yu, Y. Teng, V. C. Leung, and M. Song, "Computation offloading and content caching in wireless blockchain networks with mobile edge computing," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 11, pp. 11008–11021, 2018.

[43] P. Danzi, A. E. Kalor, C. Stefanovic, and P. Popovski, "Delay and communication tradeoffs for blockchain systems with lightweight IoT clients," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2354–2365, 2019.

[44] G. Kolumban-Antal, V. Lasak, R. Bogdan, and B. Groza, "A secure and portable multi-sensor module for distributed air pollution monitoring," *Sensors*, vol. 20, no. 2, p. 403, 2020.

[45] M. Naz, F. A. Al-zahrani, R. Khalid et al., "A secure data sharing platform using blockchain and interplanetary file system," *Sustainability*, vol. 11, no. 24, p. 7054, 2019.

[46] M. Radhika and P. Sivakumar, "Energy optimized micro genetic algorithm based LEACH protocol for WSN," *Wireless Networks*, vol. 27, no. 1, pp. 27–40, 2021.

[47] L. Zhang, F. Li, P. Wang, R. Su, and Z. Chi, "A blockchain-assisted massive IoT data collection intelligent framework," *IEEE Internet of Things Journal.*, 2021.

[48] B. M. Yakubu, M. I. Khan, N. Javaid, and A. Khan, "Blockchain-based secure multi-resource trading model for smart marketplace," *Computing*, vol. 103, no. 3, pp. 379–400, 2021.

[49] P. Kautoo, P. K. Shukla, and S. Silakari, "Trust formulization in dynamic source routing protocol using SVM," *International Journal of Information Technology and Computer Science (IJITCS)*, vol. 6, no. 8, pp. 43–50, 2014.

[50] R. D. R. Popli, "A worm hole attack detection in mobile ad-hoc network using GA and SVM," *International Journal of Engineering Applied Sciences and Technology*, vol. 5, no. 3, pp. 582–588, 2020.

[51] A. Alsarhan, M. Alauthman, E. A. Alshdaifat, A. R. al-Ghuwairi, and A. al-Dubai, "Machine learning-driven optimization for SVM-based intrusion detection system in vehicular ad hoc networks," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–10, 2021.

[52] S. Suthaharan, "Support Vector Machine," in *Machine Learning Models and Algorithms for Big Data Classification*, pp. 207–235, Springer, Boston, MA, 2016.

[53] D. Rachmawati and L. Gustin, "Analysis of Dijkstra's algorithm and A∗ algorithm in shortest path problem," in *In Journal of Physics: Conference Series*, vol. 1566, no. 1p. 012061, IOP Publishing, 2020.

[54] M. Bakshi and A. Srivastava, "Magnify lifeless nodes in WSN using shortest path ALGO for reducing energy diversion," *International Journal of Modern Communication Technologies and Research*, vol. 6, no. 6, 2018.

[55] F. Dad, N. Amin, S. T. Shah, F. Badshah, Z. U. Rahman, and ur Rahman, I, "Optimal path selection using Dijkstra's algorithm in cluster-based LEACH protocol," *Journal of Applied Environmental and Biological Sciences*, vol. 7, no. 2, pp. 194–198, 2018.

[56] M. Tripathi, M. S. Gaur, and V. Laxmi, "Comparing the impact of black hole and gray hole attack on LEACH in WSN," *Procedia Computer Science*, vol. 19, pp. 1101–1107, 2013.

[57] A. Chakrabarti and G. Manimaran, "Internet infrastructure security: a taxonomy," *IEEE Network*, vol. 16, no. 6, pp. 13–21, 2002.

[58] K. H. Yeung and W. K. Fung, "Attacking routers by packet misrouting," *WSEAS Transactions on Communications*, vol. 3, no. 2, pp. 493–498, 2004.

[59] F. Callegati, W. Cerroni, and M. Ramilli, "Man-in-the-Middle attack to the HTTPS protocol," *IEEE Security & Privacy*, vol. 7, no. 1, pp. 78–81, 2009.

Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 9966395, 13 pages
https://doi.org/10.1155/2021/9966395

Research Article

Power Prediction of Combined Cycle Power Plant (CCPP) Using Machine Learning Algorithm-Based Paradigm

Raheel Siddiqui,[1] Hafeez Anwar,[1] Farman Ullah,[1] Rehmat Ullah,[2] Muhammad Abdul Rehman,[1] Naveed Jan,[3] and Fawad Zaman[4]

[1]*Department of Electrical & Computer Engineering, COMSATS University Islamabad-Attock Campus, Pakistan*
[2]*Department of Computer Systems Engineering, University of Engineering and Technology Peshawar, Peshawar, Pakistan*
[3]*Department of Information Engineering Technology, University of Technology Nowshera, KPK, Pakistan*
[4]*Department of Electrical & Computer Engineering, COMSATS University Islamabad, Pakistan*

Correspondence should be addressed to Hafeez Anwar; hafeez.anwar@cuiatk.edu.pk
and Farman Ullah; farmankttk@ciit-attock.edu.pk

Received 29 March 2021; Revised 17 July 2021; Accepted 3 November 2021; Published 23 December 2021

Academic Editor: Javier Prieto

Power prediction is important not only for the smooth and economic operation of a combined cycle power plant (CCPP) but also to avoid technical issues such as power outages. In this work, we propose to utilize machine learning algorithms to predict the hourly-based electrical power generated by a CCPP. For this, the generated power is considered a function of four fundamental parameters which are relative humidity, atmospheric pressure, ambient temperature, and exhaust vacuum. The measurements of these parameters and their yielded output power are used to train and test the machine learning models. The dataset for the proposed research is gathered over a period of six years and taken from a standard and publicly available machine learning repository. The utilized machine algorithms are K-nearest neighbors (KNN), gradient-boosted regression tree (GBRT), linear regression (LR), artificial neural network (ANN), and deep neural network (DNN). We report state-of-the-art performance where GBRT outperforms not only the utilized algorithms but also all the previous methods on the given CCPP dataset. It achieves the minimum values of root mean square error (RMSE) of 2.58 and absolute error (AE) of 1.85.

1. Introduction

The accurate prediction of power generated by a plant helps in reducing various related issues such as power outages, economic, and technical difficulties [1, 2]. In particular, an inaccurate prediction results in the rise of per unit cost of electric power [3] due to the high fuel consumption. Hence, in this paper, we aim at achieving a precise prediction of electric power of a base load CCPP on full load conditions thus ensuring decreased cost of per unit of electric power [4].

The power of thermodynamic power stations can be calculated using complex mathematical models [5]. These models involve a vast range of assumptions and parameters to reflect the actual uncertainty of the system. However, these mathematical models are time consuming and are based on a deterministic approach [5]. On the other hand, supervised machine learning (ML) algorithms incorporate probabilistic approaches for power prediction instead of mathematical modeling [6]. With the availability of data, the prediction done with the ML approach is far more convenient, scalable, and flexible thus preventing to model the whole the system. It can also be observed in other similar approaches, for instance, predicting groundwater hardness vulnerability [7], estimating soil erosion susceptibility [8], groundwater level prediction [9], and groundwater potential prediction [10]. Our proposed ML algorithms assess historical data of a power plant, operating under a variety of environmental conditions, in order to provide optimized power forecasts in less time [11]. However, being probabilistic in nature, the predictions made by ML algorithms involve errors. Due to this reason, we propose to evaluate several algorithms for the task of

power prediction of a CCPP. Furthermore, we also search for the parameters of these algorithms where they give the least error on the current dataset.

The generated electric power of a CCPP unexpectedly fluctuates throughout the whole year due to several parameters, such as ambient temperature, atmospheric pressure, humidity, and exhaust vacuum [12]. Consequently, these parameters directly and indirectly influence the output power [13] of a CCPP. Therefore, power production can be improved and fuel consumption can be reduced by optimally controlling these parameters [14]. The primary focus of this research is to analyze the influence of ambient parameters on output power prediction rather than controlling the parameters. For this purpose, these environmental parameters are used to predict the electric power through various machine learning algorithms [4].

Nonetheless, various probabilistic approaches are previously used for CCPP power prediction including bagging and regression ANN [4, 15]. However, their prediction error is slightly high [16], due to which, this paper proposes the machine learning algorithms for the prediction of electric power of CCPP operated on full load using the previously mentioned four parameters. Gradient-boosted regression tree (GBRT), linear regression (LR), artificial neural network (ANN), and K-nearest neighbor (KNN) are used to improve the power prediction. The individual and cumulative effects of each parameter are evaluated on output power prediction using these four machine learning algorithms. These algorithms are compared with the previous research work to find improved results. The best performance among these four machine learning models is analyzed by choosing the least RMSE and AE values.

The rest of the paper is organized as follow: Section 2 outlines the related work. Section 3 introduces the proposed methodology, and the results and discussion are briefly discussed in Section 4. Finally, we conclude the paper in Section 5.

2. Related Work

Pourbeik et al. [17] proposed the basic working of a combined cycle power plant (CCPP). The main parts of the power plant are gas turbine, steam turbine, heat recovery steam generator (HRGS), and electric generator. In the gas turbine, compressed high-pressure air combined with fuel inside the combustion chamber, which makes hot pressurized air to strike the turbine blades and shaft coupled with electric generator for power generation. Due to the usage of gas turbine, the temperature of residual exhaust air is still high, and thus, it is used as an input of the heat recovery steam generator (HRSG). After that, hot air converts water into steam which rotates the steam turbine and power is generated by using both gas and steam turbines. The overall efficiency of CCPP is about 55 percent. Tfekci [4] presented to forecast the hourly based electrical power of a combined cycle power plant (CCPP) operated on base load for full load conditions. The benefit of operating in full load is to escalate the turnover of available hours. Several machine learning regression algorithms used for increasing the performance of predicting models and compared with their output results in the form of RMSE to find the best results. Firstly, it fetched the best subset that contained all input features. Secondly, it identified the best regression algorithm among the other fifteen algorithms. As a result, the bagging algorithm with the REPtree [4] algorithm to observed the best method applied to the best subset with a minimum RMSE of 3.787 and absolute error (AE) of 2.818. Yeom and Kwak [18] presented the Takai-Sugeno-Kang- (TSK-) based extreme learning machine (ELM) for power estimation. This algorithm is designed by an efficient approach to generate automatic fuzzy if then rules. It has mainly two steps. Firstly, the initial matrix of random partitions generated, and cluster centers are calculated for random clustering. These centers are used to decide the nearer part of fuzzy rules. Secondly, least square estimate (LSE) used to estimating the linear parameters of the TSK fuzzy type. The results depict that the TSK ELM method showed less RMSE of 3.93 as compared to ordinary ELM as shown in Table 1.

Lorencin et al. [16] proposed a genetic algorithm (GA) approach to a multilayer perceptron (MLP) design in order to predict of the CCPP electrical power output. A heuristic algorithm to increase the regression performance of MLP compared to those available in the literature. The GA was applied by using crossover procedures and processes based on mutation. These methods are implemented in 50 different generations for the design of 20 different chromosomes. Using Bland-Altman (BA) analysis, MLP configurations, which are devised with GA implementation, are validated. Five hidden layers of MLP, 25, 80, 65, 75, and 80 nodes using GA, respectively, are built. K-fold cross-validation is performed to assess average performance of the abovementioned MLP. The RMSE value obtained with the abovementioned MLP is 4.305, which is considerably less than the MLP provided in the existing literature, but still greater than numerous complicated algorithms, such as K star and tree-based algorithms. Bandić et al. [22] described the random forest algorithm for estimating the output power of a CCPP at full load. The analysis is conducted between twofold where in the first fold all the features are utilized, whereas in the second fold only three features are used. Random forest, random tree, and adaptive neurofussy inference system (ANFIS) algorithms are used for power prediction. Performance evaluated by taking RMSE, absolute error (AE), and correlation of predicted model. After applying all algorithms, the best result is obtained by the random forest. Results acquired on all features showed RMSE of 3.0271 MW, where 90 percent of the data used for training while the remaining 10 percent used for testing. Elfaki and Ahmed [12] presented the regression artificial neural network (ANN) to estimate the electric power of the CCPP. A total of seven experiments are performed on the ANN model. Each time the performance of the ANN model is different due to random initial weights and bias. It perceived that the performance improved by increasing the number of data records of available features, and performance oscillates when new features are added into the dataset. Consequently, higher number of training records led to increased performance. After that, correlations among features and then between features and output parameters are calculated. Two training functions are applied to the same datasets and performance is observed. In

Table 1: Achieved results in the literature on the current dataset.

Sr. no	Algorithm	RMSE	AE
1	KNN [19](2012)	N/A	3.51
2	Bagging with REPTree [4] (2014)	4.23	3.22
3	C-CRF [20] (2016)	3.97	2.97
4	TSK-ELM [18] (2018)	3.93	N/A
5	Regression ANN [12] (2018)	4.32	N/A
6	MLP with GA [16] (2019)	4.30	N/A
7	TOB Matching [21] (2020)	2.89	N/A

this study, Bayesian regularization showed better performance than the Levenberg-Marquardt method. Moreover, the output result is compared with original result, and it is almost the same with a minimal value of standard deviation. The error between the original and output results also the least. Şen et al. [13] described the vital role of input features to estimate the output power of CCPP. There are four input features used in CCPP, namely, ambient temperature, pressure, humidity, and vacuum. One main feature among all others is the ambient temperature, which causes significant variation in the performance of CCPP. Depending on the environmental conditions, the temperature also deviates between 8°C to 23°C. The efficiency of CCPP observed at 8°C is 42.7% and generated power is 227.7 MW. On the other hand, at 23°C temperature, the observed efficiency of 43.3% and generated power of 197.3 MW are recorded. This happens when the inlet air temperature of the gas turbine increases, which in turn decreases the amount of oxygen in the air per unit volume. Due to less amount of oxygen, the burning rate becomes lower in the combustion chamber. It has a negative influence on the output power of the gas turbine. Thus, combustion will be higher when the amount of oxygen is high. Therefore, for maximum power generation, a proper cooling system should be installed on the inlet side, which could reduce the temperature of air. Rashid et al. [23] proposed a novel approach of swarm optimization-based feed-forward neural network that used to design the predicting model. All input variables of the plant are used as an input of the feed-forward neural network. Particle swarm optimization (PSO) is a learning algorithm. Performance is analyzed by using mean square error (MSE). PSO trained feed-forward neural network which depicted favorable results for power estimation. MSE of the training dataset is observed to be 1.019*e*-04; on the other hand, for the testing dataset, MSE is 0.0055. Burkov [24] presented an accurate and reliable way of estimating the hourly electrical power of a combined cycle power plant. For designing, the local and global predictive models, many algorithms are used in this paper, such as additive regression, *K*-means clustering, feed-forward ANN, KNN, and conventional multivariate regression. The model's performance is analyzed by using a mean relative error (MRE%) and mean absolute error (MAE). Among all the mentioned algorithms, KNN results are found more efficient and reliable, with a relative error of less than 1. Han [25] proposed the computational intelligence algorithm like tree architecture of fuzzy neural networks that are used for power prediction. It has a benefit of selecting the minimum rules by opting neurons as nodes and significant inputs as leaves. There are two primary optimization method genetic algorithm (GA) and random signal-based learning (RSA) for prediction. GA optimized the binary structure of the networks by opting for the leaves and nodes as binary. RSA further refines the binary connection in the interval. 70% of data is used for training and 30% used for testing. Performance is accessed by using RMSE. The result of GA for testing data depicts RMSE of 3.31. Most recently, the so-called transparent open-box machine learning algorithm is used by [21] to achieve an RMSE of 2.89% on the dataset. However, some of the records from the UCI CCPP dataset are removed to achieve this reduced RMSE.

3. Proposed Methodology for Power Prediction of Combined Cycle Power Plants Using Machine Learning (ML) Algorithms

This section explains the methodology of power prediction using machine learning algorithms. The main steps in this research are feature extraction of the collected data from sensors, performance evaluation of the ML algorithms, and performance parameter calculation. The prime goal is to find the algorithm that predicts the power of a CCPP with least error. For this purpose, four ML algorithms are evaluated which are gradient-boosted regression tree (GBRT), *K* -nearest neighbor (KNN), linear regression (LR), and artificial neural networks (ANN). The prediction is done by taking four parameters related to the CCPP, which are ambient inlet air temperature, atmospheric pressure, relative humidity, and vacuum (gas turbine exhaust pressure) are used. Results are predicted using Rapid Miner [26] which is a machine learning and data mining software suite. Figure 1 depicts the proposed architecture for the prediction of CCPP power using ML algorithms.

3.1. Dataset. Table 2 depicts a part of the whole dataset where the complete dataset comprises of 9568 records gathered from an operational CCPP over a period of 6 years. This dataset is taken from the UCI machine learning repository [4, 19].

3.2. Feature Extraction. The features acquired from the sensors data are measurements per unit time, mean, variance, standard deviation cross-correlation, autocorrelation, maximum value, and minimum value.

3.2.1. Measurements per Unit Time. The hourly based data (averaged) is obtained from various sensors installed outside the plant which record ambient variables every second. The values of these variables are used without being normalized.

3.2.2. Mean Value. There are four features and the mean value of each feature is given in the dataset of CCPP. Total number of samples is 9568.

$$\mu = \left(\frac{1}{N}\right) \sum_{i=0}^{N} x_{i,} \tag{1}$$

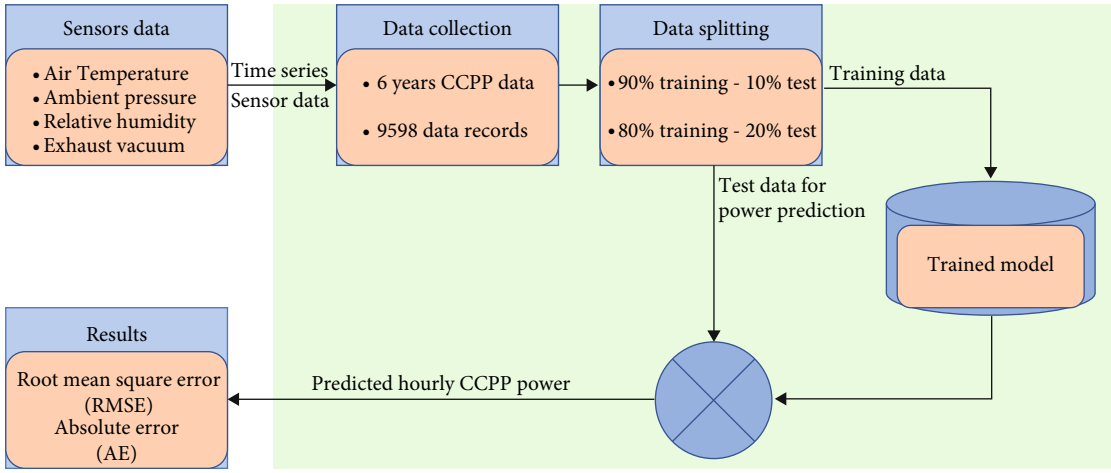


Figure 1: Proposed architecture of the prediction of CCPP power using ML algorithm.

Table 2: Input and output attributes of CCPP, where the temperature is measured in celcius, vaccum is measure in cm-Hg, pressure is measured in millibars, humidity is in percentage, and the unit of predicted power is megawatts.

Temperature	Vacuum	Pressure	Humidity	Predicted power
14.96	41.76	1024.07	73.17	463.26
25.18	62.96	1020.04	59.08	444.37
5.11	39.4	1012.16	92.14	488.56
20.86	57.32	1010.24	76.64	446.48
10.82	37.5	1009.23	96.62	473.9
26.27	59.44	1012.23	58.77	443.67
15.89	43.96	1014.02	75.24	467.35
9.48	44.71	1019.12	66.43	478.42
14.64	45	1021.78	41.25	475.98
11.74	43.56	1015.14	70.72	477.5

where N represents the total number of samples of one feature and x is the sensor output at i^{th} sample [27].

3.2.3. Variance. It is used to measure the spread of data points among themselves and from the mean. Consequently, it is calculated as the average of the squared distances from each point to the mean.

3.2.4. Standard Deviation. It finds the spread in the sensors data around the mean value.

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=0}^{n}(x_i - \mu)^2}, \tag{2}$$

where N gives the total number of records, x represents the current sensor output, and μ shows the mean value of a particular feature [27]. Table 3 shows the extracted feature details of the dataset.

3.2.5. Correlation. Correlation is a relationship of one feature with another feature. Table 4 shows that increase or decrease in the values of one feature tends to be paired with relative increment or decrement in the values of another.

$$\text{Cor}(x, y) = \left(\frac{(\text{Cov}(x, y))}{\sqrt{\text{Var}(x)\text{Var}(y)}}\right). \tag{3}$$

3.3. Algorithms. In this paper, four ML algorithms are evaluated for power prediction, which are gradient-boosted regression tree, K-nearest neighbor, artificial neural network, and linear regression. A brief explanation of each algorithm is given in the following.

3.3.1. Linear Regression (LR). Linear regression [28, 29] is used to predict the dependent variable (CCPP's power in this case) (y) based on the four independent variable (x_1, x_2, x_3, and x_4). Figure 2 shows the flow chart of the LR algorithm while its mathematical model [24] is given below.

$\theta_0, \theta_1 \cdots \theta_4$ are the five initial weights where they are assigned values between 0.5 and 2.5:

$$h_\theta(x) = \sum_{i=0}^{m}(\theta_i x_i), \tag{4}$$

where $h_\Theta(x)$ is the predicted value.

$$J(\theta_0, \theta_1) = \frac{1}{2m}\sum_{i=0}^{m}\left(h_\theta\left(x^i\right) - y^i\right)^2, \tag{5}$$

where $J(\theta)$ is the cost function. It should be minimized in order to achieve maximum prediction accuracy. y^i is the actual value.

$$\text{temp}\left(\theta_j\right) = \theta_j - \alpha\frac{\partial}{\partial\theta_j}J(\theta_0, \theta_1), \tag{6}$$

Table 3: Feature characteristics of CCPP dataset with 9568 data records.

Features	Units	Type	Min	Max	Mean	Variance	Std
Temperature	Celcius	Input	1.81	37.11	19.65	55.54	7.45
Pressure	Millibars	Input	992.89	1033.30	1013.40	35.27	5.93
Humidity	Percent	Input	25.56	100	73.30	213.17	14.6
Vacuum	cm-Hg	Input	25.36	81.56	54.30	161.49	12.70
Power	MW	Output	420.26	495.76	454.36	291.28	17.06

Table 4: Correlation matrix among input and output features.

	Temperature	Vacuum	Pressure	Humidity	Output power
Temperature	1	0.84	-0.50	-0.54	-0.98
Vacuum	0.84	1	-0.41	-0.31	-0.87
Pressure	-0.50	-0.41	1	-0.10	0.51
Humidity	-0.54	-0.31	-0.10	1	0.39
Output power	-0.98	-0.87	0.51	0.39	1

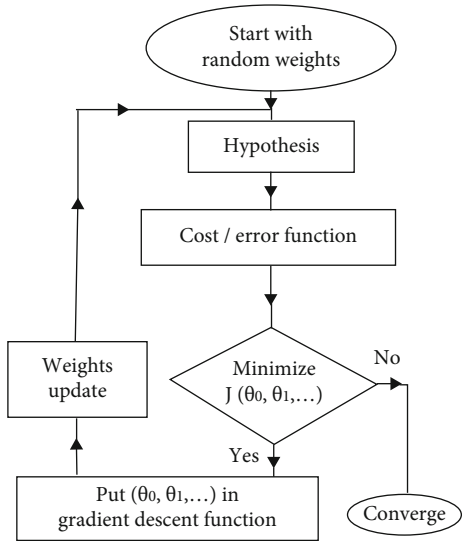


Figure 2: Flow chart of linear regression.

$$\text{temp}(\theta_j) = \theta_j - \frac{\alpha}{2m}\sum_{i=0}^{m}\left(h_\theta\left(x^i\right) - y^i\right)^2 x_j^i. \tag{7}$$

Equation (6) shows the gradient descent function where m gives the total number of sample in the dataset, α denotes the learning rate whose value ranges between 0 and 0.5. For every value of J repeat, the abovementioned equation until convergence is obtained [30].

3.3.2. Gradient-Boosted Regression Tree. In gradient-boosted regression [31–33] $(x_1, x_2, x_3, \text{and}\, x_4)$ are used as input parameters of CCPP and (y) as the output power. Figure 3 represents the tree structure of gradient-boosted regression. Learning rate α value is set as 0.1, and numbers of trees are varied for power prediction. In each tree, the depth is 20. For power prediction of CCPP, the "quantile function" is used as a distribution function. α is learning rate used to scale the tree and gradually improve the tree performance; its value is between 0 and 1. This whole process is continued until *ri* approaches to a minimum or stable value [30].

3.3.3. Artificial Neural Network. The ANN [34–36] is found to be a very useful Algorithm in machine learning [36]. ANN is an information management model that is similar to the function of biological nerves of the human brain. In Figure 4, there are 4 input parameters and one output which is the produced electric power by the CCPP. In this scenario, the ANN has two hidden layers where each layer comprises of 100 neurons. The learning rate α is 0.01. The Momentum value is set as 0.9 where in ANN, the momentum simply adds a fraction of the previous weight update to the current one. This prevents local maxima and provides smoothly optimized results. Following are the steps required to train the ANN.

(1) Initialize the weights arbitrarily:

$$w = \frac{1}{\text{total hidden layers}} \tag{8}$$

(2) Calculate the sum of inputs and weights products:

$$V = wx_i + b \tag{9}$$

(3) Analyze the activation (Sigmoid) function response:

$$h_\theta = \frac{1}{1 + e^{-V}} \tag{10}$$

(4) Take the input from the training data of (input, actual output) and enter it to the neural network.

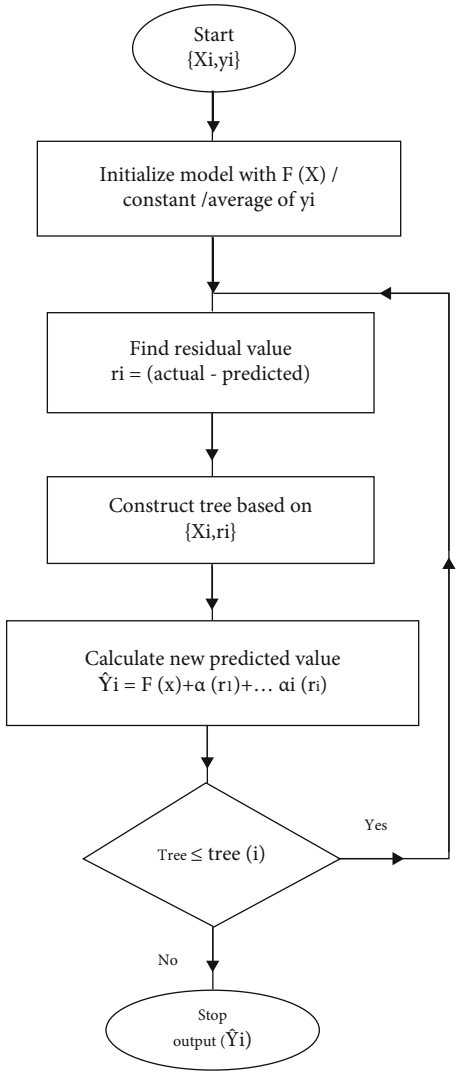


Figure 3: Flow chart of GBRT.

After that, calculate the error:

$$e_i = d_i - y_i \tag{11}$$

(5) Calculate the weight updates according to the following delta rule:

$$\Delta w_i j = \alpha e_i x_i \tag{12}$$

(6) Adjust the weights as

$$w_i j \longleftarrow w_i j(\text{old}) + \Delta w_i j \tag{13}$$

(7) Repeat steps 2–5 until the error is in an acceptable range [34]

3.3.4. K-Nearest Neighbor. KNN [36–39] stores all available scenarios and predict the numerical value based on a similarity measure such as the Euclidean distance [39] which is calculated between two points p and q as

$$\text{Dist} = \sqrt{\sum_{i}^{n}(q_i - p)2}. \tag{14}$$

For KNN, in the first step, find the K neighbor values which are near to the new feature whose value is anonymous. This nearness is found in terms of Euclidean distance between new data point and every data point present in the training set. For optimal results, K must be selected as an odd number [39]. Figure 5 depicts an example of 3-nearest neighbors where a new test sample will be labeled according to the label of the 3 nearest samples having shortest distance from the new sample [40].

3.4. Performance Metrics. For performance evaluation of the models, root mean square error (RMSE) and absolute error (AE) are used.

$$\begin{aligned} \text{RMSE} &= \sqrt{\left(\frac{\sum_1^N \left(Y - \bar{Y}\right)^2}{N}\right)}, \\ AE &= \frac{1}{N}\sum_1^N \left|Y - \bar{Y}\right|, \end{aligned} \tag{15}$$

where Y and $\bar{Y}$ are the actual and predicted output values of the CCPP plant, N is the total number of records [41].

4. Result Analysis

This section outlines the detailed results by discussing the influence of various parameters of the utilized algorithms on the predicted power. The experiments are performed in a systematic manner as outlined in the following:

(1) As a first step, we show the effect of feature selection on the predicted power achieved by each algorithm. For this, the default parameters of each algorithm are used as set in the Rapid Miner software [26]. This is helpful to find the features that have a significant effect of the generated power. In addition to that, a reduced features set will also make the rest of the analysis convenient and efficient by reducing the time taken in training a machine learning model

(2) The effect of the most crucial parameters of the respective algorithms on the predicted power is presented in terms of both RMSE and AE. Furthermore, for the training and testing of each algorithm, the dataset is split randomly into 90-10 and 80-20 where the larger number represents the percentage of dataset used for training the algorithm while the smaller is the percentage of the dataset used to test the trained algorithm. In the subsequent discussion we use 9 : 1 and 8 : 2 for the respective splits

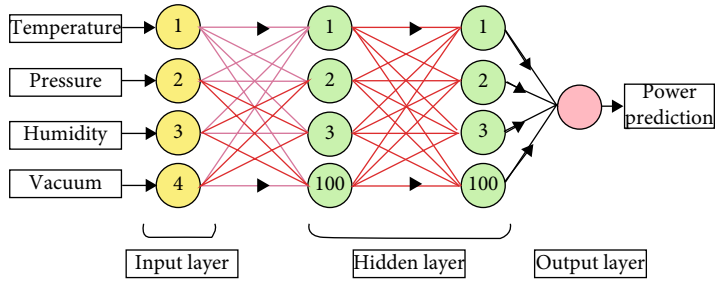


Figure 4: Structure of artificial neural network (ANN) [34].

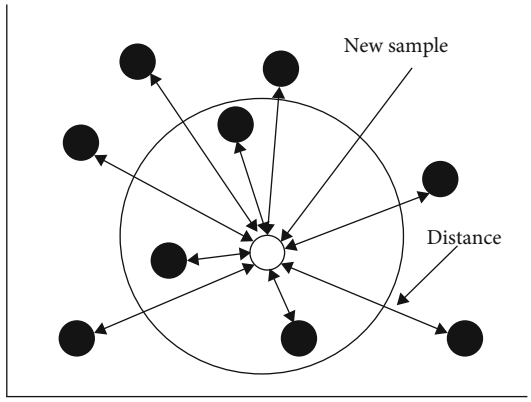


Figure 5: An example of 3-nearest neighbors [40].

(3) With the best values of the parameters for each algorithm, the actual and predicted powers for randomly selected samples from the dataset are shown

(4) Finally, the best results achieved by all the algorithms in terms of RMSE and AE are discussed along with the best results reported by the previous works on the same dataset

4.1. Effect of Features Selection on Predicted Power. In order to examine the effect of each of the input features, i.e., temperature (TEMP), vacuum (VAC), pressure (PRE), and humidity (HUM), on the predicted power, the experiments are performed by making their 15 combinations. Each of these combinations are then used to train and test each of the five algorithms for power prediction. Since, the dataset is randomly split between the training and test sets, the experiments for each of the combinations are performed 10 times. The average RMSE and AE over all the 10 runs achieved by each of the algorithms are presented in Table 5.

Columns 12 and 13 of the table are used to rank the feature combinations from worst to best. Column 12 is the average of all RMSE values achieved by all the five algorithms for a given feature combination. For instance, the mean RMSE for HUM (16.53, row 1, column 12) is calculated as the average of RMSEs achieved by LR (15.66), GBRT (17.31), KNN (18.19), ANN (15.86), and DNN (15.64). The average AE for each of the combinations is calculated in a similar manner. It should be noted that the features combinations are ranked in a descending manner with respect to their average RMSE and AE (columns 12 and 13). For instance, the highest RMSE and AE averaged over all the five algorithms is for humidity (HUM). It is noticeable that the presence of TEMP has a significant effect on the performance of all the algorithms.

Similarly, using the rest of the three features, i.e., HUM, VAC, and PRE without TEMP give high RMSE and AE values regardless of the utilized algorithm. The least RMSE of 2.63 is achieved by GBRT on the combination of TEMP-VAC-PRE whose mean RMSE and AE values are second best (column 12 and 13, second to last row). Hence, it can safely be concluded that the usage of only these three features is enough to perform the rest of the analysis. However, the complete set of the input features achieves the least RMSE and AE (column 12 and 13, last row) due to which dropping just a single feature in the name of achieving efficiency is not technically convincing. Therefore, in the subsequent experiments, all the four features are used.

4.2. Performance Evaluation of ML Algorithms Based on their Important Parameters

4.2.1. K-Nearest Neighbor (KNN). The most important parameter of the K-nearest neighbor algorithm is the value of K as explained in Section 3.3.4. We empirically select the values of K from the set of odd number ranging from 3 to 17. For training and testing, the algorithm uses two phases where in the first phase, we use 90% of the dataset as training data and 10% as test data. Similarly, in phase two, 80% is used for training and 20% of the data is used as the test set. Figure 6 shows the effect of various values of K for both the settings. The minimum RMSE value of 3.28 is achieved at $K = 5$ for 9 : 1 split, while at the same of K, the least RMSE of 3.51 is achieved when the algorithm evaluated on 8 : 2 split.

Similarly, Figure 6 shows the absolute error (AE) at various values of K for both the settings of data split. The minimum value of AE is found to be 2.374 at $K = 3$ for 9 : 1, while at the same value of K, the least AE of 2.49 is achieved at 8 : 2.

4.2.2. Gradient-Boosted Regression Tree (GBRT). The GBRT algorithm is used to predict the output power where the RMSE values are analyzed by varying the number of trees from 50 to 400, as shown in Figure 7. For 9 : 1 split, the minimum RMSE value of 2.581 is achieved for 550 trees. However, there is no significant drop of RMSE from 2.61 achieved at 200 trees to 2.581 achieved at 550 trees. Hence,

Table 5: Effect of feature selection on RMSE and AE for all the algorithms. The table is horizontally partitioned to emphasize on the combinations with and without the temperature (TEMP).

Features combinations	LR		GBRT		KNN		ANN		DNN		Mean RMSE and AE for combinations	
	RMSE	AE	RMSE	AE	RMSE	AE	RMSE	AE	RMSE	AE	RMSE-FEATURES	AE-FEATURES
HUM	15.66	13.15	17.31	13.95	18.19	14.72	15.86	13.69	15.64	13.15	16.53	13.73
PRE	14.39	11.67	15.41	11.62	16.5	12.93	15.55	13.24	14.15	11.42	15.2	12.18
PRE-HUM	13.12	10.63	13.72	10.4	14.19	1.01	13.92	11.7	12.82	10.21	13.55	8.79
VAC	8.43	6.56	5.72	3.99	6.16	4.41	7.78	6.05	7.64	6.02	7.15	5.41
VAC-HUM	8.15	6.39	4.63	3.39	7.48	5.5	7.55	5.93	7.36	5.76	7.03	5.39
VAC-PRE	7.85	6.14	4.45	3.11	6.05	4.24	7.1	5.49	6.93	5.29	6.48	4.85
VAC-PRE-HUM	7.52	5.85	4.15	2.86	5.75	4.08	6.72	5.21	6.35	4.79	6.1	4.56
TEMP	5.07	4.11	5.1	4.03	5.51	4.41	4.89	3.91	4.7	3.75	5.05	4.04
TEMP-PRE	5.01	4.09	4.9	3.71	5.15	3.93	4.67	3.77	4.53	3.63	4.85	3.83
TEMP-HUM	4.43	3.55	4.45	3.5	4.69	3.74	4.39	3.43	4.12	3.25	4.42	3.49
TEMP-PRE-HUM	4.43	3.55	4.01	3.01	4.2	3.12	4.31	3.39	4.06	3.16	4.2	3.25
TEMP-VAC	4.7	3.77	2.9	2.22	4.12	3.09	4.41	3.52	4.2	3.35	4.07	3.19
TEMP-VAC-HUM	4.27	3.4	2.82	2.11	3.79	2.86	4.19	3.33	3.84	3.01	3.78	2.94
TEMP-VAC-PRE	4.63	3.7	2.63	1.96	3.57	2.53	4.16	3.32	3.88	2.96	3.77	2.89
TEMP-VAC-PRE-HUM	4.26	3.4	2.64	1.93	3.32	2.37	4.16	3.34	3.61	2.82	3.6	2.77

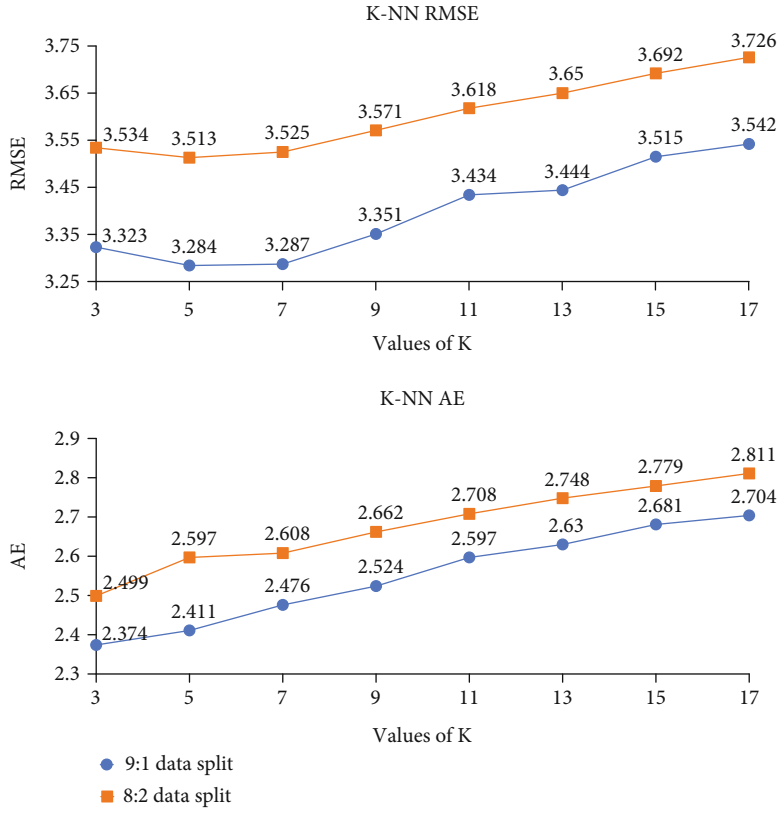


Figure 6: RMSE and AE values achieved by KNN for various values of K on both 9 : 1 and 8 : 2 splits

on the current dataset, for 9 : 1 split, 200 trees are advised for GBRT in order to optimize the usage of computational resources. Furthermore, for 8 : 2 split, the RMSE values are higher than those of 9 : 1. Figure 7 shows a similar trend in the values of AE for various values of trees and for both the settings of dataset split. The least AE for 9 : 1 split is

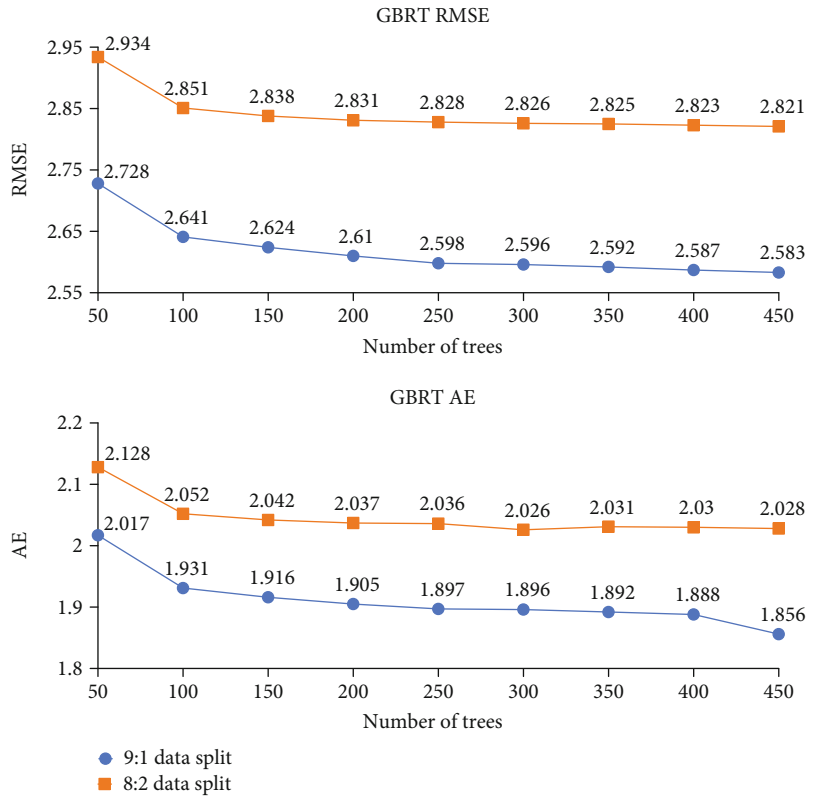


Figure 7: RMSE and AE values achieved by GBRT for various number of trees on both 9 : 1 and 8 : 2 splits.

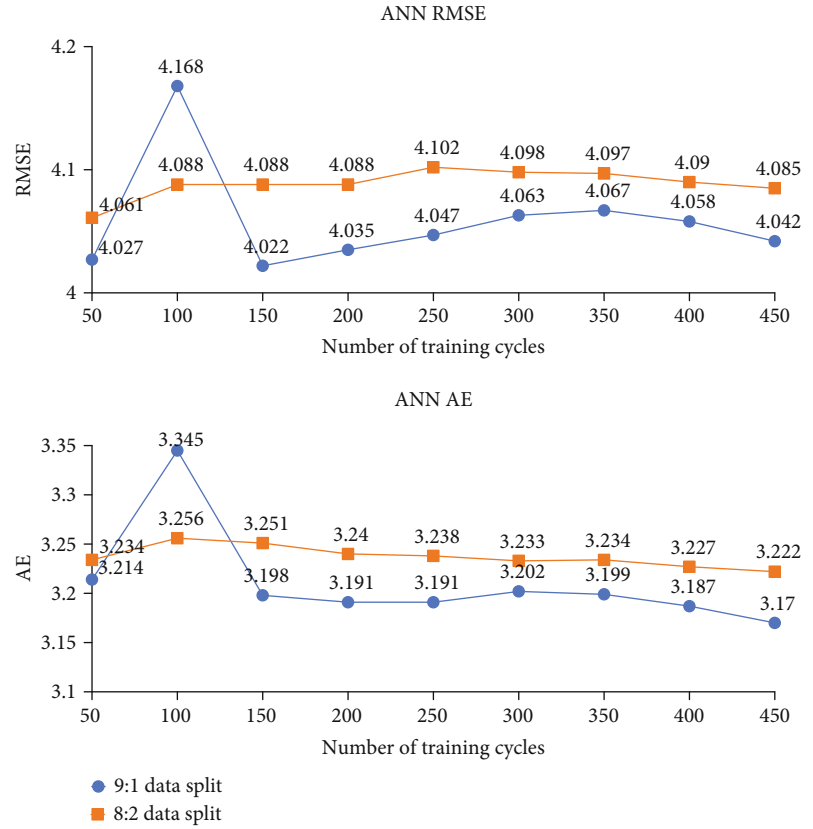


Figure 8: RMSE and AE values achieved by ANN on various number of training cycles for both 9 : 1 and 8 : 2 splits.

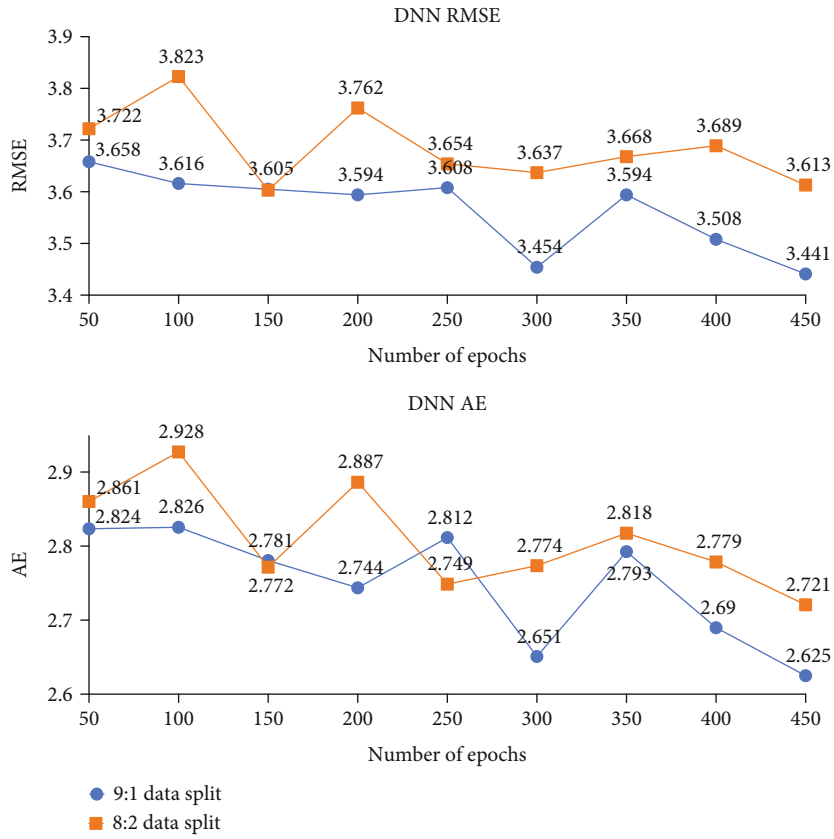


Figure 9: RMSE and AE values achieved by DNN on various training Epochs for both 9 : 1 and 8 : 2 splits.

achieved at 550 trees; however, after 250 trees, the drop is not significant and hence recommended for efficient usage of computational resources on the current dataset. Similarly, the AE values for 8 : 2 split are higher than those of 9 : 1 split.

4.2.3. Artificial Neural Network (ANN). The ANN algorithm of Rapid Miner is used for power prediction on the current dataset. We evaluate the number of training cycles while keeping all other parameters to the default values such as those of the hidden layers and the number of activation functions per layer. Figure 8 shows the achieved RMSE values for various numbers of training cycles for both 9 : 1 and 8 : 2 splits. The least RMSE value of 4.022 is achieved when the training is performed for 150 cycles with 9 : 1 dataset split. For the 8 : 2 split, the minimum RMSE of 4.085 is achieved at 550 training cycles; however the second best RMSE of 4.088 is achieved on 100 training cycles which is comparably acceptable than 4.085 achieved at 550cycles.

In Figure 8, the value of AE are depicted for various training cycles for power prediction. The minimum value of absolute error (AE) at 9 : 1 is observed as 3.15 at 500 training cycles. Using 8 : 2 data split, the minimum AE is 3.222 at 500 training cycles.

4.2.4. Deep Neural Network (DNN). We also use the deep neural network algorithm provided in the Rapid Miner suite which is a variant of the neural network. The default parameters are used while the number of training epochs are varied to observe the effect on the achieved RMSE and AE. Consequently, for 9 : 1 data split, the least RMSE of 3.441 is achieved on 450 epochs. However, the second best RMSE of 3.445 is obtained on 300 epochs and hence recommended for on the current dataset. For the data split of 8 : 2, all the RMSE values are higher than 9 : 1 split. Nontheless, the least achieved RMSE of 3.592 for 8 : 2 split is achieved on 500 epochs. The values of RMSE for both the data splits on various epochs are shown in Figure 9.

Table 6: Comparison of results achieved by each of the algorithms on both the dataset splits. GBRT outperforms the rest of the algorithms by achieving the least RMSE and AE such that the number of trees is 100.

(a)

	RMSE				
Data splits	KNN	GBRT	NN	DL	LR
Training data 90%	3.323	**2.641**	4.168	3.616	4.263
Training data 80%	3.534	**2.851**	4.088	3.823	4.347

(b)

	AE				
	KNN	GBRT	NN	DL	LR
Training data 90%	2.374	**1.931**	3.345	2.826	3.402
Training data 80%	2.499	**2.052**	3.256	2.928	3.469

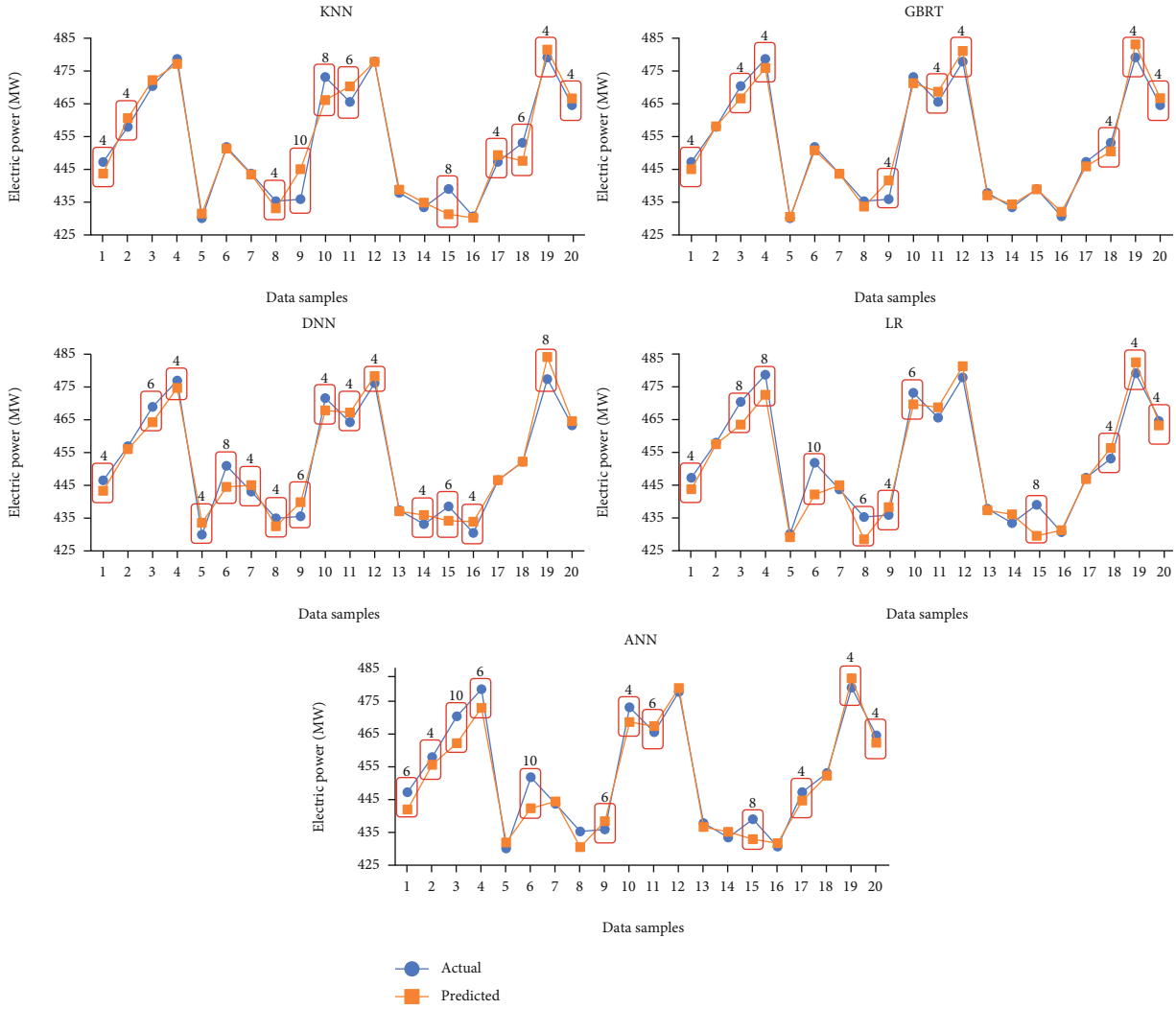


Figure 10: Actual vs. predicted power for 20 random samples of the dataset using five ML algorithms.

Similarly, in Figure 9, AE is shown for both the settings of data split and all the empirically selected values of epochs. The minimum value of AE, using 9 : 1 split, is found to be 2.625 on 450 training epochs. At 8 : 2 split, the minimum AE value is 2.694 achieved on 500 training epochs. To summarize, we show a comparison of all the five algorithms in Table 6 in terms of RMSE and AE for both the settings of data split. The results achieved by each of the algorithms on the current dataset are shown where the value of K in KNN is 3, the number of trees for GBRT is 100, number of training cycles for ANN is 100, and the number of training epochs for DNN is also 100. It can clearly be observed that GBRT outperforms the rest of the algorithms by achieving the least RMSE and AE on the current dataset.

4.2.5. Comparison of Actual and Predicted Powers for each ML Algorithm. In order to demonstrate the prediction results achieved by each algorithm, we randomly select 20 samples from the test dataset such that the actual power value is known for each of them. We then provide the input features of these 20 samples to the trained models of each of the algorithms for power prediction. It should be noted that the models are trained with the best parameters of the respective algorithms as found in the previous section. The actual power values along with the predicted values achieved by each algorithm are shown in Figure 10. The samples where the absolute difference of the actual and predicted power values are greater than 2 MW are highlighted and shown for each algorithm. It can be observed that GBRT achieves power predictions closer to the actual values and the samples where the difference is greater than 2 MW are comparatively less than other algorithms. Hence, it can be concluded that GBRT gives better power predictions than other algorithms.

4.2.6. Performance Comparison with the Literature. Table 7 shows our results in comparison on the previous methods proposed for CCPP power prediction on the same dataset. We clearly achieve superior performance from all those methods using GBRT where the number of tree is 450 and the size of the training set is 90% of the whole dataset. The achieved RMSE and AE values are 2.58 and 3.51, respectively. Similarly, our achieved RMSE and AE values on

Table 7: Results comparison with the previous methods on the CCPP dataset.

	RMSE	AE
Previous literature		
KNN (2012) [19]	N/A	3.51
Bagging with REPTree [4]	4.23	3.22
C-CRF (2016) [20]	3.97	2.97
TSK-ELM (2018) [18]	3.93	N/A
Regression ANN (2018) [12]	4.32	N/A
TOB matching (2020) [21]	2.89	N/A
Ours		
GBRT (450 trees)	**2.58**	**1.85**
KNN ($K = 3$)	3.32	2.37
DNN (450 epochs)	3.44	2.62
ANN (450 cycles)	4.04	3.17
LR	4.26	3.40

KNN and DNN are also better than those methods. Hence, at the moment, we can convincingly say that on the current dataset our results are the best.

5. Conclusion and Future Work

We perform power prediction of a CCPP on hourly basis using machine learning paradigm. In this regard, we use a publicly available dataset that is collected over a period of 6 years such that the power plant is operating on full load. The dataset measures output power as a function of four input parameters which are temperature, humidity, pressure, and vacuum. We evaluate five machine learning algorithms, namely, K-Nearest Neighbors, Linear Regression, Gradient-boosted Regression Tree, Artificial Neural Network, and Deep Neural Network of the Rapid Miner software suite. Keeping the default parameters, we evaluate the most crucial parameters of each algorithm to find the best of them that achieves minimum RMSE and AE. We also evaluate the effect of training set size and number of features on the achieved results. Consequently, GBRT outperforms the rest of the algorithm by achieving the least RMSE and AE with 450 trees while training on 90% of the dataset. Interestingly, it also exceeds in performance from all the previously proposed methods on the same datasetby achieving the least RMSE and AE.

In future, the output power can be controlled by changing the value of the parameters. Moreover, by incorporating these parameters as well as increasing the number of input parameters, the power prediction of different types of power plants can be done by using more advance machine learning algorithms.

Data Availability

The data that support the findings of this study are all briefly introduced, and all information is available in the manuscript.

Disclosure

The authors carried out the research as a part of employment at COMSATS University Islamabad, Attock Campus, UET Peshawar, UoT Nowshera, KPK.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

The second author (Hafeez Anwar) is supported by the Ernst Mach Follow-up grant awarded by the Austrian Agency for International Cooperation in Education and Research (OeAD).

References

[1] J. L. Lobo, I. Ballesteros, I. Oregi, J. del Ser, and S. Salcedo-Sanz, "Stream learning in energy IoT systems: a case study in combined cycle power plants," *Energies*, vol. 13, no. 3, 2020.

[2] A. Castillo, "Risk analysis and management in power outage and restoration: a literature survey," *Electric Power Systems Research*, vol. 107, pp. 9–15, 2014.

[3] B. Çetin, S. H. S. Erdem, S. H. Sevilgen, and A. V. Akkaya, "Electricity production cost analysis of a combined cycle power plant," *Energy Sources, Part B: Economics, Planning, and Policy*, vol. 3, no. 3, pp. 224–232, 2008.

[4] P. Tüfekci, "Prediction of full load electrical power output of a base load operated combined cycle power plant using machine learning methods," *International Journal of Electrical Power & Energy Systems*, vol. 60, pp. 126–140, 2014.

[5] U. Kesgin and H. Heperkan, "Simulation of thermodynamic systems using soft computing techniques," *International Journal of Energy Research*, vol. 29, no. 7, pp. 581–611, 2005.

[6] E. Rich and K. S. Knight, "Chapter 3," in *Artificial Intelligence*, McGraw Hill, New York, USA, 2nd Ed. edition, 1991.

[7] A. Mosavi, F. S. Hosseini, B. Choubin et al., "Susceptibility prediction of groundwater hardness using ensemble machine learning models," *Water*, vol. 12, no. 10, 2020.

[8] A. Mosavi, F. Sajedi-Hosseini, B. Choubin, F. Taromideh, G. Rahi, and A. Dineva, "Susceptibility mapping of soil water erosion using machine learning models," *Water*, vol. 12, no. 7, p. 1995, 2020.

[9] B. Choubin, F. S. Hosseini, Z. Fried, and A. Mosavi, "Application of bayesian regularized neural networks for groundwater level modeling," in *2020 IEEE 3rd International Conference and Workshop in Óbuda on Electrical and Power Engineering (CANDO-EPE)*, pp. 000209–000212, Budapest, Hungary, 2020.

[10] A. Mosavi, F. Sajedi Hosseini, B. Choubin, M. Goodarzi, A. A. Dineva, and E. Rafiei Sardooi, "Ensemble boosting and bagging based machine learning models for groundwater potential prediction," *Water Resources Management*, vol. 35, no. 1, pp. 23–37, 2021.

[11] F. Jiménez-Espadafor Aguilar, M. T. García, E. C. Trujillo, J. A. Becerra Villanueva, and F. J. Florencio Ojeda, "Prediction of performance, energy savings and increase in profitability of

two gas turbine steam generator cogeneration plant, based on experimental data," *Energy*, vol. 36, no. 2, pp. 742–754, 2011.

[12] E. A. Elfaki and A. H. Ahmed, "Prediction of electrical output power of combined cycle power plant using regression ANN model," *Engineering*, vol. 6, no. 12, pp. 17–38, 2018.

[13] G. Şen, M. Nil, H. Mamur et al., "The effect of ambient temperature on electric power generation in natural gas combined cycle power plant-a case study," *Energy Reports*, vol. 4, pp. 682–690, 2018.

[14] A. González-Díaz, A. M. Alcaráz-Calderón, M. O. González-Díaz, Á. Méndez-Aranda, M. Lucquiaud, and J. M. González-Santaló, "Effect of the ambient conditions on gas turbine combined cycle power plants with post-combustion CO_2 capture," *Energy*, vol. 134, pp. 221–233, 2017.

[15] F. Chu, J. Wang, L. Nannan, T. Tan, and F. Wang, Eds., "Prediction of ccpp output based on improved fuzzy analytical hierarchy process," in *2017 29th Chinese Control And Decision Conference (CCDC)*, pp. 3636–3641, Chongqing, China, 2017.

[16] I. Lorencin, N. Anđelić, V. Mrzljak, and Z. Car, "Genetic algorithm approach to design of multi-layer perceptron for combined cycle power plant electrical power output estimation," *Energies*, vol. 12, no. 22, 2019.

[17] P. Pourbeik, "Modeling of combined-cycle power plants for power system studies," in *2003 IEEE Power Engineering Society General Meeting (IEEE Cat. No.03CH37491)*, Toronto, ON, Canada, 2003.

[18] C.-U. Yeom and K.-C. Kwak, "A design of TSK-based elm for prediction of electrical power in combined cycle power plant," in *2018 International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS)*, Bangkok, Thailand, 2018.

[19] H. Kaya, P. Tüfekci, and F. S. Gürgen, "Local and global learning methods for predicting power of a combined gas & steam turbine," in *Proceedings of the International Conference on Emerging Trends in Computer and Electronics Engineering ICETCEE*, pp. 13–18, Dubai, UAE, 2012.

[20] G. Ahn and S. Hur, "Continuous conditional random field model for predicting the electrical load of a combined cycle power plant," *Industrial Engineering and Management Systems*, vol. 15, no. 2, pp. 148–155, 2016.

[21] D. A. Wood, "Combined cycle gas turbine power output prediction and data mining with optimized data matching algorithm," *Applied Sciences*, vol. 2, no. 3, pp. 1–21, 2020.

[22] L. Bandić, M. Hasičić, and J. Kevrić, "Prediction of power output for combined cycle power plant using random decision tree algorithms and ANFIS," in *International Symposium on Innovative and Interdisciplinary Applications of Advanced Technologies*, pp. 406–416, Sarajevo, Bosnia and Herzegovina, 2019.

[23] M. Rashid, K. Kamal, T. Zafar, Z. Sheikh, A. Shah, and S. Mathavan, "Energy prediction of a combined cycle power plant using a particle swarm optimization trained feedforward neural network," in *2015 International Conference on Mechanical Engineering, Automation and Control Systems (MEACS)*, pp. 1–5, Tomsk, Russia, 2015.

[24] A. Burkov, *The Hundred-Page Machine Learning Book, Volume 1*, Andriy Burkov Canada, 2019.

[25] C.-W. Han, *Output Power Prediction of Combined Cycle Power Plant Using Logic-Based Tree Structured Fuzzy Neural Networks*, AIP Publishing LLC, 2019.

[26] I. Mierswa and R. Klinkenberg, "Rapid miner studio," 2018.

[27] X. Wan, W. Wang, J. Liu, and T. Tong, "Estimating the sample mean and standard deviation from the sample size, median, range and/or interquartile range," *BMC Medical Research Methodology*, vol. 14, no. 1, 2014.

[28] S. Rong and Z. Bao-wen, "The research of regression model in machine learning field," *MATEC Web of Conferences*, vol. 176, article 01033, 2018.

[29] A. Schneider, G. Hommel, and M. Blettner, "Linear regression analysis: part 14 of a series on evaluation of scientific publications," *Deutsches A¨rzteblatt International*, vol. 107, no. 44, p. 776, 2010.

[30] J. H. Friedman, "Gradient boosting (regression)," 2021, https://statweb.stanford.edu/jhf/ftp/stobst.pdf.

[31] X. Li and R. Bai, "Freight vehicle travel time prediction using gradient boosting regression tree," in *2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pp. 1010–1015, Anaheim, CA, USA, 2016.

[32] K. Gao, H. Chen, X. Zhang, X. K. Ren, J. Chen, and X. Chen, "A novel material removal prediction method based on acoustic sensing and ensemble XGBoost learning algorithm for robotic belt grinding of Inconel 718," *The International Journal of Advanced Manufacturing Technology*, vol. 105, no. 1-4, pp. 217–232, 2019.

[33] C. Persson, P. Bacher, T. Shiga, and H. Madsen, "Multi-site solar power forecasting using gradient boosted regression trees," *Solar Energy*, vol. 150, pp. 423–436, 2017.

[34] R. Siddiqui, S. Umer, A. Iqbal, F. Ullah, A. Khan, and K. S. Kwak, "Estimation of solar panel output based on weather parameters using machine learning algorithms," in *Proceedings of the Korea Telecommunications Society Conference*, pp. 459–462, Korea, 2020.

[35] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter, "Continual lifelong learning with neural networks: a review," *Neural Networks*, vol. 113, pp. 54–71, 2019.

[36] X. Wenchao, Y. Pang, Y. Yang, and Y. Liu, "Human activity recognition based on convolutional neural network," in *In 2018 24th International Conference on Pattern Recognition (ICPR)*, pp. 165–170, Beijing, China, 2018.

[37] R. Goyal, P. Chandra, and Y. Singh, "Suitability of KNN regression in the development of interaction based software fault prediction models," *Ieri Procedia*, vol. 6, pp. 15–21, 2014.

[38] T. Lee, T. B. M. J. Ouarda, and S. Yoon, "KNN-based local linear regression for the analysis and simulation of low flow extremes under climatic influence," *Climate Dynamics*, vol. 49, no. 9-10, pp. 3493–3511, 2017.

[39] S. B. Imandoust and M. Bolandraftar, "Application of k-nearest neighbor (knn) approach for predicting economic events: theoretical background," *International Journal of Engineering Research and Applications*, vol. 3, no. 5, pp. 605–610, 2013.

[40] Z. Yao and W. L. Ruzzo, "A regression-based k nearest neighbor algorithm for gene function prediction from heterogeneous data," *BMC Bioinformatics*, vol. 7, Supplement1, 2006.

[41] A. Qazi, A. W. Fayaz, A. Wadi, R. G. Raj, N. A. Rahim, and W. A. Khan, "The artificial neural network for solar radiation prediction and designing solar systems: a systematic literature review," *Journal of Cleaner Production*, vol. 104, pp. 1–12, 2015.

Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 3237219, 14 pages
https://doi.org/10.1155/2021/3237219

Research Article

Source Localization of EM Waves in the Near-Field of Spherical Antenna Array in the Presence of Unknown Mutual Coupling

Oluwole John Famoriji and Thokozani Shongwe

Department of Electrical and Electronic Engineering Science, University of Johannesburg, P.O. Box 524, Auckland Park, 2006 Johannesburg, South Africa

Correspondence should be addressed to Oluwole John Famoriji; famoriji@mail.ustc.edu.cn

Received 12 July 2021; Revised 23 October 2021; Accepted 12 November 2021; Published 14 December 2021

Academic Editor: Fawad Zaman

To obtain an antenna array with isotropic radiation, spherical antenna array (SAA) is the right array configuration. The challenges of locating signals transmitted within the proximity of antenna array have been investigated considerably in the literature. However, near-field (NF) source localization of signals has hitherto not been investigated effectively using SAA in the presence of mutual coupling (MC). MC is another critical problem in antenna arrays. This paper presents an NF range and direction-of-arrival (DoA) estimation technique via the direction-independent and signal invariant spherical harmonics (SH) characteristics in the presence of mutual coupling. The energy of electromagnetic (EM) signal on the surface of SAA is captured successfully using a proposed pressure interpolation approach. The DoA estimation within the NF region is then calculated via the distribution of pressure. The direction-independent and signal invariant characteristics, which are SH features, are obtained using the DoA estimates in the NF region. We equally proposed a learning scheme that uses the source activity detection and convolutional neural network (CNN) to estimate the range of the NF source via the direction-independent and signal invariant features. Considering the MC problem and using the DoA estimates, an accurate spectrum peak in the multipath situation in conjunction with MC and a sharper spectrum peak from a unique MC structure and smoothing algorithms are obtained. For ground truth performance evaluation of the SH features within the context of NF localization, a numerical experiment is conducted and measured data were used for analysis to incorporate the MC and consequently computed the root mean square error (RMSE) of the source range and NF DoA estimate. The results obtained from numerical experiments and measured data indicate the validity and effectiveness of the proposed approach. In addition, these results are motivating enough for the deployment of the proposed method in practical applications.

1. Introduction

Antenna arrays, in which the distribution of radiating elements is over a spherical surface, remain the answer to the isotropic requirements [1]. Spherical antenna arrays (SAA) have the capability of receiving electromagnetic (EM) wave with equal strength regardless of the polarization and the direction-of-arrival (DoA). To take advantage of the highest degree of freedom, the antenna array under consideration must have the ability to determine the DoA and polarization of an incoming EM wave, impinging from all directions on the unit sphere. Although previous theoretical description has been given on SAA [2–6], more in-depth and precise description of the EM characteristics of the SAA remains an open challenge.

Multiple EM signal localization within the proximity of antenna array has been dealt with in the literature. Most works considered the sources that are in far-field (FF) of the array [8–10]. However, this paper focuses on sources that are in the near-field (NF) of SAA; in this case, it will be possible to estimate DoA and range of the source. This has not been considered in the previous works. Apart from EM systems, NF localization also applies to various areas such as sonar systems, seismic exploration, and microphone arrays. In this paper, EM systems utilizing SAA (as shown in Figure 1) [7] are considered. Spherical harmonics (SH)

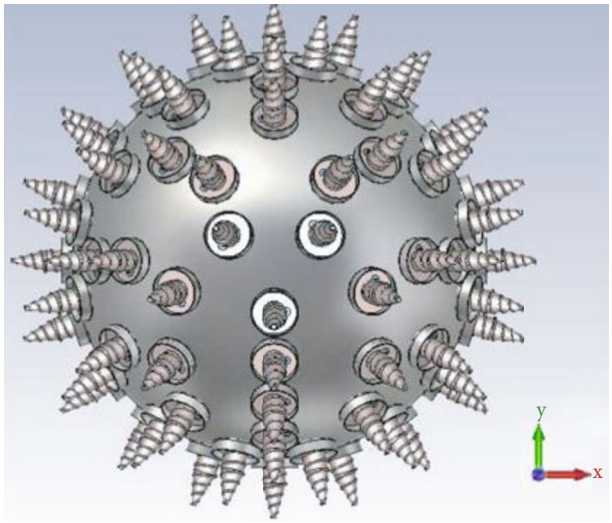


Figure 1: A typical SAA with 64 elements [7].

decomposition can be employed in the representation of reflected signals based on the coefficients, reflector locations, and mode strength matrix [11]. Radial filters have been adopted for the separation of sources, which are situated at various ranges from the system. Some algorithms have also been developed for radial filters [11, 12] and NF beamforming [13, 14] in the SH domain. However, each of the algorithms requires prior knowledge of the DoA and range of the NF source. The NF source localization of signals has hitherto not been effectively investigated using SAA in a harsh environment. Therefore, presenting an NF range and DoA estimation technique using the direction-independent and signal invariant SH features towards NF localization of EM signals using SAA becomes crucial.

Recently, Wu and Yan [15] presented a 3-dimensional far-field and near-field source localization via cross array. The method is applicable to different symmetric cross array and disallowed aperture loss. In [16], Shu et al. employed a spatially spread vector sensor to localize 3-D near-field source. This method is applicable to non-free-space propagation models at arbitrarily unknown path-loss exponent and offers an enhanced performance estimation because of the inherent extension of the vector sensor's spatial aperture in spread structure. A mixed near-field and far-field using discrete Fourier transform and orthogonal matching pursuit was used achieve a low complex localization method [10]. This method shows robustness at low SNR and does not demand any matrix decomposition. Furthermore, a one-snapshot mixed far-field and near-field localization algorithm based on discrete fractional Fourier transform is developed [17, 18]. This method simultaneously estimates and classifies mixed sources. Moreover, Kuznetsov et al. [19] performed an autocorrelation analysis and near-field localization of the radiating sources with cyclostationary properties. An autocorrelation analysis procedure for the characterization of radiated emissions from multifunctional digital electric devices in the reactive near-field region is presented and takes advantage of spatial-time localization of transmission line [19].

Furthermore, Wu and Yan [20] proposed a gridless mixed-source localization method using a low-rank matrix reconstruction. A Vandermonde structure was established for the estimation of DoA and range via the fourth-order cumulants for the development of two spatial matrices that only relate to range and DoA. This approach needs no discretization of range nor the angle regions [20]. Also, an improved near-field multiple signal classification method has been proposed in [21] for DoA estimation accuracy improvement at low SNR. The method improves accuracy by 2 degrees at that low SNR. He et al. [22] proposed a cumulant-based method for multiple near-field source localizations using an exact source-sensor spatial geometry. Here, there is no Fresnel approximation of spatial phase-delay and nonconsideration of propagation attenuation made; i.e., no simplification is made. The algorithm is noniterative, simple, and search-free; it accommodates all arbitrarily unknown propagation loss. In the same vein, Ma et al. [23] investigated the challenge of mixed far-field and near-field source localization by linear electromagnetic-vector-sensor array with phase/gain uncertainties. The estimates are able to generate the unknown gain or phase errors. This technique requires no spectral search and no restriction imposed on the electromagnetic vector sensors placement and realization of more reasonable signal type classification. Linear tripole array [24] and exact spatial propagation geometry [25] concepts have also been used for the realization of mixed near-field and far-field source localization. Moreover, near-field source localization has been dealt with in different angles using source non-circularity and based on the virtual ESPRIT idea [26], symmetric subarrays [27], two-level nested arrays [28], and subspace [29]. However, no attention has been paid to near-field source localization in spherical domain using antenna array.

There are existing methods reported in the literature for the estimation of DoA such as estimation of subspace rotational in variance technology (ESPRIT) algorithm, multiple signal classification (MUSIC), MUSIC group delay [11, 18, 19], the steered response power with phase transform (SRP-PHAT) [30–35], generalized cross-correlation (GCC) [36], adaptive eigenvalue decomposition [37], one-dimensional MUSIC [38], and order aware method [39]. MUSIC has been applied to the SH structure in NF (MUSIC-SH) [33, 40–42]. Because MUSIC-SH is sensitive to distortion in multipath, Nakamura [43] developed another technique named direct-path dominance (DPD). In NF, mode strengths depend on source range that is not a priori known. As such, DPD is only conducted in the dimension of time, and because of this, MUSIC-SH-DPD requires higher frames [43]. In the application of NF frequency smoothing, normalizing a source that is range dependent is needed. The other DoA estimation method formulated in the SH domain is the minimum variance distortionless response [44].

Some works on NF localization have been presented in the literature. He et al. [45] addressed the problem of partial NF EM source localization using an array of cross-dipole. An algorithm that is based on maximum likelihood was developed for the estimation of range and angle parameters.

The search over the polarization parameters is not required in the algorithm [45]. Guzey et al. [46] presented the localization of NF radio-controlled devices under a multipath environment using a uniform linear array (ULA). The authors developed a smooth 2-D MUSIC NF localization framework for radio control devices in a multipath fading event. The performance evaluation of symmetric subarray-based NF localization and 2-D MUSIC was compared with the developed smooth 2-D MUSIC framework [46]. Recently, Zheng et al. [47] proposed a source localization framework for mixed NF and FF sources using symmetric double nested antenna arrays. In FF sources, the DoAs are estimated using 1-D MUSIC, and the NF part was extracted from the FF using the oblique projection method. In the end, the range estimates of NF sources were obtained from the NF DoA estimates using 1-D peak searching method. The developed algorithm used the large coarray aperture to enhance the localization performance and capable of accurately differentiating the kind of sources under consideration [47].

Another NF localization has been more recently reported in [48]. The authors presented a comprehensive and generalized method for the evaluation of spatiotemporal energy that is localized in the proximity of antenna arrays of arbitrary structures and excitation of signals. Finite-difference time-domain approach was employed to compute the pointing localized energy for all structures by special consideration on the calculated fields as against the currents of the source. It is shown and demonstrated that the localized energy is fundamentally different from the classical antenna reactive energy, associated with frequency-domain Q-factors of electrically small structures. The developed concept of localized energy and the techniques developed for the investigation of its relevant new data help in the implementation of the ultrahigh gain closely packed antennas, which are useful for the emerging 5G communications. Ma et al. [49] also presented an algorithm for mixed FF and NF source localization using a linear tripole antenna array. The authors formulated special fourth-order cumulants and utilized the shift-invariant properties in cumulant domain to estimate the source parameters. The resulted estimate is used to simplify the cyclically ambiguous estimate and consequently generate parameter estimates with high accuracy.

Furthermore, with the current growth in technology and system applications, systems are becoming smaller, leading to smaller interelement distance in an array. This causes strong mutual coupling, impedance mismatch, and poor radiation characteristics. This problem will often affect signal antenna array processing and more severe in the near field.

Lately, Zheng et al. [50] proposed a new symmetric nonuniform linear array, named symmetric displaced coprime array (SDCA), for simultaneous localization of NF and FF sources. The SDCA is only decided via closed-form expression, and the corresponding coarray ranges are calculated by analysis. Besides, two optimum SDCA structures were formulated by maximizing the numbers of the unique and consecutive lags in difference coarray. The proposed SDCA configuration shows some levels of superiorities over the existing symmetric nonuniform linear arrays. Therefore, so far, the challenges of locating signals transmitted within the proximity of antenna array have been investigated considerably in the literature. However, NF source localization of signals has hitherto not been investigated effectively using SAA in a harsh environment.

From the localization methods discussed above, it is evident that most reported works only focused on FF sources. However, in NF [1, 44–50], the assumption that the waveform is planar, is not always accurate; this consequently causes a high error margin in practice. For accurate antenna array processing, the spherical wavefront from NF sources requires adequate consideration [51]. Although there are some works on NF localization, only a few reported experimental data, which is the true test of any procedure and concept.

In addition, for NF sources, some techniques have been reported in the literature for the estimation of range [40, 52–54] and DoA [7, 9–31, 45, 55]. There is a noticeable variation in the received energy at various observation points in NF EM wave propagation, because of the change in distance that exists between the source and point of observation. However, in the case of FF EM wave propagation, the distance of the source is negligible as the EM energy of the received signal at various observation points has infinitesimal or no variation. The traditional techniques for estimation of NF DoA did not take advantage of this feature of the NF EM signal propagation. For the case of range estimation, most approaches considered the environment ideal. There is performance degradation of the approaches in noisy and harsh environments because the mode strengths are sensitive to the surrounding. Furthermore, the required frequency range for consideration in the estimation of source range is a function of the same source range. Hence, a learning-based technique is more appropriate for the accurate estimation of the source range.

In contrast to the reports in the literature, the key innovation and major contributions in this article are summarized as follows. This paper presents an NF range and DoA estimation technique via the direction-independent and signal invariant SH characteristics. The energy of EM signal on the surface of SAA is successfully captured using a proposed pressure interpolation approach. The DoA estimation within the NF region is then calculated via the distribution of pressure. The direction-independent and signal invariant characteristics, which are SH features, are obtained using the DoA estimates in the NF region. We obtained the signal invariant features using SH coefficient normalization with a particular component, which corresponds to the signal strength of the source. Furthermore, we obtained the rotation-independent features by two approaches: (a) rotation of SH functions on a sphere is conducted by Wigner-D function, whereas on the other hand (b) the impact of DoA is calibrated by the normalization of SH. We equally proposed a learning scheme that uses the source activity detection and CNN to estimate the range of the source of the NF via the direction-independent and signal invariant features. Finally, to the best knowledge of the authors, this

is the first study on NF source localization of EM waves that considers SAA.

The rest of this article is sectioned as follows. The SH-based signal model and the NF criterion are presented in Section 2. Section 3 presents the proposed NF DoA estimation, while the proposed CNN-based range estimation procedure is given in Section 4. Section 5 presents the algorithm with a systematic implementation procedure. Performance evaluation and comparison between the proposed method and previous methods using experimental measured data are given in Section 6. Finally, Section 7 draws the conclusion.

2. Signal Model

Let us consider an EM signal scenario with an SAA of radius r_e and M number of antenna elements with associated source positioned at (r_s, θ_s, ϕ_s) radiating and EM wave $z(k)$, where r_s denotes the radial distance and (θ_s, ϕ_s) represents the spherical coordinates [56]. The EM signal pressure observed at (r_e, θ, ϕ) is defined as [57]

$$p(k, r_e, \theta, \phi) = \sum_{n=0}^{N} \sum_{m=-n}^{n} b_n^s(k, r_e, r_s)[Y_n^m(\theta_s, \phi_s)]^* Y_n^m(\theta, \phi)z(k), \tag{1}$$

where k denotes the wave number directly proportional to frequency f, i.e. $k = 2\pi f/c$, c represents speed of the propagating wave, a constant usually taken as the speed of light, and n and m are the order and degree, respectively; while Y denotes harmonics. The symbol $[\cdot]^*$ denotes complex conjugate representation. Sound and electromagnetic waves carry energy from one place to another and have a wavelength, frequency, and velocity. However, the velocity of sound waves is much lower than that of electromagnetic waves. Hence, $v = f\lambda$ relationship holds for both. Component of noise can be incorporated into Equation (1) to model the signal and element noise. For ease of derivation, the noise component is not included and not needed while deriving the algorithm. Also, a truncation order is strictly ensured to prevent the problem of aliasing because few SH order is needed in spatial sampling, which is related to the limited bandwidth in time domain sampling [58]. The boundary of the sphere is a function of $b_n^s(k, r_e, r_s)$ and has a relation $b_n(kr_e)$ as

$$b_n^s(k, r_e, r_s) = i^{-(n-1)}kb_n(kr_e)h_n(kr_s), \tag{2}$$

where n represents the order of SAA and $i = \sqrt{-1}$. It shows the radial characteristic of point sources can be estimated using spherical Hankel function $h_n(kr_s) \cdot b_n(kr_e)$ for a rigid sphere expressed as

$$b_n(kr_e) = 4\pi i^n \left[j_n(kr_e) - \frac{j_n'(kr_e)}{h_n'(kr_e)} h_n(kr_e) \right], \tag{3}$$

where j_n is the spherical Bessel function of the first kind and h_n is the Hankel function of the second kind, while j_n' and h_n' denote their corresponding derivatives. $Y_n^m(\theta, \phi)$ represents the SH of n order and m degree expressed as

$$Y_n^m(\theta, \phi) = \sqrt{\frac{2n+1}{4\pi}} \frac{(n-m)!}{(n+m)!} P_n^m(cos\theta)e^{im\phi}, \tag{4}$$

where $P_n^m(\cdot)$ denotes the associated Legendre equations. The spatial domain, $p(k, r_e, \theta, \phi)$, transformation to SH domain $p_{nm}(kr_e)$ is [57]

$$p(k, r_e, \theta, \phi) = \sum_{n=0}^{N} \sum_{m=-n}^{n} p_{nm}(kr_e) Y_n^m(\theta, \phi). \tag{5}$$

By juxtaposing Equations (1) and (5), the coefficient of the SH can be expressed as

$$p_{nm}(kr_e) = b_n^s(k, r_e, r_s)[Y_n^m(\theta_s, \phi_s)]^* z(k). \tag{6}$$

This holds for an ideal environment where a component of the direct path exists between the antenna and the source. However, in a practical situation, a harsh environment where multipath components exist requires special consideration because of reflections. The multipath feature can be added to Equation (6), by incorporating a reflection parameters $\mathfrak{R}$, which is defined as

$$\mathfrak{R} = \{(a_1, r_1, \theta_1, \phi_1), (a_2, r_2, \theta_2, \phi_2), (a_{\varpi}, r_{\varpi}, \theta_{\varpi}, \phi_{\varpi})\}, \tag{7}$$

where ϖ represents the number of reflections. a_l is the reflection coefficient and (r_l, θ_l, ϕ_l) is the reflection point that corresponds to gth reflection. The relationship between k and ϖ is not considered in this paper. Equation (6) can then be written as [58]

$$\begin{aligned} p_{nm}(kr_e) = {} & b_n^s(k, r_e, r_s)[Y_n^m(\theta_s, \phi_s)]^* z(k) \\ & + \sum_{l=1}^{\varpi} b_n^s\left(k, r_e, r_g\right) \left[Y_n^m\left(\theta_g, \phi_g\right) \right]^* e_g z(k). \end{aligned} \tag{8}$$

Evidently, based on Equation (8), $p_{nm}(kr_e)$ depends on r_s , (θ_s, ϕ_s), $z\,(k)$, and $\mathfrak{R}$. The relationship is given as

$$p_{nm}(kr_e) = v(r_s, \mathfrak{R}, (\theta_s, \phi_s)z(k)), \tag{9}$$

where $v(\cdot)$ is a variable representing the existing relationship between $p_{nm}(kr_e)$ and parameters. The dependency of r_e and k is not discussed exclusively, while the dependency of p_{nm} (kr_e) and the importance is later discussed in Section 4.2.

The FF and NF for SAA can be defined according to their capability to differentiate various source distances from the antenna array. The criterion of the NF defines the distance from which the radial magnitude of signal pressure of a point source is identical to the plane wave [12]. An approach used to determine the criterion of NF is already reported in the literature [37]. Considering a source range r_s and an SAA with N order and r_e radius, when $kr_s > > n$,

the mode strength of the NF, $b_n^s(k, r_e, r_s)$ is

$$b_n^s(k, r_e, r_s) \approx (-i)^{(n+1)} b_n(kr_e) \frac{e^{ikr_s}}{r_s}. \tag{10}$$

Therefore, when $kr_s >> n$, $b_n^s(k, r_e, r_s)$ and $b_n(kr_e)$ share similar characteristics [37]. Because the discrimination of source distance is conducted only on source distance itself, the NF criterion is

$$r_{\mathrm{NF}}^{'} \approx \frac{N}{k}. \tag{11}$$

Because the source is outside of SAA, r_s is required to satisfy the $r_s > r_e$ condition. Therefore, for FF and NF, the range of r_s is expressed as

$$\begin{aligned} \text{Near-field} &: r_e < r_s < \frac{N}{k}, \\ \text{Far-field} &: r_s > \frac{N}{k}. \end{aligned} \tag{12}$$

Because the aim of this work centrals on NF localization, subsequently, only NF source scenario is considered.

3. DoA Estimation from NF Source

Estimating the DoA in the FF region, the phase of the received signal at multiple elements contributes to the estimation of DoA when the elements are in close proximity (NF) [59]. The magnitude of the received signal is very similar to various elements for the FF source. In NF scenario, the variation of the received energy at different observation points on the SAA is significant. Using inverse spherical Fourier transform (SFT) on the energy distribution variation on SAA via interpolation, (a) there is variation in the distribution of energy over SAA from its peak at the corresponding point to the DoA and a minimum opposite the maxima, and (b) the profile of the location of maximum energy corresponds to DoA can be estimated via various experimental cases, such as SNR, DoA, and r_s [47].

The above points gave rise to the method that NF DoA can accurately and simply be estimated at the point where the highest energy is received on the SAA. It is practically impossible to have elements situated over the whole sphere to understand the distribution of energy for DoA estimation. This problem is solvable via observation interpolation over the whole sphere using spatial sampling and inverse SFT.

The SH coefficients $p_{nm}(kr_e)$ are obtained from the measurements at the position of the elements over the sphere. Using SFT, the signal pressure at an observation point on the sphere can be computed using inverse SFT as

$$p(k, r_e, \theta_e, \phi_e) = \sum_{n=0}^{N} \sum_{m=-n}^{n} p_{nm}(kr_e) Y_n^m(\theta_e, \phi_e) \forall (\theta_e, \phi_e) \, \epsilon \, Z^2. \tag{13}$$

The antenna element closest to the source gets larger EM intensity than the farther elements in wave propagation scenario. Therefore, the DoA estimation can be performed at the point where the highest energy is received on the sphere. This is expressed as

$$\left(\widehat{\theta}_s, \widehat{\phi}_s\right) = \arg \max_{(\theta_e, \phi_e)} |p(k, r_e, \theta_e, \phi_e)|^2. \tag{14}$$

For time-dependent signals, the method can be applied by finding the total energy on the accessible frames and applicable frequency bins. DoA estimate is the point where maximum energy is received and can be expressed as

$$\left(\widehat{\theta}_s, \widehat{\phi}_s\right) = \arg \max_{(\theta_e, \phi_e)} \sum_{\forall \tau} \sum_{\forall k_f} \left|p\left(\tau, k_f, r_e, \theta_e, \phi_e\right)\right|^2, \tag{15}$$

where k_f and τ represent the frequency bin and frame index, respectively.

4. Range Estimation of NF Source

This section presents the process involved in extracting signal invariant (SI) and direction-independent (DI) (SIDI) features in spherical domain towards the estimation of range via the DoA estimates obtained from the proposed inverse SFT technique.

4.1. SIDI Feature Extraction. The parameters on which SH coefficients $p_{nm}(kr_e)$ are dependent have been given in the signal model in Equation (9) to be $p_{nm}(kr_e) = \nu(r_s, \mathfrak{R}, (\theta_s, \phi_s), z(k))$. The work here is to reduce the dependency of $p_{nm}(kr_e)$ on parameters that are not wanted through transformation, which are $\mathfrak{R}$, (θ_s, ϕ_s), and $z(k)$. It is frequently challenging to calculate $\mathfrak{R}$ within the context of an enclosure. Note that in a practical situation, there is the existence of scattering, noise, etc. Since, $p_{nm}(kr_e)$ dependency on $\mathfrak{R}$ cannot be established effectively; it is achievable through learning obtained from the training. The SH coefficient $p_{nm}(kr_e)$ transformation problem can be mathematically formulated as

$$\begin{aligned} &\text{find} \, T, \\ \text{s.t.} \quad &q_{nm}(kr_e) = T(p_{nm}(kr_e)) \\ &q_{nm}(kr_e) = \tilde{\nu}(r_e, \mathfrak{R}), \end{aligned} \tag{16}$$

where q_{nm} denotes the wanted SIDI features, T represents the transformation on p_{nm}, which is unknown to compute q_{nm}, and the $\tilde{\nu}(\cdot)$ denotes the function (unknown) showing the nature of dependency on $(r_e, \mathfrak{R})$ parameters alone. The approach needed to get the appropriate transformation is discussed as follows.

4.2. SI Features. The SI characteristics must be invariant to the variation of the signal strength $z(k)$ of the source. This is achievable using the SH coefficient that corresponds to zero-order and zero degrees. Since $Y_0^0(\theta, \phi) = \sqrt{1/4\pi}$ from

Equation (4); the $p_{00}(kr_e)$ in Equation (8) is expressed as

$$p_{nm}(kr_e) = \sqrt{\frac{1}{4\pi}}\left[b_0^s(k, r_e, r_s) + \sum_{l=1}^{\varpi} b_0^s\left(k, r_e, r_g\right)e_g\right]z(k). \tag{17}$$

As shown, $p_{00}(kr_e)$ shares a relationship with $z(k)$. Therefore, $p_{00}(kr_e)$ is employed to normalize $p_{nm}(kr_e)$ as

$$\tilde{p}_{nm}(kr_e) = \frac{p_{nm}(kr_e)}{p_{00}(kr_e)} = \frac{b_n^s(k, r_e, r_s)[Y_n^m(\theta_s, \phi_s)]^* + \sum_{g=1}^{\varpi} b_n^s(k, r_e, r_g)\left[Y_n^m\left(\theta_g, \phi_g\right)\right]^* e_g}{\sqrt{1/4\pi}\left[b_0^s(k, r_e, r_s) + \sum_{g=1}^{\varpi} b_0^s(k, r_e, r_g)e_g\right]}. \tag{18}$$

So, the $\tilde{p}_{nm}(kr_e)$ dependency on different parameters is now formulated to be

$$\tilde{p}_{nm}(kr_e) = \hat{\nu}(r_s, \mathfrak{R}, (\theta_s, \phi_s)), \tag{19}$$

where $\hat{\nu}$ is the parameter showing the $\tilde{p}_{nm}(kr_e)$ dependency on the parameters. Therefore, signal invariant features are observed via normalization of SH coefficients with $p_{00}(kr_e)$. The directional independence (DI) is given below.

4.3. DI via Rotation. From Equation (19), it is observed that $\tilde{p}_{nm}(kr_e)$ depends on the DoA of NF source. For learning the mapping function that maps the SH features to the source range, it is preferable for these features to depend on r_s to the largest magnitude. So, DI becomes crucial. Assume $(\widehat{\theta}_s, \widehat{\phi}_s)$ to be the DoA estimate of NF computed through inverse SFT technique. If there is a rotation of the coordinate in such a way that the active DoA $(\theta, \phi) = (0, 0)$, then the features are DI. The rotation of SH is conducted anticlockwisely on Euler angles α, β, and γ around the corresponding x-, y-, and z -axes. In practice, SH rotation can be done via multiplication of it SH coefficients with matrix $\mathbf{D}(\alpha, \beta\gamma)$ modelled with Wigner-D functions using (α, β, γ) parameters [57]. Furthermore, consider the condition in which $(\alpha, \beta\gamma) = (0, -\widehat{\theta}_s, -\widehat{\phi}_s)$. Under such condition, the north pole of the sphere aligns to the DoA. The variation of spherical reflection component depends on the DoA, still, and can be established by learning from training samples. Therefore, the kind of transformation to extract the SH features that are both SI and DI (SH-SIDIR) is expressed as

$$q_{nm}(kr_e) = \mathbf{D}\left(0, -\widehat{\theta}_s, -\widehat{\phi}_s\right)\tilde{\mathbf{p}}_{nm}(kr_e). \tag{20}$$

Putting Equation (18) into Equation (20) gives

$$q_{nm}(kr_e) = \frac{b_n^s(k, r_e, r_s)[Y_n^m(\theta_s, \phi_s)]^* + \sum_{g=1}^{\varpi} b_n^s(k, r_e, r_g)\left[Y_n^m\left(\theta_g, \phi_g\right)\right]^* e_g}{\sqrt{1/4\pi}\left[b_0^s(k, r_e, r_s) + \sum_{g=1}^{\varpi} b_0^s(k, r_e, r_g)e_g\right]}, \tag{21}$$

where $\theta_g' = \theta_g - \theta_s$, $\phi_g' = \phi_g - \phi_s$, and $Y_n^m(0, 0)$ from Equation (4) is

$$Y_n^m(0, 0) = \sqrt{\frac{2n+1}{4\pi}\frac{(n-m)!}{(n+m)!}}P_n^m(1). \tag{22}$$

It is seen that the component of the direct path does not depend on the source DoA. It is also possible to obtain DI by SH division as follows.

4.4. Directional Independence via Division. DI can be established by using the SH part of DoA estimated to divide the $\tilde{p}_{nm}(kr_e)$ that corresponds to it. The SI and DI SH features (SH-SIDID) are obtained from

$$q_{nm}(kr_e) = \tilde{\mathbf{p}}_{nm}(kr_e) \ominus [\mathbf{Y}_n^m(\theta\wedge_s, \phi\wedge_s)]^*, \tag{23}$$

where $\ominus$ is the element-by-element division. Putting $\tilde{p}_{nm}(k r_e)$ of Equation (18) into Equation (23), then

$$q_{nm}(kr_e) = \frac{b_n^s(k, r_e, r_s) + \sum_{g=1}^{\varpi} b_n^s(k, r_e, r_g)\left(\left[Y_n^m\left(\theta_g, \phi_g\right)\right]^* / [Y_n^m(\theta\wedge_s, \phi\wedge_s)]^*\right)e_l}{\sqrt{1/4\pi}\left[b_0^s(k, r_e, r_s) + \sum_{g=1}^{\varpi} b_0^s(k, r_e, r_g)e_g\right]}. \tag{24}$$

It is seen now that the component of the direct path does not depend on the source DoA. Note that divisor $Y_n^m(\widehat{\theta}_s, \widehat{\phi}_s)$ could be ill-conditioned depending on the value of $Y_n^m(\widehat{\theta}_s, \widehat{\phi}_s)$. Under such a scenario, the impact of the component of the direct path to observation, $b_n^s(k, r_e, r_s)[Y_n^m(\theta_s, \phi_s)]^*$, is minimal, and the level of $q_{nm}(kr_e)$ can be put in a threshold to mitigate the impact. Hence, $q_{nm}(kr_e)$ dependency can be given as

$$q_{nm}(kr_e) = \tilde{\nu}(r_s, \mathfrak{R}). \tag{25}$$

4.5. SH-SIDID versus SH-SIDIR. Here, we state the difference in summary between SH-SIDID and SH-SIDIR features. In the extraction process of the features, we obtain the DoA $(\widehat{\theta}_s, \widehat{\phi}_s)$ estimate and signal invariant SH features $\tilde{\mathbf{p}}_{nm}(kr_e)$ first. It is seen from Equation (21) that the effective DoA of SH-SIDIR is (0, 0). We can also be ascertained from Equation (24) for SH-SIDID that the impact of $Y_n^m(\theta_s, \phi_s)$ is calibrated using $Y_n^m(\widehat{\theta}_s, \widehat{\phi}_s)$. Therefore, SH-SIDID defines mode strengths, which makes it a better choice for range information representation.

4.6. CNN-Based NF Estimation of Range. For range estimation, both SH-SIDID and SH-SIDIR features have been shown and proven above to possess the required information. The next step is the representation of the features for CNN [47] in such a way that the learning of the mapping function from the features to their respective range can be achieved. Due to the fact that the information of the source range is associated with mode strengths, $q_{nm}(kr_e)$ magnitude is given to be the input of the CNN. If N defines the SAA order under consideration and N_k represents the frequency bins, then, the feature set is in a rectangular form whose

dimension is $((N+1)^2-1)\times N_k$ as

$$\mathbf{Q}=\begin{bmatrix} |q_{-11}(k_1r_e)| & |q_{-11}(k_2r_e)| & \cdot\cdot & |q_{-11}(k_{N_k}r_e)| \\ |q_{01}(k_1r_e)| & |q_{01}(k_2r_e)| & \cdot\cdot & |q_{01}(k_{N_k}r_e)| \\ \cdot\cdot & \cdot\cdot & \cdot\cdot & \cdot\cdot \\ |q_{NN}(k_1r_e)| & |q_{NN}(k_2r_e)| & \cdot\cdot & |q_{NN}(k_{N_k}r_e)| \end{bmatrix}. \tag{26}$$

We apply log transformation to the features before feeding it into the CNN for estimation of range. The realizable source range is divided into a set of particular classes, and the function of the CNN is feature classification to a particular class via computation of the posterior probability $P_r(r_s \mid \mathbf{Q})$ of feature $\mathbf{Q}$ that belongs to a particular class of r_s.

4.7. The Architecture of the CNN. We adapted CNN to estimate range via SH-SIDID or SIDIR feature mapping to the range. For the purpose of classification, the source range is divided into many classes. There are seven classes with 10, 15, ⋯, 40 cm labels. The network is composed of nine convolutional layers, followed by two fully connected layers. The size and depth of strides present in the convolutional layers are 3×50 and 4, respectively. The activation function used for the whole layer is rectified linear unit (ReLU) [55, 60] except for the last layer where the softmax function [61] is employed to obtain the posterior probabilities. We ensured flattened convolutional layers' output prior feeding them into the fully connected layer (the size is 128). The number of the output layer is the same as that of classes, 7 in this scenario. Overfitting is prevented by using 20% dropout, early stopping with 20 epochs patience level, and validation.

4.8. Range Estimation. Employing the rule of maximum posteriori probability, the range can be computed using

$$\hat{r}_s=\arg\max_{r_s} P_r(r_s \mid \mathbf{Q}). \tag{27}$$

The range estimates from one frame are not very adequate because the frame may be noisy or has no signal. Therefore, signal activity detection (SAD) is adapted. In [62], Kim and Hahn developed a signal detection scheme via an adaptive context attention model. The model employs context information with the time of waves and deep neural networks. The multiresolution cochleagram-dependent features are provided and used to train the model, and the outputs of the trained model are the signal probability, where we obtain the hard decision label using a threshold. Therefore, SAD is employed to locate the silence frames as well as total range estimates from the signal frames.

5. Proposed Framework

This section presents the outline or summary of the whole process involved in the localization of NF source, particularly DoA and range estimation. The block diagram that encapsulates the entire process is as depicted in Figure 2. We used the inverse SFT technique to estimate DoA. SI SH features are obtained from SH coefficients. From this DoA estimate, the extraction of the SH-SIDID or SH-SIDIR features is performed. Then, the training of CNN is conducted using these features and the labels that correspond to it during training. During testing, the features are fed into the CNN as input to estimate the range successfully. Employing the appropriate frames, we performed posterior probability aggregation, which consequently gives a better range estimation. The step-by-step procedure required to conduct these task is as systematically depicted in Algorithm 1. The requirements of the algorithm are the observations $p(k, r_e, \theta, \phi)$, a trained SAD model, and trained models with SH-SIDID or SH-SIDIR.

6. Experiment, Performance Evaluation, Results, and Discussion

Some works deal with NF source localization of signals using antenna arrays, such as [1] and [45, 47–50], but none has considered SAA and the proposed method. EM signal measurement is conducted, and implementation of the algorithm is performed using MATLAB_R2018b software. We evaluated the performance of source localization by calculating the RMSE of range and DoA estimation experiment using the acquired data from SAA. A statistical analysis of SH features is equally conducted. We adopt an objective evaluation method to evaluate the outputs.

6.1. Experiment. We conducted an experiment to obtain data using an SAA [55, 63] in a similar way to our previous work reported in [55]. The SAA is positioned at the centre, and the source is situated at different directions and range using the centre of the SAA as reference. The elevation and azimuth of the source are fixed at 90° and 0 to 360°, with a step size of 90°, respectively. The range used is between 10 cm and 40 cm, with a step size of 5 cm. EM waves emitted by an antenna were used as source signals. The performance evaluation of the proposed methods for range estimation and NF DoA estimation is performed. The comparison of performance between SH-MUSIC, SHR, SRP-PHAT, and respective proposed methods for DoA and range estimation is as summarized in Table 1. It can be seen that the proposed methods performed better than SH-MUSIC4 and SRP-PHAT in the estimation of DoA and the SHR in the estimation of range.

6.2. NF DoA Estimation. The results for NF DoA estimation are given in this subsection. The performance evaluation is conducted using RMSE approach, and it is given below.

$$\mathrm{RMSE}\left[^0\right]=\sqrt{\frac{1}{G}\sum_{g=1}^{G}\left[\left(\phi_g-\phi\wedge_g\right)^2+\left(\theta_g-\theta\wedge_g\right)^2\right]}, \tag{28}$$

where (θ_g, ϕ_g) denotes the elevation and azimuth angles of g th sample and $(\widehat{\theta}_g, \widehat{\phi}_g)$ represents the corresponding estimates. G is the total samples.

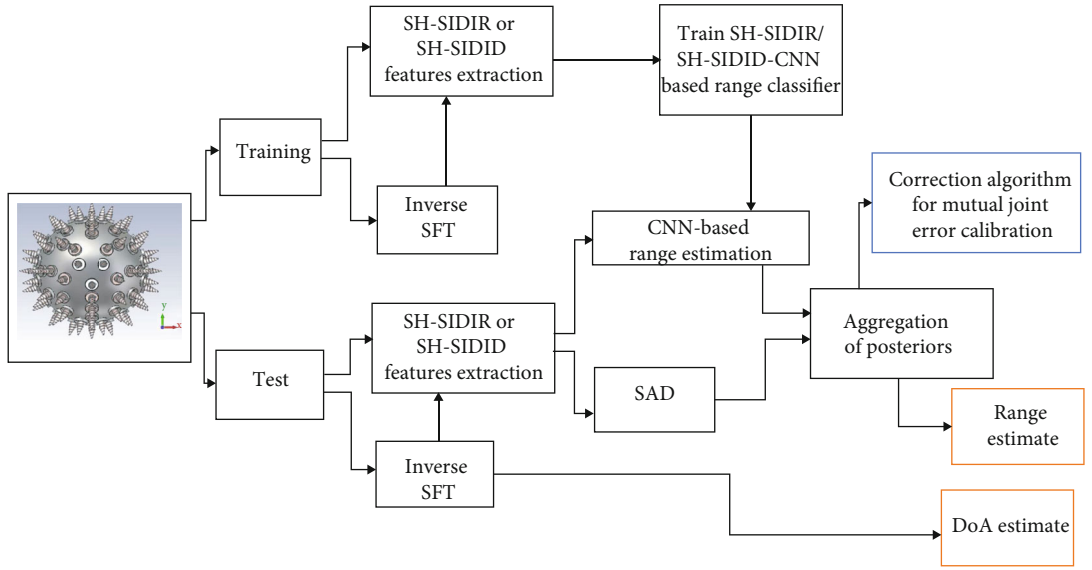


Figure 2: Procedure for NF localization of EM signals using SAA.

Requirement: $\boldsymbol{p}(k, r_e, \theta, \phi)$ observations, trained SAD model, and trained models for SH-SIDID/SH-SIDIR features mapped to range source.

(a) Spatial domain transformation to SH domain using

$$\boldsymbol{p}_{nm}(kr_e) = \boldsymbol{Y}^{\boldsymbol{H}}(\boldsymbol{\Omega})p(k, r_e, \theta, \phi)$$

(b) Using SFT to perform interpolation on the sphere

$$p(k, r_e, \theta_e, \phi_e) = \sum_{n=0}^{N}\sum_{m=-n}^{n} p_{nm}(kr_e)Y_n^m(\theta_e, \phi_e)$$

$$\forall(\theta_e, \phi_e)\, \epsilon\, Z^2$$

(c) Perform near-field DoA estimation from

$$(\widehat{\theta}_s, \widehat{\phi}_s) = arg\max_{(\theta_e,\phi_e)}|p(k, r_e, \theta_e, \phi_e)|^2$$

(d) Extract SI SH features using

$$\tilde{\boldsymbol{p}}_{\boldsymbol{nm}}(kr_e) = \frac{\boldsymbol{p}_{nm}(kr_e)}{p_{00}(kr_e)}$$

(e) Extract SH-SIDID or SH-SIDIR features from

$q_{nm}(kr_e) = \boldsymbol{D}(0, -\widehat{\theta}_s, -\widehat{\phi}_s)\tilde{\boldsymbol{p}}_{nm}(kr_e)$, or

$q_{nm}(kr_e) = \tilde{\boldsymbol{p}}_{nm}(kr_e) \ominus [\boldsymbol{Y}_n^m(\theta\wedge_s, \phi\wedge_s)]^*$ the parameters are defined in Eqns. (20) and (23)

(f) Feed the normalized $q_{nm}(kr_e)\forall k$ to the input of the trained CNN, which gives posterior probabilities $P_r(r_s \mid \boldsymbol{Q})$

(g) SAD posterior probabilities implying the presence of signal are calculated.

(h) $P_r(r_s \mid \boldsymbol{Q})$ aggregation is conducted via SAD posterior probabilities and range is consequently estimated.

(i) Obtain the complete NF source localization with the range and DoA estimates.

Algorithm 1: Proposed NF signal source localization procedure.

Table 1: Comparison of performance between the proposed method, SH-MUSIC, SRP-PHAT, and SHR on measured data.

DoA estimation Method	RMSE (°)	Range estimation Method	RMSE (cm)
Inverse SFT	15.2	SH-SIDIR-CNN	15.8
SRP-PHAT	51.3	SH-SIDID-CNN	13.4
SH-MUSIC	23.4	SHR	20.3

We made 105 test DoAs using various azimuth and seven angles of elevation, which are selected randomly within $(0, 360)^\circ$ and $(30, 150)^\circ$, respectively. Signal-to-noise ratio (SNR) is ranged between (0, 20) dB with 5 dB steps. AWGN is employed in the simulations. Fourier transform with 1024 points is utilized. In each resulting dataset, estimation of DoA is conducted by 20 adjacent frames with 100 ms frame length and overlap of 25%. The inverse SFT and SH-MUSIC methods are adapted using SH up to $N = 4$ order, while SRP-PHAT is adapted by data obtained from all the elements of SAA. The search grid is generated using

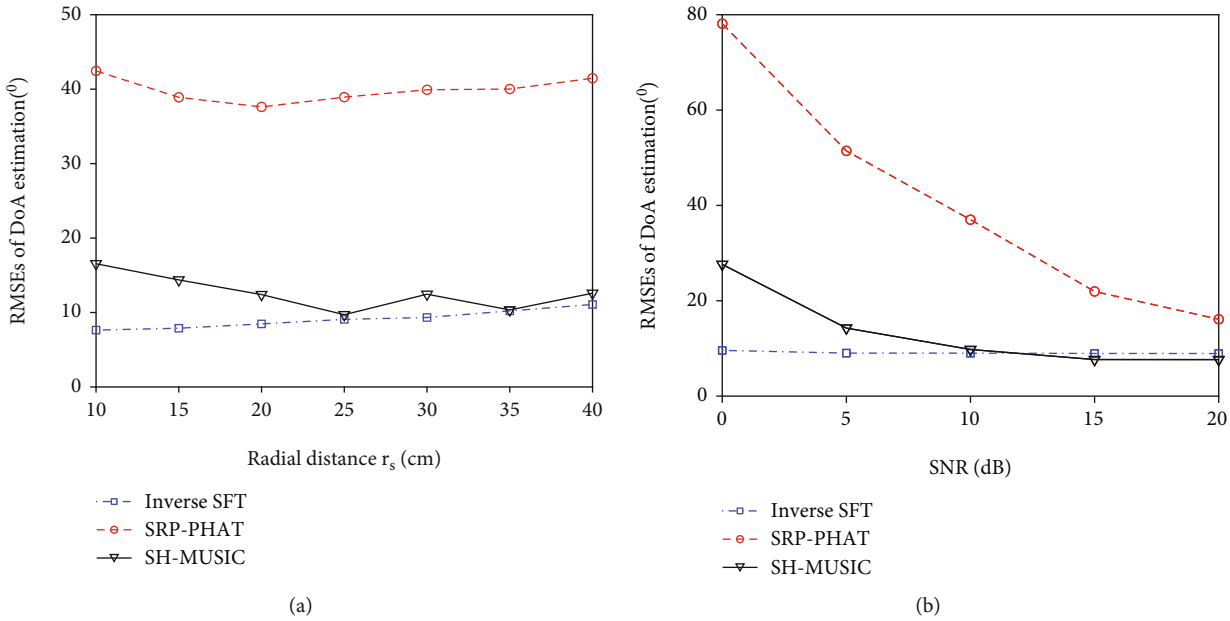


FIGURE 3: RMSE (°) of DoA estimation methods versus r_s (a) and SNR (b). The RMSE value is the average of all possible combinations of r_s and SNR.

TABLE 2: Obtained values of P_{mean} from KWT for the determination of DI and RD levels.

Requirement	Category	P_{mean}		
		SH-SIDIR	SH-SIDID	SHR
Independency	DoA	1.3082	1.8949	$4E-87$
Dependency	Range	0.7915	0.5497	0.2154

uniformly sampling azimuth (0, 360)° and elevation (30, 150)°, respectively. Consequently, we computed RMSE of all the methods for realizable combinations of SNR and r_s. The source distances of the SAA under consideration are proportionate to the critical distance that corresponds to EM conditions.

The comparison of performance obtained between the proposed inverse SFT for the estimation of DoA and previous methods (i.e., SRP-PHAT and SH-MUSIC) is as presented in Figure 3. It is revealed that the proposed inverse SFT method preforms better than SH-MUSIC and SRP-PHAT under all conditions. Also, SH-MUSIC performs better than SRP-PHAT in all cases. When noise is increased, the performance of SH-MUSIC degrades. This is because of the spurious peaks and distortion in the magnitude spectrum of SH-MUSIC in harsh environments. With high noise, there is little or negligible performance degradation in the proposed inverse SFT. With a change in r_s, both SH-MUSIC and SRP-PHAT have no particular pattern in performance. At the lower r_s, the proposed inverse SFT method performs much better, because there is a significant variation in the energy across the positions over the sphere. As r_s increases, the energy variation between various positions on the sphere reduces; hence, inverse SFT performs better in NF. Because of the noncoincidence nature of test DoAs with search grid, the least RMSE value is calculated using the angular distance existing between test DoAs and the proximate and respective DoA available on the search grid. The lowest RMSE of the selected search grid and test DoAs configuration is 6.45°. This discusses the RMSEs of the proposed method in different cases. It is observed that the proposed technique achieves RMSE that is close to the minimum under $r_s = 10$ cm, implying the better performance of the inverse SFT at lower r_s.

6.3. Statistical Discussion on SH Features. In this subsection, the discussion on SH-SIDID, SH-SIDIR, and SHR is presented using ANOVA (analysis of variance). We evaluated them under two categories: level of range dependence (RD) and level of DI. Specifically, we conducted a Shapiro-Wilk normality test [62] to know if the feature vectors are of normal distribution. Consequently, it was observed the feature vectors have no normal distribution. Therefore, a nonparametric ANOVA is performed employing a Kruskal-Wallis Test (KWT) [64, 65]. The KWT statistic H, as in [64], is usually approximated using chi-squared distribution [66]. Therefore, H is employed to calculate P value (i.e., the probability of having at least a test statistic at the extreme in a chi-squared distribution). Because the value of P is calculated for various order and degrees of SH, the mean value of P (P_{mean}) is then calculated. Hence, small P_{mean} value that falls under a significant level implies there is enough evidence for the rejection of the null hypothesis. Frequently, the 0.05 significant level is adapted as a benchmark between the significant and insignificant results.

Feature dependency of feature vectors is evaluated via KWT with the range as a categorical variable. The feature vectors are computed for different DoA and range combinations. The value of P_{mean} presented in Table 2 shows the values of various features obtained from KWT with the

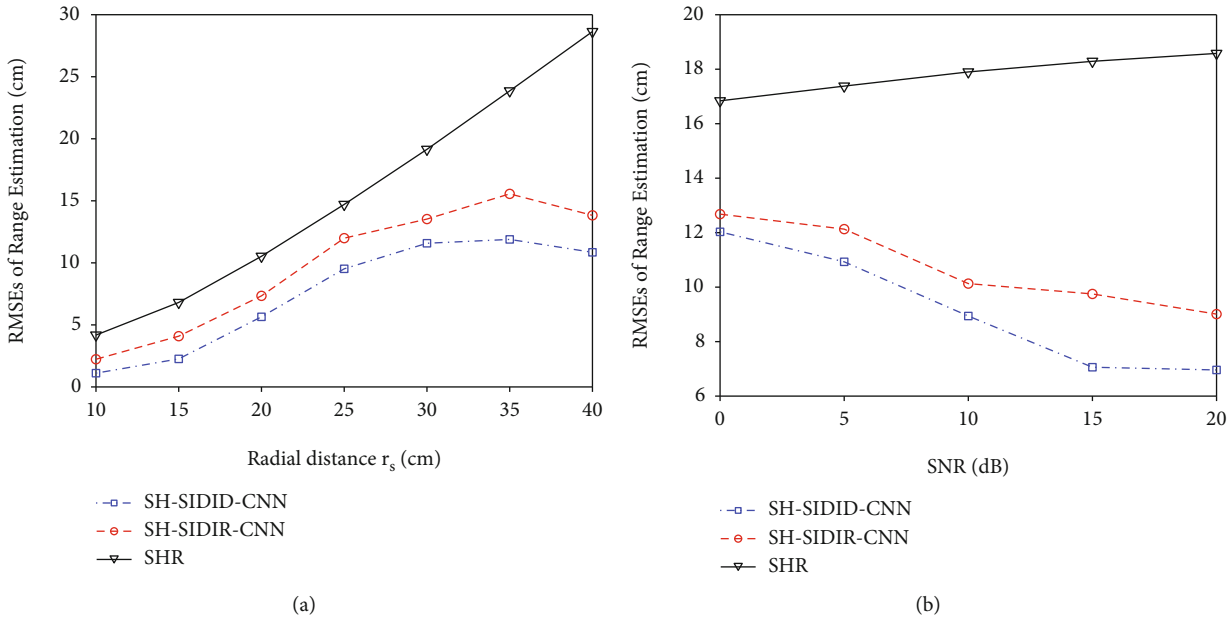


FIGURE 4: Range estimation methods versus r_s (a) and SNR (b).

categorical variable as a range. It is observed that SH-SIDIR and SH-SIDID features have appreciable statistical information, which makes them better choices for estimation of range.

DI of various feature vectors is evaluated via KWT using DoA as a categorical variable. The computation of the feature vector was performed for different combinations of DoA while keeping the range constant. Because of the requirement of DI, it becomes a desire to accept the null hypothesis. The values of P_{mean} obtained for different features from KWT is presented in Table 2. It is seen that for SH-SIDID and SH-SIDIR features, there is acceptance of the null hypothesis, while there is rejection in the SHR feature vector. Therefore, SH-SIDI and SH-SIDIR feature vectors do not depend on the DoA in a significant manner and are better choices for the estimation of range.

6.4. Range Estimation. This subsection gives the results and discussion of range estimation. The RMSE of range estimation can be computed using

$$\text{RMSE (cm)} = \sqrt{\frac{1}{G}\sum_{g=1}^{G}\left[\left(r_g - r\wedge_g\right)^2\right]}, \tag{29}$$

where r_l represents the source range of gth sample and $\hat{r}_g$ denotes the estimates corresponding to it.

Here, the condition of both training and testing is presented since the technique is learning dependent. For the training dataset, 200 SAA responses are taken by selecting DoA randomly and the size of the chamber for each r_s. There is a convolution of individual SAA response with 5 two-second durations of various EM signals with the random SNR to obtain five measurements. About 1024-point short time Fourier transform is employed. We randomly chose ten frames of 100 ms for feature extraction. Hence, the training dataset is made up of 70000 samples. For the test dataset, 50 SAA responses are taken with random DoA, chamber size, and particular condition for each r_s. There is a convolution of each SAA response with two EM signals to get two measurements, each of which is 3 sec. The 20 corresponding 100 ms with 25% overlap frames are then taken from individual measurement to get the aggregate. Hence, the test dataset is made up of 14000 test samples. We have 20 samples of that test datasets for the individual combination of SNR. The resulted features are normalized and fed into the CNN. The dataset for validation is generated similar to that of the training except that the responses from SAA are 20 for each r_s. Therefore, the validation dataset is 10% of the training dataset. Note that various seeding patterns for the generation of the random number are ensured in the training, validation, and test datasets. Therefore, there is no overlap in the experimental conditions of training, validation, and test datasets.

Comparison of performance between the proposed methods and SHR in range estimation is depicted in Figure 4. The RMSE of the varying r_s and SNR is given. It is shown how the proposed methods performed better than SHR under all scenarios. Small performance degradation is noticed in SH-SIDID-CNN and SH-SIDIR-CNN. Hence, the proposed methods show a higher level of robustness to harsh condition. It is important to note that noise has a notable effect on the performance of the proposed methods, but the performance of SHR remains almost constant. The degradation in performance of as r_s increases could be because NF mode strengths look closer when r_s increases. Therefore, it is not easy to differentiate between the r_s, which causes an increment in the value of RMSE as r_s increases.

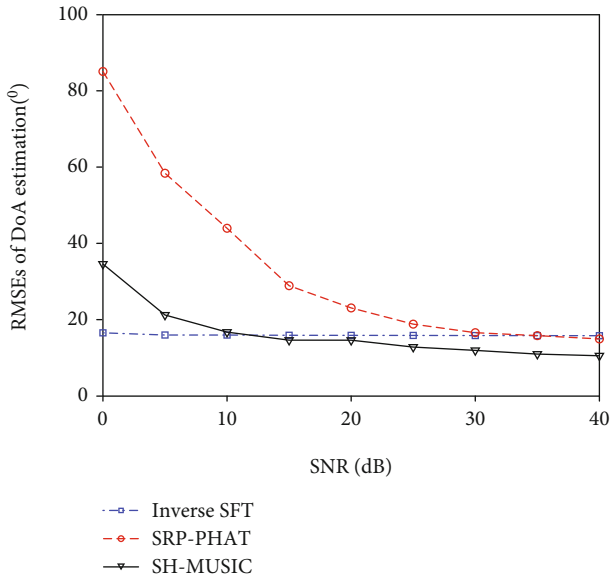


Figure 5: RMSE of DoA estimation methods versus SNR using measured data in [55].

Finally, fulfilling the growing application requirements, a specific number of elements are situated on the systems. The interelement distance is becoming shorter, causing strong mutual coupling with poor radiation performance and impedance matching. In order to incorporate the mutual coupling effect, which exist between elements, experimental measured data, which is the ground truth to systematically evaluate any procedure, is used. Therefore, experimental measurement data are further used for performance evaluation and analysis. The SAA is positioned in the middle of the chamber, and the source is situated at 74 DoAs, which are gotten from different combinations of 4 different elevations and 18 different azimuths. We selected the azimuths from 5 degrees to 365 degrees with 20 degrees as step size. For detailed information on measurement architecture/setup using SAA, readers are referred to the previous paper [55] where the measured data were first published. Far-field to near-field transformation is conducted on the data to conform with the current discussion. The comparison of performance between inverse SFT, SRP-PHAT, and SH-MUSIC was conducted. RMSE of DoA estimation methods versus SNR is plotted, and results are as presented in Figure 5. The inverse SFT exhibits greater performance than SH-MUSIC and SRP-PHAT, even in the presence of unknown mutual coupling. Furthermore, SH-MUSIC exhibits better performance than the SRP-PHAT in all scenarios. When there is increase in noise, SH-MUSIC degrades in performance. This can be attributed to the spurious peaks and distortion in the magnitude spectrum of SH-MUSIC when noise is high. When noise is high, there is negligible or little performance degradation in the proposed inverse SFT in the presence of mutual coupling.

7. Conclusion

In conclusion, a procedure for source localization of EM waves in the NF of an SAA in the presence of unknown MC using SH features has been presented. Specifically, an approach for the estimation of NF DoA by spatial pressure interpolation through inverse SFT is proposed. Through the sampling of signal pressure at some positions on the sphere, the interpolation of signal pressure at an observation point on the sphere is facilitated using inverse SFT. The proposed procedure made use of this inverse SFT characteristic for the estimation of NF DoA. The SH features, which are SIDI, were extracted from the DoA estimate. The SI is extracted via normalization of SH coefficients with a part corresponding to source signal strength. The decomposition of NF steering vector into DoA-dependent and mode strength component is facilitated by SH decomposition. With this decomposition and DoA estimates from inverse SFT technique, we obtained the DI characteristic by two techniques. In the first technique, the SH rotation technique is used to extract the feature. In the second technique, there is a division of SH coefficients by corresponding SH DoA estimate. Because there is variation in the information of the range per frequency and features obtained are impaired due to harsh environment, a CNN is adapted to map the SH features with the range of the NF source. Finally, performance evaluation was conducted using numerical simulations and estimations from measured data, and the results obtained show that the proposed procedure exhibits appreciable performance than the previous methods, even under harsh condition. These motivating results make the proposed procedure appropriate in practical cases. The authors are currently investigating the problem of multisource localization of EM waves in the NF of SAA, and this study is planned to be presented and reported in a separate follow-up article.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This research is supported in part by the University Research Committee (URC) of the University of Johannesburg, South Africa.

References

[1] B. Friedlander, "Localization of signals in the near-field of an antenna array," *IEEE Transactions on Signal Processing*, vol. 67, no. 15, pp. 2901–2907, 2019.

[2] H. Krim and M. Viberg, "Two decades of array signal processing research: the parametric approach," *IEEE Signal Processing Magazine*, vol. 13, no. 4, pp. 67–94, 1996.

[3] H. L. V. Trees, *Detection, Estimation, and Modulation Theory, Optimum Array Processing*, Wiley, Hoboken, NJ, USA, 2004.

[4] D. E. D. D. H. Johnson, *Array Signal Processing—Concepts and Techniques*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1993.

[5] E. Grosicki, K. Abed-Meraim, and Y. Hua, "A weighted linear prediction method for near-field source localization," *IEEE Transactions on Signal Processing*, vol. 53, no. 10, pp. 3651–3660, 2005.

[6] Y. D. Huang and M. Barkat, "Near-field multiple source localization by passive sensor array," *IEEE Transactions on Antennas and Propagation*, vol. 39, no. 7, pp. 968–975, 1991.

[7] B. P. Kumar, C. Kumar, V. S. Kumar, and V. V. Srinivasan, "Reliability considerations of spherical phased array antenna for satellites," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 54, no. 3, pp. 1381–1391, 2018.

[8] M. N. E. Korso, R. Boyer, A. Renaux, and S. Marcos, "Conditional and unconditional Cramer-Rao bounds for near-field source localization," *IEEE Transactions on Signal Processing*, vol. 58, no. 5, pp. 2901–2907, 2010.

[9] O. J. Famoriji, K. Akingbade, E. Ogunti, W. Apena, A. Fadamiro, and F. Lin, "Analysis of phased array antenna system via spherical harmonics decomposition," *IET Communications*, vol. 13, no. 18, pp. 5434–5454, 2019.

[10] G. Liu and X. Sun, "Efficient method of passive localization for mixed far-field and near-field sources," *IEEE Antennas and Wireless Propagation Letters*, vol. 12, pp. 902–905, 2013.

[11] I. Agrawal and R. M. Hegde, "Radial filters for near field source separation in spherical harmonic domain," in *Proceedings 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 116–120, Shanghai, China, 2016.

[12] E. Fisher and B. Rafaely, "Near-field spherical microphone array processing with radial filtering," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 2, pp. 256–265, 2011.

[13] J. Meyer and G. W. Elko, "Position independent close-talking micro-phone," *Signal Processing*, vol. 86, no. 6, pp. 1254–1259, 2006.

[14] L. Kumar and R. M. Hegde, "Near-field acoustic source localization and beamforming in spherical harmonics domain," *IEEE Transactions on Signal Processing*, vol. 64, no. 13, pp. 3351–3361, 2016.

[15] X. Wu and J. Yan, "3-D mixed far-field and near-field source localization," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 6, pp. 6833–6837, 2020.

[16] T. Shu, J. He, and V. Dakulagi, "3-D near-field source localization using a spatially spread acoustic vector sensor," *IEEE Transactions on Aerospace and Electronic Systems*, 2021.

[17] Z. Huang, B. Xue, W. Wang, F. Dong, and D. Wang, "A low complexity localization algorithm for mixed far-field and near-field sources," *IEEE Communication Letters*, 2021.

[18] Z. Huang, W. Wang, F. Dong, and D. Wang, "A one-snapshot localization algorithm for mixed far-field and near-field sources," *IEEE Communication Letters*, vol. 24, no. 5, pp. 1010–1014, 2020.

[19] Y. Kuznetsov, A. Baev, M. Konovalyuk, A. Gorbunova, and J. A. Russer, "Autocorrelation analysis and near-field localization of the radiating sources with cyclostationary properties," *IEEE Transactions on Electromagnetic Compatibility*, vol. 62, no. 5, pp. 2186–2195, 2020.

[20] X. Wu and J. Yan, "Gridless mixed sources localization based on low-rank matrix reconstruction," *IEEE Communication Letters*, vol. 9, no. 10, pp. 1748–1753, 2020.

[21] C. Guanghui, Z. Xiaoping, J. Shuang, Y. Anning, and L. Qi, "High accuracy near-field localization algorithm at low SNR using fouth-order cumulant," *IEEE Communication Letters*, vol. 24, no. 3, pp. 553–557, 2020.

[22] J. He, L. Li, and T. Shu, "Localization of near-field sources for exact source-sensor spatial geometry," *IEEE Signal Processing Letters*, vol. 27, pp. 1040–1044, 2020.

[23] H. Ma, H. Tao, and H. Kang, "Mixed far-field and near-field source localization using a linear electromagnetic-vector-sensor array with gain/phase uncertainties," *IEEE Access*, vol. 9, pp. 132412–132428, 2021.

[24] H. Ma, H. Tao, and J. Xie, "Mixed far-field and near-field source localization using a linear tripole array," *IEEE Communication Letters*, vol. 9, no. 6, pp. 889–893, 2020.

[25] J. He, L. Li, T. Shu, and T. Truong, "Mixed near-field and far-field source localization based on exact spatial propagation geometry," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 4, pp. 3540–3552, 2021.

[26] T. Shu, L. Li, and J. He, "Near-field localization for non-circular sources in the presence of sensor phase uncertainties," *IEEE Communication Letters*, vol. 10, no. 3, pp. 562–566, 2021.

[27] W. Zhi and M. Y. Chia, "Near-field source localization via symmetry subarray," *IEEE Signal Processing Letters*, vol. 14, pp. 409–413, 2007.

[28] T. Shu, L. Li, and J. He, "Near-field source localization with two-level nested arrays," *IEEE Communication Letters*, vol. 24, no. 11, pp. 2488–2493, 2020.

[29] W. Zuo, J. Xin, N. Zheng, H. Ohmori, and A. Sano, "Subspace-based near-field source localization in unknown spatially non-uniform noise environment," *IEEE Transactions on Signal Processing*, vol. 68, pp. 4713–4726, 2020.

[30] C. Knapp and G. Carter, "The generalized correlation method for estimation of time delay," *IEEE transactions on acoustics, speech, and signal processing*, vol. ASSP-24, no. 4, pp. 320–327, 1976.

[31] M. S. Brandstein and H. F. Silverman, "A robust method for speech signal time-delay estimation in reverberant rooms," in *Proceedings 1997 IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 375–378, Munich, Germany, 1997.

[32] A. Johansson, G. Cook, and S. Nordholm, "Acoustic direction of arrival estimation, a comparison between root-music and SRP-PHAT," in *Proceedings IEEE TENCON*, pp. 629–632, Chiang Mai, Thailand, November 2004.

[33] L. Kumar, A. Tripathy, and R. M. Hegde, "Robust multisource localization over planar arrays using music-group delay spectrum," *IEEE Transactions on Signal Processing*, vol. 62, no. 17, pp. 4627–4636, 2014.

[34] V. Varanasi and R. Hegde, "Robust online direction of arrival estimation using low dimensional spherical harmonic features," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 511–515, New Orleans, LA, USA, 2017.

[35] Y. Huang, J. Benesty, and G. W. Elko, "Adaptive eigenvalue decomposition algorithm for real time acoustic source localization system," in *Proceedings International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, pp. 937–940, Phoenix, AZ, USA, 1999.

[36] C. Zhang, D. Florencio, and Z. Zhang, "Why does PHAT work well in low noise, reverberative environments?," in *Proceedings*

IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 2565–2568, Las Vegas, NV, USA, 2008.

[37] E. Fisher and B. Rafaely, "The nearfield spherical microphone array," in *Proceedings IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 5272–5275, Las Vegas, NV, USA, 2008.

[38] Q. Huang and T. Chen, "One-dimensional MUSIC-type algorithm for spherical microphone arrays," *IEEE Access*, vol. 8, pp. 28178–28187, 2020.

[39] W. Gao and H. Chen, "An order-aware scheme for robust direction of arrival estimation in the spherical harmonic domain," *The Journal of the Acoustical Society of America*, vol. 146, no. 6, p. 4883, 2019.

[40] Y. R. Zheng, R. A. Goubran, and M. El-Tanany, "Robust near-field adaptive beamforming with distance discrimination," *IEEE Transactions on Speech and Audio Processing*, vol. 12, no. 5, pp. 478–488, 2004.

[41] F. Jacobsen, G. Moreno-Pescador, E. Fernandez-Grande, and J. Hald, "Near field acoustic holography with microphones on a rigid sphere (L)," *The Journal of the Acoustical Society of America*, vol. 129, no. 6, pp. 3461–3464, 2011.

[42] B. Rafaely, "Spatial sampling and beamforming for spherical microphone arrays," in *Proceedings Hands-Free Speech Communication Microphone Arrays*, pp. 5–8, Trento, Italy, 2008.

[43] S. Nakamura, "Acoustic sound database collected for hands-free speech recognition and sound scene understanding," in *Proceedings International Workshop on Hands-Free Speech Communication*, pp. 43–46, Baixas, France, 2001.

[44] R. Takeda and K. Komatani, "Sound source localization based on deep neural networks with directional activate function exploiting phase information," in *Proceedings IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 405–409, Shanghai, China, 2016.

[45] J. He, M. O. Ahmad, and M. N. S. Swamy, "Near-field localization of partially polarized sources with a cross-dipole array," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 2, pp. 857–870, 2013.

[46] N. Guzey, H. Xu, and S. Jagannathan, "Localization of near-field radio controlled unintended emitting sources in the presence of multipath fading," *IEEE Transactions on Instrumentation and Measurement*, vol. 63, no. 11, pp. 2696–2707, 2014.

[47] Z. Zheng, M. Fu, W. Wang, S. Zhang, and Y. Liao, "Localization of mixed near-field and far-field sources using symmetric double-nested arrays," *IEEE Transactions on Antennas and Propagation*, vol. 67, no. 11, pp. 7059–7071, 2019.

[48] D. Sarkar, S. M. Mikki, and Y. M. M. Antar, "Poynting localized energy: method and applications to gain enhancement in coupled antenna systems," *IEEE Transactions on Antennas and Propagation*, vol. 68, no. 5, pp. 3978–3989, 2020.

[49] G. Liu and X. Sun, "Efficient method of passive localization for mixed far-field and near-field sources," *IEEE Antennas and Wireless Propagation Letters*, vol. 12, pp. 902–905, 2013.

[50] Z. Zheng, M. Fu, W.-Q. Wang, and H. C. So, "Symmetric displaced coprime array configurations for mixed near and far field source localization," *IEEE Transactions on Antennas and Propagation*, vol. 69, no. 1, pp. 465–477, 2021.

[51] E. Fisher and B. Rafaely, "Dolph-Chebyshev radial filter for the near-field spherical microphone array," in *2009 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 169–172, New Paltz, NY, USA, 2009.

[52] H. Teutsch and G. W. Elko, "An adaptive close-talking microphone array," in *Proceedings of the 2001 IEEE Workshop on the Applications of Signal Processing to Audio and Acoustics*, pp. 163–166, New Paltz, New York, 2001.

[53] O. Koutny, J.-H. T. Jiricek, and M. Brothanek, "Source distance determination based on the spherical harmonics," *Mechanical Systems and Signal Processing*, vol. 85, pp. 993–1004, 2017.

[54] V. Varanasi, A. Agarwal, and R. M. Hegde, "Near-field acoustic source localization using spherical harmonic features," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 12, pp. 2054–2066, 2019.

[55] O. J. Famoriji, O. Y. Ogundepo, and X. Qi, "An intelligent deep learning-based direction-of-arrival estimation scheme using spherical antenna array with unknown mutual coupling," *IEEE Access*, vol. 8, pp. 179259–179272, 2020.

[56] G. Arfken and H. Weber, *Mathematical Methods for Physicists*, Elsevier, New York, NY, USA, 6th edition, 2005.

[57] M. Agmon and B. Rafaely, "Beamforming for a spherical-aperture microphone," in *2008 IEEE 25th Convention of Electrical and Electronics Engineers in Israel*, pp. 227–230, Eilat, Israel, 2008.

[58] B. Rafaely, "Analysis and design of spherical microphone arrays," *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 1, pp. 135–143, 2005.

[59] S. Chakrabarty and E. A. P. Habets, "Broadband DOA estimation using convolutional neural networks trained with noise signals," in *Proceedings IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA).*, pp. 136–140, New Paltz, NY, USA, 2017.

[60] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems.*, pp. 1097–1105, 2012.

[61] C. M. Bishop, *Neural Networks for Pattern Recognition*, Oxford Univ. Press, London, U.K., 1995.

[62] J. Kim and M. Hahn, "Voice activity detection using an adaptive context attention model," *IEEE Signal Processing Letters*, vol. 25, no. 8, pp. 1181–1185, 2018.

[63] P. Kumar, C. Kumar, S. Kumar, and V. Srinivasan, "Active spherical phased array design for satellite payload data transmission," *IEEE Trans. Antennas Propagat.*, vol. 63, no. 11, pp. 4783–4791, 2015.

[64] S. S. Shapiro and M. B. Wilk, "An analysis of variance test for normality (complete samples)," *Biometrika*, vol. 52, no. 3/4, pp. 591–611, 1965.

[65] W. H. Kruskal and W. A. Wallis, "Use of ranks in one-criterion variance analysis," *Journal of the American Statistical Association*, vol. 47, no. 260, pp. 583–621, 1952.

[66] B. Lantz, "The impact of sample non-normality on ANOVA and alternative methods," *British Journal of Mathematical and Statistical Psychology*, vol. 66, no. 2, pp. 224–244, 2013.

[67] H. Aoyama, "Mutual coupling matrix estimation and null forming methods for MBF antennas," *IEICE Transactions on Communications*, vol. E88-B, no. 6, pp. 2305–2312, 2005.

[68] Y. Peled and B. Rafaely, "Method for dereverberation and noise reduction using spherical microphone arrays," in *IEEE*

International Conference on Acoustics, Speech and Signal Processing, pp. 113–116, Dallas, TX, USA, 2010.

[69] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[70] R. V. Hogg and J. Ledolter, *Engineering Statistics*, Macmillan, New York, NY, USA, 1987.

Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 2835638, 8 pages
https://doi.org/10.1155/2021/2835638

Research Article

Reconfigurable Reflectarray Antenna: A Comparison between Design Using PIN Diodes and Liquid Crystals

Muhammad Inam Abbasi,[1] **Muhammad Yusof Ismail**,[2] **Muhammad Ramlee Kamarudin**,[2] **and Qammer H. Abbasi**[3]

[1]*Centre for Telecommunication Research & Innovation (CETRI), Faculty of Electrical and Electronic Engineering Technology (FTKEE), Universiti Teknikal Malaysia Melaka (UTeM), Melaka 76100, Malaysia*
[2]*Faculty of Electrical and Electronic Engineering (FKEE), Universiti Tun Hussein Onn Malaysia (UTHM), 86400 Batu Pahat, Johor, Malaysia*
[3]*James Watt School of Engineering, College of Science and Engineering, University of Glasgow, Glasgow, UK*

Correspondence should be addressed to Muhammad Inam Abbasi; muhammad_inamabbasi@yahoo.com

Received 27 July 2021; Revised 24 September 2021; Accepted 15 October 2021; Published 29 October 2021

Academic Editor: Farman Ullah

This work presents the design and analysis of active reflectarray antennas with slot embedded patch element configurations within an *X*-band frequency range. Two active reflectarray design technologies have been proposed by digital frequency switching using PIN diodes and analogue frequency tuning using liquid crystal-based substrates. A waveguide simulator has been used to perform scattering parameter measurements in order to practically compare the performance of reflectarray designed based on the two active design technologies. PIN diode-based active reflectarray unit cell design is shown to offer a frequency tunability of 0.36 GHz with a dynamic phase range of 226°. On the other hand, liquid crystal-based design provided slightly lower frequency tunability of 0.20 GHz with a dynamic phase range of 124°. Moreover, the higher reflection loss and slow frequency tuning are demonstrated to be the disadvantages of liquid crystal-based designs as compared to PIN diode-based active reflectarray designs.

1. Introduction

A reflectarray, as suggested by the name, is a flat reflecting array of resonant patch elements that can be used for a number of applications where high gain antennas are required. Some of the main characteristics of a reflectarray antenna are its lower cost, lower mass, and smaller stowage volume, which is generally demanded in the spacecraft antennas in order to reduce payload weight and required shroud space to minimize overall launch cost. Conventionally, high-gain applications have counted on parabolic reflectors and phased arrays [1]. Nevertheless, due to the curvature of their surface, parabolic reflectors are challenging to be manufactured in many cases at higher microwave frequencies [2]. The shape of the parabolic reflector also causes an increase in the weight and size of the antenna. Moreover, it has also been established in [3] that wide-angle electronic beam scanning cannot be achieved using a parabolic reflector. On the other hand, high-gain phased array antennas offer the opportunity to electronically scan the main beam along wide-angle positions provided that they are equipped with controllable phase shifters. However, the main shortcoming of phased array antennas is their large hardware footprint, as each element of an array or subarray needs to be connected to a dedicated transceiver module. These modules are usually high profile, thus making phased array antennas a costly solution for high-gain applications.

Direct Broadcast Satellites (DBS) and Multibeam Antennas (MBA) are also considered potential applications of reflectarrays apart from recent investigations on their applicability in 5G communication systems [4–7]. Reflectarrays can also be used as amplifying arrays by adding an amplifier in each of the unit cells [8]. Despite many potentials, the main shortcomings of a reflectarray antenna are its limited

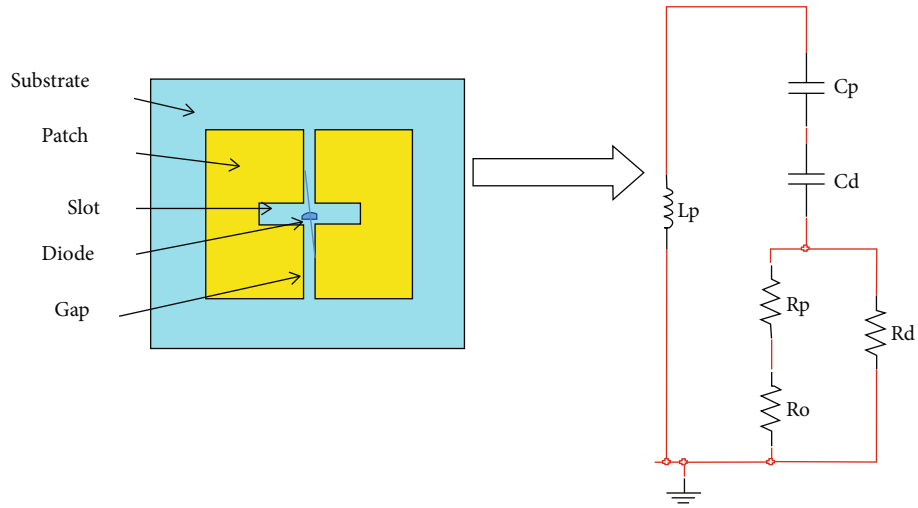


Figure 1: PIN diode-based unit cell with equivalent circuit representation.

bandwidth and high loss performance as compared to the parabolic reflector antennas [9–11]. Researchers have proposed a number of configurations in the past few years for the bandwidth and loss performance improvement of reflectarrays [12–16]. However, considerable efforts are still required to improve the bandwidth performance of reflectarrays. In order to steer the main beam of an active reflectarray, the reflected phase from each of the resonant elements can be controlled. Therefore, the reflected beam can be directed in the desired direction, which makes a reflectarray capable of achieving a wide-angle electronic beam scanning. Such a beamforming approach can have many advantages over traditional tunable antenna array architectures, including a significant reduction in hardware required per element and increased efficiency [17]. There has been considerable research in beam steering of reflectarray antennas such as the use of nonlinear dielectric materials [18–20] and the integration of Radio Frequency Micro-Electro-Mechanical Systems (RF MEMS) as switches [21, 22], using aperture-coupled elements where the tuning circuit can be located on the nonresonating surface of the element in order to control the contributed phase from each element [23] and using mechanical movement of the antenna [24].

In this work, slot embedded patch element configurations have been proposed for active reflectarray designs. PIN diodes have been proposed to be incorporated directly on the resonant elements for frequency switching in reflectarrays, while liquid crystals have been proposed in separate designs to be used as a substrate for tunable reflectarray design. The 3D EM computer simulation software results of CST MWS and Ansoft HFSS have been verified using waveguide scattering parameter measurements. Detailed comparisons between different performance parameters of the two design technologies are discussed in this work. Commercially available computer models of CST Microwave Studio and Ansoft HFSS were used to design unit cell patch elements with proper boundary conditions in order to analyze the scattering parameters of an infinite reflectarray. Initially, a reflectarray with a rectangular patch element was designed to resonate at 10 GHz using Rogers RT/Duroid 5880 ($\varepsilon r = 2.2$ and $\tan\delta = 0.0010$) as a substrate with a thickness of 0.3818 mm. Then, rectangular slot configurations are introduced in the patch element, and the effect on the performance of the reflectarray was observed. Reflectarray unit cells consisting of two patch elements were used for the waveguide scattering parameter measurements [25].

2. Frequency Switchable Reflectarray Design Using PIN Diodes

In this work, apart from the rectangular slot, a vertical gap was introduced in the resonant slot embedded patch element for the practical implementation of PIN diodes. This gap provides an option to connect the diode in a way that it can have different potentials at the two connecting ends. Moreover, the vertical gap does not affect the resonance performance of the unit cell as the maximum surface currents were observed to be along the width of the patch and slots.

PIN diodes were integrated in the gap introduced on the slot embedded patch element. Scattering parameter measurements were carried out for a unit cell that comprised two patch elements with dimensions of $L_p \times W_p = 9.4\,\text{mm} \times 10\,\text{mm}$ each, which were printed on a substrate of $L_s \times W_s = 15\,\text{mm} \times 30\,\text{mm}$ (L_p and W_p are the length and width of the patch element, respectively, while L_s and W_s are the length and width of the substrate, respectively). The slot length was kept at 0.6 mm, while the width was 0.5 W_p. The vertical gap was introduced with a 0.6 mm width in order to fit the PIN diode. Figure 1 shows the unit cell of the PIN diode-based planar reflector with its equivalent circuit representation. Equivalent circuits can also be used for the characterization of the planar reflector unit cells based on the lumped components. In this design, L_p, C_p, and R_p represent inductance, capacitance, and resistance of a passive planar reflector unit cell, respectively, while C_d, R_o, and R_d are used to represent the on and off states of the PIN diodes. For the electronic switching of PIN diode-based design, a GaAS MA4GP907 PIN diode manufactured by

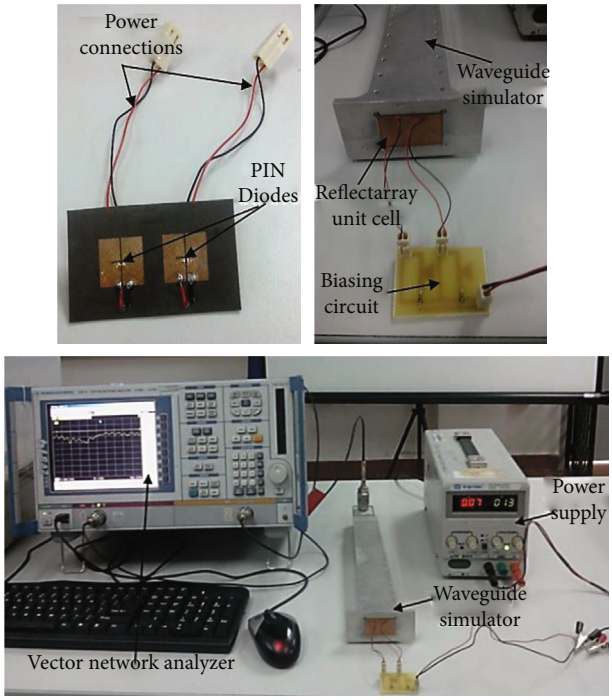


Figure 2: Fabricated unit cell, biasing circuit, and complete measurement setup.

MACOM was used. This PIN diode has a series capacitance of 0.025 pF and low series resistance of 4.2 Ω.

Figure 2 shows the fabricated unit cell, the biasing circuit connected with the unit cell, and the complete setup for scattering parameter measurements. As shown in Figure 2, the PIN diodes were soldered on the surface of the patch element and were powered by a power supply using a biasing circuit. A high-accuracy SMT fabrication facility was used to solder the PIN diodes on the resonant patch structure as accurately as possible. 1.33 V were supplied and a 100 Ω resistor was used to forward bias the diodes while no negative voltage (0 V) was required to reverse bias the diodes. RF choke was implemented using quarter-wavelength segments and radial stub on the biasing circuit. DC block capacitors were not required in this case because there is no physical connection between the RF source (network analyzer) and DC source (power supply).

Reflection loss and reflection phase were measured within an *X*-band frequency range, and a close agreement between measured and simulated results was observed. Figure 3 shows a comparison between measured and simulated reflection loss curves for fabricated samples. It can be observed from Figure 3 that in the off state of the PIN diode, the measured resonant frequency is close to the simulated resonant frequency. The fabricated unit cell resonated at 9.40 GHz with a reflection loss of 2.60 dB while the simulations for the off state of the PIN diode provided a resonant frequency of 9.38 GHz with 1.61 dB reflection loss. When the PIN diodes were switched on, a clear change in frequency was observed for the fabricated samples. In the on state, the measured resonant frequency was observed to be 9.04 GHz with a reflection loss of 3.91 dB. In comparison,

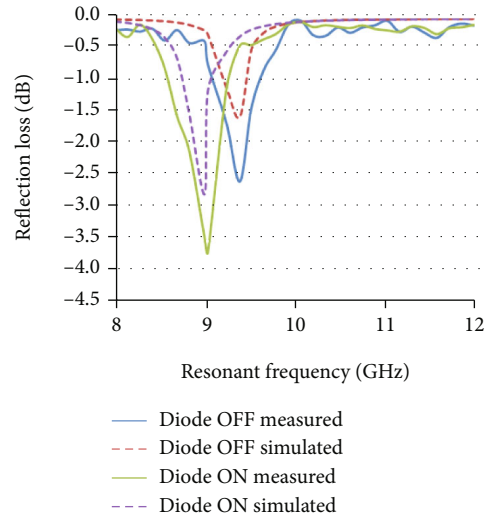


Figure 3: Comparison between simulated and measured reflection loss curves of PIN diode-based active reflectarrays.

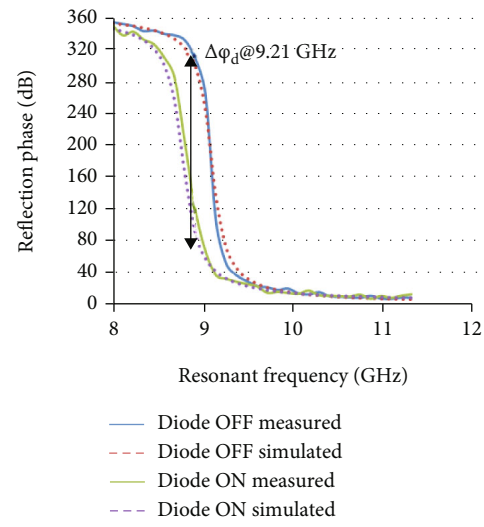


Figure 4: Comparison between simulated and measured reflection phase curves of PIN diode-based active reflectarrays.

the simulation results for the on state of the PIN diode exhibited a reflection loss of 2.88 dB at a resonant frequency of 8.99 GHz.

The highest discrepancy between measured and simulated reflection loss was observed to be 0.99 dB and 1.03 dB in off and on states of PIN diodes, respectively. Moreover, extra noise or ripples with a maximum level of 0.25 dB were observed. The reason for this discrepancy is due to fabrication quality, including the soldering of PIN diodes and the difference between actual material properties and the properties given in the datasheet. Furthermore, the diode was intentionally placed tilted in order to optimize the reflection loss and reflection phase results in simulations. The optimization was carried out keeping in mind the maximum current distribution on the surface of the patch. However, in measurements, it was not possible to place the diode at the

Table 1: Comparison between simulated and measured frequency tunability and dynamic phase range.

Results		Frequency tunability Fr (GHz)	Dynamic phase range $\Delta\varphi d$ (°)
Simulated	CST MWS	0.41	231
	Ansoft HFSS	0.39	228
Measured results		0.36	226

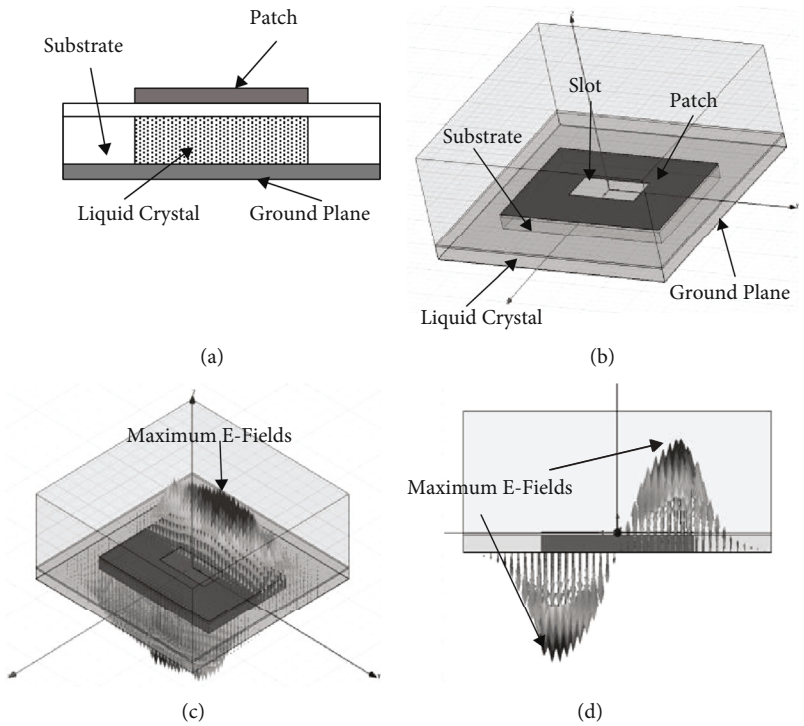


Figure 5: Active reflectarray unit cell design: (a) perspective view of a rectangular slot embedded patch element unit cell; (b) illustrative side view; (c) E-fields from a perspective view; (d) E-fields from a side view.

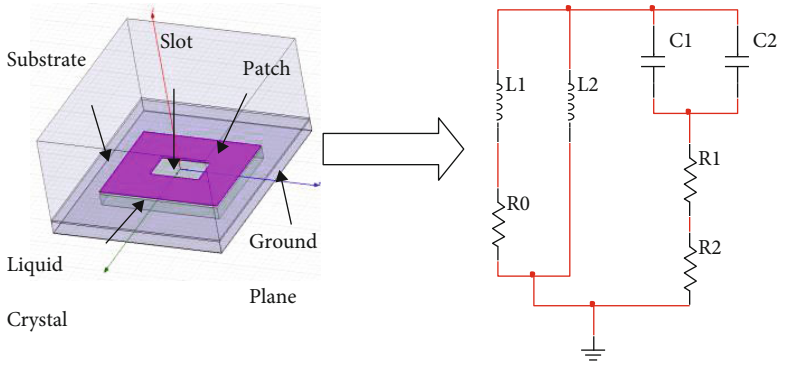
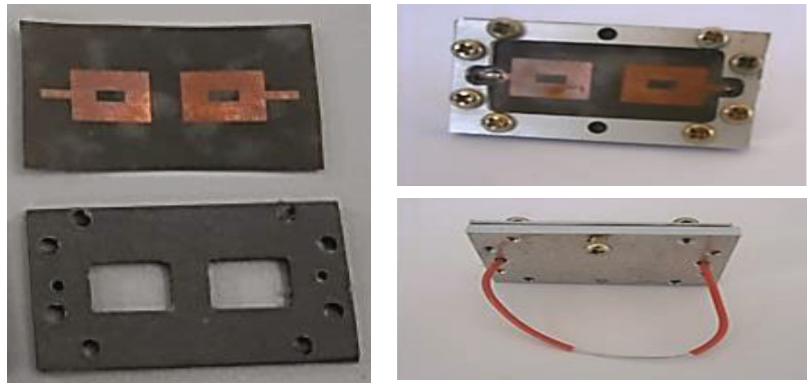


Figure 6: LC-based unit cell with equivalent circuit representation.

exact location as in simulations, because of the diode dimensions and fabrication complexity. The positioning of the diode can also be a possible reason to add up to the discrepancy in the results as shown in Figure 3. The discrepancy can be minimized with the more careful fabrication of unit cells and a thorough investigation of the actual material properties of the substrate used after going through the fabrication process. Figure 4 shows a comparison between the measured and simulated reflection phases. A close agreement between the measured and simulated phases can be observed from



(a) (b)

Figure 7: Fabricated samples: (a) unit cell patch element and LC cavity; (b) rectangular slot embedded path and back side of complete unit cell assembly in an encapsulator.

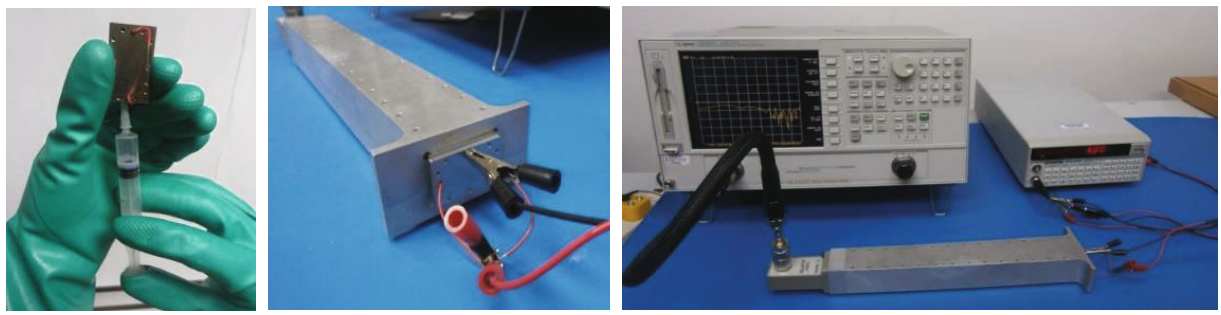


Figure 8: LC filling in a cavity, a unit cell inserted in a waveguide simulator, and complete measurement setup for scattering parameter measurements.

Figure 4, except for the ripples found towards the edges of the measured curves. These ripples can be linked to the same sources, which caused a discrepancy in the reflection loss curves. Table 1 provides a comparison between simulated and measured results of the frequency tunability and dynamic phase range. The dynamic phase range was calculated at the central frequency of two resonant curves in off and on states of PIN diodes as shown in Figure 4. It can be observed from Table 1 that a maximum frequency tunability of 0.36 GHz and a dynamic phase range of 226° were demonstrated by PIN diode base unit cell measurements. The results are in close agreement with the results obtained by 3D EM simulators of CST MWS and Ansoft HFSS, which practically validates the proposed design.

3. Reconfigurable Reflectarray Design Using Liquid Crystal Substrates

The change in the molecular orientation of liquid crystals (LC) can be done by applying a bias voltage [26, 27]. This change in molecular orientation gives rise to the dielectric anisotropy ($\Delta\varepsilon$) of LC, which makes them suitable to be used as a tunable dielectric substrate in reflectarrays. $\Delta\varepsilon$ can be explained as

$$\Delta\varepsilon = \varepsilon_{\parallel} - \varepsilon_{\perp}, \tag{1}$$

where $\varepsilon\perp$ and $\varepsilon\parallel$ are the magnitude of the dielectric constant measured perpendicular and parallel to the applied electric field. The reflection phase and resonant frequency of reflectarrays can be tuned for various values with the help of an external tunable bias voltage [16].

The basic design topology of a unit cell reflectarray with periodic boundary conditions has been used in Ansoft HFSS to represent a single patch element as an infinite array. The resonant patches, as shown in Figures 5(a) and 5(b), have been fabricated on a thin supporting layer of Rogers RT/Duroid 5880 with a substrate thickness of 0.127 mm. K-15 nematic LC has been deposited within a cavity made in the dielectric substrate of Rogers RT/Duroid 5880 with a substrate thickness of 0.787 mm backed by a ground plane. The dimensions of the resonant patch element are kept 8.4 mm × 11.8 mm ($L \times W$) for resonance within an X-band frequency range. It can be observed from Figures 5(c) and 5(d) that the E-fields are sinusoidally distributed with maxima at the corners of the resonant patch element. Therefore, the surface currents will be maximum in the centre of the patch element along the direction of field excitation (x-axis).

Figure 6 shows the equivalent circuit representation of an LC-based unit cell planar reflector design. Apart from the basic RLC circuit, extra inductance, capacitance, and resistance have to be considered because of the introduction of liquid crystal under the patch element within the solid substrate cavity.

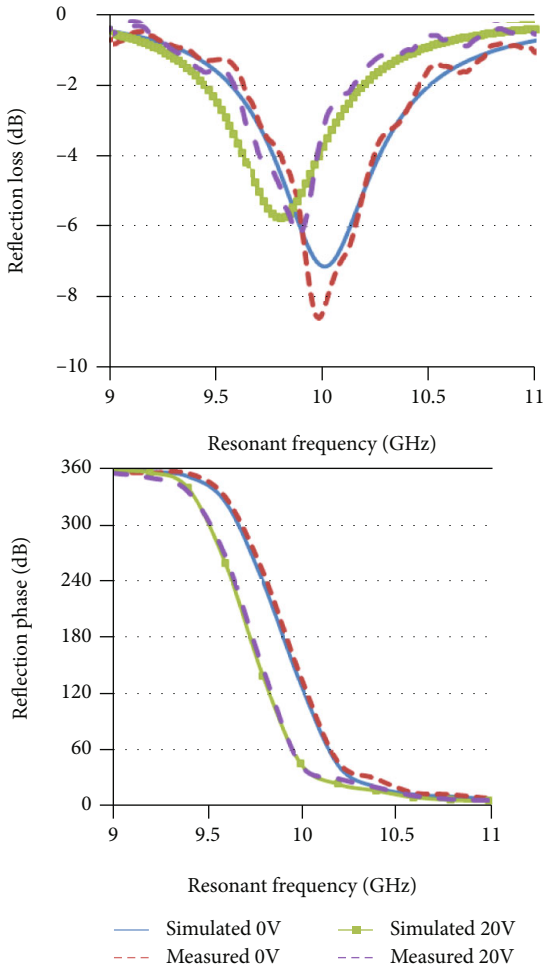


Figure 9: Comparison of measured and simulated results for a rectangular slot embedded patch element unit cell reflectarray.

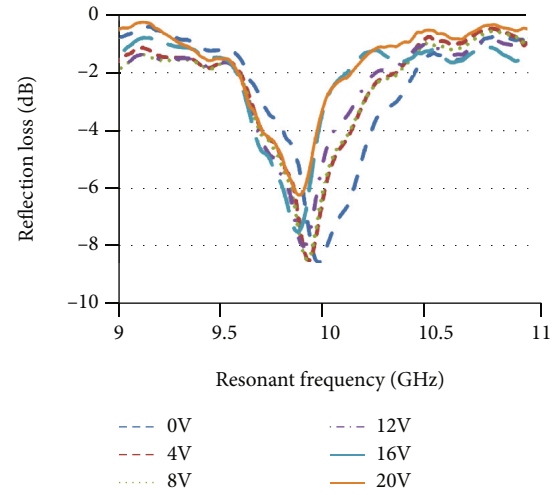


Figure 10: Measured reflection loss curves for different voltage levels applied to LC of unit cell designs.

Table 2: Measured performance comparison between the PIN diodes and liquid crystal-based planar reflectors at an *X*-band frequency range.

Technology	Frequency tunability (GHz)	Dynamic phase range ($\Delta\varphi d$) (°)	Static phase range ($\Delta\varphi s$) (°)	Tunable loss factor (dB)
PIN diodes	0.36	226	270	1.43
Liquid crystal	0.20	124	260	1.91

In order to design a frequency tunable reflectarray unit cell, the properties of K-15 nematic LC have been exploited. For this type of LC, a voltage variation from 0 V to 20 V can be applied to change the orientation of K-15 nematic LC molecules from perpendicular ($\varepsilon r = 2.7$ and $\tan\delta = 0.04$) to parallel ($\varepsilon r = 2.9$ and $\tan\delta = 0.03$). Different rectangular slot embedded unit cell patch elements have been fabricated for *X*-band frequency range operations, as shown in Figure 7(a). Encapsulations made of aluminium shown in Figure 7(b) have been used to keep intact different parts of unit cells, and a connecting wire has been used to electrically short the two patches in order to apply the desired voltage. Figure 8 shows the measurement procedure and LC filling inside the cavity constructed under the resonant patch element. The complete assembly of unit cell patch elements filled with LC has been inserted in the aperture of the waveguide, and scattering parameter measurements have been carried out using a waveguide simulator with a vector network analyzer while the voltage from 0 V to 20 V has been supplied by a function generator to the resonant patch elements.

The scattering parameter measurements of the LC-based rectangular slot embedded patch element unit cells have been carried out, and a comparison between simulated and measured results has been presented in Figure 9. The simulated and measured results provided a close agreement with a variation in measured resonant frequency from 10 GHz to 9.88 GHz with an increase in voltage from 0 V to 20 V. Moreover, a dynamic phase range of 103° measured from the reflection phase curve has been demonstrated by the proposed design of a reconfigurable LC-based unit cell.

In order to further investigate the proposed design, different voltage levels have been applied to K-15 nematic LC and the effect on reflection loss and resonant frequency has been observed, as shown in Figure 10. It can be observed that each increment in voltage level contributes to a small frequency tunability, which reaches to 180 MHz at 20 V level. It can also be observed from Figure 10 that the reflection loss also decreases from 8.5 dB at 10 GHz to 6.2 dB at 9.88 GHz with an increase in voltage from 0 to 20 V. The decrease in reflection loss is because of a decrease in loss tangent value of K-15 nematic LC material from 0.04 to 0.03 with an increase in voltage from 0 to 20 V.

4. Comparative Summary

Active planar reflectors using PIN diodes and liquid crystals provided interesting results for the design for switchable planar reflectors for beam shaping realization in an *X*-band

Table 3: General comparison between the PIN diodes and liquid crystal base planar reflector design ("+," "0," and "-" symbols refer to good, neutral, and poor, respectively).

Technology	Control	Complexity (cost)	Loss	Switching time
PIN diodes	Digital	0	+	+
Liquid crystal	Analogue	0	-	-

frequency range. The measured results for PIN diode embedded configurations provided a maximum frequency tunability of 0.36 GHz with a dynamic phase range of 226° as compared to those for liquid crystal-based cells, which demonstrated a maximum frequency tunability of 0.20 GHz with 124° of the dynamic phase range. Moreover, the static phase range, which is a measure of phase errors in a planar reflector design, provided a higher range of 270° in the case of PIN diode-based design as compared to the static phase range of 260° demonstrated by liquid crystal-based design. Although a range of frequency tunability options is achievable for liquid crystal-based cells by applying different voltage levels, huge time consumption between 102 S and 840 S is one of the main disadvantages of this design. On the other hand, PIN diode embedded unit cells can provide a fast frequency switching depending on the switching time of the PIN diodes. The diode used in this work (MACOM MA4GP907) has a switching time of 2-3 nS, which provides PIN diode embedded design and edge over liquid crystal-based planar reflector design.

Furthermore, the PIN diode embedded planar reflector unit cell exhibited a maximum measured reflection loss of 3.91 dB, which is much lesser as compared to the 8.56 dB of reflection loss observed for liquid crystal-based unit cell design. The tunable loss factor, which is measured as the difference of reflection loss between the two extreme tunable frequencies, is also higher (1.91 dB) in the case of liquid crystal-based design as compared to PIN diode-based design (1.43 dB). A higher tunable loss factor can be attributed to the properties of the liquid crystals used in the design. Table 2 provides the measured performance comparison for the planar reflector design using the two technologies.

Additionally, as far as the complexity of design is concerned, a uniform deposition layer is required in the case of LC-based design in order to achieve full anisotropy of LC molecules. Moreover, keeping the liquid crystal fully filled inside the cavity is a challenging task and requires a perfect design of the encapsulator. On the other hand, because of the tiny size of the PIN diodes, it was difficult to handle and solder the diodes on the resonant patch elements. However, professional skills and equipment can help to resolve these problems. A general summary of the outcomes of the two design techniques is provided in Table 3.

5. Conclusions

Slot embedded patch element configurations have been identified as a potential configuration for the improved design of passive and reconfigurable reflectarray antennas. The slot embedded patch elements also provide an extra parameter, which is the dimensions of the slot for the control of the resonant frequency and reflection phase of the reflectarray antenna. PIN diode-based design provides a number of advantages over liquid crystal-based design in terms of frequency tunability, dynamic phase range, static phase range, and tunable loss factor. However, liquid crystal-based design provides an edge over PIN diode-based active reflectarray design because of analogue control, which provides the option of tunability over a range of frequencies. It can be concluded from this investigation that there is a trade-off between the performance parameters and continuous tunability achieved by LC-based design. Further investigations are required to improve the frequency tunability and reflection loss performance of reconfigurable reflectarray antennas by investigating the properties of materials and the applied electronic components.

Data Availability

Data is available on request. The corresponding author can be contacted for any relevant data.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

Research funding for this work is fully provided by the Ministry of Higher Education, Malaysia, under Best Project of Fundamental Research Grant Scheme (FRGS, VOT 0983), Prototype Research Grant Scheme (PRGS, VOT 0904), and Research Acculturation Collaborative Effort (RACE, VOT 1119).

References

[1] G. D. G. Berry, R. G. Malech, and W. A. Kennedy, "The reflectarray antenna," *IEEE Transactions on Antennas and Propagation*, vol. 11, no. 6, pp. 645–651, 1963.

[2] J. Huang and J. A. Encinar, *Reflectarray Antennas*, vol. 30, John Wiley & Sons, New York, USA, 2007.

[3] J. A. Encinar, M. Arrebola, M. Dejus, and C. Jouve, "Design of a 1-metre reflectarray for DBS application with 15% bandwidth," in *First European Conference on Antennas and Propagation (EUCAP)*, pp. 1–5, Nice, France, 2006.

[4] M. H. Dahri, M. H. Jamaluddin, M. Inam, and M. R. Kamarudin, "A review of wideband reflectarray antennas for 5G communication systems," *IEEE Access*, vol. 5, pp. 17803–17815, 2017.

[5] M. H. Dahri, M. Inam, M. H. Jamaluddin, and M. R. Kamarudin, "A review of high gain and high efficiency reflectarrays for 5G communications," *IEEE Access*, vol. 6, no. 1, pp. 5973–5985, 2018.

[6] M. H. Dahri, M. H. Jamaluddin, M. Khalily, M. Inam, R. Selvaraju, and M. R. Kamarudin, "Polarization diversity and adaptive beamsteering for 5G reflectarrays: a review," *IEEE Access*, vol. 6, no. 1, pp. 19451–19464, 2018.

[7] M. Inam, M. H. Dahri, M. H. Jamaluddin, N. Seman, M. R. Kamarudin, and N. H. Sulaiman, "Design and characterization of millimeter wave reflectarrays for 5G communication

systems," *International Journal of RF and Microwave Computer Aided Engineering*, no. article e21804, 2019.

[8] M. E. Bialkowski, A. W. Robinson, and H. J. Song, "Design, development and testing of X-band amplifying reflectarrays," *IEEE Transactions on Antennas and Propagation*, vol. 50, no. 8, pp. 1065–1076, 2002.

[9] M. Y. Ismail and M. Inam, "Resonant elements for tunable reflectarray antenna design," *International Journal of Antennas and Propagation*, vol. 2012, 6 pages, 2012.

[10] M. E. Biallowski and J. Encinar, "Reflectarray: potential and challenges," *International Conference on Electromagnetics in Advanced Applications*, pp. 1050–1053, 2007.

[11] D. M. Pozar and S. D. Targonski, "A shaped-beam microstrip patch reflectarray," *IEEE Transactions on Antennas propogation*, vol. 47, no. 7, pp. 1167–1173, 1999.

[12] J. Huang and J. Encinar, *Reflectarray Antennas: Broadband Techniques*, Wiley, Interscience, New York, 2007.

[13] M. Y. Ismail and M. Inam, "Performance improvement of reflectarrays based on embedded slots configurations," *Progress In Electromagnetics Research C*, vol. 14, pp. 67–78, 2010.

[14] M. Fazaelifar, S. Jam, and R. Basiri, "Design and fabrication of a wideband reflectarray antenna in Ku and K bands," *AEU - International Journal of Electronics and Communications*, vol. 95, pp. 304–312, 2018.

[15] G. T. Chen, Y. C. Jiao, and G. Zhao, "A reflectarray for generating wideband circularly polarized orbital angular momentum vortex wave," *IEEE Antennas and Wireless Propagation Letters*, vol. 18, pp. 182–186, 2019.

[16] M. Karimipour and I. Aryanian, "Demonstration of broadband reflectarray using unit cells with spline-shaped geometry," *IEEE Transactions on Antennas and Propagation*, vol. 67, pp. 3831–3838, 2019.

[17] S. V. Hum, M. Okoniewski, and R. J. Davies, "Realizing an electronically tunable reflectarray using varactor diode-tuned elements," *IEEE Microwave and Wireless Components Letters*, vol. 15, pp. 422–424, 2005.

[18] S. Costanzo, F. Venneri, and G. D. Massa, "Liquid crystal-based reconfigurable reflectarray cells for 5G systems," in *2019 23rd International Conference on Applied Electromagnetics and Communications (ICECOM)*, pp. 1–3, Dubrovnik, Croatia, 30 Sept.-2 Oct. 2019.

[19] L. Cai, Z. H. Jiang, and W. Hong, "Evaluation of reconfigurable reflectarray antenna element at 19 GHz based on highly anisotropic liquid crystal material," in *2019 IEEE International Conference on Computational Electromagnetics (ICCEM)*, pp. 1–3, Shanghai, China, March 2019.

[20] M. Y. Ismail, M. Inam, and J. Abdullah, "Design optimization of reconfigurable reflectarray antenna based on phase agility technique," in *Proceedings of the 3rd Conference on Antenna & RCS Measurement (ATMS '10),*, vol. 33, pp. 1–5, Delhi, India, 2010.

[21] O. Bayraktar, O. A. Civi, and T. Akin, "Beam switching reflectarray monolithically integrated with RF MEMS switches," *IEEE Transactions on Antennas and Propagation*, vol. 60, no. 2, pp. 854–862, 2012.

[22] F. Tabarani, T. Chaloun, T. Purtova, M. Kaynak, and H. Schumacher, "0.25μm BiCMOS system-on-chip with four transceivers for Ka-band active reflectarrays," in *2015 SBMO/IEEE MTT-S International Microwave and Optoelectronics Conference (IMOC)*, pp. 1–5, Porto de Galinhas, Brazil, November 2015.

[23] M. Riel and J. J. Laurin, "Design of an electronically beam scanning reflectarray using aperture-coupled elements," *IEEE Transactions on Antennas and Propagation*, vol. 55, no. 5, pp. 1260–1266, 2007.

[24] M. I. Abbasi, M. H. Dahri, M. H. Jamaluddin, N. Seman, M. R. Kamarudin, and N. H. Sulaiman, "Millimeter wave beam steering reflectarray antenna based on mechanical rotation of array," *IEEE Access*, vol. 7, pp. 145685–145691, 2019.

[25] M. Inam and M. Y. Ismail, "Reflection loss and bandwidth performance of X-band infinite reflectarrays: simulations and measurements," *Microwave and Optical Technology Letters*, vol. 53, no. 1, pp. 77–80, 2011.

[26] E. Lueder, *Liquid Crystal Displays: Addressing Schemes and Electro-Optical Effects*, John Wiley and Sons, New York, 2nd edition, 2010.

[27] J. Goodby, P. J. Collings, T. Kato, C. Tschierske, H. Gleeson, and P. Raynes, *Handbook of Liquid Crystals*, John Wiley and Sons, New York, 2nd Edition edition, 2014.

Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 9933111, 19 pages
https://doi.org/10.1155/2021/9933111

Research Article

A Hybrid Deep Neural Network for Electricity Theft Detection Using Intelligent Antenna-Based Smart Meters

Ashraf Ullah,[1] Nadeem Javaid,[1] Adamu Sani Yahaya,[1] Tanzeela Sultana,[1] Fahad Ahmad Al-Zahrani,[2] and Fawad Zaman[3]

[1]*Department of Computer Science, COMSATS University Islamabad, Islamabad 44000, Pakistan*
[2]*Computer Engineering Department, Umm AlQura University, Mecca 24381, Saudi Arabia*
[3]*Department of Electrical and Computer Engineering, COMSATS University Islamabad, Islamabad 44000, Pakistan*

Correspondence should be addressed to Nadeem Javaid; nadeemjavaidqau@gmail.com

Received 12 March 2021; Revised 16 June 2021; Accepted 10 August 2021; Published 26 August 2021

Academic Editor: Daniele Pinchera

This paper presents a hybrid model, named as hybrid deep neural network, which combines convolutional neural network, particle swarm optimization, and gated recurrent unit, termed as convolutional neural network-particle swarm optimization-gated recurrent unit model. The major aims of the model are to perform accurate electricity theft detection and to overcome the issues in the existing models. The issues include overfitting and inability of the models to handle imbalanced data. For this purpose, the electricity consumption data of smart meters is taken from state grid corporation of China. An electric utility company gathers the data from the intelligent antenna-based smart meters installed at the consumers' end. The dataset contains real-time data with missing values and outliers. Therefore, it is first preprocessed to get the refined data followed by feature engineering for selection and extraction of the finest features from the dataset using convolutional neural network. The classification of electricity consumers is performed by dividing them into honest and fraudulent classes using the proposed particle swarm optimization-gated recurrent unit model. The proposed model is evaluated by performing simulations in terms of several performance measures that include accuracy, area under the curve, $F1$-score, recall, and precision. The comparison between the proposed hybrid deep neural network and benchmark models is also performed. The benchmark models include gated recurrent unit, long short term memory, logistic regression, support vector machine, and genetic algorithm-based gated recurrent unit. The results indicate that the proposed hybrid deep neural network model is more efficient in handling class imbalanced issues and performing electricity theft detection. The robustness, accuracy, and generalization of the model are also analyzed in the proposed work.

1. Introduction

The rapid growth of energy consumers has increased the energy demand, which requires efficient generation and distribution of energy at the grid level. In this regard, smart grid [1], with the incorporation of advanced metering infrastructure (AMI), monitors the energy consumption patterns of consumers. AMI establishes a bidirectional communication between consumers and grid to balance supply and demand of energy [2]. Despite of demand management challenge, smart grid faces two types of losses during energy transmission. The first type is technical losses (TLs); whereas the second type is nontechnical losses (NTLs). The former occurs due to energy loss in power distribution lines and transformers. The latter is also known as commercial loss and it occurs due to unregistered connections [3], unpaid bills [4], tampering the antenna-based smart meters, etc., [5]. Moreover, the major reason for NTLs is electricity theft, caused by fraudulent consumers.

NTLs further lead to revenue loss in the economy of countries [6]. For instance, the electric utilities of Brazil and India face about 4.5 billion dollars loss annually [7]. The utilities of USA face around 6 billion dollars loss annually [8]. It is necessary for the power utilities to overcome the revenue loss by detecting NTLs. Therefore, several strategies are being used by the power utilities. The integration of AMI

in power grids provides many advanced and automatic electricity theft detection (ETD) methods. However, calculating such losses and figuring out their exact locations are considered as the most crucial tasks [9]. The inefficiency and less profitability of power utilities are other major concerns. These issues cause extra burden for honest consumers by adding extra charges in their actual utility bills. The aforementioned losses lead to other issues as well, such as hampering and inflation of the industrial routine and load shedding [10].

Several strategies and methods are presented in literature to handle issues related to ETD. These methods are commonly based on hardware, game theory, and data driven [1]. The hardware-based methods, termed as the state-based methods as well [11], perform their operations with the utilization of physical devices, such as transformers, radio-frequency identification tags, sensors, and other electrical equipment. The state-based methods calculate the difference between energy generation at power utility side and energy consumption at consumers' side. These methods achieve high efficiency in theft detection; however, the maintenance and safety of physical devices are major concerns [12]. On the other hand, in the game theory-based methods [13], a game is created between the participants (utilities and energy consumers). Both of the participants compete with each other in order to increase the utility and to get more benefits [14]. Although the game theory-based methods are more efficient contrary to the state-based methods; however, they are based on assumptions and are not able to perform efficient ETD.

In literature, data-driven-based methods are presented to perform efficient ETD. These methods utilize the machine learning (ML) techniques and models. Some of them are decision tree (DT) [3], artificial neural network (ANN) [15], support vector machine (SVM) [16], etc. These techniques use both labeled and unlabeled data to give optimal ETD results, as they have efficient learning capabilities. However, handling the imbalanced class data is challenging for these methods [17]. The traditional classifiers become biased towards the majority class if the data is imbalanced. Therefore, data balancing is required before classification in order to avoid biasness of a classifier and to achieve optimal ETD results. Data balancing is performed using resampling techniques, which balance the data in different classes present in a dataset. The abbreviations used in this work are presented in Table 1.

1.1. Problem Definition and Statement. The increased electricity demand has led to several issues, not just in underdeveloped countries but also in developed countries. The increase in poverty rate is one of the issues, which forces the people to perform electricity theft. In the energy sector, people are adopting illegal means of using electricity to fulfill their demands, i.e., electricity theft. Therefore, ETD is an important thing and needs immediate attention to avoid ever-increasing electricity theft rate. Keeping this in mind, many data analysis techniques, such as SVM, logistic regression (LR), and gated recurrent unit (GRU), have been presented in the literature. However, efficient results have not been achieved yet because of the limitations in these techniques [15]. Some of these limitations are poor learning rate, limited generalization capability, etc. However, the biggest issue being faced is the class imbalanced issue. There exists a huge difference in the number of instances in both classes, i.e., honest and theft consumers' classes. Electricity theft leads to a revenue loss of billion dollars annually for electric utilities [1] and poses serious threats to the country's economy. In addition, electric utilities face electricity losses, which are further classified into TLs ad NTLs; the latter being the most difficult to tackle. NTLs are caused by meddling either with the smart antennas or the smart meters installed at the consumers' end. Therefore, to deal with the class imbalanced issue and to avoid NTLs, an efficient model is presented in this work, termed as particle swarm optimization GRU (PSO-GRU).

1.2. Contributions. This work presented a new variant of neural networks (NNs), named as new hybrid deep neural network (HDNN), in order to address the class imbalanced and overfitting problems in ETD. This work extends the idea present in [18]. The following are the primary contributions of this work:

(i) A metaheuristic model, known as particle swarm optimization (PSO), is used with conventional GRU and convolutional neural network (CNN) to fine tune the parameters and to improve the learning rate, which makes the proposed PSO-GRU model more generalized in terms of training and testing in order to solve the overfitting issue

(ii) A real-world electricity consumption (EC) dataset provided by State Grid Corporation of China (SGCC) [19] is used

(iii) Data normalization and preprocessing are done using local average method and min-max normalization technique, respectively, and

(iv) A comparison is made between the proposed and existing models, which proves the model's efficiency in terms of ETD

The rest of the paper is organized as follows. Section 2 gives an overview of the related work done for ETD. The proposed HDNN model is described in Section 3. The simulations performed in the proposed work are discussed in Section 4. In the end, Section 5 concludes the paper and presents the future work.

2. Related Work

In literature, many systems and approaches are presented for ETD. Most of them are based on hardware and game theory. However, maintenance and data diversity problems are still faced by these approaches. It is observed that the ML techniques are better than the abovementioned ETD methods due to no maintenance requirements and their ability to handle data diversity. However, various existing machine

Table 1: List of abbreviations.

AMI	Advanced metering infrastructure
ANN	Artificial neural network
AUC	Area under curve
CFSFDP	Clustering technique by fast search and find of density peaks
CNN	Convolutional neural network
DNN	Deep neural network
DT	Decision tree
EBT	Ensemble bagged tree
EC	Electricity consumption
ETD	Electricity theft detection
GA	Genetic algorithm
GMM	Gaussian mixture model
MODWPT	Maximal overlap discrete wavelet packet transform
GRU	Gated recurrent unit
HDNN	Hybrid deep neural network
LR	Logistic regression
LSTM	Long short term memory
MEPCO	Multan electric power company
MIC	Maximum information coefficient
ML	Machine learning
NaN	Not a number
NN	Neural network
NTLs	Non-technical losses
PCA	Principal component analysis
PSO-GRU	Particle swarm optimization GRU
RE	Relative entropy
RUSBoost	Random undersampling boosting
SMOTE	Synthetic minority oversampling technique
SGCC	State Grid Corporation of China
SVM	Support vector machine
TLs	Technical losses

and deep learning techniques proposed in the literature face the overfitting problem [17].

The authors in [3] proposed a classification technique based on ensemble bagged tree (EBT) for detecting NTLs in power grids. The proposed technique handles the electricity loss issue in Multan Electric Power Company (MEPCO), Pakistan. In the proposed work, the technique is validated in terms of various performance metrics and is found more efficient than the existing techniques. In [20], the authors proposed a hybrid model for ETD that is based on long short-term memory (LSTM) and CNN. The model is compared with other models, and the results show that it beats the existing models and achieves high accuracy. The proposed model used in [21] is based on the relative entropy (RE) along with principal component analysis (PCA). This work is aimed at detecting the electricity losses, which occur in the vicinity of AMI, using the reconstructed data. The model is evaluated for sensitivity and specificity, and results indicate the good performance of the model for ETD.

In [22], the authors used fuzzy logic technique for the detection of suspicious electricity consumers. The selected time series data is linked with consumers, and fuzzy sets of suspicion are created. Based on these fuzzy sets, a threshold value is decided, which helps in the detection of suspicious consumers. The proposed technique's performance is examined in terms of curve membership function, and the results show that it performs better than benchmark techniques. Similarly, the authors in [23] presented fuzzy logic technique to detect electricity theft and to increase the reliability of the power grid. The proposed technique is evaluated by presenting sixteen real-world scenarios. Efficiency of the technique is evaluated in terms of classifying honest and fraudulent consumers. However, integrating the renewable sources with power grids is not handled well. Similarly, the authors in [24] extracted the EC behavior patterns of the users and detected the abnormal consumption behavior. The authors in [25] proposed a deep learning model to overcome the issues related to NTLs in smart grids. This model considered an unlabeled data and an adversarial model to mitigate the

overfitting issue. It performs better than the existing models. In [26], a hybrid approach is presented for ETD, which is based upon Gaussian mixture model (GMM) and LSTM. In this work, actual time series data is considered and some improvements are made in the LSTM structure. The simulations are carried out to show the performance of the proposed approach in terms of ETD.

In [27], the authors proposed a maximal overlap discrete wavelet packet transform (MODWPT) based model for feature extraction and random undersampling boosting (RUSBoost) technique to detect NTLs in the power grids. A comparison of the proposed and existing techniques is done, and results indicate the efficient performance of the proposed technique for NTL detection. The model is compared with the benchmark techniques, and the results show that the proposed technique outperforms the existing techniques in terms of NTL detection. The authors in [28] established a relationship between commercial losses and characterization of irregular consumers using black hole algorithm. Two different datasets are used in this work provided by Brazilian electric utility, and theft categorization is performed.

The authors in [29] presented a clustering-based approach for the detection of electricity thefts. The approach is based on maximum information coefficient (MIC) and clustering technique by fast search and find of density peaks (CFSFDP). Irish smart meter dataset is used in this work for carrying out the simulations. It is observed that the proposed model performs efficient theft detection. Similarly, a clustering approach is used by the authors in [30] to divide the consumers into clusters on the basis of load consumption and perform efficient short-term load forecasting. The authors in [31] adopted the LSTM method for forecasting EC of the consumers based on the recent past consumption profiles. The continuous monitoring of the profiles helps in efficient ETD. The simulation results prove the model's efficiency. The authors in [32] used big data analytics for forecasting the EC along with the corresponding price. The simulations are performed to prove the model's efficiency in terms of price forecasting. The authors in [33] proposed a fault-tolerant model to preserve the privacy of users and perform data aggregation at the smart grids. Table 2 summarizes the related work in a tabular form for better understanding.

3. Proposed System Model

In this section, the proposed system model is described along with the dataset used in this work. Furthermore, different techniques used in this work are discussed. The proposed system model is shown in Figure 1.

3.1. Description of the Proposed Model. The proposed HDNN-based model consists of several steps, discussed as follows. Initially, the data is gathered from the intelligent antenna-based smart meters installed at the consumers' end, shown in the lower box of the proposed system model. Each smart home has a smart meter and an intelligent antenna, which helps in recording the EC data. The data is saved and is made publicly available by SGCC. Afterward, the dataset is preprocessed in order to normalize the data and to remove redundant and irrelevant data. Also, the outliers are removed to get more refined data. The most important features are obtained by performing feature engineering process. In this process, feature selection and extraction are done. Then, the classification of normal and fraudulent consumers is done. Figure 2 shows the flow of data acquired from the smart homes having intelligent antenna-based smart meters. In Table 3, the identified limitations are mapped with their respective solutions and validations.

(1) *Description of the Dataset*. In this work, real-time EC data of the users is used, provided by SGCC [19]. The data is gathered from the consumers' side using intelligent antenna-based smart meters. The dataset consists of 1,035 features. A subset containing data of 3000 consumers is selected from the whole dataset, in which 2480 are normal consumers; while the remaining 520 consumers are fraudulent. It can be clearly observed that the dataset is imbalanced, due to which ETD is highly affected. In this work, the data is balanced using SMOTE [20], which balanced the number of fraudulent and normal consumers. In addition, the dataset is divided into 75% and 25%, respectively, for training and testing purposes. Table 4 gives a detailed description of the dataset used in the proposed work

(2) *Synthetic Minority Oversampling Technique*. It is considered as one of the oversampling techniques, which increases the data points in the minority class (fraudsters) in order to handle imbalanced data problem. In SMOTE, the synthesized data points are generated in the minority class. In this work, the highly imbalanced data is balanced by using SMOTE. In SMOTE, if (x_1, x_2) depicts a sample of a minority class, then (x_1', x_2') is selected as its nearest neighbors. The synthesized or fake data points are generated by the following equation.

$$(X_1, X_2) = (x_1, x_2) + \text{random}(0, 1) * \Delta, \tag{1}$$

where random $(0, 1)$ presents a number that is chosen between 0 and 1 and Δ denotes the Euclidean distance between the minority class and its neighboring class sample. It is calculated in

$$\Delta = \left(x_1' - x_1\right), \left(x_2' - x_2\right). \tag{2}$$

3.2. Data Preprocessing. The major aim of preprocessing step is to get the most refined data from the whole dataset. In this step, the missing or Not a Number (NaN) values are recovered by local average method, which is given in the following equation [7].

Table 2: Summary of related work.

Proposed solutions	Performance metrics	Limitations
EBT [3]	Sensitivity, specificity, false-positive rate, and $F1$-score	Increased computational time
CNN and LSTM [20]	$F1$-score, area under curve (AUC), precision, and recall	No normalization and parameter tuning
PCA and RE [21]	ROC, specificity, and sensitivity	PCA only works for linear data
Fuzzy logic [22]	Generalized bell curve membership function	Increased computational time
Fuzzy logic [23]	Accuracy, $F1$-score, AUC	Issues related to renewable sources are not handled
Semisupervised deep neural network (DNN) [25]	Precision, true and false-positive rates, recall, and $F1$-score	High false-positive rate
LSTM and GMM [26]	AUC, MCC, recall, and accuracy	Data imbalance is not handled
MODWPT and RUSBoost [27]	$F1$-score, AUC, and precision	Oversampling issue is not tackled
Blackhole algorithm [28]	Average execution time and convergence	High false-positive rate
MIC and CFSFDP [29]	$F1$-score, precision, and recall	Low precision and recall
LSTM [31]	Accuracy, sensitivity, and specificity	Overfitting is not handled well
LSTM and regression [32]	$F1$-score, recall, and precision	Low $F1$-score

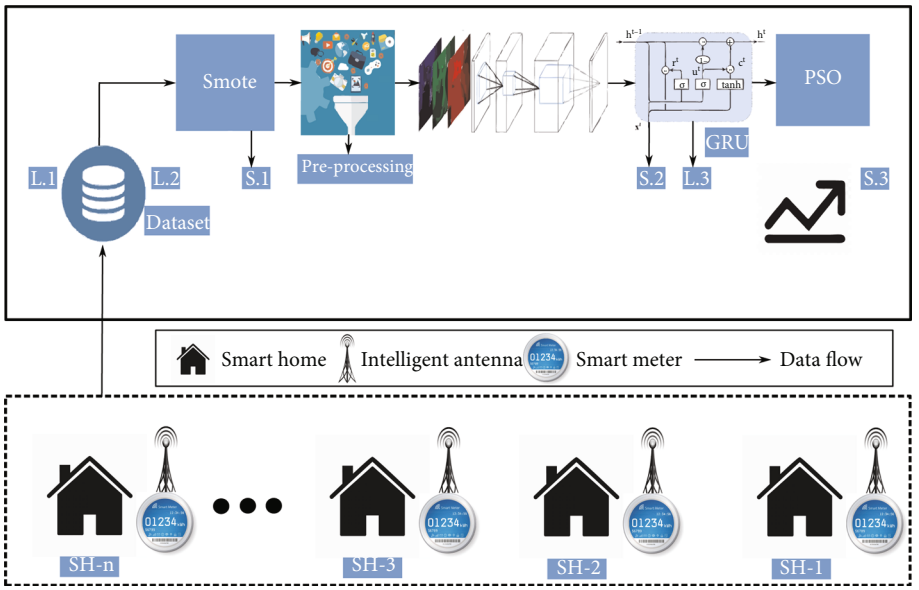


Figure 1: The proposed system model to process data coming from intelligent antenna-based smart meters for users' classification.

$$f(x_i) = \begin{cases} \sum_{k=i-1}^{i+5} P_k \mu_k * \text{Average}_{\text{local}} & \text{if } x_i = \text{NAN}, \\ x_i & \text{if } x_i \notin \text{NAN}, \end{cases} \quad (3)$$

when the value of NAN is continuous, then 0.10 will be the value of P_k. x_i represents the EC of a user at specific time interval. μ_k has a binary value, i.e., either 0 or 1, which is based on threshold k and is calculated using the following equation.

$$\mu_k = \begin{cases} 1 & \text{if } x_k \geq \text{Average}_{\text{local}}, \\ 0 & \text{if } x_k < \text{Average}_{\text{local}}, \end{cases} \quad (4)$$

where $\text{Average}_{\text{local}}$ is computed in the following equation

$$\text{Average}_{\text{local}} = \frac{1}{10} * \sum_{i-5}^{i+5} f(x_i). \quad (5)$$

The min-max normalization technique is used to transform and normalize the dataset in the range of [0, 1]. The minimum value is transformed into 0; whereas the maximum value is transformed into 1 and other values are transformed between 0 and 1. The min-max technique is calculated by the equation given below,

$$B = \frac{A - \min(A)}{\max(A) - \min(A)}. \quad (6)$$

A refers to the actual value of the features and B indicates the value after normalization. While $\max(A)$ is the

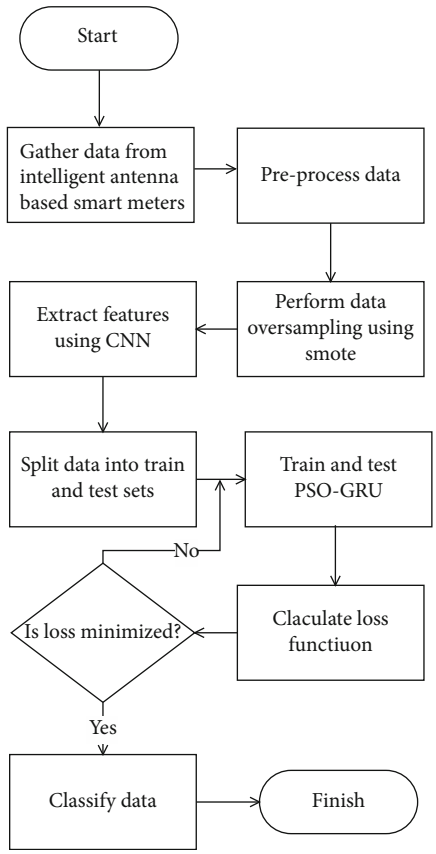


Figure 2: The flow of data gathered from intelligent antenna-based smart meters.

maximum value of A, and min (A) is the minimum value of A. The equations used above are adopted from [20].

3.3. Feature Engineering. Once the data is preprocessed and normalized, the feature engineering process is performed. This process includes two steps: one is feature selection and another is feature extraction. The former selects the most relevant data features from the whole dataset to reduce both overfitting and training time and to improve accuracy; whereas the latter extracts the selected features for data dimensionality reduction and removal of data redundancy. With the feature engineering process, the performance of the model is enhanced.

In this work, feature engineering is done using a DNN, termed as CNN. The idea of CNN was primarily presented in [20]. The typical CNN architecture has various layers, which include convolution, pooling, and fully connected layers. The first convolution layer contains many convolution filters, which are termed as kernels and they perform mapping operation. The convolution layer is mathematically given in the following equation [34]

$$y_{\text{conv}}\left(X_t^{\text{ft}}\right) = \sigma\left(\sum_{\text{ft}=1}^{\text{ft}} W_t^{\text{ft}} * X_t^{\text{ft}} + b_t^{\text{ft}}\right), \tag{7}$$

where σ refers to the activation function and $*$ represents the convolution operation. W_t^{ft} and b_t^{ft} represent the learnable parameters in the fth feature filter. The next layer in CNN is the pooling layer, which comes after the convolution layer. The main objectives of this layer are to extract the meaningful features and to perform the downsampling of each feature map to achieve dimensionality reduction. It also reduces the execution time of the network. The pooling layer consists of two common functions, which are as follows.

(i) *Average Pooling*. This function calculates the average number of features in the feature map

(ii) *Maximum Pooling*. This function calculates the maximum number of optimal features in the feature map

In CNN, the third fully connected layer performs the final classification. In ETD, the data is classified into honest and fraudulent consumers' classes. The mathematical representation of the fully connected layer is given in Equation (8), taken from [34],

$$Y_{\text{ft}}(X_t) = \sigma(W_t.X_t + b_t), \tag{8}$$

where W represents weight and b represents bias.

Function that is used in CNN to predict the final output is known as the Softmax function. The output is given in binary form, either 0 or 1 [20]. Equation (9) provides a complete mathematical form of CNN, as given in [34]. Figure 3 gives an overview of the architecture of CNN,

$$A(W,b) = \frac{1}{m}\sum_{i=1}^{m}\left(\frac{1}{2}\left|y_{W,b}(x_i) - y_i\right|^2\right). \tag{9}$$

The parameters involved in Equation (9) are initialized with some random number using the normal distribution. m denotes the total number of training example. Initially, the weight W_i, j is assigned randomly, then later it is updated using the gradient descent method. Equations (10) and (11) correspond to $W_{ij}^{(l)}$ and $b_i^{(l)}$,

$$W_{ij}^{(l)} = W_{ij}^{(l)} - \alpha\frac{\partial}{\partial W_{ij}^{(l)}}J(W,b), \tag{10}$$

$$b_i^{(l)} = b_i^{(l)} - \alpha\frac{\partial}{\partial b_i^{(l)}}J(W,b), \tag{11}$$

where α represents the learning rate, ∂ denotes partial derivative, $W_{ij}^{(l)}$ is the connection weight between ith neuron in the lth layer and jth neuron in the $(l+1)$ th layer. b is the bias of the ith neuron in the lth layer. Equations (10) and (11) are repeated until the optimal value of objective function $A(W,b)$ is achieved. The above mathematical representations are motivated from [34]. The hyperparameters of CNN used in this work and their values are given in Table 5.

3.4. Gated Recurrent Unit. It is considered as a variation of LSTM and recurrent neural network (RNN) and is a subclass of DNN. It resolves the vanishing gradient problem of RNN

Table 3: Mapping of limitations with solutions and validations.

Limitations	Proposed solution	Validation
Inability to handle imbalanced data (L.1)	Synthetic minority oversampling technique (SMOTE) (S.1)	SMOTE has balanced the data, shown in Table 4 (V.1).
Overfitting (L.2)	CNN-GRU (S.2)	The validation is given in Figures 7 and 8 (V.2)
High false-positive rate (L.3)	PSO (S.3)	The validation is given in Figures 9 and 10 (V.3)

Table 4: Description of dataset.

Hyperparameters	Values
Original dataset size	42372
No. of normal users	38757
No. of fraudulent users	3615
Selected dataset size	5960
No. of selected normal users	2480
No. of selected fraudulent users before SMOTE	520
No. of selected fraudulent users after SMOTE	2480

by using two gates: update gate and reset gate. These gates determine that how much information is required to pass to the future. In the updated gate, the past or previous information needed to be passed to the future is determined. Equation (12) gives the formula to calculate the output of the updated gate z_t for time series data, taken from [35].

$$z_t = \sigma(W^z x_t + U^z h_{t-1}), \tag{12}$$

where x_t shows an input that is given to the network unit and is multiplied by its weight W^z. The h_{t-1} maintains the previous information and is multiplied by its weight U^z as well. Then, these weights are summed up and the result is squashed between 0 and 1 by applying the sigmoid function. The reset gate decides that how much previous information is required to be neglected. Equation (13) gives the mathematical form of the reset gate, taken from [35].

$$z_t = \sigma(W^r x_t + U^r h_{t-1}), \tag{13}$$

where x_t is multiplied by its weight W^r, and the h_{t-1} is multiplied by its weight U^r. Figure 4 shows the architectural view of GRU. In Table 6, the hyperparameters of GRU used in this work along with their values are presented.

3.5. Particle Swarm Optimization. It is a population-based stochastic technique that handles the local optima issue by covering the search space with global optimum solutions. The traditional ML techniques, such as GRU, LR, and SVM, are mostly stuck into local optima. That is why it is not suitable to utilize such techniques for ETD due to their poor ETD performance. In this work, a hybrid technique is made by integrating PSO with GRU to perform efficient and accurate ETD. The proposed technique overcomes the local optima issue very efficiently. PSO performs the searching operation via swarm particles, which are updated in every next iteration. The best optimal solution is achieved by moving each particle in the direction of previous best $\text{pbest}(i,t)$ and global best $\text{gbest}(t)$ solutions in the swarm [18]. Equations (14) and (15) give the mathematical form of calculating pbest and gbest, respectively.

$$\mathbf{pbest}(\mathbf{i},\mathbf{t}) = \arg\ \min_{k=1,\cdots,t} \min\ [f(P_i(k))], i \in 1,2,\cdots,N_p, \tag{14}$$

$$\mathbf{gbest}(\mathbf{t}) = \arg\ \min_{i=1_{k-1,\cdots,t},\cdots,N_p} [f(P_i(k))], \tag{15}$$

where i indicates the particle index, t gives the current iteration number, N_p gives the total number of the particles, f represents the fitness function, and P tells the position. The velocity V of a particle is updated using the following equation.

$$\begin{aligned} V_i(t+1) = {} & \omega V_i(t) + c_1 r_1(\mathbf{pbest}(\mathbf{i},\mathbf{t}) - p_i(t)) \\ & + c_2 r_2(\mathbf{gbest}(\mathbf{t}) - p_i(t)), \end{aligned} \tag{16}$$

where ω is a weight of inertia that is used to balance both global and local exploitation. Whereas r_1 and r_2 indicate the uniformly distributed random variables that are in the range of [0,1]. While c_1 and c_2 represent the positive constant parameters, which are also known as acceleration coefficients. The hyperparameters of PSO used in this work and their values are given in Table 7. The above given mathematical formulations of PSO are taken from [18]. Algorithm 1 gives the pseudocode of PSO.

The efficiency of the presented hybrid model is optimized by passing three parameters of GRU to PSO. Based on these parameters, the training and testing processes of the model are optimized for given dataset. As a result, the model becomes accurate and more robust. These parameters are discussed below.

(i) *Hidden Layer*. It is considered the most important layer of GRU. It is positioned between input layer and output layer. This layer primarily performs the computational operations. Moreover, the weights are given to input values by this layer. After successfully accessing optimal input sets, the results are passed to the output layer for final predictions

(ii) *Batches*. They determine the number of training samples required to compute training and testing loss. Generally, the loss is calculated by the predefined loss function

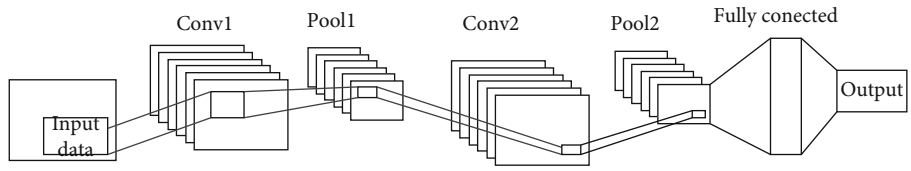


Figure 3: CNN architecture.

Table 5: Hyperparameters selection for CNN through PSO.

Hyperparameters	Selected values	Range of values
Dropout	0.2	0.1, 0.2, 0.3, 0.4
Batch size	25	25, 50, 64, 128
Optimizer	Adam	Adam, SGD, RMSE
Kernel size	3	1, 3, 5
Epochs	6	4, 6, 8, 10
No. of neurons	500	200, 300, 400, 500
No. of hidden layers	5	3, 5, 7, 9, 11
No. of convolution layers	32	4, 8, 16, 32
No. of fully connected layers	1	1, 2, 3, 4

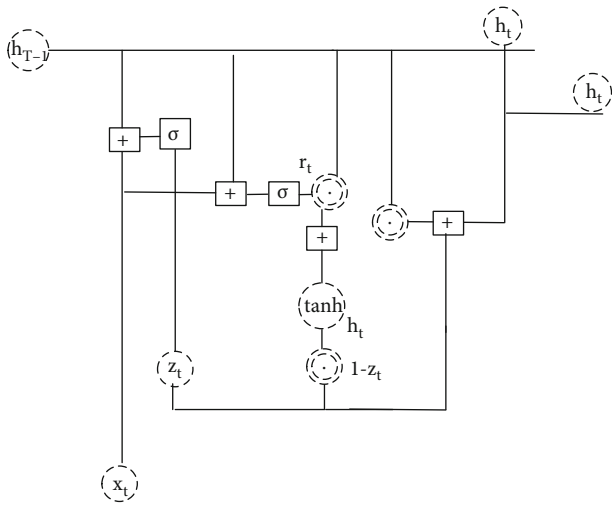


Figure 4: GRU architecture.

Table 6: Hyperparameters selection for GRU through PSO.

Hyperparameters	Selected values	Range of values
Learning rate	0.01	0.1, 0.01, 0.001
Batch size	25	25, 50, 64, 128
Epochs	4	4, 6, 8, 10
Optimizer	Adam	Adam, SGD, RMSE'
No. of neurons	300	100, 200, 300, 400
No. of fully connected layers	1	1, 2, 3, 4
No. of hidden layers	5	3, 5, 7, 9, 11

Table 7: Hyperparameters of PSO.

Hyperparameters	Values
Population size	5960
C1 and C2	2, 2

(iii) *Learning Rate*. It establishes the adjustment of the network weights, which are needed regarding the loss gradient

3.6. Particle Swarm Optimization-Gated Recurrent Unit Binary Classification. A hybrid deep model is presented in this work that is based on PSO-GRU. In the existing models, overfitting and parameter tuning are the main problems. These problems further lead to increase in false-negative rate and false positive rate (which are given in Equations (19) and (20)). Therefore, a hybrid PSO-GRU model is presented to overcome the aforementioned issues. The pseudocode of the model is given in Algorithm 2. The parameters of GRU are tuned using PSO. Afterward, the well-tuned model is used for classification. The main purpose of PSO is to improve the learning of the GRU network and to solve overfitting problem. In this whole process, the data is initially preprocessed. In this phase, local average method is used for recovering missing values; whereas min-max normalization is applied to scale the data. Afterward, data balancing is performed using SMOTE oversampling technique, which balances the provided data in different classes by generating synthetic samples of the minority class. If the classifier is trained on imbalanced data, then it is biased towards the majority class; therefore, the data is balanced using SMOTE technique. Afterward, useful features are extracted from the dataset by CNN. The extracted features are then passed to GRU for training. The parameters of GRU are tuned using PSO. Finally, the fine-tuned model is used to perform classification.

4. Simulation Results and Discussion

Several simulations are conducted to assess the performance of the proposed hybrid model. The simulation results, performance metrics, and benchmark models are discussed in this section.

4.1. Performance Metrics. The performance of the proposed model is examined by considering several performance measures, which include AUC, accuracy, *F*1-score, recall, and precision. The training and testing loss and accuracy are also calculated to assess the performance of the proposed and the

1: Initialization
2: **for** each particle i =1,…,Np, **do.**
3: (a) Initialize the particle's position using uniform distribution as $p_i(0)$ and *U(LB, UB)* where UB and LB represent the upper bound and lower bound of the search space, respectively.
4:
(b) Initialize **pbest** to its initial position:**pbest**(i,0) = $p_i(0)$
5: (c) Initialize **gbest** to the minimal value of the swarm.
:gbest(0) = $p_i(0)$.
6: (d) Initialize Velocity: Vi U(—UB-LB—),—UB-LB—
7: **end for.**
8: Repeat until the Termination Criteria Is Met
9: **for** each particle i =1,…,NP, **do.**
10: (a) Pick random numbers: r1,r2 U(0,1).
11: (b) Update the particle's velocity. See Equation (16).
12: **end for.**
13: **if** f[p_i(t)] < f[**pbest(i,t)**], **then.**
14: Update the best Known Position of Particle I: **Pbest(I,T)** = p_i(T)
15: **if** f[p_i(t)] < f[**gbest(t)**], **then.**
16: Update the swarm's best Known Position: **Gbest(T)** = p_i(T)
17: **end if.**
18: **end if.**
19: Output **gbest(t)** that holds the best found solution.
20: End.

Algorithm 1. Pseudocode of PSO.

1: Initialization
2: Load the dataset D_s from the smart meter S_m
3: Perform pre-processing steps
4: Split D_s into D_{st} (train dataset), D_{ste} (test dataset)
5: Adjust the parameter of the classifier such that
$b \in B, h \in H, n \in N$ and $e \in Ep$. Where b is the number of the batches, h is the number of hidden layers, n is the number of neurons in the hidden layer and e is the number of epoch set for training and testing phases 6: Pass the *h, b and lr* parameters to the PSO 7: Steps in training phase:
8: Build $g_ruPSO \in PSO - GRU$ with the network parameters *b, h* and *lr* for the D_s
9: Repeat
10: At the n^{th} epoch do
11: Train the g_ruPSO to fetch b from D_s
12: Proposed model's performance is measured in terms of AUC, accuracy, precision, recall and F1-score
13: Update the performance of the *queue* ∈ Q. where $Q \in$
P$Push_{opt}$or$Q \in$ PPop_{opt}
14: End
15: The loss function is calculated until loss function $ls \leq Cn$. Where Cn is the convergence threshold
16: If early stopping is performed, then
17: Proceed
18: Else, go to step 8
19: Steps in testing phase:
20: Fetch D_{ste} from the D_s
21: Test is performed by g_ruPSO on the D_{ste} to generate the final results through plots
22: Compare the proposed and benchmark models on the basis of D_{ste} in terms of performance measures for final prediction
23: End

Algorithm 2. Pseudocode of the proposed HDNN model.

benchmark models. A detailed description of the performance metrics is given below.

(1) *Area Under Curve*. This performance metric is used for the validation of model by considering an AUC between two integrals. Moreover, it provides the accumulative performance of the binary classes. Generally, the value of AUC is either 0 or 1. Where 0 value means that the performance of the model is poor; whereas 1 indicates the best performance.

Equation (17) is used to compute the value of AUC [20], as given below

$$\text{AUC} = \frac{\sum \text{Rank}_{i \epsilon \text{positiveclass}} - M(1+M)/2}{M * N}, \quad (17)$$

where the rank values of samples are indicated by Rank_i. M represents the total positive samples, and N represents the negative ones.

(2) *F1 -Score*. It is referred as *F*-measure as well. It calculates the testing accuracy of the model and assesses the testing score using recall and precision. The *F*1 -score is calculated using the following equation [20]

$$F1 - \text{Score} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}. \quad (18)$$

(3) *Recall and Precision*. The recall and precision are expressed using the following equations

$$\text{Recall} = \frac{\text{TruePositive}}{\text{TruePositive} + \text{FalseNegative}}, \quad (19)$$

$$\text{Precision} = \frac{\text{TruePositive}}{\text{TruePositive} + \text{FalsePositive}}, \quad (20)$$

where recall calculates the number of true positives in all of the classifier's results. True positive means correctly classified energy thieves. False-positive means honest electricity consumers misclassified as thieves. False-negative means energy thieves misclassified as honest consumers and true negative means correctly classified honest consumers. Whereas precision is the measure of relevance of the classifier's results. If precision is high, it means that classifier's relevant results are more than the irrelevant results.

4.1.1. Case Study A. In this case study, the SGCC dataset is considered, which is publically available on internet. The brief description about dataset is given in Section III-A1 [Page 3].

4.2. Description of Existing Models with their Performance. This section presents the existing models along with their performance based on the abovementioned performance metrics.

(1) *Support Vector Machine Model.* SVM is widely used in ETD for binary classification [36]. The performance metrics used for SVM, and their results are given in Table 8. It performs better than LR model; however, its performance is worse than LSTM, GRU, and PSO. It means that SVM is less accurate in handling the imbalanced data.

(2) *Logistic Regression Model.* LR is a popular classifier that is widely used for both classification and regression. In literature, it is also used for ETD [37]. In this work, the performance of LR is examined in terms of aforementioned performance metrics. The results of LR are given in Table 9, which show that it performs worse than all of the benchmark techniques. The reasons behind this are overfitting issue and inability of LR to handle imbalanced ETD data

(3) *Long Short Term Memory Model.* LSTM is a DNN model and is widely used for feature extraction and classification in ETD [38, 39]. The performance of LSTM is checked in terms of abovementioned metrics. The results are presented in Table 10, which show that LSTM performs better than SVM and LR and worse than the proposed model. The proposed model performs better than LSTM because it does not face overfitting problem, which degrades the performance of LSTM

(4) *Gated Recurrent Unit Model.* GRU is also a DNN model [40]. It is an advanced version of the LSTM. Its results are shown in Table 11, which are better as compared to the benchmarks. It means that GRU is capable of handling imbalanced data while avoiding overfitting

(5) *Genetic Algorithm Model.* Genetic algorithm (GA) is a metaheuristic technique. Its performance is assessed based on various performance metrics, as given in Table 12. The results show that GA performs better than the benchmark models: SVM, LR, LSTM, and GRU. GA is also found to be more accurate and robust than the benchmarks because of its better learning capability

4.3. Results. The simulations are performed to evaluate the proposed and benchmark models by considering the aforementioned performance measures along with loss and accuracy. Moreover, the models are integrated for the performance evaluation. The SVM and LR obtain 0.68% and 0.63% of ACU score, respectively. SVM has higher AUC score as compared to LR because of using kernel trick to cope with the nonlinear data. In contrast, LR has lowest AUC score of 0.63% because it has only one hidden layer, which did not handle high dimensional data effectively and stuck in local minima. The performance results of SVM and LR are given in Tables 8 and 9, respectively.

Figures 5 and 6 show the accuracy and loss values of combined CNN-LSTM model. The CNN is utilized to extract abstract and latent features from EC data with the help of convolutional and pooling layers. Whereas LSTM extracts temporal patterns and classifies consumers' records into normal and abnormal data patterns. From Figure 5, the training accuracy is observed as 81%; whereas the testing

Table 8: Results of SVM.

Performance evaluation metrics	Results
AUC	68%
*F*1-score	75.89%
Accuracy	76.67%
Precision	77.43%
Recall	78.82%

Table 9: Results of LR.

Performance evaluation metrics	Results
AUC	63%
*F*1-score	67.93%
Accuracy	72.09%
Precision	69.02%
Recall	73.63%

Table 10: Results of LSTM.

Performance evaluation metrics	Results
AUC	81.7%
*F*1-score	79.2%
Accuracy	80.11%
Precision	81.20%
Recall	81.22%

Table 11: Results of GRU.

Performance evaluation metrics	Results
AUC	83.7%
*F*1-score	82.98%
Accuracy	83%
Precision	83.98%
Recall	84.23%

Table 12: Results of CNN-GA-GRU.

Performance evaluation metrics	Results
AUC	87%
*F*1-score	87.22%
Accuracy	87%
Precision	88.1%
Recall	88.72%

accuracy of the model is 75.5%. Both accuracies increase for an increasing number of epochs, which shows that model gives good results on a large number of epochs. However, the model is trained only for four epochs due to limited resources. There is a 6% difference between training and testing accuracies curves, which indicates that the model is stuck into an overfitting problem. On the other hand, Figure 6 shows that the training and testing losses of the

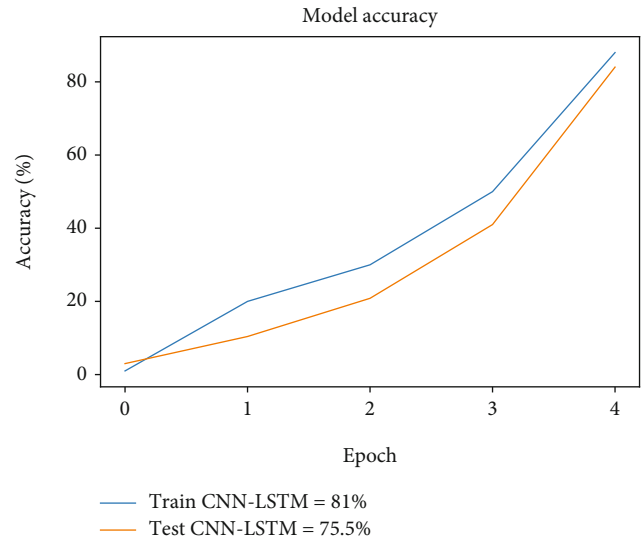


Figure 5: CNN-LSTM accuracy.

model decrease as number of epochs increase. After the 4th epoch, the training and testing loses are 19% and 24.5%, respectively.

This implies that the epoch is the main controlling parameter that decides the optimal point where model achieves higher performance. However, the CNN-LSTM obtains 75.5% test accuracy, which is neither satisfactory nor acceptable in ETD. This is happened due to the inappropriate selection of hyperparameters. For deep learning models, the suitable selection of hyperparameters has great influence on the performance results.

Figures 7 and 8 show accuracy and loss values of combined CNNGRU model on training and testing datasets. The CNN is used to extract optimal features while classification task is performed through GRU model. The GRU model has reset and update gates that extract more relevant information from extracted high variance features through CNN and remove the noisy and redundant features. This process makes the performance of CNN-GRU better than CNN-LSTM. There is a 4% difference between accuracy curves and a 6.97% difference between loss curves on training and testing datasets, which indicate that the CNN-GRU model is stuck into an overfitting problem. The inappropriate tuning of hyperparameters leads to an overfitting problem where the model gives good results on seen data as compared to unseen data.

In literature, there are different techniques to tune the hyperparameters of ML and deep learning models like random search, grid search, gradient-based optimization, and evolutionary algorithms. Each one of these methods has its pros and cons. In this study, we utilize evolutionary algorithms PSO and GA to find optimal hyperparameters of the CNN-GRU model. These algorithms make a search space of hyperparameters and try to find the optimal combination where the model gives high values of performance indicators.

The PSO is merged with CNN-GRU for hyperparameters tuning to enhance the performance of the proposed model. The results are shown in Figures 9 and 10. The

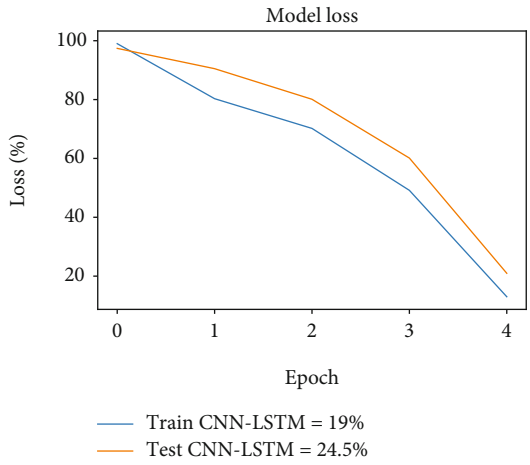


Figure 6: CNN-LSTM loss.

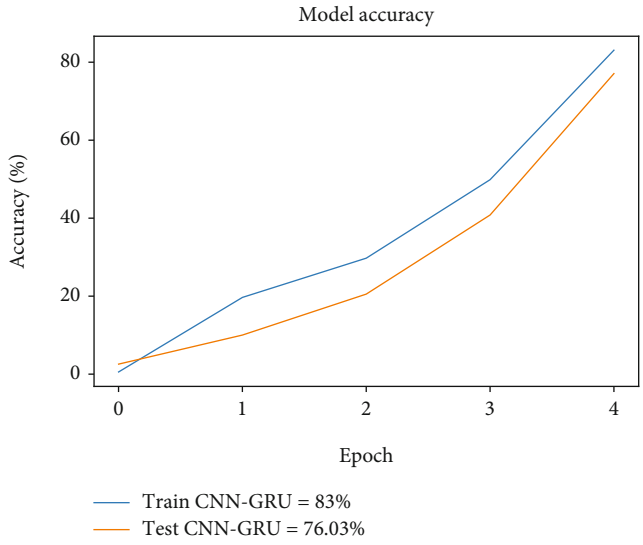


Figure 7: CNN-GRU accuracy.

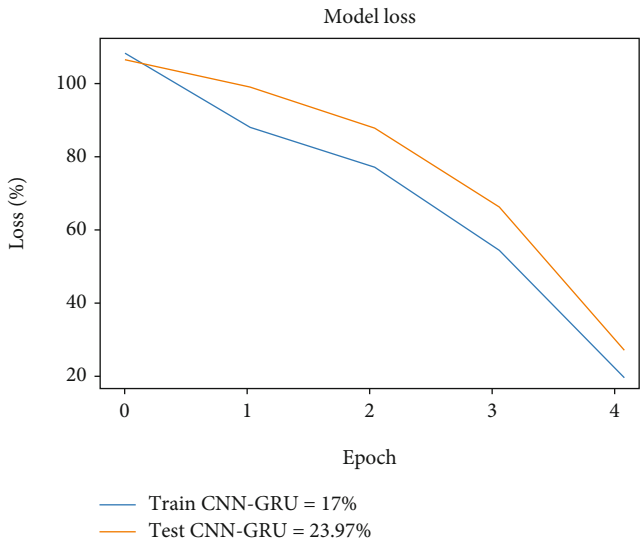


Figure 8: CNN-GRU-loss.

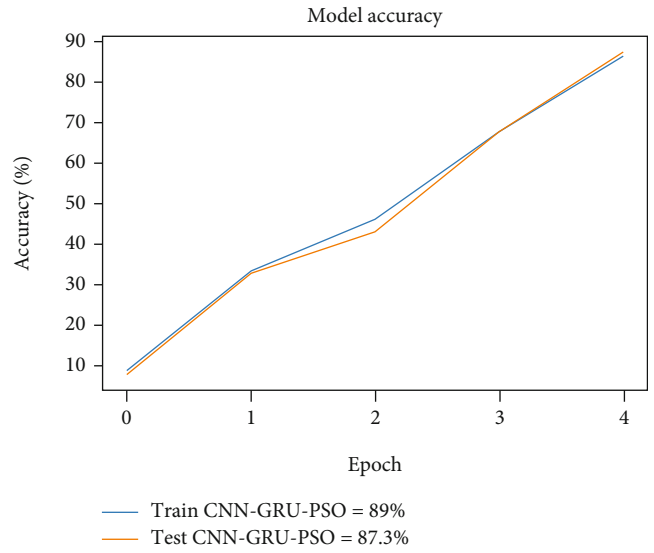


Figure 9: CNN-GRU-PSO accuracy.

former shows the training and testing accuracy of CNN-GRU, which is increased with the increase in the number of epochs. Whereas the training and testing loss are presented in the latter figure and the results show that the loss decreases with the increase in the number of epochs. Furthermore, CNN-GRU is integrated with GA to find optimal combination of hyperparameters. The performance of combined CNN-GRU-GA is presented in Figures 11 and 12. The training and testing accuracies of CNN-GRU-GA are shown in Figure 11. The figure shows that both training and testing accuracy are approximately 87% and 86.3%, respectively. Figure 12 presents the training and testing loss of CNNGRU-GA, which keep decreasing with the increasing number of epochs. As shown in the figure, the training loss and the testing loss are 13% and 13.7%, respectively, which are approximately the same. Next, training and testing accuracy and loss of hybrid CNNGRU-PSO model are evaluated in Figures 9 and 10, respectively. Here, PSO and GA are used for tuning the hyperparameters of the CNNGRU model. The results exhibit that the PSO obtains optimal set of parameters as compared to GA because the PSO require less number of parameters and less execution time. In PSO, each solution has its own local best, which leads it towards global best after each iteration. Whereas GA has crossover and mutation steps that create diversity in newly generated offsprings and prevent the model from falling into local optima problem. However, in this case, PSO performs better as compared to GA and gives optimal combination hyperparameters where CNN-GRU gives good results. The former shows that both training and testing accuracy of CNN-GRU-PSO increase with the increasing number of epochs. The training accuracy is 89%; while testing accuracy is 87.3%. Whereas in latter, the training and testing loss are given, which are 11% and 12.7%, respectively. Moreover, CNN is combined with GRU and the performance of CNN-GRU is evaluated in terms of training and testing accuracy and loss. In Figures 5–10, the performance of the combined models is depicted.

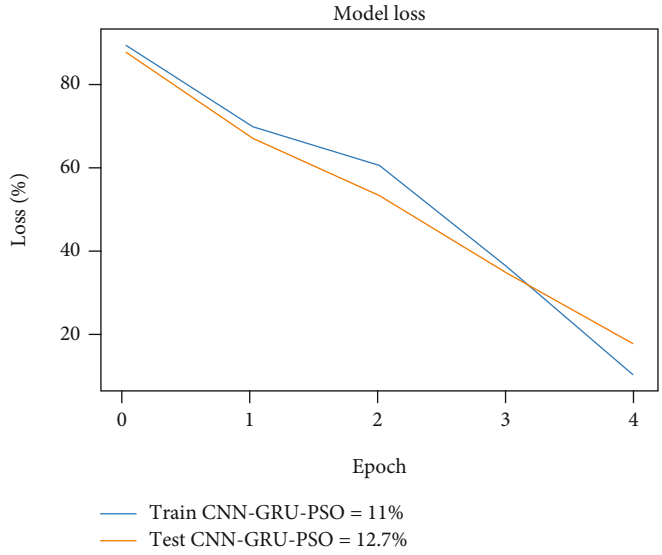


Figure 10: CNN-GRU-PSO loss.

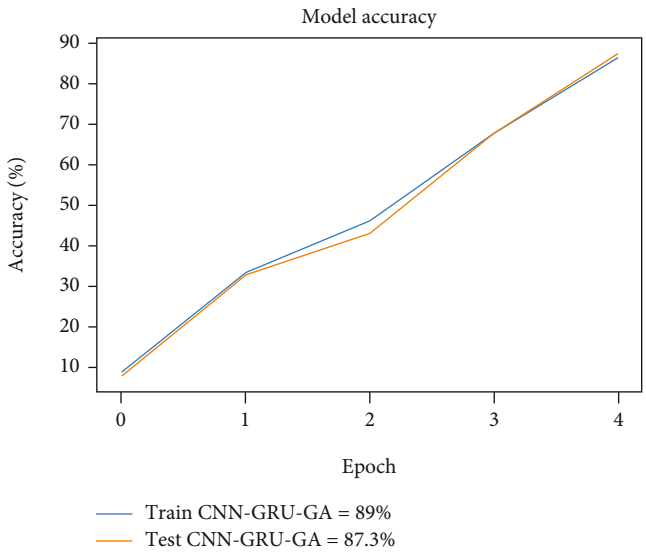


Figure 11: CNN-GRU-GA accuracy.

It is observed that the CNN-GRU-PSO has the maximum accuracy as compared to the other models. Similarly, the CNN-GRU-PSO model has a minimum loss, which shows the model's generalization. Figures 13 and 14 show the combined performance of the used and existing models for AUC. In former, AUC is calculated using both the false positive rate and true positive rate. The results indicate that the proposed CNN-GRU-PSO achieves high AUC score. Whereas the other models have a low AUC score. Moreover, it is shown that the proposed model beats the existing ones regarding AUC in the presence of imbalanced dataset. The GRU module in proposed model has strong ability to learn temporal correlation from long-term electricity load profile of consumers. It also maintains the context of previous EC information, which helps out to handle any nonmalicious (weather condition, family structure, etc.) change in EC profile. Moreover, the integration of PSO for parameters tuning further enhance the performance of the proposed model towards efficient ETD. Furthermore, the proposed model is compared with the benchmark models in terms of mentioned performance metrics, and the result is shown in Figure 15. The result indicates that the hybrid CNN-GRU-PSO model is more robust, accurate, efficient, and more generalized than the benchmarks, as given in Table 13.

Table 14 describes the running time of the proposed and baseline models. The SVM model takes 220 s during the training phase, which is higher than all other schemes. The selected SGCC dataset is high dimensional and not linearly separable. SVM draws $n-1$ hyperplanes and then picks an optimal hyperplane of high margin for distinguishing two classes (n represents the number of dimensions).

So, that is why SVM takes higher time as compared to other models.

The LR takes lowest execution time because of its simplex layering structure. It has only one hidden layer of neural network (NN). So, it needs less weights to learn and consumes less time as compared to other deep learning models. The CNN-GRU takes 56 seconds running time in training phase, which is 10 seconds less than CNNLSTM because of less gated configuration as compared to LSTM. The proposed CNN-GRU-PSO has higher execution time as compared to other models because of using PSO for tuning the hyperparameters of both CNN and GRU models concurrently.

4.3.1. Case Study B. In this case study, the PRECON dataset is used, which is publically available on internet. The dataset is collected by Pakistan Residential EC company. This dataset contains the EC history of 42 residential houses for 365 days. In dataset, the EC of each user is recorded after one minute time period. However, in this work, the data granularity is reduced into half hour for ease. EC of 30 minutes is aggregated into single value for all dataset. All the consumers

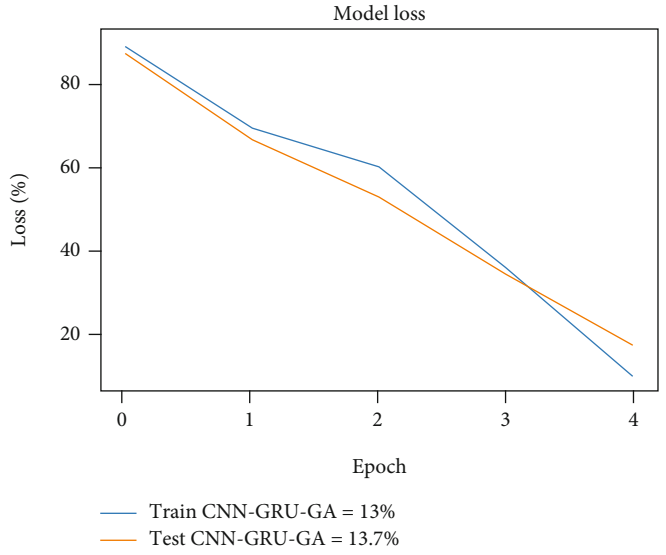


Figure 12: CNN-GRU-GA loss.

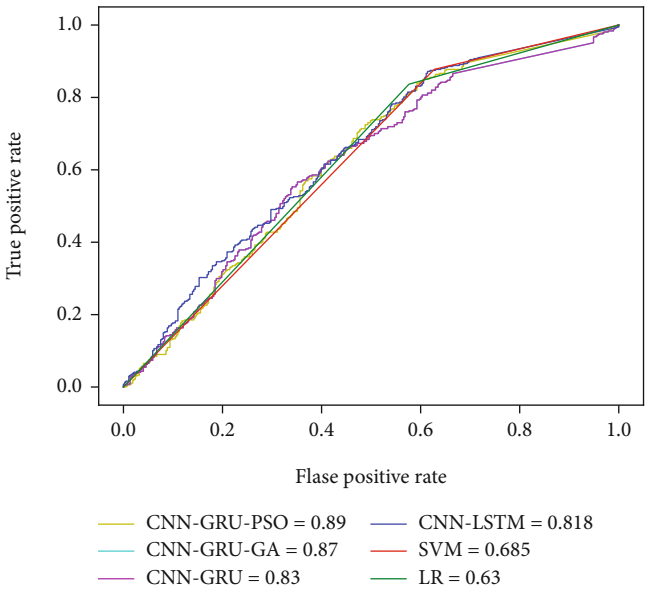


Figure 13: Area under the curve.

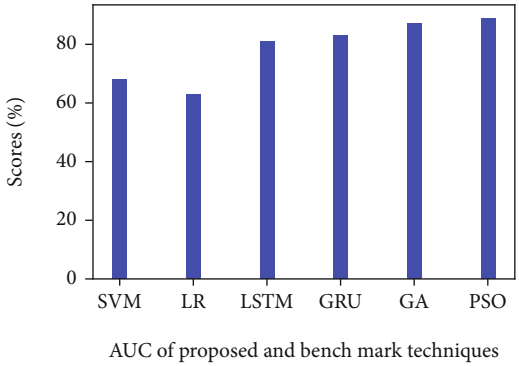


Figure 14: Area under the curve.

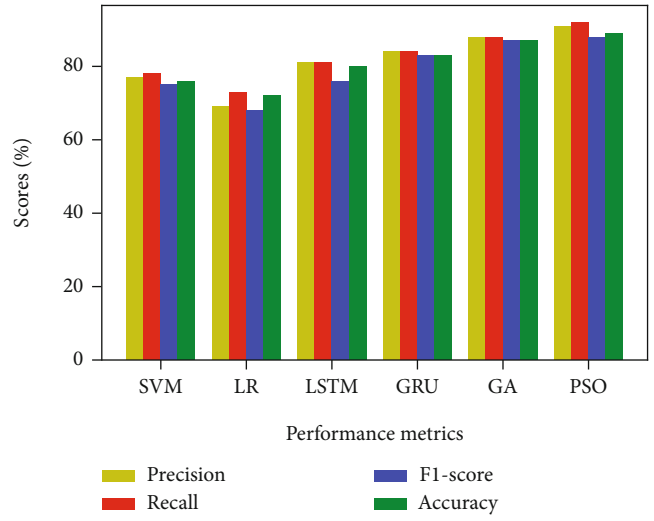


Figure 15: Combined performance evaluation.

Table 13: Results of CNN-PSO-GRU.

Performance evaluation matrix	Results
AUC	89%
*F*1-score	88.54%
Accuracy	89%
Precision	91.23%
Recall	92.72%

which are participated in the experiment are assumed as honest consumers. So, the dataset contains the EC of only normal consumers. For ETD, malicious samples need to be generated as well. For this, six theft attacks are applied on benign samples in order to generate attack samples. These attacks are introduced by Jokar et al. in [1]. After applying theft attacks, the generated theft samples are drastically increased than benign class samples. Now, the dataset

Table 14: Execution time of the proposed and baseline models.

Models	Execution time (seconds)
SVM	220
LR	35
CNN-LSTM	65
CNN-GRU	56
CNN-GRU-PSO	89

becomes imbalance. For data balancing, SMOTE is used to synthesize the minority benign class.

4.4. Results and Discussion. The simulations are performed on PRECON dataset, and the performance is measured through different performance metrics. Figure 16 depicts the loss of the proposed hybrid model. The decline in loss is noted on every epoch. The epoch is the main controlling parameter during the training phase. The optimal epoch value is 25, which is founded by PSO during parameter tuning. It is observed that the proposed model efficiently learns the EC patterns on each iteration by using the strong feature learning and temporal correlation abilities of the CNN and GRU models, respectively. Similarly, Figure 17 shows the proposed model accuracy. The accuracy tells about how accurately data samples are classified. The higher accuracy means higher correct predictions. The accuracy is increasing gradually on test and train data after each epoch. The optimal epoch value found by PSO is 25. Figure 18 illustrates $F1$-score, which is the harmonic mean of precision and recall. It helps the model to accurately identify the energy thieves. The higher $F1$-score is beneficial for power utilities to recover maximum revenue. The AUC score of the proposed and baseline models is presented in Figure 19. The AUC measures the separability between the positive and negative classes. The proposed model obtains 0.95 of AUC score, which is higher than all benchmark models. This implies that the proposed model efficiently distinguishes two classes and reduces the miss classification rate to a minimal level. Furthermore, Table 15 describes the performance results of baseline models and the proposed model on the PRECON dataset. It is seen from the results that the SVM and LR also perform well because the dataset has 48 features after reducing the data granularity to half hour. The SVM and LR capable to deal this data dimensional effectively and learn EC pattern for efficient ETD. The regular LSTM captures temporal correlation from historical EC data and reduces high FPR. The hybrid of CNN-LSTM and CNN-GRU achieves satisfactory performance towards efficient ETD due to using the CNN feature extraction ability and storing temporal correlations abilities of LSTM and GRU models. However, the proposed CNNGRU-PSO outperforms the other models because of utilizing PSO for hyperparameter tuning. The best selection of hyperparameters boosts the performance of the classification model. PSO finds the optimal set of parameters, where the models obtain the highest performance. It is concluded that the proposed model efficiently performs on both case studies. The performance results indicate that our method is an ideal solution for efficient ETD.

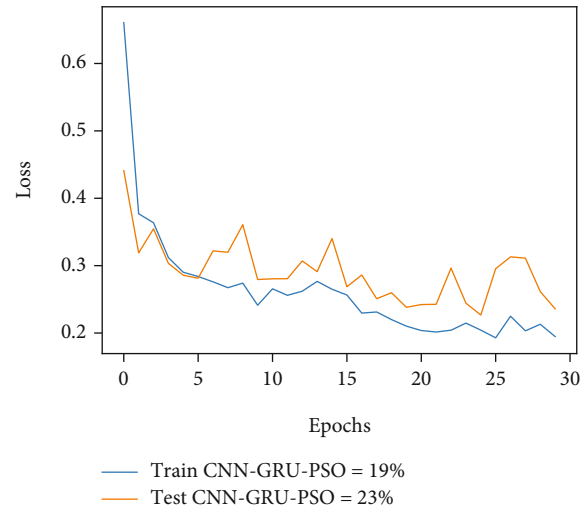


Figure 16: CNN-GRU-PSO loss on PRECON dataset.

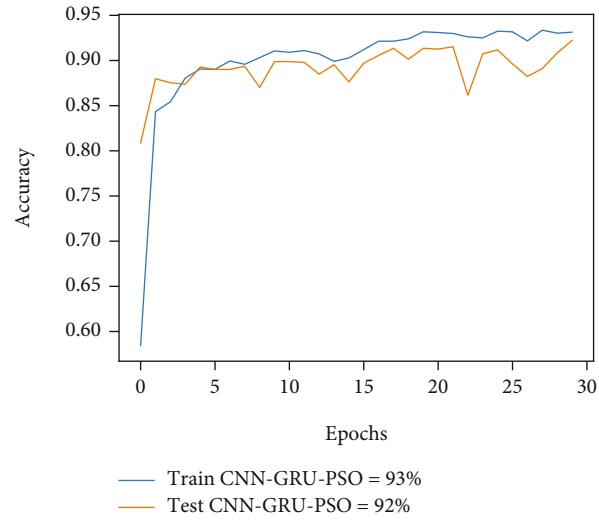


Figure 17: CNN-GRU-PSO accuracy on PRECON dataset.

4.5. Statistical Test Evaluating the Significant of Method Result. Statistical tests are performed to check the significant and robustness of different learning algorithms. These tests determine whether one learning algorithm outperforms another learning algorithm by chance (due to randomness) or in real. The core principal of these tests is null hypothesis. The null hypothesis is constituted according to the problem. For instance, for classification task, the null hypothesis would be the difference between the mean performance of two algorithms is probably real or not. Numerous statistical tests are available to judge the significant of different case studies. However, in this work, our intention is to compare and judge the performance of different classification algorithms. A good statistical test has capable to judge the randomness in different classifiers' results, random variation in classification error, and randomness in the selection of

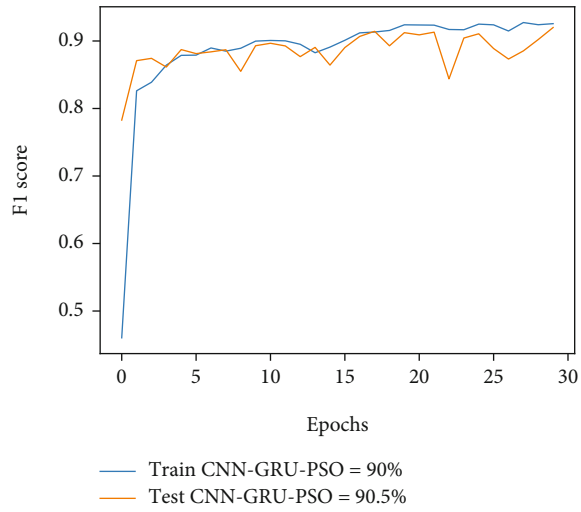


Figure 18: CNN-GRU-PSO *F*1-score on PRECON dataset.

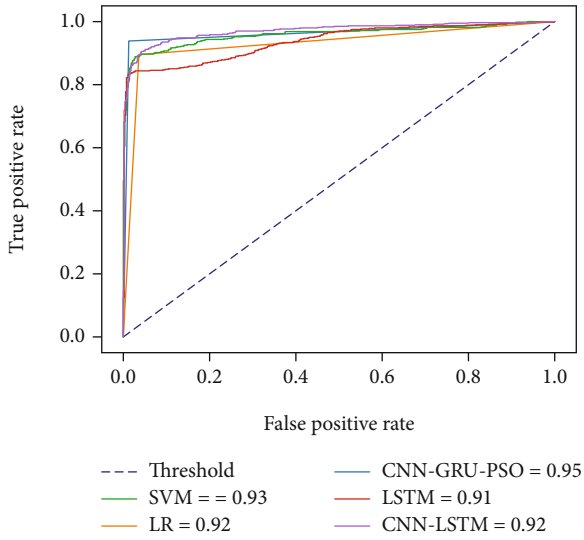


Figure 19: AUC score of CNN-GRU-PSO and baseline models on PRECON dataset.

Table 15: Comparison analysis of proposed model with benchmark schemes on PRECON dataset.

Models	Accuracy	Precision	Recall	*F*1-score	AUC
SVM	0.94	0.91	0.92	0.92	0.93
LR	0.93	0.90	0.92	0.91	0.92
LSTM	0.83	0.92	0.91	0.91	0.91
CNN-LSTM	0.91	0.92	0.93	0.92	0.92
CNN-GRU	0.92	0.93	0.93	0.93	0.93
CNN-GRU-PSO	0.93	0.94	0.95	0.94	0.95

test data. In this regard, three suitable statistical tests are opted that are closely related to the classification task. The detailed description of these test is given as follows.

(1) *5x2cv Paired t Test*. A well-known statistical test for evaluating the performance of classification and regression models. It is introduced by Dietterich in 1998 [41]. In this study, this test is conducted to judge the performance of different classifiers. It consist of twofold cross-validation with five repeats. For each fold, the classifier is trained and the results are recorded. Afterward, the 5x2 paired t test is applied on the final result to accept or reject the null hypothesis. In this case, the null hypothesis defines as the difference between the mean performance of two algorithms is probably real or not. In this test, the p value is calculated against each t value. The p denotes the probability value, which decides that the result of your sample data is occurred by chance or not. The p value ranges from 0 to 1. The smallest p values are suitable. In general, the p value should be less than 0.05 (5%). However, this test is suitable where sample size is small and the mean is known

(2) *5x2cv Combined f Test*. This basic principal of this test is similar to the 5x2cv paired t test except some improvement. This test is introduced by Alpaydin [42]. It is suitable for large sample size and capable to compare more than two populations at the same time. This test is conducted to compare the variance of two population while the 5x2cv paired t test judges the mean of two populations. The working mechanism of this test is similar to the 5x2cv paired t test. The two-fold cross-validation is conducted with five repeats. The results of each fold are recorded and then f test is applied to evaluate the null hypothesis. The null hypothesis is based on the p value. The null hypothesis is accepted if p value is smaller than 0.05 (5%) otherwise rejected

(3) *McNemar's Test*. This test is nonparametric and distribution free statistical test. This test is conducted to check the disagreement of the classifiers on the performance results. In this test, a contingency table is constructed to judge its homogeneity. According to our scenario, this test judges how two or more classifier agree or disagree on the same sample. For this, the contingency table is designed according to the confusion matrix of two classifiers. The structure of contingency shown in Table 16 is given below. The table is finalized by filling the required values, and these values are derived from the confusion matrix. Afterward, the McNemar's test statistic is calculated from the contingency table by using following formula

$$Ms = \left(\frac{(\text{Yes/No} - \text{No/Yes})^2}{(\text{Yes/No}) + (\text{No/Yes})} \right), \tag{21}$$

where Ms denotes McNemar's test statistic. Similar to abovementioned test, a null hypothesis is formulated. The null hypothesis ($H0$) is defined as the classifiers have similar proportion of errors in test set or vice versa. The p value is

Table 16: Contingency table.

	Cf2 correct	Cf2 incorrect
Cf2 correct	Yes/yes	Yes/no
Cf2 incorrect	No/yes	No/no

Table 17: Statistical test on proposed with existing models.

	5x2cv paired t test	
Models	t-statistic	p value
Proposed and SVM	-0.678	0.528
Proposed and LR	0.223	0.489
Proposed and LSTM	0.449	0.672
Proposed and CNNLSTM	0.295	0.780
	5x2cv paired f test	
Proposed and SVM	1.253	0.078
Proposed and LR	1.968	0.049
Proposed and LSTM	2.035	0.215
Proposed and CNNLSTM	2.295	0.0236
	McNemar's test	
Proposed and SVM	1.789	0.0569
Proposed and LR	2.365	0.0422
Proposed and LSTM	4.369	0.1258
Proposed and CNNLSTM	3.269	0.056

calculated to accept or reject the $H0$. The $H0$ is rejected if the value of p is less than 0.005 and vise versa.

Table 17 describes the results of different statistical tests. It is seen that the results of 5x2cv paired f test outperform the other tests because it is suitable for large population size. The value of probability (p) is almost lesser that 0.005 (5%). The smallest value of p indicates that the models' results are that occurred by chance. The t test does not perform well because the available dataset has large in both population and sample size. The McNemar's test also yields better value of p and proves that the models' results are not occurred by chance. All the results are real and do not depend on any noise factor.

5. Conclusion and Future Work

This work presents a HDNN based model in order to detect electricity theft in the smart grid. For this, dataset is taken from SGCC, which provides the real EC data gathered using intelligent antenna-based smart meters installed at the consumers' end. The proposed model works in several steps. In the preprocessing step, the raw data is normalized, and the outliers and missing values are handled. The preprocessing is done by the local average method and min-max normalization technique. Then, the feature engineering step is performed using CNN. Once the most relevant and normalized data is obtained, the classification process is done using PSO-GRU integrated CNN. In this step, the normal and fraudulent consumers are classified. The proposed model is validated in terms of several performance metrics like accuracy, recall, precision, AUC, and $F1$-score. Moreover, comparison of the proposed and existing hybrid models is done. The models include CNN-GRU, CNN-LSTM, and CNN-GRU-GA.

The comparison results show the efficiency, accuracy, robustness, and generalization of the proposed hybrid model for handling imbalanced class issue in terms of ETD. Despite that, our proposed method is an ideal solution towards efficient ETD. However, it has incurred little bit higher computational cost because the proposed model's modules are integrated in a sequential manner (CNN-GRU-PSO). First, CNN takes time while capturing potential features from high-dimensional EC data. Second, GRU processes the CNN's extracted features map for final classification. Meanwhile, PSO tunes the hyperparameters of both CNN and GRU models. This working flow of the proposed model consumed a little bit higher execution time as compared to the existing methods. Moreover, the proposed method has a lack in some complex real-world scenarios by accurately identifying the electricity thieves due to the addition of simulated theft data (in minority class using SMOTE). For the future, more robust techniques will be utilized to efficiently handle the overfitting issue.

Data Availability

The datasets used in this study are openly available in [henryRD- lab/ElectricityTheftDetection] at [23] [Page 3, Section III-A1].

Conflicts of Interest

The authors declare no conflict of interest.

References

[1] P. Jokar, N. Arianpoo, and V. C. M. Leung, ""ETD in AMI using customers" consumption patterns," *IEEE Transactions on Smart Grid*, vol. 7, no. 1, pp. 216–226, 2015.

[2] S. Kazmi, N. Javaid, M. J. Mughal, M. Akbar, S. H. Ahmed, and N. Alrajeh, "Towards optimization of metaheuristic algorithms for IoT enabled smart homes targeting balanced demand and supply of energy," *IEEE Access*, vol. 7, pp. 24267–24281, 2017.

[3] M. S. Saeed, M. W. Mustafa, U. U. Sheikh, T. A. Jumani, and N. H. Mirjat, "Ensemble bagged tree based classification for reducing non-technical losses in multan electric power company of Pakistan," *Electronics*, vol. 8, no. 8, p. 860, 2019.

[4] H. O. Henriques, R. L. S. Correa, M. Z. Fortes, B. S. M. C. Borba, and V. H. Ferreira, "Monitoring technical losses to improve non-technical losses estimation and detection in LV distribution systems," *Measurement*, vol. 161, article 107840, 2020.

[5] Z. Aslam, F. Ahmed, A. Almogren, M. Shafiq, M. Zuair, and N. Javaid, "An attention guided semi-supervised learning mechanism to detect electricity frauds in the distribution systems," *IEEE Access*, vol. 8, pp. 221767–221782, 2020.

[6] L. Raggi, F. Trindade, V. C. da Cunha, and W. Freitas, "Non-technical loss identification by using data analytics and customer smart meters," *IEEE Transactions on Power Delivery*, vol. 35, no. 6, pp. 2700–2710, 2020.

[7] K. M. U. Ghori, R. A. Abbasi, M. Awais, A. Ullah, and L. Szathmary, "Performance analysis of different types of machine learning classifiers for non-technical loss detection," *IEEE Access*, vol. 8, pp. 16033–16048, 2019.

[8] K. M. U. Ghori, M. Imran, A. Nawaz, R. A. Abbasi, A. Ullah, and L. Szathmary, "Performance analysis of machine learning classifiers for non-technical loss detection," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–16, 2020.

[9] A. Aldegheishem, M. Anwar, N. Javaid, N. Alrajeh, M. Shafiq, and H. Ahmed, "Towards sustainable energy efficiency with intelligent electricity theft detection in smart grids emphasising enhanced neural networks," *IEEE Access*, vol. 9, pp. 25036–25061, 2021.

[10] P. F. M. Simões, R. C. Souza, R. F. Calili, and J. F. M. Pessanha, "Analysis and short-term predictions of non-technical loss of electric power based on mixed effects models," *Socio-Economic Planning Sciences*, vol. 71, article 100804, 2020.

[11] Z. Yan and W. He, "Electricity theft detection base on extreme gradient boosting in AMI," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–9, 2021.

[12] Z. Qu, H. Li, Y. Wang, J. Zhang, A. A. Siada, and Y. Yao, "Detection of electricity theft behavior based on improved synthetic minority oversampling technique and random forest classifier," *Energies*, vol. 13, no. 8, article 2039, 2020.

[13] O. Samuel, N. Javaid, A. Khalid et al., "Towards real-time energy management of multi-microgrid using a deep convolution neural network and cooperative game approach," *IEEE Access*, vol. 8, pp. 161377–161395, 2020.

[14] Z. Aslam, N. Javaid, A. Ahmad, A. Ahmed, and S. M. Gulfam, "A combined deep learning and ensemble learning methodology to avoid electricity theft in smart grids," *Energies*, vol. 13, no. 21, article 5599, 2020.

[15] O. I. Abiodun, A. Jantan, A. E. Omolara, K. V. Dada, N. A. E. Mohamed, and H. Arshad, "State-of-the-art in artificial neural network applications: a survey," *Heliyon*, vol. 4, no. 11, article e00938, 2018.

[16] K. Wang, C. Xu, Y. Zhang, S. Guo, and A. Y. Zomaya, "Robust big data analytics for electricity price forecasting in the smart grid," *IEEE Transactions on Big Data*, vol. 5, no. 1, pp. 34–45, 2017.

[17] Z. A. Khan, M. Adil, N. Javaid, M. N. Saqib, M. Shafiq, and J.-G. Choi, "Electricity theft detection using supervised learning techniques on smart meter data," *Sustainability*, vol. 12, no. 19, article 8023, 2020.

[18] A. Ullah, N. Javaid, O. Samuel, M. Imran, and M. Shoaib, "CNN and GRU based deep neural network for electricity theft detection to secure smart grid," in *2020 International Wireless Communications and Mobile Computing (IWCMC)*, Limassol, Cyprus, 2020.

[19] "State grid corporation of China," https://github.com/henryRDlab/ElectricityTheftDetection.

[20] M. Hasan, R. N. Toma, A. A. Nahid, M. M. M. Islam, and J. M. Kim, "Electricity theft detection in smart grid systems: a CNN-LSTM based approach," *Energies*, vol. 12, no. 17, p. 3310, 2019.

[21] S. K. Singh, R. Bose, and A. Joshi, "Energy theft detection for AMI using principal component analysis based reconstructed data," *IET Cyber-Physical Systems: Theory & Applications*, vol. 4, no. 2, pp. 179–185, 2019.

[22] J. V. Spiric, S. S. Stankovic, and M. B. Docic, "Identification of suspicious electricity customers," *International Journal of Electrical Power & Energy Systems*, vol. 95, pp. 635–643, 2018.

[23] K. V. Blazakis, T. N. Kapetanakis, and G. S. Stavrakakis, "Effective electricity theft detection in power distribution grids using an adaptive neuro fuzzy inference system," *Energies*, vol. 13, no. 12, p. 3110, 2020.

[24] A. Grewal, M. Kaur, and J. H. Park, "A unified framework for behaviour monitoring and abnormality detection for smart home," *Wireless Communications and Mobile Computing*, vol. 2019, Article ID 1734615, 16 pages, 2019.

[25] X. Lu, Z. Yu, Z. Wang, Y. Yi, L. Feng, and F. Wang, "Knowledge embedded semi-supervised deep learning for detecting non-technical losses in the smart grid," *Energies*, vol. 12, no. 18, p. 3452, 2019.

[26] N. Ding, H. X. Ma, H. Gao, Y. H. Ma, and G. Z. Tan, "Real-time anomaly detection based on long short-term memory and Gaussian mixture model," *Computers & Electrical Engineering*, vol. 79, article 106458, 2019.

[27] N. F. Avila, G. Figueroa, and C.-C. Chu, "NTL detection in electric distribution systems using the maximal overlap discrete wavelet-packet transform and random undersampling boosting," *IEEE Transactions on Power Systems*, vol. 33, no. 6, pp. 7171–7180, 2018.

[28] C. C. O. Ramos, D. Rodrigues, A. N. de Souza, and J. P. Papa, "On the study of commercial losses in Brazil: a binary black hole algorithm for theft characterization," *IEEE Transactions on Smart Grid*, vol. 9, no. 2, pp. 676–683, 2016.

[29] K. Zheng, Q. Chen, Y. Wang, C. Kang, and Q. Xia, "A novel combined data-driven approach for electricity theft detection," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 3, article 18091819, 2018.

[30] C. Li, X. Zheng, Z. Yang, and L. Kuang, "Predicting short-term electricity demand by combining the advantages of ARMA and XGBoost in fog computing environment," *Wireless Communications and Mobile Computing*, vol. 2018, Article ID 5018053, 18 pages, 2018.

[31] G. Fenza, M. Gallo, and V. Loia, "Drift-aware methodology for anomaly detection in smart grid," *IEEE Access*, vol. 7, pp. 9645–9657, 2019.

[32] K. Wang, C. Xu, and S. Guo, "Big data analytics for price forecasting in smart grids," in *2016 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6, Washington, DC, USA, 2016.

[33] H. Liu, G. Tianlong, Y. Liu, J. Song, and Z. Zeng, "Fault-tolerant privacy-preserving data aggregation for smart grid," *Wireless Communications and Mobile Computing*, vol. 2020, Article ID 8810393, 10 pages, 2020.

[34] A. Jamil, T. A. Alghamdi, Z. A. Khan et al., "An innovative home energy management model with coordination among appliances using game theory," *Sustainability*, vol. 11, no. 22, p. 6287, 2019.

[35] G. Micheli, E. Soda, M. T. Vespucci, M. Gobbi, and A. Bertani, "Big data analytics: an aid to detection of non-technical losses in power utilities," *Computational Management Science*, vol. 16, no. 1-2, pp. 329–343, 2019.

[36] M. Adil, N. Javaid, U. Qasim, I. Ullah, M. Shafiq, and J.-G. Choi, "LSTM and bat-based RUSBoost approach for electricity theft detection," *Applied Sciences*, vol. 10, no. 12, p. 4378, 2020.

[37] H. Gul, N. Javaid, I. Ullah, A. M. Qamar, M. K. Afzal, and G. P. Joshi, "Detection of non-technical losses using SOSTLink and bidirectional gated recurrent unit to secure smart meters," *Applied Sciences*, vol. 10, no. 9, p. 3151, 2020.

[38] M. Ismail, M. F. Shaaban, M. Naidu, and E. Serpedin, "Deep learning detection of electricity theft cyber-attacks in renewable distributed generation," *IEEE Transactions on Smart Grid*, vol. 11, no. 4, pp. 3428–3437, 2020.

[39] A. Thengade and R. Dondal, "Genetic algorithm-survey paper," in *MPGI National Multi Conference*, Citeseer, 2012.

[40] J. Genlin, "Survey on genetic algorithm," *Computer Applications and Software*, vol. 2, no. 1, pp. 69–73, 2004.

[41] T. G. Dietterich, "Approximate statistical tests for comparing supervised classification learning algorithms," *Neural Computation*, vol. 10, no. 7, pp. 1895–1923, 1998.

[42] E. Alpaydin, "Combined 5×2 cv F test for comparing supervised classification learning algorithms," *Neural Computation*, vol. 11, no. 8, article 18851892, 1999.

Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 8588646, 11 pages
https://doi.org/10.1155/2021/8588646

Research Article

Congestion-Aware Routing Algorithm for NoC Using Data Packets

Khurshid Ahmad,[1] Muhammad Athar Javed Sethi,[1] Rehmat Ullah,[1] Imran Ahmed,[2] Amjad Ullah,[3] Naveed Jan,[4] and Ghulam Mohammad Karami[5]

[1]*Department of Computer Systems Engineering, University of Engineering and Technology, Peshawar 25000, Pakistan*
[2]*Center of Excellence in IT, Institute of Management Sciences, Peshawar 25000, Pakistan*
[3]*Department of Electrical Engineering, University of Engineering and Technology, Peshawar 25000, Pakistan*
[4]*Department of Information Engineering Technology, University of Technology, Nowshera 24100, Pakistan*
[5]*SMEC International Pvt. Limited, Kabul 1007, Afghanistan*

Correspondence should be addressed to Ghulam Mohammad Karami; ghulam.karami@smec.com

Received 17 May 2021; Revised 26 July 2021; Accepted 10 August 2021; Published 19 August 2021

Academic Editor: Sungchang Lee

Network on Chip (NoC) is a communication framework for the Multiprocessor System on Chip (MPSoC). It is a router-based communication system. In NoC architecture, nodes of MPSoC are communicating through the network. Different routing algorithms have been developed by researchers, e.g., XY, intermittent XY, DyAD, and DyXY. The main problems in these algorithms are congestion and faults. Congestion and faults cause delay, which degrades the performance of NoC. A congestion-aware algorithm is used for the distribution of traffic over NoC and for the avoidance of congestion. In this paper, a congestion-aware routing algorithm is proposed. The algorithm works by sending congestion information in the data packet. The algorithm is implemented on a 4×4 mesh NoC using FPGA. The proposed algorithm decreases latency, increases throughput, and uses less bandwidth in sharing congestion information between routers in comparison to the existing congestion-aware routing algorithms.

1. Introduction

Semiconductors revolutionized the microelectronics world in all aspects that include the military, computers, medical field, telecommunication, and aerospace. The increasingly lower cost per transistor increases the transistor's integration up to 20 million transistors per chip because of the industrial revolution that leads to System on Chip (SoC) in the application area of military, computer peripheral, DSP, communication, and multiprocessor. SoC's functional unit connectivity is a complex task because it is more important for the system's overall performance [1, 2].

SoC is a bus-based system, and because of nonscalability, it cannot fulfill the requirement of performance. In SoC, the bandwidth of the bus is shared between devices, and it is insufficient. Based on this reason, in 2000, a new generic interconnection template was proposed, which addresses the performance and scalability requirement for SoC, using a switching network [3, 4]. In this network, data is moved from one terminal to another in a small format called a packet. A packet contains a header, payload, and tail flits. The header comprises destination information. The switching element is called a router. When the router receives the packet, it forwards it to a neighbor router based on its destination information.

Like a computer network, the routing algorithm in NoC can forward data flit (packet in NoC) from source to destination. For the avoidance of congestion and distribution of traffic over the network, the routing algorithms are used in NoC [5]. Different techniques are used by researchers in

the development of routing algorithms to control congestion in NoC [6].

A congestion-aware routing algorithm is developed by sending congestion information in data packets across NoC, which controls congestion. The proposed algorithm decreases latency, increases throughput, and uses less bandwidth in sharing congestion information between routers in comparison to existing congestion-aware algorithms.

2. Literature Review

The employed routing algorithm and topology of NoC affect the overall performance. The NoC routing algorithm is the same as the routing algorithm of a computer network with area and cost constraints [5]. The routing algorithm is the set of rules used by the NoC router to traverse a packet from a source to a destination [7–9]. XY, West first, and random routing algorithms are oblivious congestion algorithms [10, 11]. In these algorithms, routing decisions do not depend on the network's status which affects the performance of NoC. Unlike oblivious congestion algorithms, congestion-aware algorithms consider network congestion information when sending data flit from source to destination [12–14].

The DyAD routing algorithm integrates the benefit of adaptive and deterministic routing algorithms [15]. If no congestion is in a neighbor router, then routing is done through deterministic routing. Otherwise, adaptive routing is used for sending flits across NoC. In DyAD, they are flipping between routing schemes based on local congestion information. If the source sends data flit to a destination, the source router selects deterministic routing based on local congestion information. However, if congestion occurs in the neighbor router, then the latency of the flit is increased significantly. In the DyXY routing algorithm, data flit travels through the shortest path [16]. When there are several shortest routes from source to destination, then path selection is based on local congestion information. The DyXY algorithm's weakness is when there are two paths between source and destination (e.g., path1 and path2); then based on congestion information, if there is no congestion in the neighbor router of path1, the DyXY algorithm selects path1 to traverse data flit. However, if in path1 the router next to the neighbor router is congested, then the algorithm chooses the wrong path [17].

The EDXY routing algorithm overcomes the problem of the DyXY algorithm [18]. To share congestion information between routers over row (or column), a congestion wire is used. Through congestion wire, the congestion flags are spread on the path of row (or column), which tells the adjacent row (or column) that this row is congested or not. For implementation, extra hardware is used, which increases area and power consumption. The function of the FADyAD algorithm is the same as DyAD. FADyAD has the same problem as DyAD [19].

In 2014, a congestion controlling routing algorithm based on a dynamic routing table in NoC was developed [20]. After every 10 seconds, routers broadcast their routing table to neighbor routers. At the time of sharing the routing table, no data flits are sent which affects the latency. Another issue is that a more congested node is declared faulty. In regional congestion-aware routing, congestion information is shared between routers when links are idle [21]. When links are not idle, no congestion information is shared between routers. When a network is loaded, and congestion occurs at any router; then, delivery of data flit to a destination is compromised [22].

In MCAR, without a data packet, two types of messages propagate between routers for congestion avoidance [23]. One is normal (0), and the other is congested (1). Each router has four neighbors (north, south, east, and west). For example, when the input port on the north side of the router is crowded from normal, the output port on the north side of the router sends a congested (1) message to the north side router without taking any turns as this causes delay [24].

The congestion-aware, fault-tolerant, and process variation adaptive routing algorithm (CFPA) was developed in 2019 [25]. CFPA avoids congestion based on two routing tables (queuing delay table and propagation delay table) which are maintained in each router of NoC. Overhead in CFPA is periodic broadcasting queuing delay information through queuing delay message (QD_MSGs), and extra memory is required for storage. One other issue is the calculation for finding a path which requires more processing.

In MCAR, when sending a normal or congested message to a neighbor, data flit moving in that direction waits until a normal or congested message is sent. Here is the possibility to attach a normal or congested message to data flits and send it to a neighbor router. In CFPA, without sending QD_MSGs, the delay information is periodically attached to the data flit and sent to a neighbor.

3. Methodology

3.1. Congestion Information Calculation. In [26], a delay model is proposed in which input and output buffer delay and calculation of the router delay are not considered. In the CFPA routing algorithm, a new model of delays is proposed [20]. In this model, they consider propagation delay of the packet and queuing delay in its input port and output port to calculate congestion information. Equation (1) shows the total delay:

$$T_D = P_D + Q_D, \tag{1}$$

where Q_D is the queuing delay and P_D is the propagation delay. The propagation delay is calculated based on links from the sender router output port to receiving router input port. The queuing delay is the delay that how many packets stay in the sender router output port and an input port of the receiving router.

In the proposed routing algorithm, both propagation delay and queuing delay for congestion information calculation are used. Each router calculates its propagation delay

and queuing delay. The total delay is the time in which the flit is staying on the router.

3.2. Propagation Delay. The propagation delay is the time interval taken by the packet when passing through the router. Propagation delay calculation starts when a packet enters into an input port up to when the packet is sent through the output port without waiting in a queue. P_D is the propagation delay of the router, and the components of propagation delay are as below:

(i) Input port delay $= P_i$

(ii) Delay at RC (routing competition) unit of router $= P_{rc}$

(iii) Delay at the output port $= P_{out}$

(iv) When packet moves from the input port to the RC unit and from RC unit to output port, the delays $= 2P_{dl}$

Total propagation delay calculated from Equation (2) is the sum of all the delays at the input, output, routing competition unit, and a link between these components:

$$P_D = \sum_{i=0}^{3} (P_i + P_{rc} + P_{out} + 2P_{dl})_i. \quad (2)$$

3.3. Queuing Delay. Queuing delay refers to the time that a packet can take in the queue of the input port and output port of the router. A router has four input ports and four output ports for communication with neighbor routers. The total queuing delay is the sum of all the time that flit can take on the input and output ports. So, the total queuing delays are

$$Q_D = \sum_{i=0}^{3} (Q_{IDi} + Q_{ODi}). \quad (3)$$

In Equation (3), Q_D is the total queuing delay of the router. Q_{IDi} and Q_{ODi} are the delays at their input port and output port, respectively. The components of the queuing delay of a router are illustrated in Figure 1. The total delay is calculated from Equation (1), and the average of T_D is used as congestion information in the proposed routing algorithm. On this base, the router was declared to be congested or not congested. This T_{TD} is calculated on the router and embedded in a flit to share congestion information between the routers.

3.4. Flit Pattern. An NoC flit (flow control unit or flow control digit) is the smallest unit of a packet that carries data from source to destination. The packet is divided into flit. Flit size influences the capability of NoC. When flit size is small, it requires a smaller memory/buffer for storage over a network or on the router. When buffer sizes are small, it shrinks the size of the router, so the size of a network on chip is decreased, but it increases the power consumption. On the other hand, when the flit size is large, the size of the router is increased, which increases the overall area of a network on chip. Flit size for different NoC structures varies, starting from 8 bits to 256 bits. The flit size is 64 bits in the proposed algorithm.

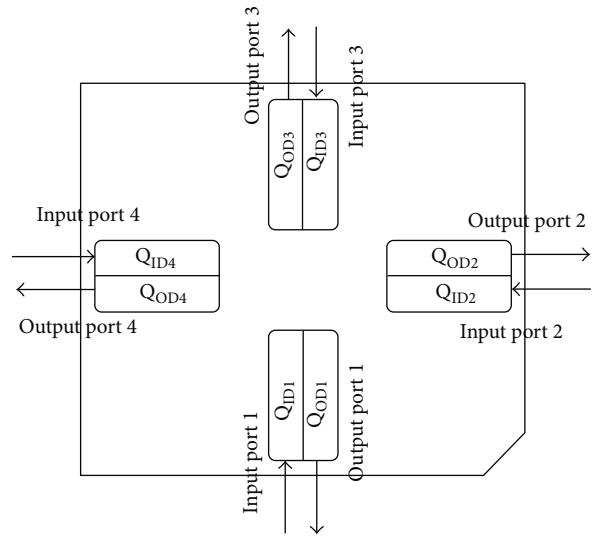


Figure 1: Q delay components.

The communicating data between two nodes is first divided into a packet, and then, the packet contains multiple flits. Flit contains head flit, body flit, and tail flit. Figure 2 shows the structure of the flit. In Figure 2, the first three bytes are head flit, the next 4 bytes are body flit, and the last one is tail flit.

3.5. Head Flit. Head flit contains the source address, destination address, and congestion information. Figure 3 shows a structural organization of the head flit. The first byte is the source address. In the source address, the first four bits show the location of a router on the x-axis, and the last four bits show the location on the y-axis in NoC. The second byte specifies the destination address. The source and the destination address are the same or not changing when the flit is on a route over the network. The last byte of the head flit contains congestion information that is shared between two neighbor routers.

The congestion information is detached and attached when the flit moves from one router to another. The first four bits of congestion information contain the total delay of a current router which is calculated from Equation (1). The last four bits have the conservation status of the router. Each router has four neighbors. The last four bits of congestion information have the congestion status of neighbor routers, and each bit shows the status of one router. Each bit has to state either 0 or 1, 1 for congested and 0 for an uncongested router.

3.6. Body Flit. The body flit has the data, which is sent from source to destination via the network. The size of the body flit is 4 bytes. The data in flit is placed in sequence order.

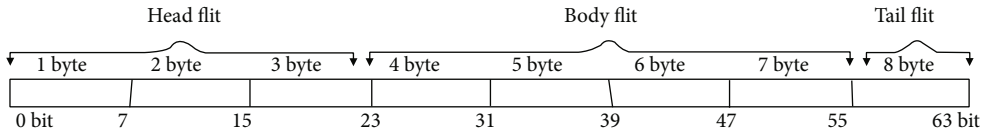


Figure 2: Flit structure.

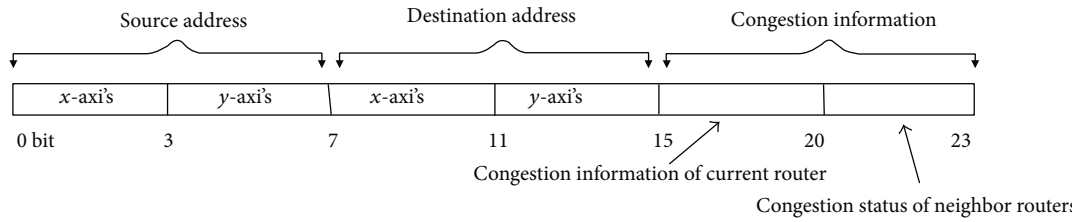


Figure 3: Head flit.

3.7. Tail Flit. Tail flit contains ending information of data and the sequence number of flit. The size of the tail flit is one byte. The seven bits starting from 0 to 6 in tail flit are the sequence number of flit. For example, the data size is 256 bits, which will be sent from source to destination. This data will be sent through eight different flits (flit-0 to flit-7). The flit sequence number shows the flit number on which the data is rearranged at its destination. The maximum number of flit in a single packet is up to 2^7. The size of data in a packet is up to $2^7 \times 4$ bytes.

The last bit of the tail flit shows the ending of a packet. When the bit is on, it means this is the last flit. Otherwise, it is not the last flit of sent data to the destination.

3.8. Proposed Routing Algorithm. The proposed routing algorithm is a congestion-aware routing in which the congestion information is shared between the routers through data packets/flits. The routing decision on sending data from source to destination is taken based on the congestion information of neighbor routers. This algorithm calculates router congestion information according to Equation (1), then shares it with the neighbor router when the packet/flit moves from the current router to the neighbor router. The congestion information is embedded in the head flit and moves to the neighbor router on the route. The router detaches the packet's congestion information and attaches its congestion information and sends the packet through the route.

The proposed routing algorithm for routing decisions relies on the destination address and congestion information table (CITb). CITb stores the congestion information of the neighbor router and next-door neighbor router congestion status. The node connectivity between routers in mesh 4×4 topology is the following.

(i) Corner router = 2

(ii) Boundary router = 3

(iii) Centre router = 4

Each router stores four neighbors' congestion information and their neighbor congestion status based on the CITb. Let router R connect to four routers on his north, south, east, and west. Router R CITb contains congestion information of the northside route and the northside router neighbor's congestion status. Router R also stores congestion information of all other side routers and the congestion status of their neighbors. Table 1 shows the CITb. The CITb is updated when the packet passes through the router. This routing algorithm has two parts: congestion information sharing and routing decisions.

Table 1: Congestion information table (CITb).

S.No.	Neighbor router location	Congestion information of the neighbor router	Congestion status of the next-door neighbor router			
			North	East	West	South
1	North	4 bits	1 bit	1 bit	1 bit	1 bit
2	East	4 bits	1 bit	1 bit	1 bit	1 bit
3	West	4 bits	1 bit	1 bit	1 bit	1 bit
4	South	4 bits	1 bit	1 bit	1 bit	1 bit

3.9. Congestion Information Sharing. In this section, congestion information sharing is elaborated. The congestion information sharing algorithm calculates, stores, and shares the congestion information with the neighbor router. The congestion information sharing process cannot require a congestion propagation network. The routers' congestion information is shared when the head flit passes from the router to its neighbor. Figure 4 shows the steps of congestion information sharing for each packet.

The congestion information sharing process relies on the CITb. When the head flit is received on the router, it reads and writes the congestion information byte of the head flit. In reading, it collects congestion information and stores congestion information in CITb with respect to the location of the sender router. In writing, the router calculates congestion information of its own and

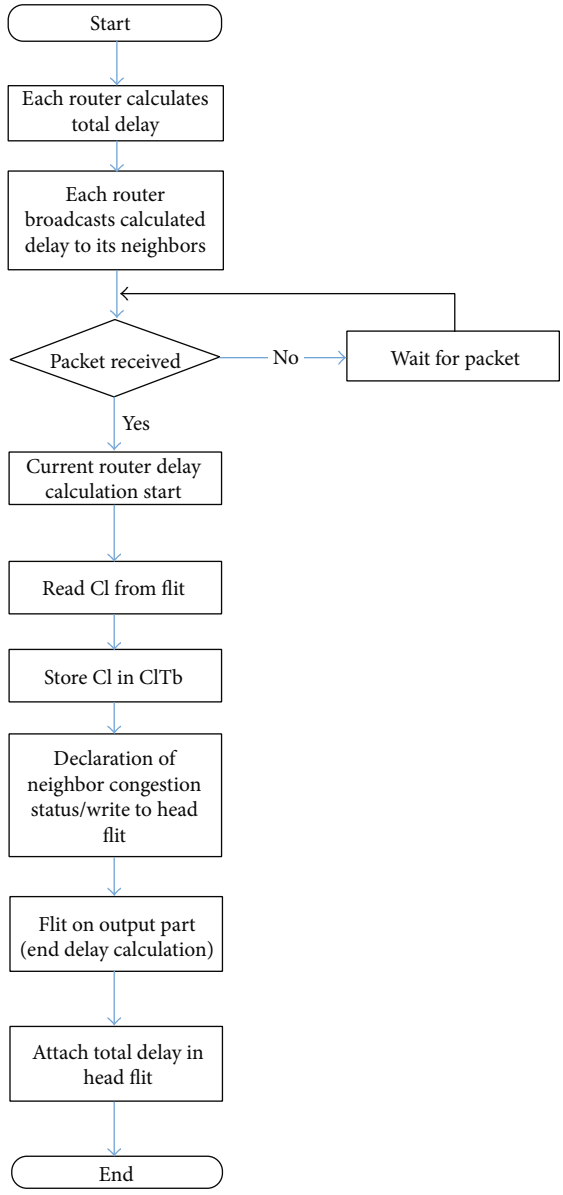


Figure 4: Congestion information sharing steps.

congestion status of its neighbor from stored congestion information in the CITb.

The congestion information of the router and the total delay of the router is calculated through Equation (1). The congestion status of the router means the router is congested or not. The router declares if its neighbor is congested or not and is dependent on the congestion information of that neighbor router which is stored in the CITb. The router will be declared congested when delay/100 > 0.70.

When the router is declared congested, its status is set to 1; otherwise, it is zero. The congestion status of the neighbor router is attached in the congestion information byte of the head flit whenever routing decisions are finalized, and congestion information of the router is attached, when the head flit is on the output port of the router. After the attachment of congestion information byte data in the head flit, the flit is sent to the neighbor router on the route.

3.10. Routing Decision. The routing decision of the proposed routing algorithm depends on the destination address and congestion information stored in the CITb. The routing decision of the proposed routing algorithm has two levels. In the first level, it finds the two alternative paths for flit, which can send the packet to a destination. There are two possible paths to a destination, one on the x-axis and the other on the y-axis. In this level, it compares the destination address of the flit with its address. In comparison, first, it compares the x-axis then the y-axis for finding the two alternative paths. In the second level, it selects the route for flit between these two alternative paths. The selection of paths between these two alternative routes depends on the CITb data. In order to finalize routing decisions, it checks the neighbor router's congestion status of alternative paths from the CITb. Comparison of the congestion status of the neighbor router of both alternative paths has three possibilities.

(i) When both neighbor routers are uncongested, select the route with low delay information of the neighbor router

(ii) When between these two neighbor routers one is congested and another is uncongested, then an uncongested router route is selected for flit routing

(iii) When both neighbor routers are congested, then compare the congestion status of the next-door neighbor of both routes

Comparing the congestion status of next-door neighbor routers has three possibilities.

(i) When both next-door neighbor routers of both routes are congested, select the route with low delay information of the neighbor router

(ii) When both next-door neighbor routers of both routes are uncongested, select the route with low delay information of the neighbor router

(iii) When one next-door neighbor router is uncongested and another is congested, the route is selected on which next-door neighbor router is uncongested

Figure 5 shows that the flit is routed from route $P(1,1)$ to destination router $Q(4,4)$. The route from P to Q is shown through blue arrow lines, and R-D is the delay of the router.

4. Results and Discussion

For the experimental setup, a 4×4 mesh topology NoC having flexible routers is modeled. For routing purposes, the proposed routing algorithm is used in simulator tool

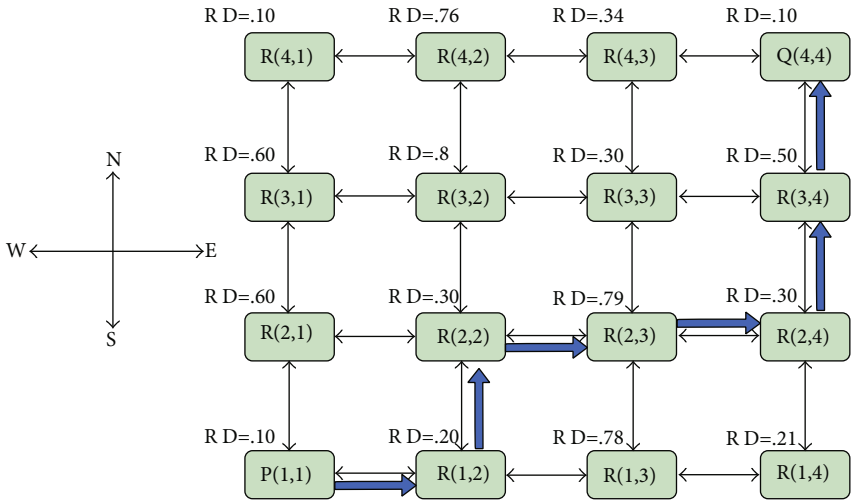


Figure 5: Routing from P to Q.

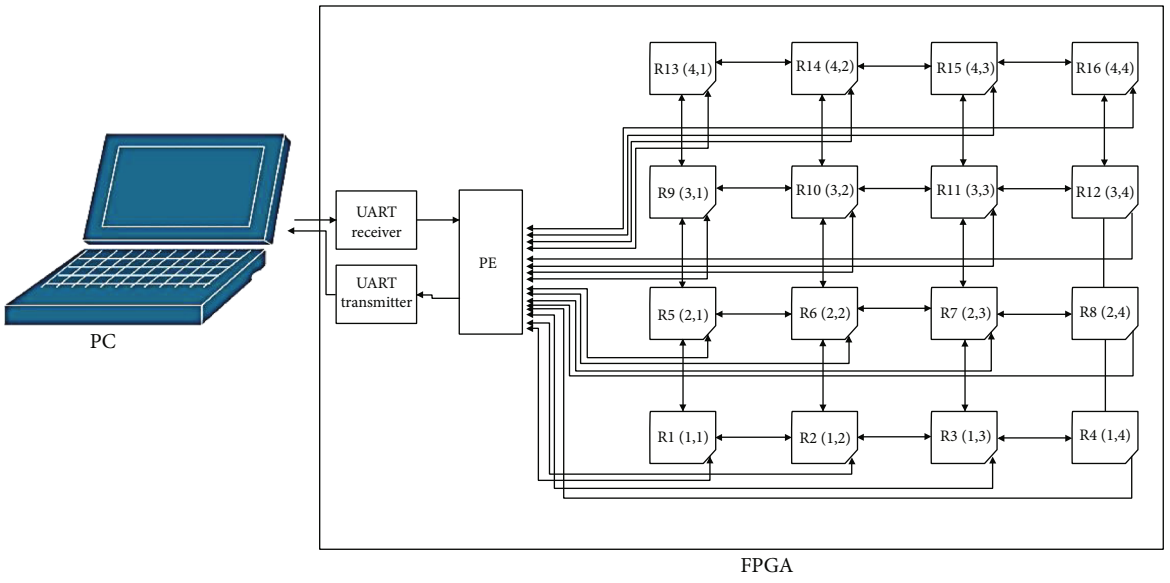


Figure 6: Experimental setup.

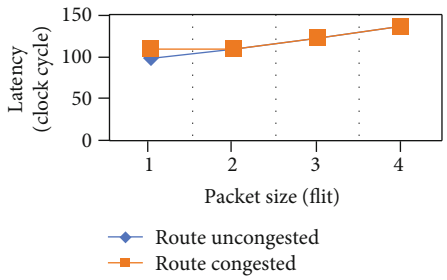


Figure 7: Latency from $R_{1\text{-}1}$ to $R_{1\text{-}2}$.

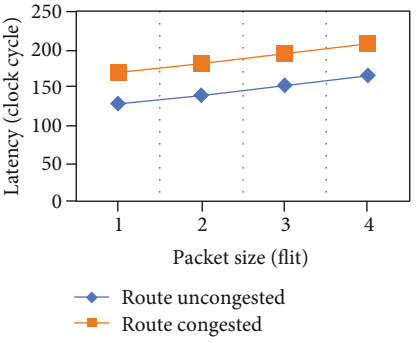


Figure 8: Latency from $R_{1\text{-}1}$ to $R_{3\text{-}1}$.

ISE 14.7 webpack through Verilog language. Model NoC is implemented on the Spartan 6 XC6SLX9 FPGA development kit. UART is an intermediate node between NoC and PC for receiving and sending data from PC to FPGA kit. UART is also embedded/implemented in the FPGA kit and connected to NoC routers. Figure 6 shows the experimental setup and connectivity of PC, UART, PE, and 4×4 NoC.

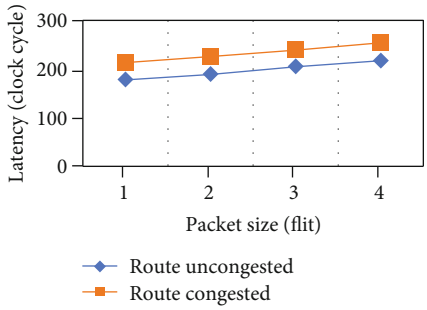


Figure 9: Latency from $R_{1\text{-}1}$ to $R_{4\text{-}1}$.

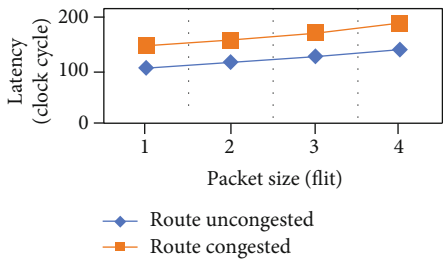


Figure 10: Latency from $R_{1\text{-}1}$ to $R_{1\text{-}2}$.

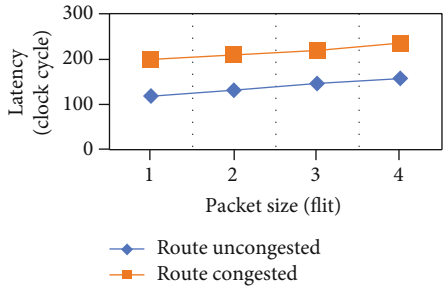


Figure 11: Latency from $R_{1\text{-}1}$ to $R_{2\text{-}2}$.

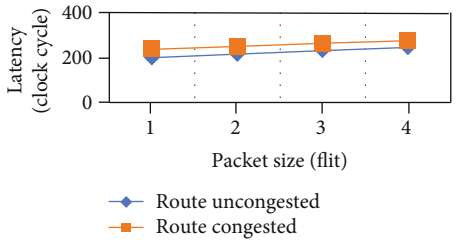


Figure 12: Latency from $R_{1\text{-}1}$ to $R_{3\text{-}2}$.

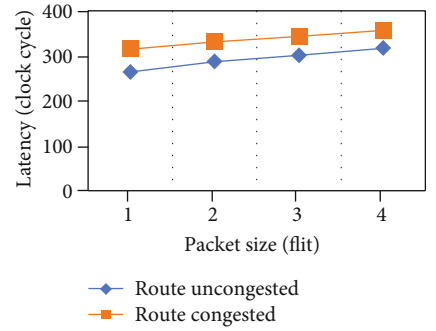


Figure 13: Latency from $R_{1\text{-}1}$ to $R_{4\text{-}2}$.

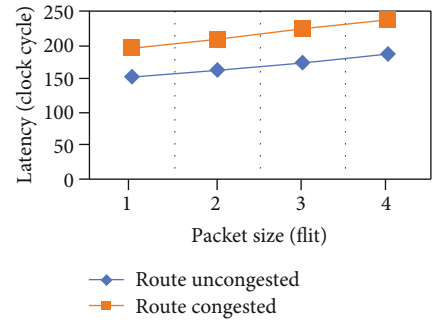


Figure 14: Latency from $R_{1\text{-}1}$ to $R_{1\text{-}3}$.

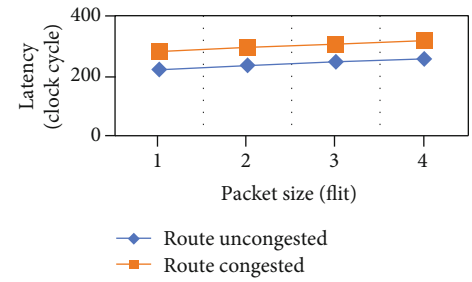


Figure 15: Latency from $R_{1\text{-}1}$ to $R_{2\text{-}3}$.

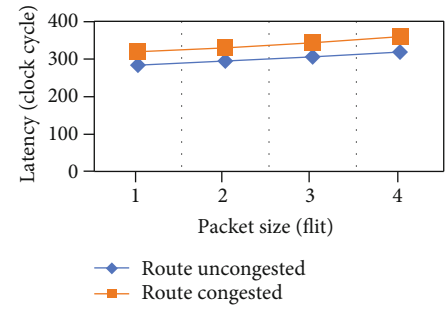


Figure 16: Latency from $R_{1\text{-}1}$ to $R_{3\text{-}3}$.

For the result evaluation, an experiment was conducted by sending several packets having a size from 1 flit to 4 flits from source $R_{1\text{-}1}$ to all other destinations of NoC shown in Figure 5, and the latency was monitored in both situations, without congestion and with congestion. The flit size is 64 bits, and the size of the buffer is also 64 bits. One buffer in a router can store one flit at a time. The channel width is 8 bits, and in one clock cycle, 12.55 packets can be processed. The latency was analyzed in the worst case. The latency was explored through the Chip Scope Pro Analyzer. Average latency from $R_{1\text{-}1}$ to all other routers are shown in line graphs in Figures 7–21 and in bar graphs in Figures 22–36. In Figures 21–36, the blue line in the line graph and blue bar in the bar graph show the average latency from $R_{1\text{-}1}$ to all other routers when the route from $R_{1\text{-}1}$ to destination is uncongested. The red line and red bar show average latency

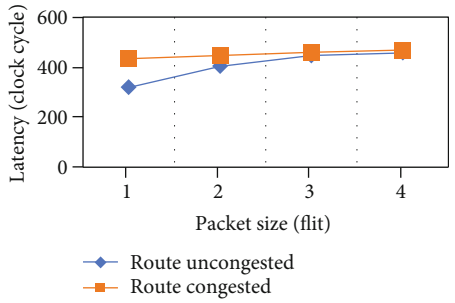


Figure 17: Latency from $R_{1\text{-}1}$ to $R_{4\text{-}3}$.

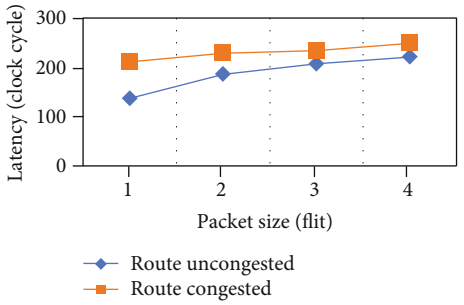


Figure 18: Latency from $R_{1\text{-}1}$ to $R_{1\text{-}4}$.

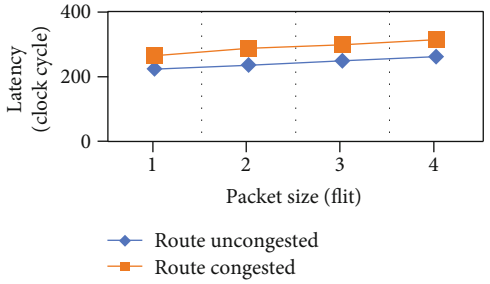


Figure 19: Latency from $R_{1\text{-}1}$ to $R_{2\text{-}4}$.

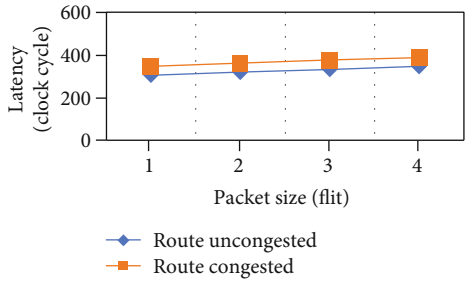


Figure 20: Latency from $R_{1\text{-}1}$ to $R_{3\text{-}4}$.

when the route from $R_{1\text{-}1}$ to destination is congested. The latency depends on the route length or the number of nodes between source and destination, congestion status of the route, and packet size.

The node between $R_{1\text{-}1}$ to $R_{1\text{-}2}$ and $R_{2\text{-}1}$ is 0. The highest number of nodes in this mesh topology from $R_{1\text{-}1}$ to $R_{4\text{-}4}$ is 5

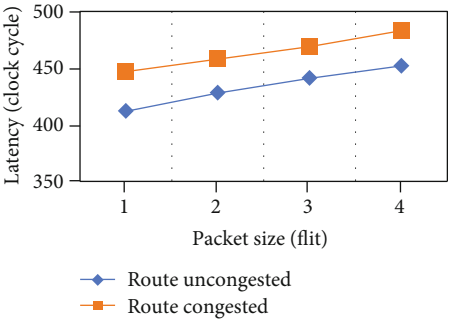


Figure 21: Latency from $R_{1\text{-}1}$ to $R_{4\text{-}4}$.

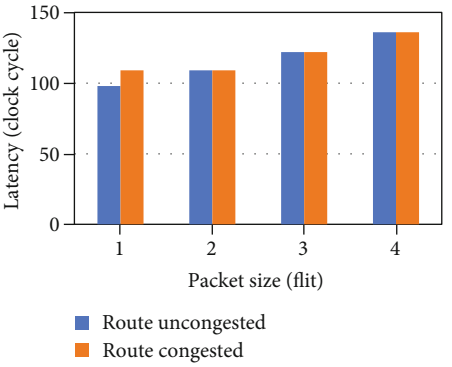


Figure 22: Latency from $R_{1\text{-}1}$ to $R_{2\text{-}1}$.

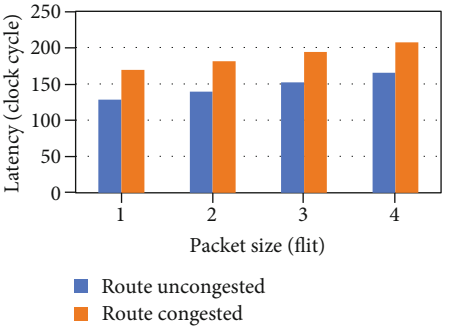


Figure 23: Latency from $R_{1\text{-}1}$ to $R_{3\text{-}1}$.

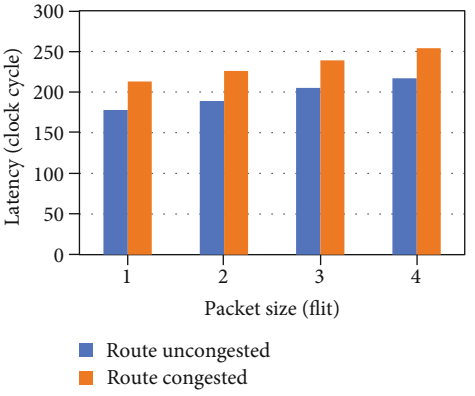


Figure 24: Latency from $R_{1\text{-}1}$ to $R_{4\text{-}1}$.

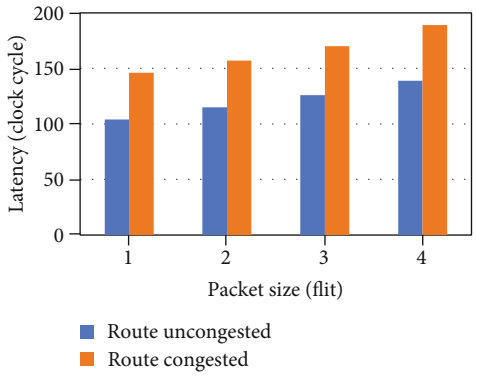


Figure 25: Latency from $R_{1\text{-}1}$ to $R_{1\text{-}2}$.

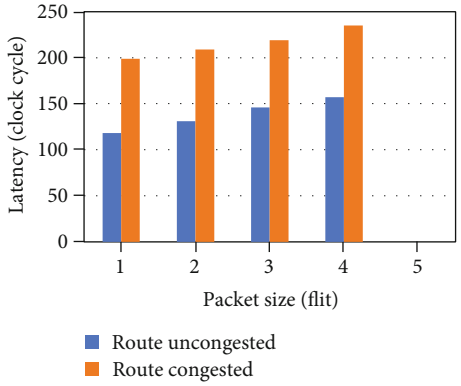


Figure 26: Latency from $R_{1\text{-}1}$ to $R_{2\text{-}2}$.

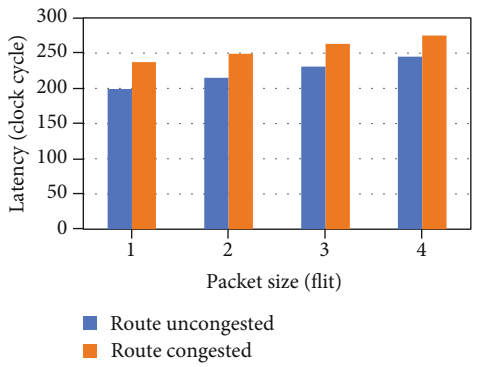


Figure 27: Latency from $R_{1\text{-}1}$ to $R_{3\text{-}2}$.

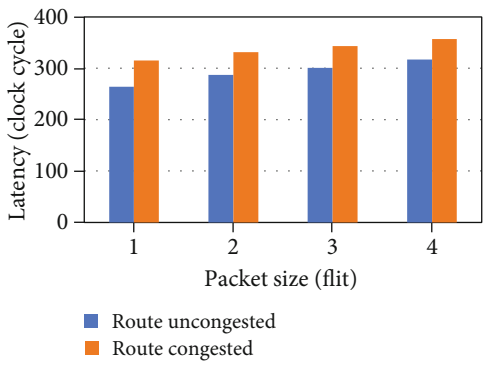


Figure 28: Latency from $R_{1\text{-}1}$ to $R_{4\text{-}2}$.

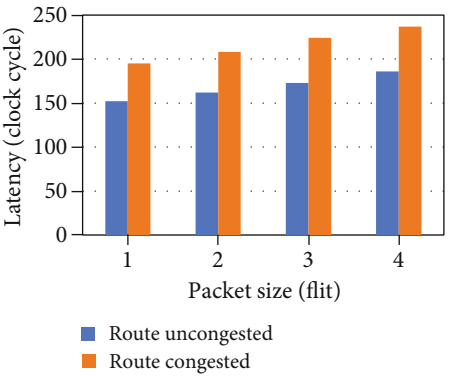


Figure 29: Latency from $R_{1\text{-}1}$ to $R_{1\text{-}3}$.

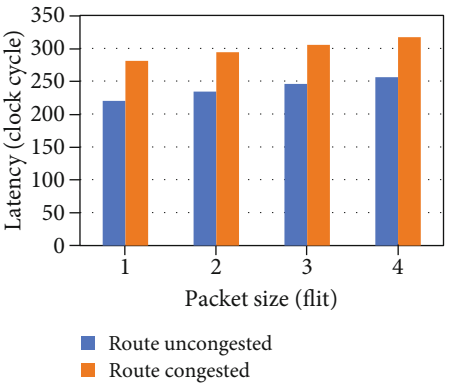


Figure 30: Latency from $R_{1\text{-}1}$ to $R_{2\text{-}3}$.

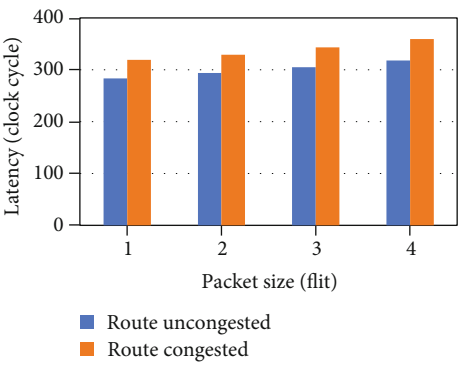


Figure 31: Latency from $R_{1\text{-}1}$ to $R_{3\text{-}3}$.

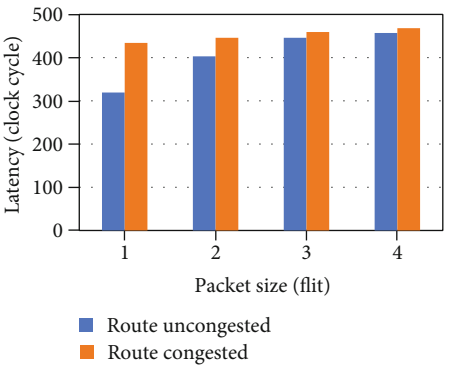


Figure 32: Latency from $R_{1\text{-}1}$ to $R_{4\text{-}3}$.

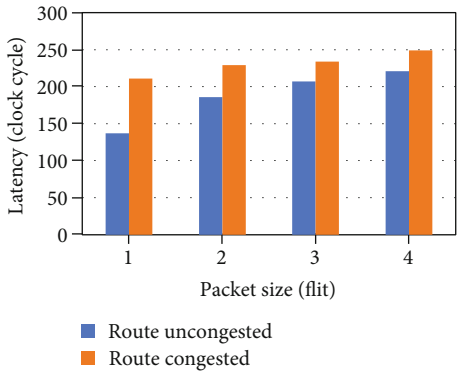


Figure 33: Latency from $R_{1\text{-}1}$ to $R_{1\text{-}4}$.

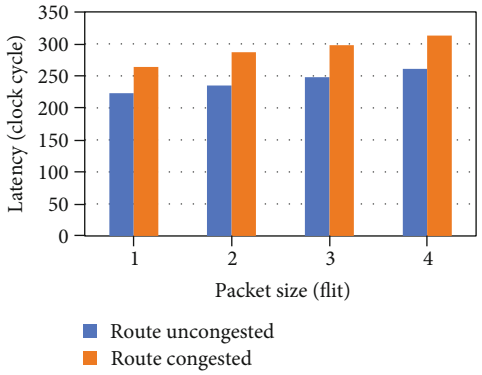


Figure 34: Latency from $R_{1\text{-}1}$ to $R_{2\text{-}4}$.

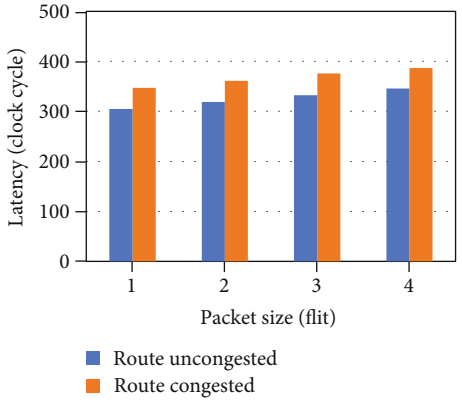


Figure 35: Latency from $R_{1\text{-}1}$ to $R_{3\text{-}4}$.

when a route is uncongested. The latency is increased when the number of nodes between source and destination increases. The latency of congested routes is higher than uncongested route latency because of computation in a router for finding an uncongested route from source to destination. In Figures 21–36, a congested route means finding an alternate uncongested route by the router based on shared congestion information when the first shortest route to a destination is congested.

In the experiment for the calculation of results, packet size is from 1 flit to 4 flits, and the size of flit is 64 bits.

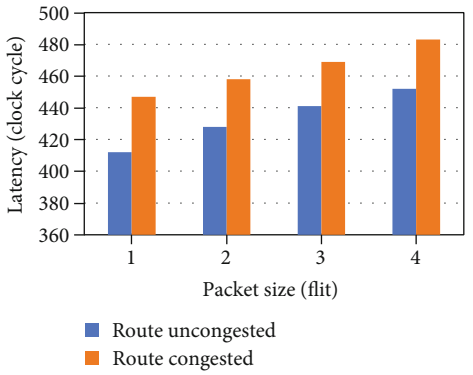


Figure 36: Latency from $R_{1\text{-}1}$ to $R_{4\text{-}4}$.

The latency in Figures 21–36 increases with the packet size.

5. Conclusion

In this paper, a congestion-aware routing algorithm is proposed in which the congestion information between routers is shared through a data flit. The routing decision in the router from source to destination is based on the congestion information of neighbor routers. The proposed routing algorithm is a congestion-aware adaptive routing algorithm. The routing algorithms in the literature use a separate network to share congestion information which increases the area of NoC. In few routing algorithms, congestion information is transmitted through the data line, and during congestion information sharing time, the data flits are not sent. In the proposed routing algorithm, the congestion information between routers is shared when the date flit traverses over the network. The proposed routing algorithm distributes the traffic over the network to avoid congestion in the router of NoC. The route from source to destination is dependent on basis of the congestion status of the NoC router. The packet/flit in the proposed routing algorithm is sent on an uncongested route or the route which has low congestion compared to other routes to a destination. From the experiment results, the average latency is calculated in both conditions, congested route and uncongested route in the worst case. The latency decreases up to approximately 50-70% when we increase the buffer size in router and channel width.

Data Availability

The data that support the findings of this study are available from the corresponding author, Ghulam Mohammad Karami (ghulam.karami@smec.com), upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

[1] R. R. Tummala and V. K. Madisetti, "System on chip or system on package?," *IEEE Design & Test of Computers*, vol. 16, no. 2, pp. 48–56, 1999.

[2] T. A. Claasen, "An industry perspective on current and future state of the art in system-on-chip (SoC) technology," *Proceedings of the IEEE*, vol. 94, no. 6, pp. 1121–1137, 2006.

[3] W. J. Dally and B. Towles, "Route packets, not wires: on-chip inteconnection networks," in *Proceedings of the 38th annual design automation conference*, pp. 684–689, Las Vegas, NV, USA, 2001.

[4] S. Kumar, A. Jantsch, J. P. Soininen et al., "A network on chip architecture and design methodology," in *Proceedings IEEE computer society annual symposium on VLSI. New Paradigms for VLSI Systems Design. ISVLSI 2002*, pp. 117–124, Pittsburgh, PA, USA, 2002.

[5] G. Adamu, P. Chejara, and A. B. Garko, "Review of deterministic routing algorithm for network-on-chip," in *2nd international conference on science, technology and management*, pp. 741–745, India, 2015.

[6] M. A. J. Sethi, F. A. Hussin, and N. H. Hamid, "Review of network on chip architectures," *Recent Advances in Electrical & Electronic Engineering (Formerly Recent Patents on Electrical & Electronic Engineering)*, vol. 10, pp. 4–29, 2017.

[7] A. V. de Mello, L. C. Ost, F. G. Moraes, and N. L. V. Calazans, *Evaluation of Routing Algorithms on Mesh Based NoCs*, PUCRS, Av. Ipiranga, 2004.

[8] A. Mejia, M. Palesi, J. Flich et al., "Region-based routing: a mechanism to support efficient routing algorithms in NoCs," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 17, no. 3, pp. 356–369, 2009.

[9] S. D. Chawade, M. A. Gaikwad, and R. M. Patrikar, "Review of XY routing algorithm for network-on-chip architecture," *International Journal of Computer Applications*, vol. 43, pp. 975–8887, 2012.

[10] A. Intel, *Touchstone Delta System Description*, Intel Corporation, 1991.

[11] C. J. Glass and L. M. Ni, "The turn model for adaptive routing," *ACM SIGARCH Computer Architecture News*, vol. 20, no. 2, pp. 278–287, 1992.

[12] M. Ebrahimi, M. Daneshtalab, P. Liljeberg, J. Plosila, and H. Tenhunen, "CATRA-congestion aware trapezoid-based routing algorithm for on-chip networks," in *2012 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pp. 320–325, Dresden, Germany, 2012.

[13] K. Ahmad and M. A. J. Sethi, "Review of network on chip routing algorithms," *EAI Endorsed Transactions on Context-aware Systems and Applications*, vol. 7, no. 22, p. 167793, 2020.

[14] S. Ganesan, V. Muthuswamy, G. Sannasi, and K. Arputharaj, "A comprehensive analysis of congestion control models in wireless sensor networks," in *Sensor Technology: Concepts, Methodologies, Tools, and Applications*, pp. 1194–1214, IGI Global, 2020.

[15] J. Hu and R. Marculescu, "DyAD: smart routing for networks-on-chip," in *Proceedings of the 41st annual Design Automation Conference*, pp. 260–263, New York, NY, USA, 2004.

[16] M. Li, Q.-A. Zeng, and W.-B. Jone, "DyXY: a proximity congestion-aware deadlock-free dynamic routing method for network on chip," in *Proceedings of the 43rd annual Design Automation Conference*, pp. 849–852, New York, NY, USA, 2006.

[17] G. Sangeetha, M. Vijayalakshmi, S. Ganapathy, and A. Kannan, "A heuristic path search for congestion control in WSN," in *Industry Interactive Innovations in Science, Engineering and Technology*, pp. 485–495, Springer, 2018.

[18] P. Lotfi-Kamran, A.-M. Rahmani, M. Daneshtalab, A. Afzali-Kusha, and Z. Navabi, "EDXY - a low cost congestion-aware routing algorithm for network-on-chips," *Journal of Systems Architecture*, vol. 56, no. 7, pp. 256–264, 2010.

[19] G. Sangeetha, M. Vijayalakshmi, S. Ganapathy, and A. Kannan, "An improved congestion-aware routing mechanism in sensor networks using fuzzy rule sets," *Peer-to-Peer Networking and Applications*, vol. 13, no. 3, pp. 890–904, 2020.

[20] B. Wang, H. Cai, F. Qu, and Y. Yang, "The congestion controlling algorithm based on dynamic routing table on network on chips," in *10th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM 2014)*, pp. 381–385, Beijing, China, 2014.

[21] C. Chen, Q. Li, N. Li, H. Liu, and Y. Dai, "Link-sharing: regional congestion aware routing in 2D NoC by propagating congestion information on idle links," in *2018 IEEE 3rd International Conference on Integrated Circuits and Microsystems (ICICM)*, pp. 291–297, Shanghai, China, 2018.

[22] M. Selvi, S. S. Kumar, S. Ganapathy, A. Ayyanar, H. K. Nehemiah, and A. Kannan, "An energy efficient clustered gravitational and fuzzy based routing algorithm in WSNs," *Wireless Personal Communications*, vol. 116, no. 1, pp. 61–90, 2021.

[23] R. Xie, J. Cai, X. Xin, and B. Yang, "MCAR: non-local adaptive network-on-chip routing with message propagation of congestion information," *Microprocessors and Microsystems*, vol. 49, pp. 117–126, 2017.

[24] S. Munuswamy, J. M. Saravanakumar, G. Sannasi, K. N. Harichandran, and K. Arputharaj, "Virtual force-based intelligent clustering for energy-efficient routing in mobile wireless sensor networks," *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 26, pp. 1444–1452, 2018.

[25] S. T. Muhammad, M. Saad, A. A. El-Moursy, M. A. El-Moursy, and H. F. Hamed, "CFPA: congestion aware, fault tolerant and process variation aware adaptive routing algorithm for asynchronous networks-on-chip," *Journal of Parallel and Distributed Computing*, vol. 128, pp. 151–166, 2019.

[26] S. T. Muhammad, M. A. El-Moursy, A. A. El-Moursy, and H. F. Hamed, "Architecture level analysis for process variation in synchronous and asynchronous networks-on-chip," *Journal of Parallel and Distributed Computing*, vol. 102, pp. 175–185, 2017.

Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 5552012, 22 pages
https://doi.org/10.1155/2021/5552012

Research Article

Self-Organized Efficient Spectrum Management through Parallel Sensing in Cognitive Radio Network

Muddasir Rahim[1], **Riaz Hussain**[1], **Irfan Latif Khan**[1], **Ahmad Naseem Alvi**[1], **Muhammad Awais Javed**[1], **Atif Shakeel**[1], **Qadeer Ul Hasan**[1], **Byung Moo Lee**[2] **and Shahzad A. Malik**[1]

[1]*Department of Electrical and Computer Engineering, COMSATS University Islamabad, Pakistan*
[2]*Department of Intelligent Mechatronics Engineering and Convergence Engineering for Intelligent Drone, Sejong University, Seoul 05006, Republic of Korea*

Correspondence should be addressed to Byung Moo Lee; blee@sejong.ac.kr

Received 22 February 2021; Revised 20 April 2021; Accepted 23 July 2021; Published 15 August 2021

Academic Editor: Stefano Savazzi

In this paper, we propose an innovative self-organizing medium access control mechanism for a distributed cognitive radio network (CRN) in which utilization is maximized by minimizing the collisions and missed opportunities. This is achieved by organizing the users of the CRN in a queue through a timer and user ID and providing channel access in an orderly fashion. To efficiently organize the users in a distributed, ad hoc network with less overhead, we reduce the sensing period through parallel sensing wherein the users are divided into different groups and each group is assigned a different portion of the primary spectrum band. This consequently augments the number of discovered spectrum holes which then are maximally utilized through the self-organizing access scheme. The combination of two schemes augments the effective utilization of primary holes to above 95%, even in impasse situations due to heavy primary network loading, thereby achieving higher network throughput than that achieved when each of the two approaches are used in isolation. By efficiently combining parallel sensing with the self-organizing MAC (PSO-MAC), a synergy has been achieved that affords the gains which are more than the sum of the gains achieved through each one of these techniques individually. In an experimental scenario with 50% primary load, the network throughput achieved with combined parallel sensing and self-organizing MAC is 50% higher compared to that of parallel sensing and 37% better than that of self-organizing MAC. These results clearly demonstrate the efficacy of the combined approach in achieving optimum performance in a CRN.

1. Introduction

The demand for spectrum has grown exponentially over the last two decades due to unprecedented growth and proliferation of wireless devices, systems, and services. With the advent of massive Internet of Things- (IoT-) based applications, this problem has further been aggravated. The availability of spectrum is very limited, primarily due to static spectrum allocation policies, where the spectrum is exclusively allocated to licensed users, which prohibits the unlicensed users to access that portion of the spectrum. According to the Federal Communication Commission (FCC), there is a huge variation in the spatial and temporal spectrum utilization ranging from 15% to 85% in the band below 3 GHz [1], leading to virtual spectrum scarcity [2, 3]. To overcome the imbalance between the high spectrum demand and limited spectrum availability, FCC has suggested dynamic spectrum access. Cognitive radio (CR) has emerged as a key technology that realizes the principle of dynamic spectrum access (DSA), by enabling simultaneous access to the underutilized portion of the licensed spectrum for users other than those of the licensed network [4, 5]. The users in the cognitive radio networks (CRNs) are required not only to locate the opportunities in the primary network but also have to ensure that users of the licensed network are not affected by them. This requires additional

functionality of spectrum handover or spectrum mobility as the primary user appears on the channel being opportunistically used by the CRN user. A wide range of techniques have been deployed for spectrum handover such as bioinspired techniques [6], Markov modeling [7], and supervised machine learning techniques [8].

Efficient spectrum utilization is the prime objective in CRN [9]. It requires an efficient discovery mechanism as well as maximal utilization of the discovered opportunities. To improve the channel utilization, the coordination among the CRN users, also called secondary users (SUs), is provided by introducing a common control channel (CCC) [10, 11]. However, it increases the overheads, which reduces the actual available transmission time to the SUs. To minimize these overheads for increased channel utilization, there have been various noteworthy research contributions in the recent past. To make the discovery efficient, the sensing time needs to be reduced and it has to be exhaustive and accurate too; this raised contradicting challenges [12–15]. Likewise, the access mechanism needs to be efficient so that opportunities are neither missed, wasted, nor inefficiently tapped [16, 17].

In an ad hoc CRN, this becomes even more challenging as there is no central entity to govern its users. In random access, unused channels of primary users (PUs) may remain underutilized either due to collision or remain unattempted. To minimize the wastage of discovered opportunities, a self-organizing collision-free CRN MAC has been proposed [18]. To create the pool of available resources, SUs perform cooperative sensing [19]. In another research, a parallel sensing scheme has been devised to maximize the channel time for data transmission [20]. Both schemes improve the performance in their respective domains. However, since both of these schemes require a coordinated effort among the CRN users and that too in an ad hoc network scenario, there is an associated overhead with each one of these two schemes.

To improve the efficiency, the existing MAC approaches for CRN either target to improve the utilization of the discovered holes or target the efficient discovery mechanism. Combining the two goals penalize one another due to excessive overhead, particularly in the ad hoc CRNs employing common control channel for coordination among the SUs. The goal is to provide a unified common control channel frame structure that facilitates to improve the utilization, maximize the hole discoveries, and extend the transmission time in a transmission cycle and adjusts dynamically to the varying network conditions.

In this work, parallel sensing along with self-organizing medium access control (PSO-MAC) has been proposed, which is an extension of work in [18, 20]. PSO-MAC provides parallel channel sensing and self-organizing channel access mechanism through a single unified frame structure, which improves channel utilization as well as idle channel discovery process and hence increases the overall network throughput. Parallel channel sensing can also be used with contention-based MAC protocols. The primary motivation for this work lies in the development of a unified frame structure that allows to harness the benefits of both approaches and yet keeps the overheads to a minimum. This requires removing redundancies, careful sequencing of different phases, and provisioning of dynamic adjustment according to the situation. The goal is to maximize the discovered resources and utilization of these resources while leaving maximum time for the secondary users for data transmission. The secondary users can transmit their data in the transmission cycle only after the allocation of a channel to the user and the rest of the period is overhead. Further, the existing parallel sensing scheme reduces the duration of the sensing phase through parallelism.

Increasing the number of groups increases parallelism that further decreases the duration of the sensing phase. However, for a given number of secondary users in a CRN, it is at the cost of a higher probability of a primary band being skipped in sensing and thus reducing the number of discoveries. Through dynamic adjustment of the number of groups according to the prevailing situation in terms of load on the primary network and number of secondary users, the optimal trade-off can be achieved between the parallelism and the overhead. The enhanced unified frame structure along with optimization of the number of parallel sensing groups in the proposed PSO-MAC provides the synergy for optimum performance gains.

1.1. Contributions. The key contributions of this paper are highlighted below.

(1) A unified frame structure for CRN MAC has been proposed that combines parallel sensing with the organized medium access to improve the utilization and maximize the hole discoveries

(2) The proposed PSO-MAC has been carefully designed to keep the overhead of the frame structure of the common control channel to a minimum so that the transmission time in a transmission cycle is extended and a synergic effect is achieved. The order of different phases has been critical to keep the flexibility of the frame structure for varying network conditions

(3) We have developed a model to evaluate overall performance gain in terms of throughput with due consideration of the overhead of the frame structure. Comparison with existing schemes is based on this model

(4) Parallel sensing has been optimized by dynamically adjusting the number of groups in parallel sensing according to the changing network conditions

The rest of the paper is organized as follows. The literature review is discussed followed by the proposed PSO-MAC. Then, the performance metrics are explained and simulation environment and results are evaluated. Finally, conclusion is presented.

2. Literature Review

Many efforts have been dedicated to optimize the spectrum sensing in CRN [20–33]. Spectrum sensing schemes are mainly classified into two main categories: (a) narrow band and (b) wide band.

In the narrow-band spectrum sensing, the SUs sense any particular channel to see if it is available or busy. The availability or unavailability of a channel is determined through (i) energy detection, which measures the energy of the signal and matches it to a given threshold; if the signal energy is higher than the predefined threshold, then it is assumed that the PU is present; otherwise, the channel is considered as idle [21, 22]; (ii) matched filter detection, which compares the received signal with the pilot signal obtained from a similar transmitter [23]; and (iii) machine learning-based sensing methods, which are considered as a classification problem that may be supervised or unsupervised [24, 25].

Developing a reliable and highly efficient spectrum sensing technique is a challenging task. The sensing performance of an individual SU may be affected by noise, shadowing, and fading causing uncertainty. Cooperative sensing schemes enhance the performance of the CRN by improving the accuracy of spectrum sensing [26]. When multiple CRs sense the PU detection then their individual results are shared among the CRN to evaluate the best PU channel by employing different techniques such as unanimous cooperative decision and consensus-based cooperative sensing but at the cost of additional overhead.

Cooperative spectrum sensing (CSS) schemes with adaptive sensing window with due consideration of bit error rate (BER) to improve spectrum utilization with an increase in SNR have been proposed in [27]. In [28], the authors proposed an optimal channel sensing with heavy-tailed idle times for CRN, which were modeled as hyperexponential distribution (HED). The authors in [29] proposed an autonomous compressive sensing algorithm that enabled the local SUs to choose the number of compressive samples automatically to reduce the sensing duration along with reduced complexity of data processing. Double threshold (DT) cooperative spectrum sensing mechanism with hard-soft combining has been proposed in [30] to enhance the reliability of spectrum sensing and reduce the communication overhead. Similarly, the authors in [31] have proposed a hybrid double threshold cooperative spectrum sensing mechanism in CRN. Sensing performance is improved by exploiting both the local and global detection in terms of energy and binary values at each SU, given the interference caused to the PU at an acceptable level. An additional threshold is used when the local sensing decision at each SU is not available.

The development in wireless communication systems requires high data rates which can be achieved with high bandwidth. From this aspect, SUs need to sense a wide range of the primary spectrum to find the pool of available channels. In the case of wide band, the whole spectrum is divided into multiple channels and then they are sensed using the narrow-band spectrum sensing schemes, either randomly, sequentially, or in parallel [32]. In [20], the authors have proposed cooperative parallel sensing in CRN to discover the vacant primary channels. The idea is to make SU groups according to their IDs. Spectrum is equally divided into the same number of portions, and each group of SUs is assigned a portion to scan for the vacant channels. Sensing information is shared in a separate sharing phase. Due to parallel sensing of the assigned portion of the spectrum, the total sensing time is reduced and the maximum number of vacant PU channels is discovered. Parallel sensing improves the spectrum utilization and throughput by minimizing the sensing time and maximizing the discovered resources. However, determining the number of groups to be formed to optimize the parallelism and coverage of the PU spectrum is not discussed.

The valuable resources discovered through cooperative parallel sensing then require an efficient MAC mechanism that can reduce the number of collisions and ensure no resource left untapped. In contention-based MAC protocols, the SU contends for medium access in a random fashion, leading to a waste of resources either due to collisions or inability to tap these resources. This adversely affects the spectrum utilization. Existing work has shown collisions can be greatly reduced by providing some form of contention-free channel access [18, 34–37].

In [35], a contention-free reporting scheme-based MAC protocol has been proposed to improve the sensing time and network throughput. Any SU willing to join the CRN and CSS process can randomly select a slot in a slotted ALOHA manner and send random access ready to send (RARTS). If a slot is selected by only one SU, the RARTS is successfully received at the fusion center (FC) and the FC responds with the random access clear to send (RACTS). The key concept of this paper is that the SU joining the network participates in the sensing phase and reports the sensing result to the FC, which leads to saving the sensing time and collision-free reporting. In this scheme, the reporting is contention free, yet the random sensing results in duplicate sensing of some of the spectrum portion and some spectrum portion being missed. These adversaries waste energy and cause lost opportunities.

In [36], the authors have proposed a backoff (BO) algorithm-based cognitive radio MAC (BO-CRMAC). In this scheme, the collided SUs are allowed to recontend in the same cycle for an idle channel. The BO-CRMAC reduced the access delay and improves the overall utilization of the idle channel. However, if fair play is not ensured, all SUs are tempted to target the initial channels to have multiple reattempt options open and this could eventually choke the system. This scheme works well and shows improvement in the utilization when the pool of available resources is huge. Utilization of the discovered pool of resources improves and eventually gets better in BO-CRMAC than random access. The scheme provides more utilization gain when the number of channels is greater than the number of SUs; however, when the number of channels is less than the number of SUs, its performance is even poor than random access. This is a matter of concern as the actual challenge lies where the resources are limited and contenders are large in number.

The authors in [37] have proposed CR intelligent MAC (CR-i-MAC) with the aim to make the medium access contention-free, minimizing the loss of discovered idle channel and overhead in the context of CRN MAC. The process of CR-i-MAC is done in three phases, i.e., (i) the sensing and sharing, (ii) the contention, and (iii) the transmission phase. The contention phase is hybrid in the proposed scheme: a cooperative approach to overcome the hidden node problem

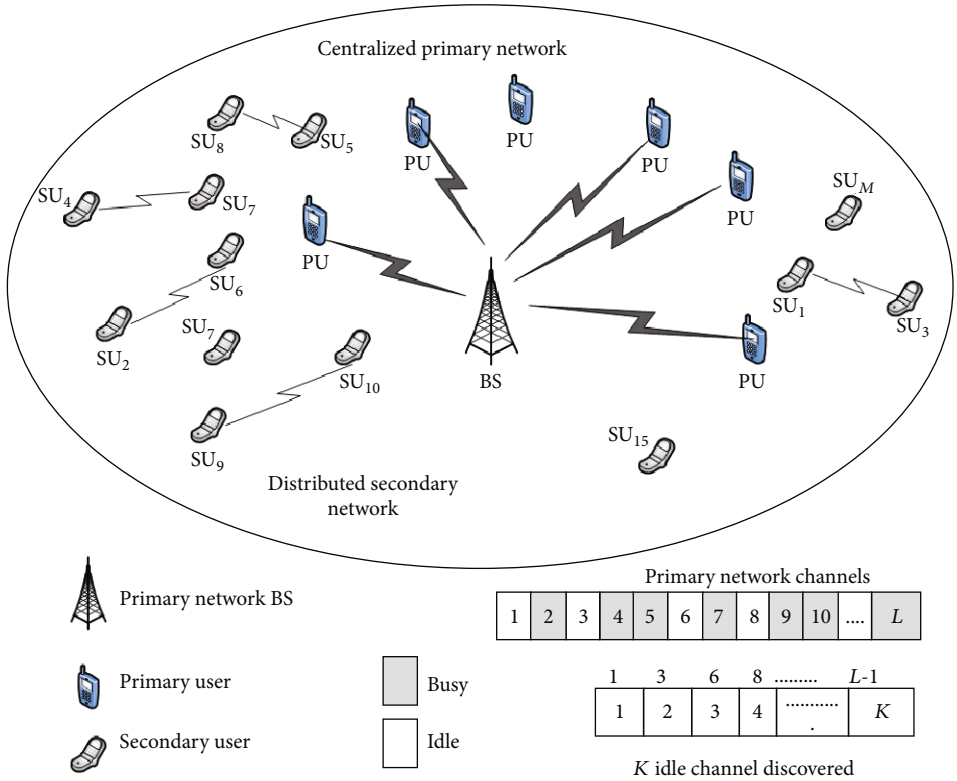


Figure 1: Network model.

and contention-free approach when multiple secondary users select the same channel. When only a single SU accesses the channel, the contention-free access is granted to it. The rest of the users determine their channel through an algorithm that eliminates the already allotted channels and SUs already granted access. The computational cost and scalability issues are raised as the network size increases.

To address this problem, the authors in [18] have proposed a self-organizing distributed (SOD) CR-MAC protocol. Loss of opportunities due to collisions and nonattempt is avoided by organizing and queuing the SUs through a timer value and directing them in an orderly fashion to use the discovered holes. In SOD, contention-free spectrum access is provided, consequently improving spectral efficiency.

The possibility to unify the parallel sensing approach with the self-organizing MAC scheme has been quite enticing and served to be the main motivation for this work. Therefore, the proposed PSO-MAC develops a unified frame structure that captures the benefits of the two approaches as presented in [18, 20]. Further, PSO-MAC incorporates the optimization of number parallel sensing groups to maximize the discovered resources as well as the utilization of discovered resources while keeping the overhead minimum.

3. PSO-MAC Protocol for CRN

In this section, the network model along with the frame structure of PSO-MAC and its algorithms is discussed.

3.1. Network Model. We have considered a centralized primary network having L licensed channels available to N PUs. There exists a distributed secondary network, also called CRN, with M SUs as shown in Figure 1. SUs access the channel only when PUs are not using the channel. Therefore, the number of channels available to SUs at any given time is K, where $[0 \leq K \leq L]$ and is a function of load on the primary network (ρ), which is the ratio of average number of channels being used by the primary network to the total number of primary channels. The ON-OFF process is considered to model the primary channel, where "ON" represents channel is busy and "OFF" represents channel is idle. Spectrum sensing is assumed as perfect by ignoring miss detection and false alarms. The detailed list of notations used in this paper is presented in Table 1.

3.2. PSO-MAC Frame Structure. The frame structure of the proposed PSO-MAC has been built upon the frame structures proposed in [18, 20] (see Figures 9 and 3 of [18, 20], respectively) and is shown in Figure 2. It consists of five phases, i.e., synchronization phase (T_i), organization phase (T_o), parallel sensing phase (T_{ps}), sharing phase (T_s), and transmission phase (T_{tx}). PSO-MAC frame structure is different from the standard MAC frame in a way that it has an additional organization phase and the sensing phase is replaced by an efficient parallel sensing phase.

The sequence of these phases has carefully been adjusted that eliminates redundancies and allows dynamic adjustments to work optimally according to the prevailing network conditions. This arrangement enables to maximize the gains

Table 1: List of notations.

Description	Parameter
Number of PU	N
Number of SU	M
Number of primary channel	L
Number of available channel	K
System capacity	C
Primary traffic load	ρ
Cycle time	T_c
Synchronization phase	T_i
Organization phase	T_o
Sensing phase	T_s
Parallel sensing phase	T_{ps}
Sharing phase	T_{sh}
Transmission phase	T_{tx}
Short interframe time	aSIFSTime
Slot time	aSlotTime
RTS time	aRTSTime
CTS time	aCTSTime
Number of bits in user ID	b_{ID}
Timer value	b_{TValue}
Number of slots required	$b_{SlotsReq}$
Bit duration	T_{bit}
Group ID	G_{id}
Number of groups	N_g

from the combination of the two approaches as mentioned above while keeping the incurred overhead to a minimum as well. For example, placing the organization phase before the parallel sensing phase allows the dynamic adjustment of the number of parallel groups that optimizes the sensed spectrum with the overhead so that discoveries and utilization are optimized. This is as the number of active SUs is known by the end of the organization phase.

3.2.1. Idle Phase. Each cycle starts with an idle phase T_i. In this phase, the SUs synchronize with each other and can share and collect the information from the common control channel. The length of this phase is

$$T_i = a\text{SIFSTime} + 2 \times a\text{SlotTime}, \quad (1)$$

where aSIFSTime and aSlotTime are the short interframe space time and the slot time, respectively, as in [36].

3.2.2. Organization Phase. PSO-MAC introduces a self-organization (T_o) phase, where SUs organize themselves in a queue in an ad hoc network. This organization of SUs facilitates the collision-free access to the channel and enables maximum utilization of the available channels. The organization is achieved through a user ID and a timer for each SU. Step 1 of Algorithm 1 shows the pseudocode of the organization method. An organization phase comprises multiple subslots and each subslot contains the information of each SU, i.e., user ID, its timer value, and the number of slots required by the user along with its RTS and CTS information. The length of this phase is

$$T_o = C \times \left(\left(b_{ID} + b_{TValue} + b_{SlotsReq}\right) \times T_{bit} + a\text{RTSTime} + a\text{SIFSTime} + a\text{CTSTime}\right), \quad (2)$$

where b_{ID}, b_{TValue}, and $b_{SlotsReq}$ are the number of bits in user ID, timer value, and number of slots required, respectively. T_{bit} is the duration of a bit and depends on the data rate (R) of the network [18].

The user ID is acquired by the SU at the time of entry into the CRN. For this purpose, the new entrant observes the announcement made by active SUs in the subslots of the organization phase (T_o). Any subslot without any information announcement is assumed to be vacant, and the user ID is considered available for contention. The SUs contend for the user ID, and with multiple SUs seeking entry in the network, the collision in ID acquisition is possible. However, since the entry into the network is a one-time process, the collision only delays the entry of the SU into the network and does not result in a waste of resources, i.e., the vacant primary channel. In case of collision, the new entrant has to contend for a vacant ID in the next T_c. Once it gets the successful access to a subslot of the T_o phase, then that subslot would be its user ID in the CRN.

The timer is started by an SU as soon as it is ready to transmit the data. The SU with higher timer value at the beginning of the T_c gets the priority in the queue. The timer value is announced by the SUs which might give rise to some fairness issues.

3.2.3. Parallel Sensing Phase. Once the SUs are organized in the T_o phase, distributed SUs sense for idle channels in the sensing phase T_{ps}. It allows multiple SUs to sense for the vacant channel in a parallel manner. The parallel sensing phase of the PSO-MAC is shown in Figure 2, having two groups of SUs. These groups are based on their IDs, e.g., SUs with even-numbered IDs in one group and SUs with odd-numbered IDs in another group. The grouping criteria and number of groups in an actual scenario are optimally adjusted according to the network conditions, and the group ID (G_{id}) is determined as follows:

$$G_{id} = \left(SU_{id} \bmod N_g\right) + 1, \quad (3)$$

where SU_{id} is SU's ID and N_g is the number of groups. For example, if an SU has $SU_{id} = 231$ and the number of groups $N_g = 6$, then its $G_{id} = (231/6) + 1 = 4$ and this SU will scan the 4^{th} portion of the primary spectrum. Note that the primary spectrum is divided into N_g portions, i.e., 6 in this case. In this manner, each group senses L/N_g number of primary channels in parallel. The SUs sense the primary channel

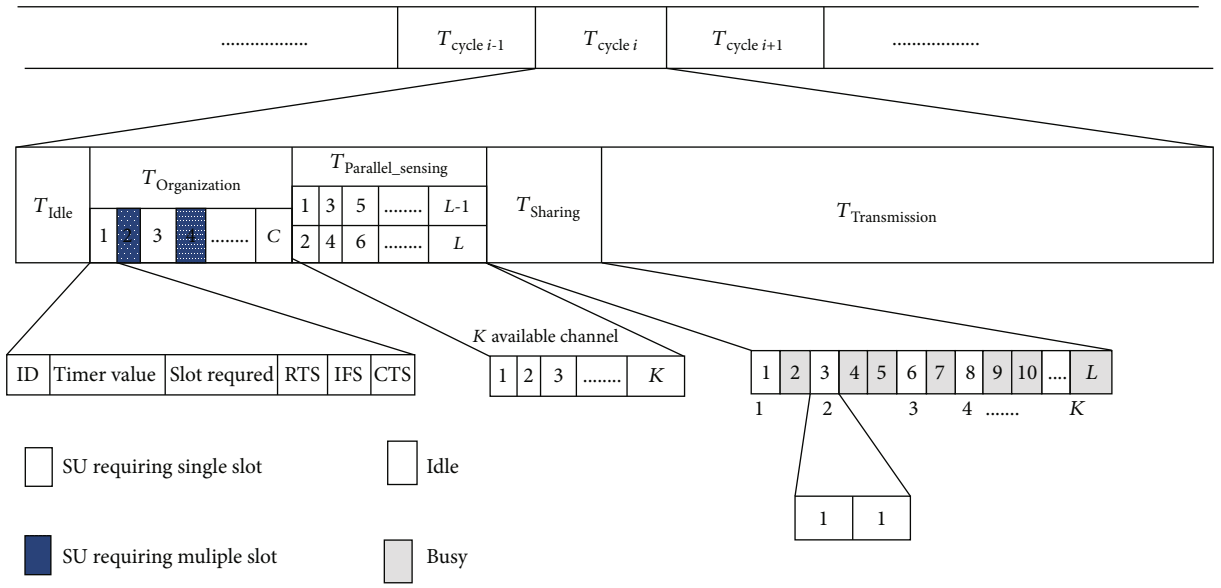


Figure 2: PSO-MAC frame structure.

Input: number of primary users N, number of secondary users M, number of primary channels L, number of discovered idle channel K, number of parallel groups N_g, primary traffic load ρ
Output: channel utilization
1. **Step 1: organization phase;**
2. Assign IDs as in [18];
3. Start timer value when data is ready;
4. Share IDs and timer values among all SUs;
5. **for** all M SUs **do**
6. if all M have different timer value **then**
7. SUs position in queue based on descending order of timer values;
8. **else**
9. SU with lower ID get higher preference;
10. **end**
11. **end**
12. SUs organized in a queue;
13. **Step 2: parallel sensing phase**;
14. Number of channels scan by each group is L/N_g;
15. **for** all N_g**do**
16. Scan the specific portion of L
17. **end**
18. **Step 3: sharing phase**;
19. **for** all L slots **do**
20. Share the sensing result;
21. **if** channel sensed idle **then**
22. Set the second bit of subslot to 0;
23. **else**
24. Set the second bit of subslot to 1;
25. **end**
26. **end**
27. **Step 4: transmission phase**;
28. **if** SU in queue and channel sensed idle **then**
29. Transmit the data of SU standing at the first position in queue;
30. **end**
31. **if** SU needs multiple slots **then**
32. Use round-robin scheduling
30. **end**

Algorithm 1: PSO-MAC algorithm.

based on their group ID (G_{id}) and repeat the sensing after the number of group N_g till L.

3.2.4. Sharing Phase. Sharing phase, T_s, is where the spectrum sensing results of different groups are shared among the SUs both within and outside the group. The sensing of the channel is done in a parallel manner; however, the status of the channel cannot be shared in parallel. So the total number of subslot required in the sharing phase is the same as the number of primary channels, i.e., L. The length of this phase is

$$T_s = L \times 2T_{bit}. \quad (4)$$

Each subslot in the sharing phase carries two-bit information. The first bit represents whether the channel is sensed "1" or skipped "0," and the second bit represents the status of the sensed channel according to the reporting user, i.e., "0" for idle and "1" for busy, as shown in Table 2. All the users in a group sensing a particular channel report their findings through these two bits. The decision about the presence or absence of a primary user on a channel is made locally by each user through these bits. If there are conflicting reports for a channel, i.e., some reporting the channel idle and other as busy, then the channel is considered as busy by the user. This provides complete protection to the primary user against the interference from the CRN. The channel is considered as not sensed if the first bit is "0" regardless of the value of the second bit which is marked as "x."

3.2.5. Transmission Phase. All the SUs are organized in a queue after the organization phase (T_o) and after the channel state information is shared among the distributed SUs in the sharing phase, where K vacant channels from the primary band have been identified, the channel access is smooth and efficient, i.e., without contention and channel skipping. The first SU in the queue gets access to the first vacant channel in the list the second SU gets the second vacant channel and so on. If all the SUs have got access to a channel and the list of vacant channels is not exhausted, then the SUs demanding multiple channels can get access to additional channels in the same T_c using a scheduling mechanism such as round-robin scheduling, because preferably used to access the remaining idle channels as all the SUs have already been organized. On the other hand, if an SU does not get access to a channel in the current T_c, it gets automatic priority in the next T_c by the virtue of its higher timer value. Since the access is collision-free, the SUs getting access to the channel can start their data transmission after the T_s phase, i.e., the T_{tx} phase.

3.3. PSO-MAC Algorithm. The challenge in a CRN is to maximize channel utilization through sensing of the maximum vacant primary channels for data transmission. In contention-based random channel access, there exists a possibility of collision, where multiple SUs access the same channel, as well as the possibility of an idle channel being skipped, i.e., no SU attempting to access a particular channel. In either case, the result is a waste of valuable resources and eventually degrades the performance of CRN.

Table 2: Sharing phase subslot and channel status.

Subslot bit no.	Value	Channel state
First bit	1	Channel sensed idle
Second bit	0	
First bit	1	Channel sensed busy
Second bit	1	
First bit	0	Channel not sensed
Second bit	x	

PSO-MAC protocol reduces both the duplicated and the missed idle channel sensing through parallel sensing and maximizes the successful utilization by organizing the SUs in a queue and enabling the channel access without contention. In the organization phase, all SUs share their timer values and the number of slots required. The timer value indicates the age of data, so the SU with the highest timer value gets top priority in the queue. When two SUs have the same timer value, then priority in the queue is decided through user ID, which is acquired at the time of entry into the CRN. All the SUs are aware of their exact location in the queue at the end of the organization phase.

Once the SUs are organized in a queue, the parallel sensing phase is initiated. PSO-MAC prefers parallel sensing instead of random sensing to maximize the sensing of vacant channels and minimize the sensing time. In parallel sensing, the primary band is divided into the same number of portions as the number of groups N_g and every member of the same group senses the specific portion of the primary band. PSO-MAC offers a dynamic grouping of the SUs for parallel sensing with due consideration of the load on the primary network and the number of active SUs in the CRN. This feature is missing in the existing work, but it is important as increasing the number of groups does not keep on increasing the gain. The sensing time is reduced through parallel sensing, but at the same time, the increased parallelism reduces the probability of idle channel discovery as chances of a portion of a band with no SU falling in that group are increased. Consequently, the gain achieved through parallelism in sensing is overwhelmed by the loss due to skipping of portions of bands during sensing. By determining the optimal number of groups (N_g) according to the number of channels (L), load (ρ) on the primary network, and the active number of SUs (M), we can maximize the product of available time and available channels at the disposal of CRN; i.e., the spectrum utilizable time (U) by the CRN.

$$U = E[K] \times T_{tr}, \quad (5)$$

where $E[K]$ is the mean of discovered idle channels, which is determined through (10) of [20], i.e.,

$$E[K] = (1-\rho)\, L \left[1 - \left(1 - \frac{1}{N_g}\right)^M\right], \quad (6)$$

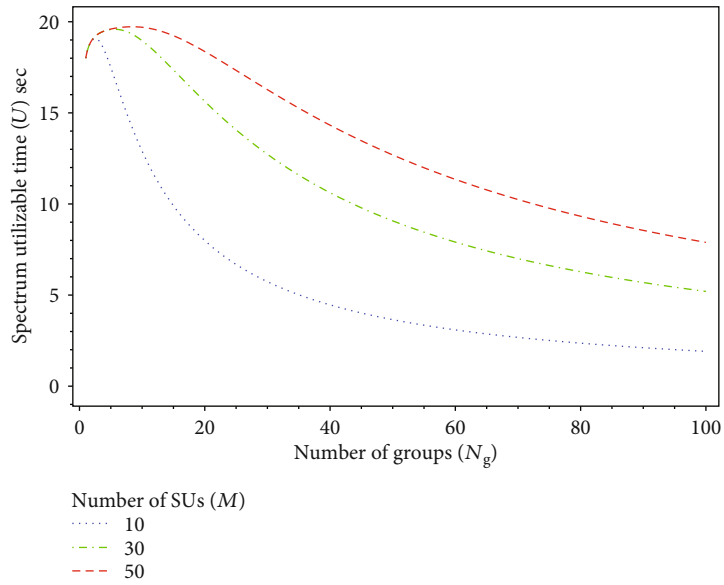


FIGURE 3: Spectrum utilizable time for SUs in the CRN.

and T_{tr} is the transmission time available to the SUs after the overhead in a transmission cycle (T_c), i.e.,

$$T_{tr} = T_c - \left(T_i + T_o + T_{ps} + T_{sh}\right). \tag{7}$$

While T_c, the idle time duration (T_i), the duration of the organization phase (T_o), and sharing phase (T_s) are fixed, the duration of parallel sensing phase (T_{ps}) is a function of N_g, i.e.,

$$T_{ps} = \frac{L \times \tau_s}{N_g}, \tag{8}$$

where τ_s is the duration of the sensing slot, i.e., the time taken by an SU to sense a channel.

Figure 3 shows the net utilizable time, i.e., the product of number of discovered idle channels and the time available for useful transmission on each discovered idle channel, for the CRN. Three plots are shown for identical network conditions ($L = 100$, $\rho = 0.8$, $T_c = 1$ sec, $T_i = 54$ nsec, $T_o = 37\ \mu$ sec, $\tau_{sh} = 37$ nsec, $T_{sh} = L \times \tau_{sh}$, and $\tau_s = 1$ msec), but with different numbers of SUs in the CRN (i.e., $M = 10$, 30, and 50). Initially, the net spectrum utilizable time (U) increases by increasing the number of groups (N_g). However, this trend does not continue with further increase in number of groups; rather, it starts to fall down sharply and the peak in each case is reached at a different value of N_g.

In order to ascertain the optimum number of groups, as per the network conditions and number of SUs, we determine the maxima of $U(N_g)$. The minima and maxima points are obtained through first derivative (U'), i.e., by solving

$$\frac{d}{dN_g} U = 0, \tag{9}$$

and to single out the maxima, the second derivative test is applied; i.e., if $U''|_{g_i} < 0$, then g_i is the point of maxima and the optimal number of groups (N_g^*) is

$$N_g^* = \text{round}(g_i). \tag{10}$$

For the three cases being considered, i.e., $M = 10$, 30, and 50, the optimal number of groups are 3, 6, and 8, respectively. In the scenario, where $L = 100$, $\rho = 0.8$, and $T_c = 1$ sec, the maximum spectrum utilizable time is 20 sec. Without using the parallel sensing mechanism, i.e., $N_g = 1$, the spectrum utilizable time that can be made available to the CRN for data transmission is 17.999 sec; however, with the optimal number of groups, the spectrum utilizable time for the CRN can be extended by 5.5% to 18.9973 sec, by 8.8% to 19.583 sec, and by 9.6% to 19.7243 sec for $M = 10$, 30, and 50, respectively. This gain becomes even more prominent at low primary traffic.

After parallel sensing, the sensing results are shared among the SUs in the sharing phase. This phase concludes with the list of idle channels in the current cycle. The SUs arranged in the queue access the vacant channel without contention for data transmission. The first available channel is allocated to the first SU in the queue and so on until all the idle channels are occupied by the SUs or all SUs get the required slots. When an SU requires multiple slots, it can get access to additional slots, provided all other SUs have got at least one, in the same transmission cycle in a round-robin fashion. This feature further improves channel utilization. The complete procedure of the proposed scheme is shown in Algorithm 1.

Parallel sensing along with the self-organized access allows the CRN to continue to offer a certain level of service to the secondary users even when the primary network has heavy traffic load and the opportunities for the secondary network are scarce.

4. Performance Metrics

The performance of the proposed PSO-MAC is analyzed in terms of the number of idle channels discovered, successful channel allocations to the SU, and the network throughput compared with the parallel sensing scheme of [20] and self-organizing distributed cognitive radio MAC protocol (SOD CR-MAC) [18]. In this section, we define these performance metrics in the context of PSO-MAC.

4.1. Discovered Idle Channels. For a sensing scheme in the CRN, the ultimate goal is to maximize the pool of available resources. Sensing techniques are employed to discover the idle channels; "discovered idle channels" is defined as the number of vacant channels of the PU band that are being successfully declared unused or idle by the sensing mechanism. These discovered idle channels can be used by the SUs for the data transmission. The probability distribution of idle channel discovery is [38]

$$p_K(k) = \binom{L}{k}(1-\rho)^k(\rho)^{(L-k)}, \tag{11}$$

where k is the number of discovered idle channels, L is the total number of channels in the primary band, and ρ is the primary traffic load. So the average number of idle channels discovered in a random sensing scheme is

$$E[K] = \sum_{k=0}^{L} k p_K(k). \tag{12}$$

For a random sensing scheme, such as RSO-MAC, the average number of channels discovered idle is

$$E^{\mathrm{RSO}}[K] = L(1-\rho)\left(1 - \frac{L-j}{L}\right)^M, \tag{13}$$

where j is the number of channels that every SU is required to sense during the sensing phase.

The average number of channels sensed during the parallel sensing is obtained as [20]

$$E[S] = L\left[1 - \left(1 - \frac{1}{N_g}\right)^M\right], \tag{14}$$

and the average number of discovered idle channels for a parallel sensing scheme such as PRA-MAC is

$$E^{\mathrm{PRA}}[K] = L\left[1 - \left(1 - \frac{1}{N_g}\right)^M\right](1-\rho), \tag{15}$$

where N_g is the number of parallel groups. The number of groups in [20] is fixed and randomly chosen, whereas in the proposed PSO-MAC scheme, the number of groups is adaptive according to (5) and is determined by evaluating (9), applying second derivative test and (10). The goal is to reduce the sensing time through parallelism and yet maximize the number of channels to be sensed. Covering the entire band during sensing would increase the probability of finding all the vacant channels. So for PSO-MAC, the average number of discovered idle channels is

$$E^{\mathrm{PSO}}[K] = L\left[1 - \left(1 - \frac{1}{N_g^*}\right)^M\right](1-\rho), \tag{16}$$

where N_g^* is the optimum number of groups, determined through (10).

4.2. Channel Utilization. In a CRN, the availability of discovered resources is dynamic and generally ephemeral, so maximizing their utilization is even more important than in traditional communication systems. For example, random access wastes many opportunities due to collisions and non-attempt on the channels. Utilization of the discovered resources, which is the ratio of the successfully utilized channels to the total number of discovered idle channels, in random access and in organized access has been studied in [18]. Using the Monte Carlo simulations, the probability of successful channel utilization (P_{su}) is determined, which we use to analyze the network throughput.

4.3. Network Throughput. The network throughput of the CRN depends on idle channel discovery, successful channel utilization, and overhead of the frame structure. Higher number of discovered resources and their maximal utilization would definitely increase the network throughput. However, the PSO-MAC scheme is designed for an ad hoc network, which has no infrastructure cost but lays the onus of coordination on the distributed nodes. The coordination among the users is achieved through a common control channel (CCC) with a frame structure that enables several critical functions for CRN operation; this arrangement does incur an associated overhead that needs to be analyzed. Network throughput provides a suitable measure to encompass all aspects and compare various schemes.

The cycle time consists of number of phases that include idle time, organization period, sensing duration, sharing time, contention period in case of random access, and the transmission time.

$$T_c = T_i + T_o + T_s + T_{sh} + T_{tr} + T_{ct}, \tag{17}$$

where T_{tr} is the data transmission time and the rest of the contributing factors in T_c are overhead so

$$T_{tr} = T_c - (T_i + T_o + T_s + T_{sh} + T_{ct}). \tag{18}$$

The transmission time (T_{tr}) for random sensing-organized MAC (RSO-MAC) [18], parallel sensing-random access MAC (PRA-MAC) [20], and parallel sensing-organized MAC (PSO-MAC) schemes is, respectively, as follows:

$$T_{tr}^{\mathrm{RSO}} = T_c - (T_i + T_o + T_s + T_{sh}), \tag{19}$$

$$T_{\text{tr}}^{\text{PRA}} = T_{\text{c}} - \left(T_{\text{i}} + T_{\text{ps}} + T_{\text{sh}} + T_{\text{ct}}\right), \tag{20}$$

$$T_{\text{tr}}^{\text{PSO}} = T_{\text{c}} - \left(T_{\text{i}} + T_{\text{o}}, T_{\text{ps}}^{*} + T_{\text{sh}}\right), \tag{21}$$

where

$$\begin{aligned} T_{\text{s}} &= \tau_{\text{s}} \times L, \\ T_{\text{sh}} &= \tau_{\text{sh}} \times L, \\ T_{\text{ps}} &= \tau_{\text{s}} \times \frac{L}{N_{\text{g}}}, \\ T_{\text{ps}}^{*} &= \tau_{\text{s}} \times \frac{L}{N_{\text{g}}^{*}}. \end{aligned} \tag{22}$$

N_{g}^{*} is the optimal number of groups as obtained through (10), and T_{ct} is the duration of the contention phase as required in random access schemes.

In PSO-MAC, the parallel sensing scheme is used to reduce the sensing time and efficiently discover the maximum vacant channels by dividing the SUs into an optimal number of groups. Additionally, the discovered channels are accessed in an organized manner, which eliminates the need for contention and the probability of successful utilization of channels $P_{\text{su}}^{\text{PSO}} \approx 1$ [18], thus leading to improved network throughput. The network throughput for PSO-MAC is obtained as follows:

$$\text{Throughput}^{\text{PSO}} = P_{\text{su}}^{\text{PSO}} \frac{E^{\text{PSO}}[K] \times T_{\text{tr}}^{\text{PSO}} \times R}{T_{\text{c}}}, \tag{23}$$

where $E^{\text{PSO}}[K]$ is the expected number of idle channels discovered in the PSO-MAC and is obtained using (16), whereas $T_{\text{tr}}^{\text{PSO}}$, which is the transmission time after the user gets access to the channel, is determined through (21). R is the channel data rate and T_{c} is the cycle time duration.

The corresponding expressions of throughput for RSO-MAC [18] and PRA-MAC [20] are as given below:

$$\text{Throughput}^{\text{RSO}} = P_{\text{su}}^{\text{RSO}} \frac{E^{\text{RSO}}[K] \times T_{\text{tr}}^{\text{RSO}} \times R}{T_{\text{c}}}, \tag{24}$$

$$\text{Throughput}^{\text{PRA}} = P_{\text{su}}^{\text{PRA}} \frac{E^{\text{PRA}}[K] \times T_{\text{tr}}^{\text{PRA}} \times R}{T_{\text{c}}}. \tag{25}$$

The values of $E^{\text{RSO}}[K]$ and $T_{\text{tr}}^{\text{RSO}}$ are obtained using (13) and (19), respectively. The values of $P_{\text{su}}^{\text{RSO}}$ and $P_{\text{su}}^{\text{PRA}}$ are obtained in [18] through Monte Carlo simulations. $E^{\text{PRA}}[K]$ and $T_{\text{tr}}^{\text{PRA}}$ are determined using (15) and (20), respectively, where the number of parallel groups N_{g} is randomly chosen and not by determining the optimal value.

Table 3: Simulation parameters.

Parameter	Value
Number of channels in the primary network (L)	20~100
Primary traffic load (ρ)	0~1
Number of secondary users (M)	2~100
Number of SU groups (N_{g})	2~50
Idle time duration for synchronization (T_{i})	54 μs
Slot duration in sensing (τ_{s})	1 ms
Cycle time (T_{c})	1 s
Slot duration in sharing (τ_{sh})	37 ns
T_{SIFT}	16 μs
T_{RTS}	24 μs
T_{CTS}	24 μs

5. Simulation Scenario, Analysis, and Comparison

In this section, we present the simulation-based performance analysis of the proposed PSO-MAC protocol and compare results with the following two schemes:

RSO-MAC. The authors in [18] used the self-organized contention-free access scheme, while the sensing of the channel is random (RSO-MAC).

PRA-MAC. In [20], the authors proposed the parallel sensing with random access MAC (PRA-MAC). PRA-MAC sensing uses parallel sensing of the primary band; however, access of channel is random which is not efficient.

First, we provide the simulation setup and parameters, and in the following subsection, the results and the comparison are discussed.

5.1. Simulation Parameters. For the simulation scenario, we select different values of number of channels (L) ranging from 20 to 100 and the primary traffic load (ρ) is varied from 0 to 1. The number of SUs (M) fluctuates between 2 and 100 in different scenarios. For comparison, the number of groups (N_{g}) is varied from 2 to 50, but the optimum number of groups for PSO-MAC is obtained by determining the point of maxima of $U(N_{\text{g}})$ (5) taking first derivative, applying second derivative test and (10) as elaborated in Section 3.3.

The PSO-MAC protocol is simulated in MATLAB and Monte Carlo simulations have been performed to evaluate the various performance parameters, and results are obtained on averaging 10^6 experiments.

The detailed list of parameters used in the simulation is shown in Table 3.

5.2. Results and Discussions. In this subsection, we analyze the performances of PSO-MAC, PRA-MAC, and RSO-MAC protocols for the considered system model and compare the results in terms of the discovered idle channels, idle channel utilization, and the network throughput.

Figure 4 shows the comparison of the number of idle channels discovered by the proposed PSO-MAC scheme with those in the baseline schemes, i.e., PRA-MAC and RSO-

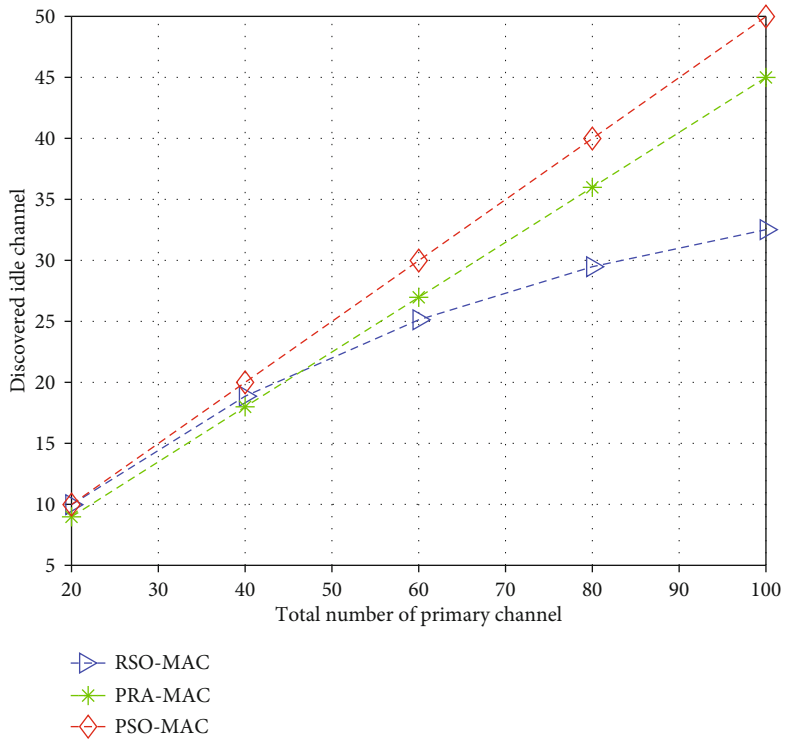


Figure 4: Discovered idle channel with varying primary channels.

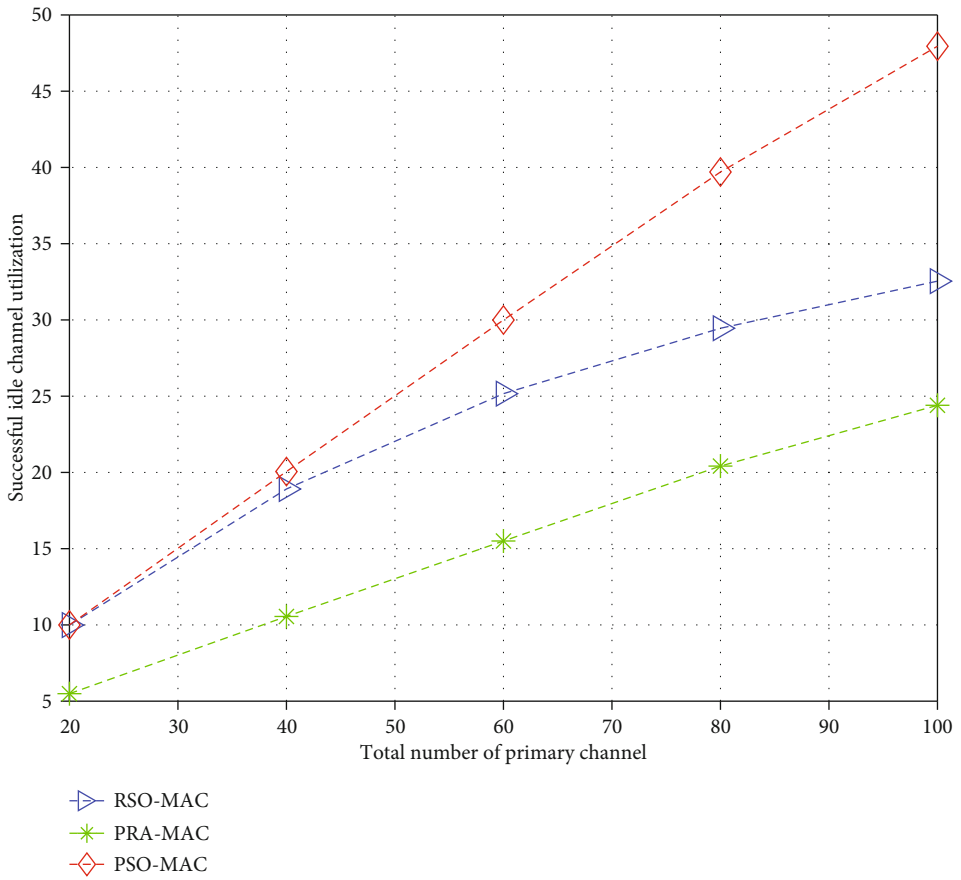


Figure 5: Idle channel utilization with varying primary channels.

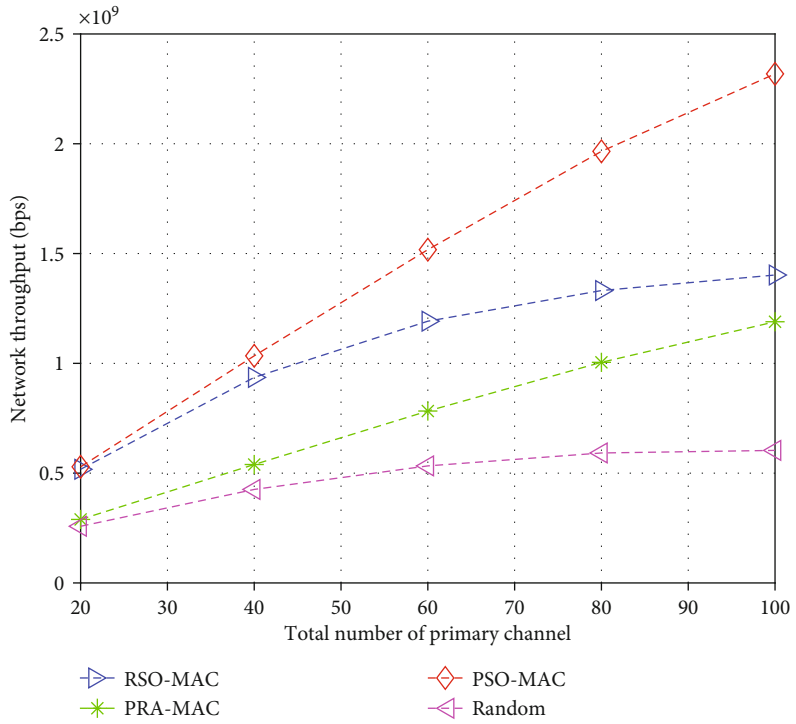


Figure 6: Network throughput with varying primary channels.

MAC. These simulations are conducted at a 50% load on the primary network and $M = 10$. Idle channel discovery performance for PSO-MAC is far better than PRA-MAC and RSO-MAC for the entire range of the number of primary channels. In RSO-MAC, the users randomly pick and sense the channel and its performance quickly degrades, whereas in PRA-MAC, through parallel sensing, the performance remains uniform that results in successfully discovering 90% of the idle channels. PSO-MAC by optimally dividing the sensing task among the SUs gets almost 100% discoveries of the vacant channels. For example, at 50% load on the primary network with 80 channels, there are 40 vacant channels and PSO-MAC has successfully discovered the 40 channels as evident from Figure 4.

Figure 5 shows the variation of idle channel successful utilization with the varying number of the primary channels. Not all the discovered idle channels are successfully utilized by the SUs, and it depends on the access control mechanism. In PSO-MAC, through contention-free access, the waste is negligible; however, this is not the case with other MAC schemes. These results are obtained with simulation parameters set at $M = 10$, the primary traffic load $\rho = 50\%$, and the average number of channels required by each SU to be 5. In PSO-MAC, channels are sensed using parallel sensing and accessed with a self-organized queue in which contention is not required. This leads to the maximum utilization of discovered idle channels and outperforms the PRA-MAC and the RSO-MAC. In PRA-MAC, the parallel sensing improves the discovered resources; however, the random channel access results in wasted opportunities due to collisions and nonattempt. In RSO-MAC, channels are accessed in an organized fashion; thus, despite fewer idle channels being discovered, it performs better than PRA-MAC as all discovered opportunities are tapped. PSO-MAC by virtue of contention-free access successfully utilizes twice as many channels as with PRA-MAC under the identical network conditions. Comparing RSO-MAC with PSO-MAC, initially the gap is small, but it widens as the opportunities for CRN users grow as the discoveries are limited in RSO-MAC.

Network throughput in the CRN can be enhanced by maximizing the discovery of idle channels from the primary band and reducing the overhead in T_c, which extends the transmission time available for the secondary users. Figure 6 illustrates the comparison of the network throughput achieved for the three schemes as well as for the case when both the sensing and access mechanism are random. It is to be noted that the PSO-MAC establishes its superiority, as the network throughput gains are higher with PSO-MAC than the sum of the gains achieved when each one of the other two schemes is employed independently. Considering the IEEE 802.11a channel data rates and the primary network with 80 channels and at $\rho = 50\%$, the network throughput is 1.9 Gbps, 1.35 Gbps, and 1 Gbps for PSO-MAC, RSO-MAC, and PRA-MAC, respectively.

It is also interesting to evaluate the performance with different values of the primary traffic load. We conduct the simulations with parameters $L = 100$, $M = 10$, channels required by the SUs are uniformly distributed between 1 to 10, and the primary traffic load ρ is varied from 0 to 1.

The impact of the primary load on the discovered idle channels is shown in Figure 7 where the PSO-MAC discovered the maximum number of idle channels. The performance is even better than PRA-MAC as not only sensing is done in parallel but the number of groups is adjusted

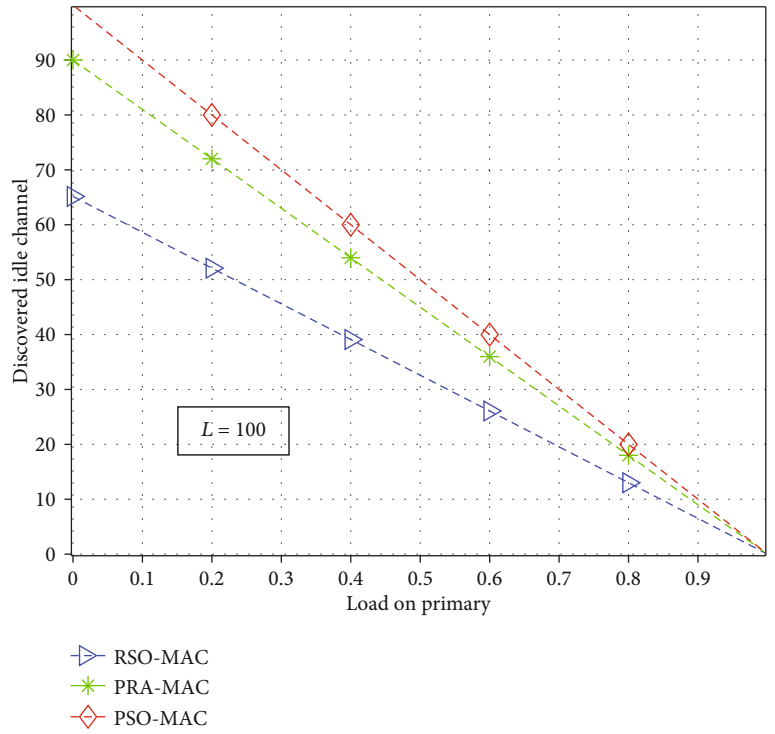


Figure 7: Discovered idle channel with varying primary load.

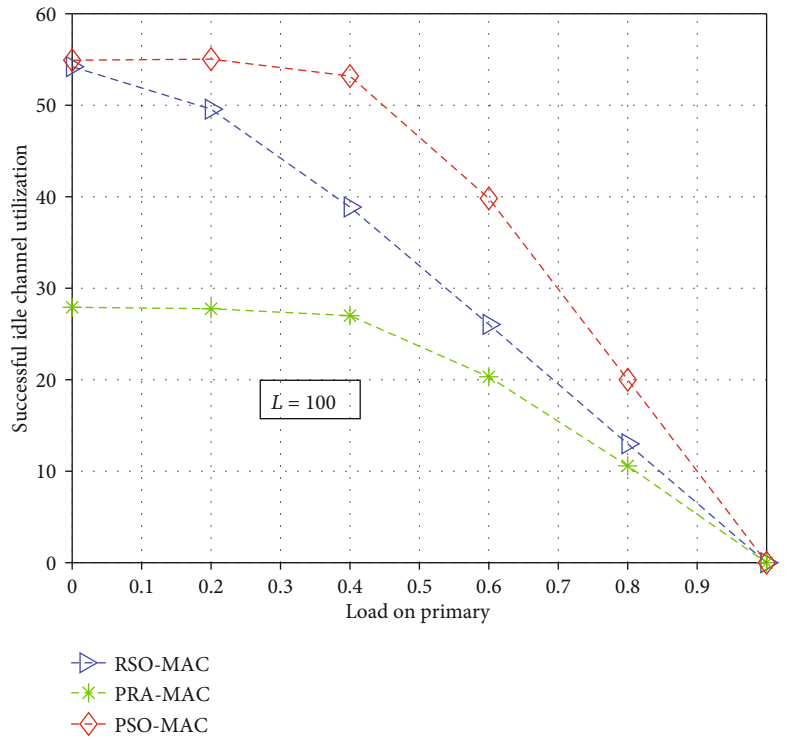


Figure 8: Idle channel utilization with varying primary load.

according to the load on the primary network. The worst performance with respect to idle channel discovery is with RSO-MAC as there is no coordination among the SUs for sensing and is done randomly.

Figure 8 shows the idle channel utilization for the three schemes with a varying load on the primary network. With no load on the primary network ($\rho = 0$), the performance of RSO-MAC is approximately the same as the PSO-MAC

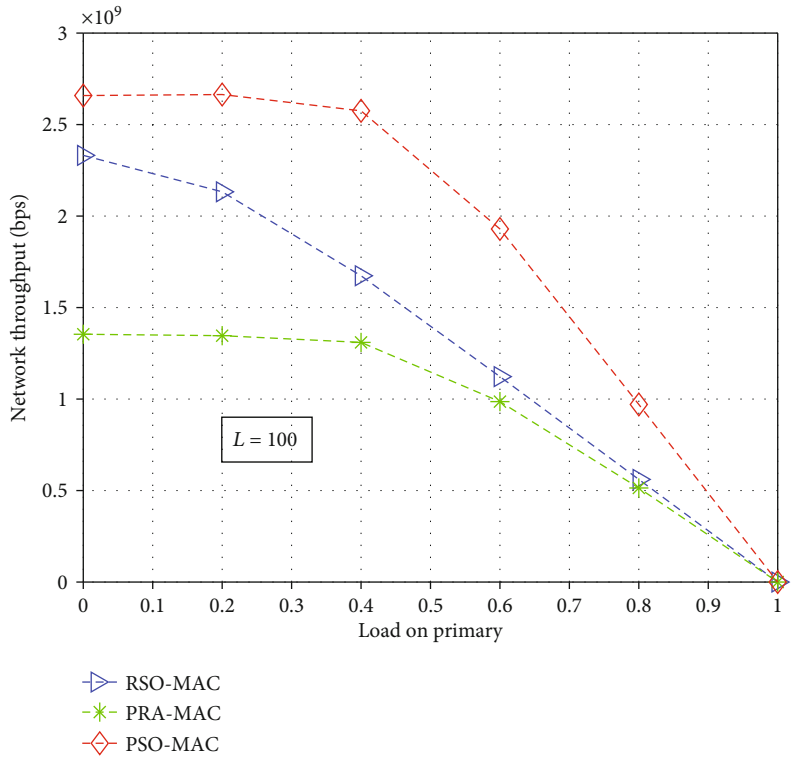


Figure 9: Network throughput with varying primary load.

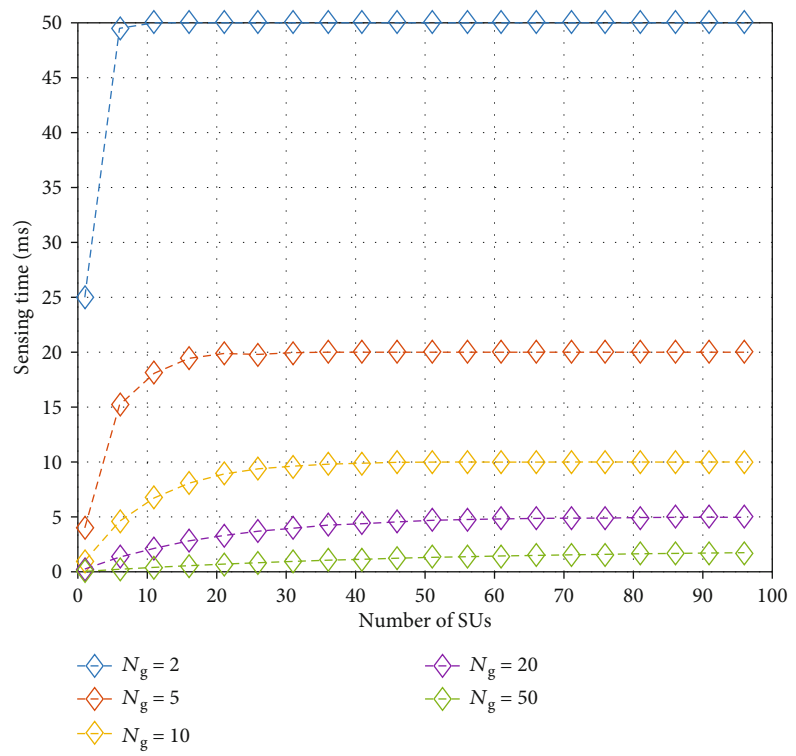
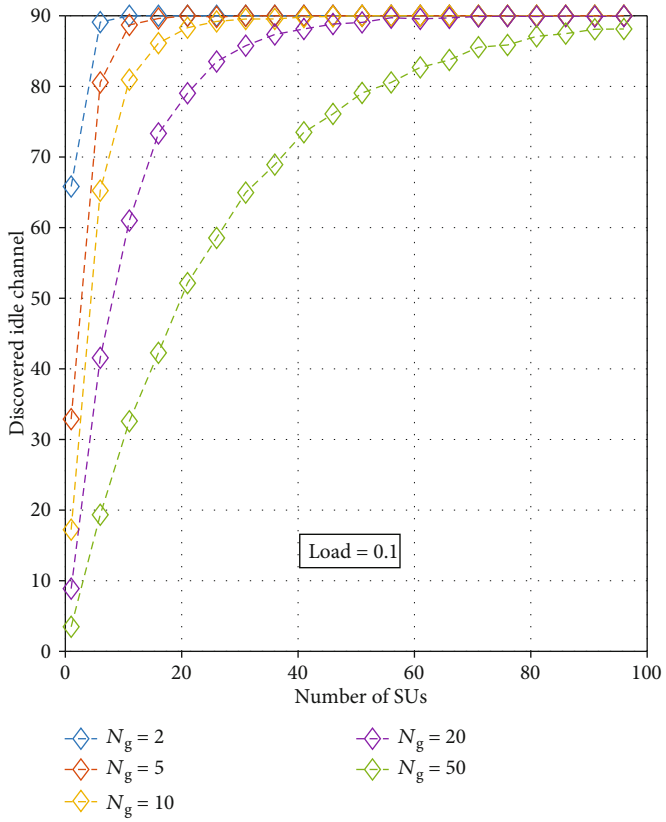
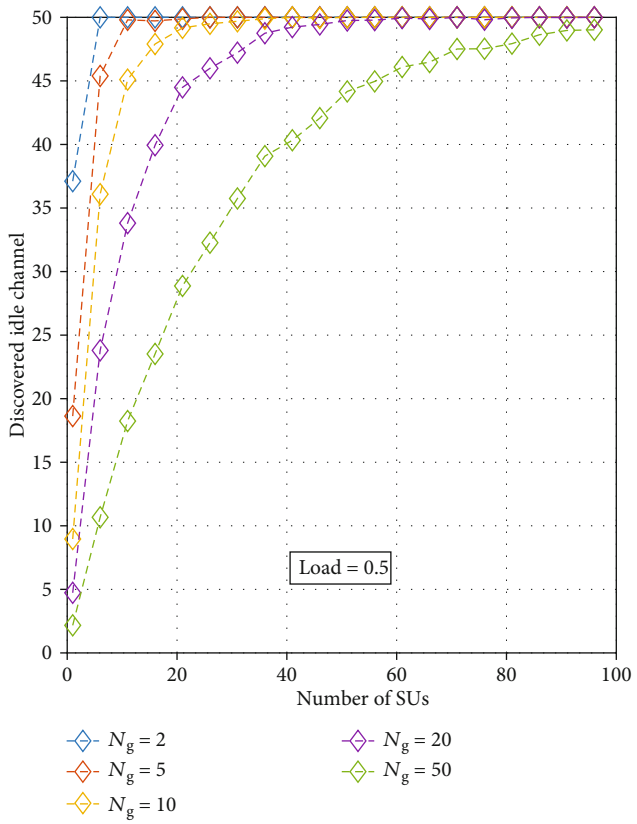


Figure 10: Sensing time with varying number of SUs.

as the entire band is available for the SUs. However, as the load increases, the performance of RSO-MAC declines sharply due to the poor discovery mechanism. The performance of PRA-MAC remains below par despite having better sensing and discovery process. This is due to the unorganized, contention-based random access on the vacant



(a)



(b)

Figure 11: Continued.

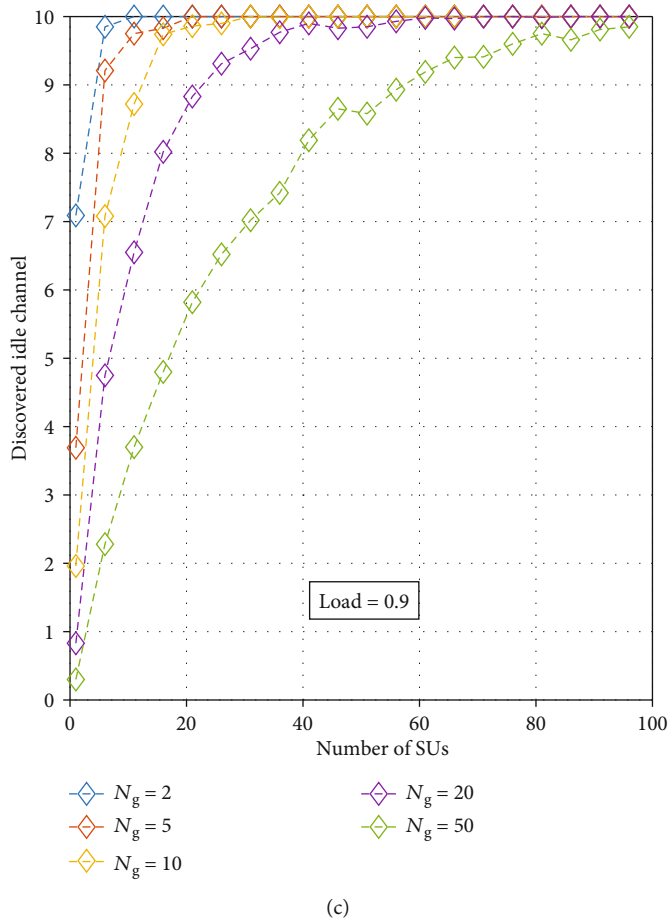


Figure 11: Discovered idle channel with varying numbers of SUs.

primary channels. The utilization is best with PSO-MAC due to optimized sensing and organized access; however, as the load on the primary network increases, the utilization drops as the available opportunities are limited.

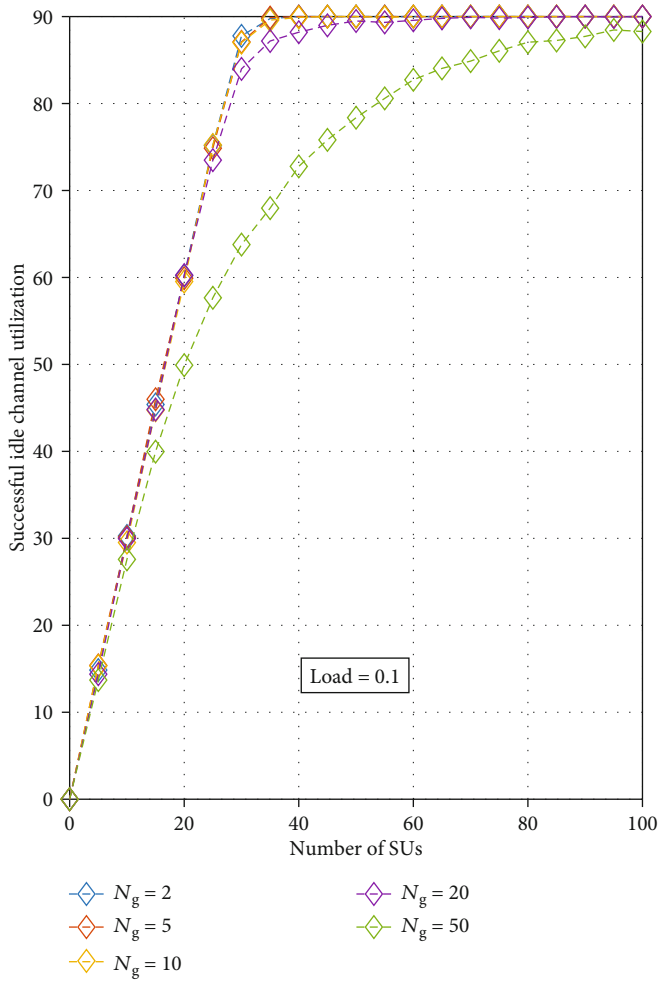
The network throughput with variation in primary traffic load is shown in Figure 9. The PSO-MAC achieves the highest network throughput throughout the entire range of primary traffic load. This is by the virtue of optimized parallel sensing and organized contention-free access.

The effect of parallel grouping on duration of sensing phase has also been investigated further. In Figure 10, when the number of SUs increases in the CRN, the probability of a group having no member is reduced. As a result, less number of portions of the primary band is skipped in sensing and the sensing time is increased. However, it becomes constant after the CRN has a certain number of SUs (divided into N_g groups) as the probability of a group having no members approaches 0. It also shows the favorable effect of increasing the number of groups on sensing time. With number of primary channels $L = 100$, the sensing time is 50 ms, 20 ms, 10 ms, 5 ms, and 2 ms for $N_g = 2$, $N_g = 5$, $N_g = 10$, $N_g = 20$, and $N_g = 50$, respectively. However, this is at the cost of reduced number of discovered resources as revealed in Figure 11. This reduction in discovered resources with increase in number of groups is primarily due to the fact that with large number of groups, the probability of a group with no members in it increases. The portion of spectrum assigned to be sensed by a group with no members is not covered during the sensing phase, and consequently, all opportunities of this band are lost.

The comparison of the three subplots in Figure 11 shows that as the load on the primary network increases, the number of discovered resources for each of the groups of SUs decreases (depicted in scaling of y-axis in the three subplots where the limits are 0-90, 0-50, and 0-10, respectively). This behavior call for adjustments in the number of groups in parallel sensing phase so that the overall sensing time is reduced while discoveries are not compromised at the same time. In PSO-MAC, the number of parallel groups is dynamically adjusted according to equations (5) and (9), which is dependent on the load on the primary network and number of secondary users.

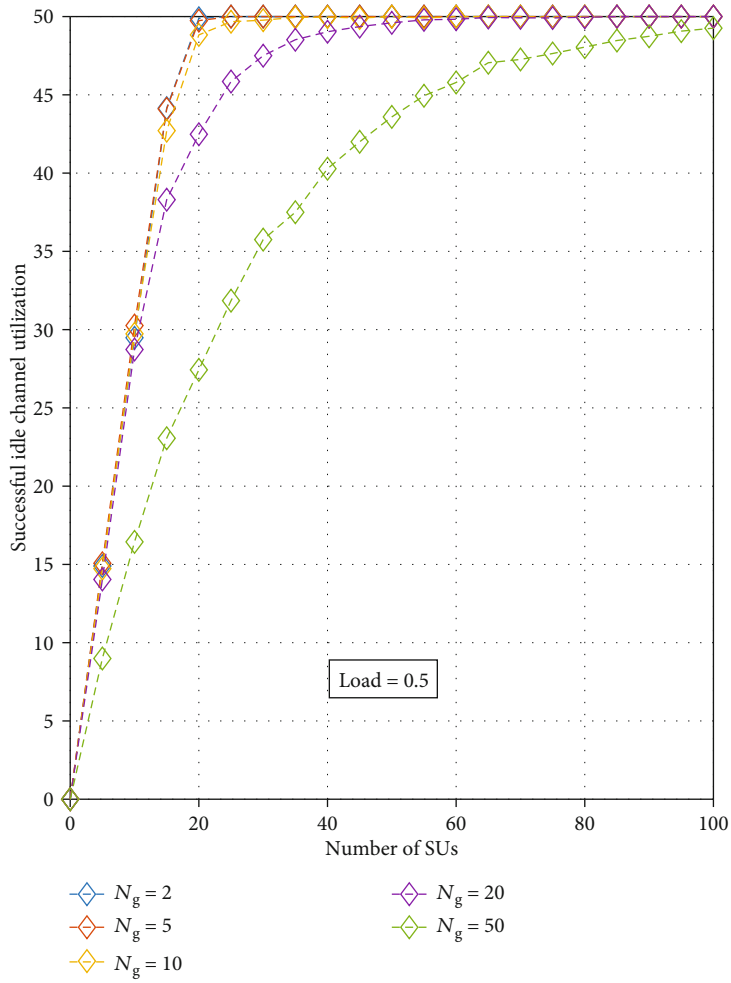
The utilization of idle channels for three values of primary traffic loads, i.e., 0.1, 0.5, and 0.9, is shown in Figures 12(a)–12(c), respectively. It can be seen that when the primary traffic is low, the higher number of SU groups provides better utilization. And as the load on the primary network increases, the number of groups should be smaller for the same number of SUs.

Finally, we compare the actual utilization of the vacant resources of the primary network by the CRN taking into consideration the overhead of the scheme, discovered



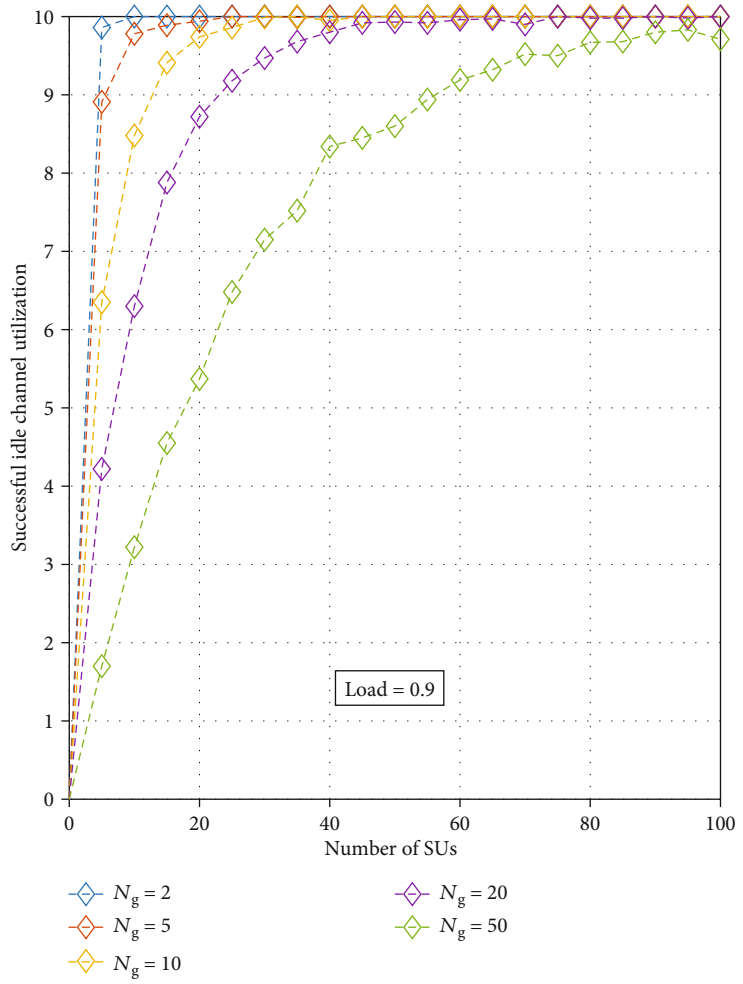
(a)

Figure 12: Continued.



(b)

Figure 12: Continued.



(c)

Figure 12: Idle channel utilization with varying numbers of SUs.

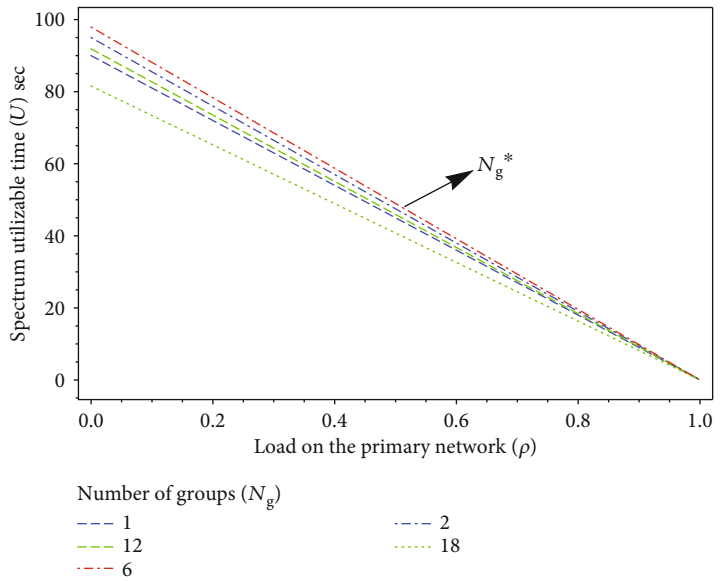


Figure 13: Utilizable time for the CRN with different numbers of groups.

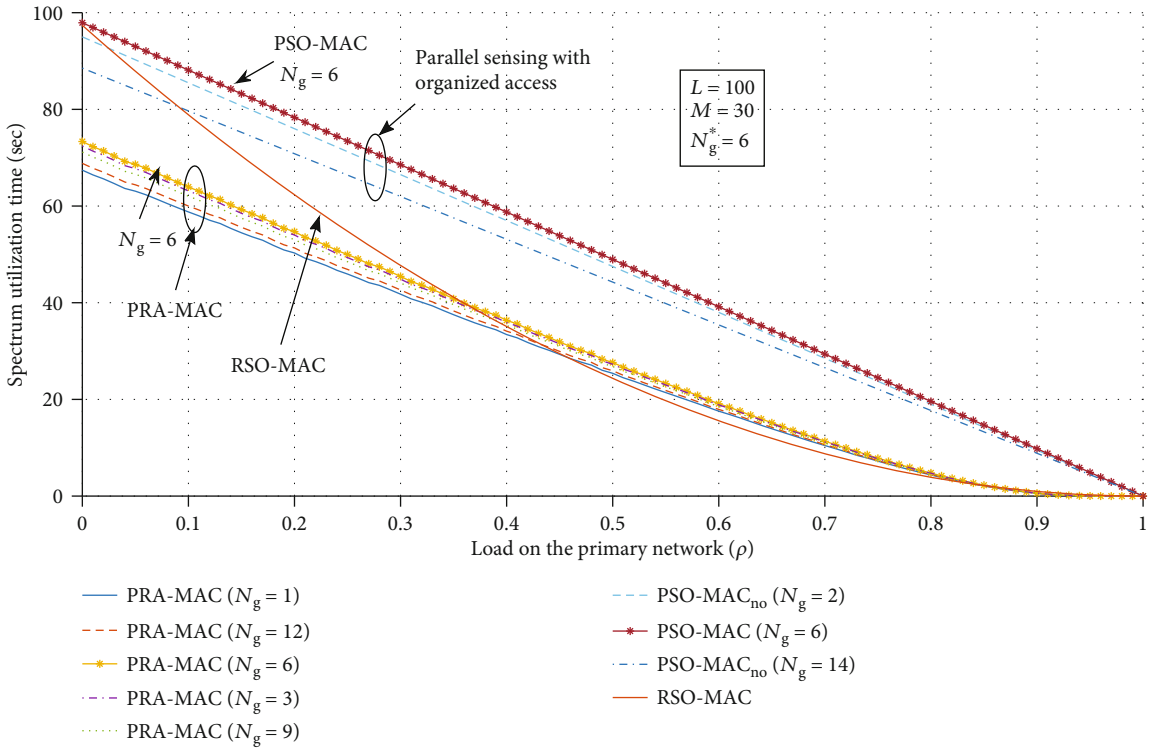


Figure 14: Successful spectrum utilization time.

resources, and successful utilization of these resources. For PSO-MAC, the optimal number of groups is determined as elaborated in the algorithm; in order to demonstrate the advantage of dividing the SUs into an optimal number of groups, we include the plots with nonoptimal number of groups, i.e., parallel sensing, organized access but number of groups randomly chosen.

Considering $M = 30$ and $L = 100$, the optimum number of groups as obtained through (10) is $N_g^* = 6$. Figure 13 demonstrates that the PSO-MAC maximizes the utilizable time from the primary network to the CRN by using the optimum number of groups. Any other value of N_g, smaller or greater, as used in PRA-MAC, reduces the utilizable time for the CRN. With 6 groups of SUs, each SU is required to sense $100/6 \approx 16$ channels. In order for a fair comparison of PSO-MAC with RSO-MAC, which employs the random sensing, each SU in RSO-MAC is also required to sense 16 channels, i.e., the value of j in (13) is 16.

The successful utilization of this spectrum utilizable time, acquired for SUs through sensing, depends upon the access scheme. In PSO-MAC and RSO-MAC schemes, where there is no contention, the successful utilization is very high. However, when these acquired resources are accessed through random access, as in PRA-MAC, the utilization drops sharply with the increase in load on the primary network.

As shown in Figure 14, the PRA-MAC despite accumulating large resources through parallel sensing wastes a lot of opportunities for data transmission due to random access. RSO-MAC by virtue of organized access does not miss too many opportunities, but since the sensing is performed randomly, the resources that it manages to accumulate a limited set of resources for its users in CRN are very limited. As a result, the successful utilization of the resources falls sharply with an increase in load on the primary network. By combining the parallel sensing with organized access, the utilization is significantly improved even for PSO-MAC$_{no}$, even for the case where the number of groups is randomly chosen and is a nonoptimal value. However, when the number of groups is optimized in PSO-MAC, the results clearly demonstrate its superior performance in comparison to other schemes.

6. Conclusions and Future Work

In this work, parallel sensing with self-organizing medium access control (PSO-MAC) has been proposed for an ad hoc CRN. Parallel sensing not only reduces the overhead time but also makes the discovery of idle channels more efficient through the division of the sensing task among the secondary users; this minimizes duplication as well as channel skipping. The self-organization of the secondary users in an ad hoc network maximizes the utilization of the discovered opportunities by reducing collisions and nonattempts on any discovered idle channel. Both these schemes, the parallel sensing and the self-organization MAC, require coordinated efforts among the secondary users which could result in significant overhead. This overhead decreases the valuable time available to the secondary users to use for the transmission of data.

To minimize this overhead and to maximize the useful time in a transmission cycle, an enhanced frame structure has been designed that unifies these two mechanisms while keeping the overhead to a minimum. This improved frame

structure enables to achieve the synergy where the gain of this unified approach is more than the sum of the gains of the two schemes working in isolation. This is clearly evident from the comparison of the network throughput achieved with the unified scheme to that for each of the two approaches separately. It can be noticed that the performance gains with the proposed PSO-MAC increase with the increase in the number of channels for the given load on the primary network and given the number of secondary users. This is primarily due to the careful design of the frame structure which includes a sequence of different phases, provision of dynamic adjustment of the frame according to the load on the primary network, and the number of secondary users. Another contributing factor is the optimization of the parallel sensing phase wherein the number of groups is dynamically adjusted according to the load on the primary network and the number of secondary users of the cognitive radio network. This dynamic adjustment makes the best compromise between the sensing duration in the frame and the portion of the primary band to be sensed so that the utilization is maximized.

There are several possible directions to extend this work further. For this work, the channel is considered busy if there are conflicting reports about the channel to provide the maximum protection to the primary user against interference from the CRN; other schemes such as the majority decision can increase the probability of finding a vacant channel, albeit at the cost of some interference to the primary network. Another challenging possibility could be to incorporate fairness guarantees in the self-organization scheme. Other possible extensions may include location-based grouping in parallel sensing and channel allocation using machine learning techniques such as reinforcement learning and deep learning.

Data Availability

Data is available on request from the corresponding author.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This work has been supported by (i) Higher Education Commission, Pakistan, through NRPU-10257 and (ii) Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Korea government (MSIT) (Grant No. NRF-2020R1F1A1048470) (Grant No. NRF-2019R1A4A1023746).

References

[1] Force, Spectrum Policy Task, *Spectrum Policy Task Force Report ET Docket No. 02-135*, US Federal Communications Commission, 2002.

[2] Y.-C. Liang, K.-C. Chen, G. Y. Li, and P. Mahonen, "Cognitive radio networking and communications: an overview," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 7, pp. 3386–3407, 2011.

[3] D. Datla, A. M. Wyglinski, and G. J. Minden, "A spectrum surveying framework for dynamic spectrum access networks," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 8, pp. 4158–4168, 2009.

[4] J. Mitola and G. Q. Maguire, "Cognitive radio: making software radios more personal," *IEEE Personal Communications*, vol. 6, no. 4, pp. 13–18, 1999.

[5] S. Haykin, "Cognitive radio: brain-empowered wireless communications," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 2, pp. 201–220, 2005.

[6] H. Anandakumar and K. Umamaheswari, "A bio-inspired swarm intelligence technique for social aware cognitive radio handovers," *Computers & Electrical Engineering*, vol. 71, pp. 925–937, 2018.

[7] A. Shakeel, R. Hussain, A. Iqbal, I. L. Khan, Q. u. Hasan, and S. A. Malik, "Analysis of efficient spectrum handoff in a multi-class hybrid spectrum access cognitive radio network using markov modelling," *Sensors*, vol. 19, no. 19, p. 4120, 2019.

[8] H. Anandakumar and K. Umamaheswari, "Supervised machine learning techniques in cognitive radio networks during cooperative spectrum handovers," *Cluster Computing*, vol. 20, no. 2, pp. 1505–1515, 2017.

[9] G. C. Deepak, K. Navaie, and Q. Ni, "Radio resource allocation in collaborative cognitive radio networks based on primary sensing profile," *IEEE Access*, vol. 6, pp. 50344–50357, 2018.

[10] P. Pawelczak, S. Pollin, H. S. So, A. Bahai, R. V. Prasad, and R. Hekmat, "Comparison of opportunistic spectrum multichannel medium access control protocols," in *IEEE GLOBECOM 2008-2008 IEEE global telecommunications conference*, New Orleans, LA, USA, 2008.

[11] H. W. So, J. Walrand, and J. Mo, "McMAC: a multi-channel MAC proposal for ad-hoc wireless networks," in *Proc. of IEEE WCNC*, Hong Kong, China, 2007.

[12] H. M. Almasaeid, "Maximizing achievable transmission time in cognitive radio networks under sensor-aided crowdsourced spectrum sensing," *The Computer Journal*, vol. 62, no. 10, pp. 1477–1489, 2019.

[13] H. A. Shah, K. S. Kwak, M. Sengoku, and S. Shinoda, "Reliable cooperative spectrum sensing through multi-bit quantization with presence of multiple primary users in cognitive radio networks," in *2019 34th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC)*, Jeju, South Korea, 2019.

[14] T. C. Thanuja, K. A. Daman, and A. S. Patil, "Optimized spectrum sensing techniques for enhanced throughput in cognitive radio network," in *2020 International Conference on Emerging Smart Computing and Informatics (ESCI)*, Pune, India, 2020.

[15] A. Tohamy, U. S. Mohammed, and T. A. Khalaf, "Cooperative sensing using maximum a posteriori as a detection technique in cognitive radio network," in *2019 36th National Radio Science Conference (NRSC)*, Port Said, Egypt, 2019.

[16] S. Mishra, S. S. Singh, and B. S. P. Mishra, "A comparative analysis of centralized and distributed spectrum sharing techniques in cognitive radio," in *Computational Intelligence in Sensor Networks*, pp. 455–472, Springer, Berlin, Heidelberg, 2019.

[17] M. Ozturk, M. Akram, S. Hussain, and M. A. Imran, "Novel QoS-aware proactive spectrum access techniques for cognitive radio using machine learning," *IEEE Access*, vol. 7, pp. 70811–70827, 2019.

[18] I. L. Khan, R. Hussain, A. Iqbal et al., "Design and evaluation of self organizing, collision free MAC protocol for distributed cognitive radio networks," *Wireless Personal Communications*, vol. 99, no. 2, pp. 1081–1101, 2018.

[19] H. Anandakumar and K. Umamaheswari, "An efficient optimized handover in cognitive radio networks using cooperative spectrum sensing," *Intelligent Automation & Soft Computing*, vol. 23, pp. 1–8, 2017.

[20] I. L. khan, R. Hussain, A. Shakeel et al., "Efficient idle channel discovery mechanism through cooperative parallel sensing in cognitive radio network," *EURASIP Journal on Wireless Communications and Networking*, vol. 2018, no. 1, 2018.

[21] M. Z. Alom, T. K. Godder, M. N. Morshed, and A. Maali, "Enhanced spectrum sensing based on energy detection in cognitive radio network using adaptive threshold," in *2017 International Conference on Networking, Systems and Security (NSysS)*, Dhaka, Bangladesh, 2017.

[22] Y. Arjoune, Z. El Mrabet, H. El Ghazi, and A. Tamtaoui, "Spectrum sensing: enhanced energy detection technique based on noise measurement," in *2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)*, University of Nevada, Las Vegas, USA, 2018.

[23] F. Salahdine, H. El Ghazi, N. Kaabouch, and W. F. Fihri, "Matched filter detection with dynamic threshold for cognitive radio networks," in *2015 international conference on wireless networks and mobile communications (WINCOM)*, Marrakech, Morocco, 2015.

[24] B. Khalfi, A. Zaid, and B. Hamdaoui, "When machine learning meets compressive sampling for wideband spectrum sensing," in *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC)*, Valencia, Spain, 2017.

[25] D. Wang and Z. Yang, "An novel spectrum sensing scheme combined with machine learning," in *2016 9th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, Datong, China, 2016.

[26] I. F. Akyildiz, B. F. Lo, and R. Balakrishnan, "Cooperative spectrum sensing in cognitive radio networks: a survey," *Physical communication*, vol. 4, no. 1, pp. 40–62, 2011.

[27] Y. Lu, D. Wang, and M. Fattouche, "Cooperative spectrum-sensing algorithm in cognitive radio by simultaneous sensing and BER measurements," *EURASIP Journal on Wireless Communications and Networking*, vol. 2016, no. 1, 2016.

[28] S. Senthilmurugan and T. G. Venkatesh, "Optimal channel sensing strategy for cognitive radio networks with heavy-tailed idle times," *IEEE Transactions on Cognitive Communications and Networking*, vol. 3, no. 1, pp. 26–36, 2017.

[29] X. Zhang, Y. Ma, Y. Gao, and W. Zhang, "Autonomous compressive-sensing-augmented spectrum sensing," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 8, pp. 6970–6980, 2018.

[30] A. Kumar, S. Saha, and K. Tiwari, "A double threshold-based cooperative spectrum sensing with novel hard-soft combining over fading channels," *IEEE Wireless Communications Letters*, vol. 8, no. 4, pp. 1154–1158, 2019.

[31] Q. Vien, H. X. Nguyen, R. Trestian, P. Shah, and O. Gemikonakli, "A hybrid double-threshold based cooperative spectrum sensing over fading channels," *IEEE Transactions on Wireless Communications*, vol. 15, no. 3, pp. 1821–1834, 2016.

[32] Q. Lu, S. Yang, and F. Liu, "Wideband spectrum sensing based on riemannian distance for cognitive radio networks," *Sensors*, vol. 17, no. 4, p. 661, 2017.

[33] Y. Arjoune and N. Kaabouch, "A comprehensive survey on spectrum sensing in cognitive radio networks: recent advances, new challenges, and future research directions," *Sensors*, vol. 19, no. 1, p. 126, 2019.

[34] A. Shakeel, R. Hussain, A. Iqbal, I. Latif Khan, Q. Ul Hasan, and S. Ali 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.

[35] Z.-H. Wei, B.-J. Hu, E.-J. Xia, and S.-H. Lu, "A contention-free reporting scheme based MAC protocol for cooperative spectrum sensing in cognitive radio networks," *IEEE Access*, vol. 6, pp. 38851–38859, 2018.

[36] S. Pandit and G. Singh, "Backoff algorithm in cognitive radio MAC protocol for throughput enhancement," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 5, pp. 1991–2000, 2015.

[37] J. S. P. Singh and M. K. Rai, "Cognitive radio intelligent-MAC (CR-i-MAC): channel-diverse contention free approach for spectrum management," *Telecommunication Systems*, vol. 64, no. 3, pp. 495–508, 2017.

[38] S. Lim and T.-J. Lee, "A self-scheduling multi-channel cognitive radio MAC protocol based on cooperative communications," *IEICE Transactions on Communications*, vol. E94-B, no. 6, pp. 1657–1668, 2011.

Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 9915420, 25 pages
https://doi.org/10.1155/2021/9915420

Research Article

A Novel Modified Sparrow Search Algorithm with Application in Side Lobe Level Reduction of Linear Antenna Array

Qiankun Liang[1], **Bin Chen**[1], **Huaning Wu**[1], **Chaoyi Ma**[2] **and Senyou Li**[2]

[1]*Naval University of Engineering, College of Electronic Engineering, Wuhan 430033, China*
[2]*China Academy of Information and Communications Technology, Beijing 100191, China*

Correspondence should be addressed to Huaning Wu; wuhuaning007@163.com

Received 18 March 2021; Revised 17 May 2021; Accepted 4 June 2021; Published 28 June 2021

Academic Editor: Fawad Zaman

Antenna arrays play an increasingly important role in modern wireless communication systems. However, how to effectively suppress and optimize the side lobe level (SLL) of antenna arrays is critical for communication performance and communication capabilities. To solve the antenna array optimization problem, a new intelligent optimization algorithm called sparrow search algorithm (SSA) and its modification are applied to the electromagnetics and antenna community for the first time in this paper. Firstly, aimed at the shortcomings of SSA, such as being easy to fall into local optimum and limited convergence speed, a novel modified algorithm combining a homogeneous chaotic system, adaptive inertia weight, and improved boundary constraint is proposed. Secondly, three types of benchmark test functions are calculated to verify the effectiveness of the modified algorithm. Then, the element positions and excitation amplitudes of three different design examples of the linear antenna array (LAA) are optimized. The numerical results indicate that, compared with the other six algorithms, the modified algorithm has more advantages in terms of convergence accuracy, convergence speed, and stability, whether it is calculating the benchmark test functions or reducing the maximum SLL of the LAA. Finally, the electromagnetic (EM) simulation results obtained by FEKO also show that it can achieve a satisfactory beam pattern performance in practical arrays.

1. Introduction

The latest development trend of wireless communication systems is to realize the antenna array system with strong directivity and maneuverability, so that it can radiate and receive energy to the maximum extent in a specific direction and reduce the waste of energy by suppressing the SLL in noninteresting directions [1]. According to the pattern multiplication theorem, the pattern of an antenna array can be obtained by multiplying the element pattern by the array factor. By adjusting the spacing, excitation amplitude, and phase of array factor, the antenna array can have the characteristics of high gain, narrow beam, low SLL, and easy electrical scanning [2]. In the field of scientific research and engineering practice, most of the problems encountered can be attributed to solving optimization problems, and the design and optimization of the antenna array are no exception [3]. With the development of computer technologies and computational electromagnetics, some intelligent optimization algorithms that simulate the behavior mechanism of biological groups or the laws of natural phenomena have begun to appear in the vision of many scholars. With its unique advantages in solving large-scale, nonlinear, and other complex optimization problems, the design and optimization technology of antenna arrays based on an intelligent optimization algorithm has always been a research hotspot in the field of EM optimization [4].

In the past few decades, various intelligent optimization algorithms have been implemented to optimize and design antenna arrays, such as genetic algorithm (GA) [5], particle swarm optimization (PSO) [6], bees algorithm (BA) [7], biogeography-based optimization (BBO) [8], firefly algorithm (FA) [9], cat swarm optimization (CSO) [10], cuckoo optimization algorithm (COA) [11], backtracking search algorithm (BSA) [12], symbiotic organisms search (SOS) [13], grey wolf optimization (GWO) [14], extended GWO (GWO-E) [15], spider monkey optimization (SMO) [16], gravitational search algorithm (GSA) [17], invasive weed

optimization (IWO) [18], elephant swarm water search algorithm (ESWSA) [19], grasshopper optimization algorithm (GOA) [20], and equilibrium optimization algorithm (EOA) [21]. These algorithms have been successfully applied in this field. However, according to no free lunch (NFL) theorem, no algorithm can perform best on all kinds of optimization problems [22]. Therefore, finding and researching more efficient algorithms are still a problem worthy of attention in the field of EM optimization.

SSA is a new intelligent optimization algorithm proposed by Xue and Shen in 2020. It is mainly inspired by the foraging and antipredation behaviors of sparrows. It has the characteristics of simple implementation, few adjustment parameters, and high expansibility and has been successfully applied in many scientific research and engineering practice fields [23]. For example, [24] proposed an improved SSA to solve the path planning problem of UAV under constraint and proved that the route generated by this method is better than that by the other four algorithms in the same environment. Wang and Xianyu used SSA to solve the optimal configuration model of distributed power supply for the first time and verified the effectiveness and superiority of this method through the IEEE 33 distribution system [25]. Reference [26] established the SSA model based on empirical mode decomposition to optimize the parameters of the kernel extreme learning machine (KELM) and achieved higher prediction accuracy in blood glucose prediction. Kumaravel and Ponnusamy improved the control parameters of the power controller based on SSA to optimize the power flow management of the smart grid system, which realized the real-time energy management in the microgrid [27]. In [28], the improved model based on SSA can track the distributed maximum power point more accurately and quickly and has good robustness, thus effectively solving the problem of power mismatch loss in a photovoltaic microgrid system. Zhu and Yousefi introduced an adaptive strategy on the basis of SSA and applied it to the optimization of proton exchange membrane fuel cell (PEMFC) stack model parameters. The validity of the proposed method in determining the maximum output power and optimal operating state of the stack is verified through four cases [29]. With the help of the adaptive SSA, Liu and Rodriguez took a residential building as an example and took the energy load demand and investment cost as the optimization objectives to get the best combination scheme of the integrated sustainable energy systems, which achieved the purpose of reducing energy consumption and saving economic expenses [30].

LAA is known as the basic and one of the most practical geometric configurations of antenna array in which all elements are arranged in a straight line. Aimed at the design and optimization of LAA, a new modified SSA is proposed in this paper. The main contributions are briefly highlighted as follows:

(i) Proposition of the MSSA to further enhance the performance of SSA

(ii) Application of the algorithm to the electromagnetics and antenna community for the first time

(iii) Design of a few different scenarios of LAA for the maximum SLL reduction by optimizing the element spacing and excitation amplitude

(iv) Consideration of an additional constraint on the total length of the antenna array to preserve the features of the beam pattern

(v) Comparison of the results with several classical and well-known algorithms in function optimization and antenna array design problem

(vi) EM simulation with Altair FEKO 2019 to test the validity of the experiment results in practical conditions

The rest of this paper is structured as follows. Section 2 describes the basic principle of SSA, introduces the improvement strategies, and gives the pseudocode of the modified algorithm. Section 3 verifies its effectiveness and convergence performance. In Section 4, the mathematical description of the LAA optimization problem is given, and the simulation results are discussed and analyzed, and then, the EM verification experiment is carried out. Finally, Section 5 draws the conclusion of this paper.

2. The Sparrow Search Algorithm and Its Modification

2.1. Standard Sparrow Search Algorithm. A sparrow is an intelligent social animal, which keeps alert and stays a safe distance at all times. They also show great unity when encountering the enemy. According to their different behavior rules, sparrows can be divided into three roles: producers, scroungers, and scouters. Assuming that there are N sparrows in a D-dimensional search space, the position of the ith sparrow can be expressed as $X_i = [X_{i,1}, X_{i,2}, \cdots, X_{i,D}]$, $i = 1, 2, \cdots, N$. The following is an introduction of the three updating methods.

2.1.1. Producers. Producers are sparrows with better fitness values in the population. They have a wide search range and are responsible for searching and providing foraging directions for the whole population. The mathematical expression of producers is described as follows:

$$X_{i,j}^{t+1} = \begin{cases} X_{i,j}^{t} \cdot \exp\left(-\dfrac{i}{\alpha \cdot t_{\max}}\right), & R_2 < \mathrm{ST}, \\ X_{i,j}^{t} + Q \cdot L, & R_2 \geq \mathrm{ST}, \end{cases} \tag{1}$$

where t represents the current iteration number and $t_{\max}$ represents the maximum number of iterations. $X_{i,j}$ denotes the position of the ith sparrow in the jth dimension. α is a uniform random number in the range $(0, 1]$, and R_2 and ST represent the alarm value and the safety threshold respectively, where $R_2 \in [0, 1]$ and $\mathrm{ST} \in [0, 1]$. Q is a random number with normal distribution. L is a one-dimensional matrix with all elements of 1. When $R_2 < \mathrm{ST}$, it means that the surrounding environment is safe and they can search for food

extensively. When $R_2 \geqq \mathrm{ST}$, it means that there is danger at this time, and all sparrows have to fly to other safe areas quickly.

2.1.2. Scroungers. Scroungers are sparrows except all the producers and keep an eye on the producers. If they find that the producers have found better food, they will immediately leave their present position to fight for food and make themselves the producers. The position of the scroungers is updated as follows:

$$X_{i,j}^{t+1} = \begin{cases} Q \cdot \exp\left(\dfrac{X_{\text{worst}} - X_{i,j}^{t}}{i^2}\right), & i > \dfrac{N}{2}, \\ X_p^{t+1} + \left|X_{i,j}^{t} - X_P^{t+1}\right| \cdot A^{+} \cdot L, & i \leq \dfrac{N}{2}, \end{cases} \tag{2}$$

where X_{worst} represents the global worst position of the current population and X_p represents the best position occupied by the producers in the current iteration process. A is a one-dimensional matrix with elements of 1 or -1, and $A^{+} = (AA^{T})^{-1}$. If $i > N/2$, it indicates that the ith scrounger with low fitness value is in an unfavorable position and needs to expand his flight range to obtain food; if $i \leq N/2$, the ith scrounger will find a random place near the optimal location and perform local search.

2.1.3. Scouters. The scouters are randomly generated between the producers and the scroungers and can perceive whether there is danger in the foraging area. The model of scouters can be formulated as follows:

$$X_{i,j}^{t+1} = \begin{cases} X_{\text{best}}^{t} + \beta \cdot \left|X_{i,j}^{t} - X_{\text{best}}^{t}\right|, & F_i > F_g, \\ X_{i,j}^{t} + K \cdot \left(\dfrac{\left|X_{i,j}^{t} - X_{\text{worst}}^{t}\right|}{(F_i - F_w) + \varepsilon}\right), & F_i = F_g, \end{cases} \tag{3}$$

where X_{best} represents the global optimal position of the current population, and it is also the safest location. As a step size control parameter, β is a random number subject to standard normal distribution. $K \in [-1, 1]$ indicates the direction of movement of the sparrows. F_i, F_g, and F_w represent the fitness value of the ith sparrow, the global optimal, and the worst fitness value of the current population, respectively. ε is a minimal constant that avoids zero division error. $F_i > F_g$ indicates that the sparrow is at the edge of the population, vulnerable to predators, and needs to move to a safe area. $F_i = F_g$ indicates that the sparrow is in the middle of the population, but it is aware of the danger and needs to be close to other sparrows to reduce the risk of predation.

2.2. Modified Sparrow Search Algorithm. Compared with other representative intelligent optimization algorithms in recent years, although SSA has strong competitiveness in convergence speed, accuracy, and stability [31], it is still inevitable to fall into local optimum at the later stage of iterations, resulting in insufficient convergence accuracy [32]. In order to further improve the performance of the algorithm, this paper proposes a new modified algorithm based on chaotic adaptive inertia weight and improved boundary constraint.

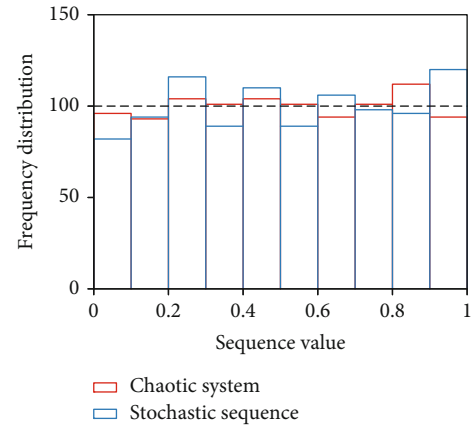


FIGURE 1: Statistical histogram of an ordinary random sequence and chaotic sequence.

2.2.1. Homogeneous Chaotic System. The quality of initial solution directly affects whether the algorithm can find the optimal solution. Chaos is a kind of random phenomenon with ergodicity, inherent regularity, and long-term unpredictability [33]. Within the search range of feasible solutions, chaotic sequences are widely used in population initialization of optimization algorithms; this is because they can traverse all states without repetition [34]. The research in Reference [35] shows that the homogeneous chaotic system has better random effect in variable initialization. Its function is expressed as

$$\begin{cases} \mu(t+1) = 3.5\mu(t)^2 + 3.3\mu(t) - 0.265, \\ \bar{X}(t+1) = \dfrac{1}{\pi} \arcsin\left(-\dfrac{7}{4}\mu(t+1) - \dfrac{33}{40}\right), \end{cases} \tag{4}$$

where $\mu \in [-73/70, 1/10]$ is the initialization sequence and $\bar{X}$ is completely chaotic in $[-1/2, 1/2]$. The formula for transforming chaotic sequence $\bar{X}$ into the solution space is as follows:

$$X = \frac{\mathrm{Ub} + \mathrm{Lb}}{2} + \bar{X}(\mathrm{Ub} - \mathrm{Lb}), \tag{5}$$

where Ub and Lb, respectively, represent the upper and lower boundary values of the optimized variables.

The statistical histogram obtained from the numerical statistics of ordinary random sequences and sequences generated by the homogeneous chaotic system is shown in Figure 1. It can be seen from the figure that the chaotic system has better homogenization, that is, better randomness. Hence, when the diversity of the sparrows increases, the quality of initial solution can be improved.

2.2.2. Adaptive Inertia Weight. All intelligent optimization algorithms include two processes of global exploration and local exploitation. An efficient algorithm needs to balance the global exploration capability and the local exploitation capability [36]. From the basic principle of SSA in Section 2.1, it is not difficult to find that the producer's search ability plays a vital role in whether the algorithm can find the optimal solution. Therefore, inertia weight w is introduced to adjust it adaptively. The mathematical expression is shown in

$$w = w_{\min} + (w_{\max} - w_{\min}) \cdot \text{rand}\,(1) \cdot \exp\left(-\frac{t}{t_{\max}}\right). \quad (6)$$

Take $w_{\max} = 0.1$ and $w_{\min} = 0.01$. The schematic diagram is illustrated in Figure 2.

As can be noticed, as the number of iterations increases, the inertia weight decreases adaptively. The update formula of the producers' position is modified as follows:

$$X_{i,j}^{t+1} = \begin{cases} X_{i,j}^{t} \cdot \exp\left(-\dfrac{i}{w \cdot \alpha \cdot t_{\max}}\right), & R_2 < \text{ST}, \\ X_{i,j}^{t} + Q \cdot L, & R_2 \geq \text{ST}. \end{cases} \quad (7)$$

By introducing the adaptive inertia weight, sparrow individuals can search favorable regions in the global range with a larger step size in the early stage of search and strengthen the ability of global exploration; in the later stage of the search, a smaller weight w can ensure sparrows to do fine search near the extreme points and strengthen the ability of local exploitation, so that the algorithm has a greater probability of converging to the global optimal value.

2.2.3. Improved Boundary Constraint. In the standard SSA, the processing strategy for sparrows overstepping the boundary is generally given by

$$X_{i,j}^{t+1} = \begin{cases} \text{Ub}, & X_{i,j}^{t+1} > \text{Ub}, \\ \text{Lb}, & X_{i,j}^{t+1} < \text{Lb}. \end{cases} \quad (8)$$

In this method, the individual that oversteps the boundary is simply assigned to the boundary value of the search range, which is equivalent to giving up the individual's search information. In the iterative process, if more individuals overstep the boundary, the positions of sparrows will accumulate more boundary values, resulting in the decrease in population diversity, which directly affects the convergence accuracy of the algorithm [37]. The strategy of improved boundary constraint handling is shown in Algorithm 1. Sparrows that overstep the boundary will randomly determine a position near the optimal position of the population, which enhances the diversity of the population and improves the global optimization ability of the algorithm to a certain extent.

The SSA modified by the above strategies is named MSSA, and the pseudocode outlining the steps of its implementation is shown in Algorithm 2.

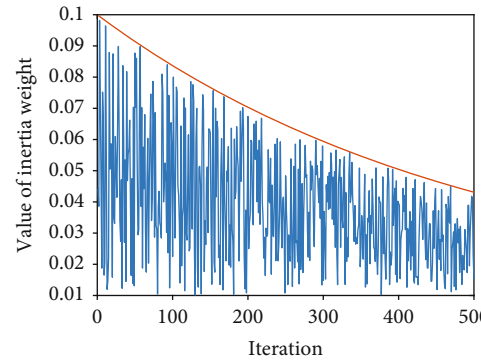


Figure 2: Schematic diagram of inertia weight.

3. Performance Analysis

In this section, MATLAB is used to verify and analyze the computational performance of MSSA on benchmark test functions. In order to verify the effectiveness and superiority of the modified algorithm, classical PSO and representative algorithms in recent years, including PSOGSA, WOA, GOA, and MTDE, are adopted for comparative experiments. All the parameters of these algorithms are in accordance with the original papers, as shown in Table 1.

The population size of all algorithms is set to 30, and the number of iterations is 500, for fairness. All numerical experiments are implemented on Intel(R) Core(TM) i5-9400U CPU with 2.90 GHz and 8 GB RAM.

3.1. Benchmark Test Functions. In order to comprehensively evaluate the global and local optimization ability of the algorithm, the experiment uses three types of benchmark test functions, in which the unimodal function has only a global optimal value but no local optimal value, so it can better test the local exploitation ability of the algorithm; the multimodal function has many local optimal values, so it can test the ability of the algorithm to jump out of the local extreme value and the global exploration ability. Table 2 shows the benchmark test functions for the experiment.

3.2. Experimental Results and Analysis. Considering the randomness of algorithm operation, 50 tests were run independently in order to make the results more convincing and universal. The best value and the worst value can show the exploration ability of the algorithm, and the mean value and standard deviation can show the accuracy and stability of the algorithm. Therefore, the best value, the worst value, and the mean value of 50 experimental results were statistically analyzed, and the standard deviation was calculated. The numerical results are shown in Table 3, where the optimal value is expressed in bold.

As can be seen from the results in Table 3, for unimodal functions F1~F3, the accuracy and stability of MSSA are greatly improved as compared to the other six algorithms, and the results of multiple optimizations can converge to the optimal value of the functions with the smallest standard deviation. For multimodal functions F4~F6, MSSA as well as SSA has excellent performance, and their ability to escape

Input: $X_{i,j}^{*}$ (Position that overstep the boundary), Ub (Upper boundary), Lb (Lower boundary), X_{best} (Global optimal position), X_{good} (Current optimal position)
Output: $X_{i,j}^{*}$ (The new position)

if $X_{i,j}^{*} < Lb$ || $X\ i,j > Ub$ **then**
  $temp = X_{best} + | X_{best} - X_{good} | \cdot rand(1)$;
  if $Lb \le temp \le Ub$ **then**
    $X_{i,j}^{*} = temp$;
  else
    $X_{i,j}^{*} = X_{good}$;
  end if
end if
return $X_{i,j}^{*}$

Algorithm 1: Improved boundary constraint.

Input: N (Population size), D (Dimension size), P_{Num} (Producers size), S_{Num} (Scouters size), ST (Safety threshold), t (Initial iteration), t_{max} (Maximum iterations)
Output: X_{best} (Global optimal position), F_{best} (Fitness of global optimal position)

/∗Initializing∗/
Randomly generate the positions of N sparrows $X_{i,j}$ by homogeneous chaotic system (i=1,2,…,N, j=1,2,…,D);
Calculate the fitness of each sparrow F_i;
Find X_{best} and F_{best};
/∗Iterating∗/
while $t < t_{max}$ **do**
  Sort the F_i and find global worst position X_{worst};
  $R_2 = rand(1)$;
  for $i = 1 : P_{Num}$
    Evaluate and calculate the adaptive weight w;
    Update the position $X_{i,j}$ by (7);
    Check and adjust position that overstep the boundary by Algorithm 1;
  end for
  Sort the F_i and find the best position of producers X_p;
  for $i = (P_{Num} + 1) : N$
    Update the position $X_{i,j}$ by (2);
    Check and adjust position that overstep the boundary by Algorithm 1;
  end for
  for $i = 1 : S_{Num}$
    Update the position $X_{i,j}$ by (3);
    Check and adjust position that overstep the boundary by Algorithm 1;
  end for
    Evaluate and update X_{best} and F_{best};
    $t = t + 1$;
end while
return X_{best} and F_{best};

Algorithm 2: Modified sparrow search algorithm.

local optimal solutions and robustness are obviously better than those of the remaining algorithms. For fixed-dimensional multimodal functions F7~F9, all algorithms can find the optimal value of the functions. MSSA's performance is inferior to that of MTDE for function F7, but better than that of PSOGSA, GOA, and SSA. For function F8, the standard deviation of MSSA is the smallest, and MSSA performs best for function F9.

In order to directly reflect the convergence characteristics of the modified algorithm on the benchmark test functions, the average convergence curves of the seven algorithms are presented in Figure 3. The logarithmic axis is adopted for the y-axis to make data fluctuation much clearer. It can be seen from Figure 3 that MSSA not only has higher calculation accuracy but also significantly improves the convergence speed.

Table 1: Parameters of different algorithms.

Algorithm	Parameter	Value
PSO [38, 39]	Individual learning factor $c1$	2
	Social learning factor c_2	2
	Inertia weight w	1.05
PSOGSA [40]	Individual learning factor c_1	0.5
	Social learning factor c_2	1.5
	Weighting function w	$[0, 1]$
	Gravitational constant G	1
	Alpha α	20
WOA [41]	Variable a	$2 \longrightarrow 0$
	Constant defining logarithmic spiral shape b	1
GOA [42]	Minimum reduction factor $c_{\min}$	0.00004
	Maximum reduction factor $c_{\max}$	1
	Attraction intensity f	0.5
	Attractive length scale l	1.5
MTDE [43]	Number of portions divided by iterations WinIter	20
	Gbest-history size H	5
	Nonlinear decreased coefficient a_2	$0.001 \longrightarrow 2$
	Dimension-dependent value Mu	$\log(D)$
	Mean value of improved scale factors μ_f	0.5
	Variance of improved scale factors σ	0.2
SSA [23]	Number of producers P_{Num}	20%
	Number of scouters S_{Num}	10%
	Safety threshold ST	0.8

Aimed at observing the search process more intuitively, Figure 4 shows the search trajectories of sparrows on the three types of benchmark test functions of the MSSA. It can be seen that in the early iterations, sparrows are scattered over a large search space. As the number of iterations increases, the adaptive weight factor improves the local optimization ability of the algorithm, and precocity is avoided in the end. Consequently, most sparrows are able to gather in the optimal solution and its nearby area (red box in Figure 4).

3.3. Comparison of the Improved Factors. Tests are conducted to observe the performance of the three introduced improved factors. The standard SSA with the modifications of the homogeneous chaotic system function and the improved boundary constraint is named CSSA, while the SSA with the modifications of adaptive inertia weight and the improved boundary constraint is named ASSA. Figure 5 shows the convergence curves of the improved factors on the three types of benchmark test functions. It can be found from the figure that MSSA has better performance by combining those improved factors. The reason may be attributed to the fact that the adaptive inertia weight can speed up the local convergence rate. Chaos initialization and the improved boundary constraint can increase the diversity of population to a certain extent, which lays a good foundation for the improvement of the convergence accuracy.

3.4. Statistical Testing. The Wilcoxon rank sum test is performed to statistically identify significant differences between the two algorithms. The best values, worst values, mean values, and standard deviations of all benchmark test functions in Table 3 are adopted as test samples. A nonparametric test is done by comparing MSSA with PSO, PSOGSA, WOA, GOA, MTDE, and SSA respectively. It is generally considered that the algorithm is significantly different from others if the p value is less than 0.05. The results from Table 4 show that all the p values are less than 0.05, except for the result of MSSA versus SSA. This indicates that there are significant differences between MSSA and most algorithms at the 5% significance level. In the calculation of the above benchmark test functions, although the performance of MSSA is better than that of SSA, there is no significant difference between them.

3.5. Algorithm Complexity. The computational cost of the intelligent optimization algorithm is mainly determined by the evaluation of fitness function. Assuming that the maximum iteration is T, the complexity of MSSA can be written as $O(NT)$ since there is only one inner loop in the algorithm, where N is the population size. It can be seen that the computational complexity is linear with N and T. According to the above modifications, MSSA does not change the algorithm framework of SSA, so they have the same complexity, and the proposed algorithm does not reduce the solution efficiency.

3.6. Analysis of Asymptotic Property. For the minimization optimization problem, the fitness asymptotic property of MSSA is analyzed in this section. Suppose that the finite nonempty set $\mathbf{S}$ is the solution space and the optimal solution set $\Omega = \{x^* | x^* \in \mathbf{S}, f(x^*) < \varepsilon\}$, where f is the objective function and ε is the acceptable objective function value. Let the optimal sparrow position obtained by the algorithm in the tth iteration be x_t^*, and $\{d_t | d_t = f(x_t^*) - f(x^*),\ 1 \le t \le T\}$ is the nonnegative random process generated by the algorithm. Since the algorithm gets the optimal sparrow individual through the idea of survival of the fittest; hence, $P(f(x_{t+1}^*) - f(x_t^*) > 0) = 0$. In addition, because the sparrow population updates its position in three ways randomly, when $f(x_t^*) > f(x^*)$, $P(f(x_{t+1}^*) = f(x_t^*)) \neq 1$. Therefore, $P(f(x_{t+1}^*) - f(x_t^*) < 0) > 0$. Let $E[f(x_{t+1}^*) - f(x_t^*)] = -\tau_{t+1}$, and $\tau_{t+1} > 0$ can be obtained from the above analysis. Hence,

$$\begin{aligned} E[d_{t+1} - d_t] &= E[(f(x_{t+1}^*) - f(x_t^*)) - (f(x_t^*) - f(x^*))] \\ &= E[f(x_{t+1}^*) - f(x_t^*)] = -\tau_{t+1}. \end{aligned} \tag{9}$$

That is, $E(d_{t+1}) = E(d_t) - \tau_{t+1}$.

Let $\tau = \min\{\tau_1, \tau_2, \cdots, \tau_T\}$, and there is $E(d_{t+1}) \le E(d_t) - \tau$. Therefore, the fitness asymptotic property of MSSA can be proven, which means that the algorithm asymptotically converges to the optimal solution.

Table 2: Benchmark test functions.

Function's type	Function's name	Function's equation	D	Search space
Unimodal	Sphere	$F_1(x) = \sum_{i=1}^{D} x_i^2$	30	$[-100,100]$
	Schwefel 2.22	$F_2(x) = \sum_{i=1}^{D} \lvert x_i \rvert + \prod_{i=1}^{D} \lvert x_i \rvert$	30	$[-10, 10]$
	Schwefel 1.2	$F_3(x) = \sum_{i=1}^{D} \left(\sum_{j=1}^{i} x_j \right)^2$	30	$[-100,100]$
Multimodal	Rastrigin	$F_4(x) = \sum_{i=1}^{D} \left[x_i^2 - 10 \cos(2\pi x_i) + 10 \right]$	30	$[-5.12,5.12]$
	Ackley	$F_5(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{D} \sum_{i=1}^{D} x_i^2} \right) - \exp\left(\frac{1}{D} \sum_{i=1}^{D} \cos(2\pi x_i) \right) + 20 + e$	30	$[-32, 32]$
	Griewank	$F_6(x) = \frac{1}{4000} \sum_{i=1}^{D} x_i^2 - \prod_{i=1}^{D} \cos\left(\frac{x_i}{\sqrt{i}} \right) + 1$	30	$[-600,600]$
Fixed-dimension multimodal	Goldstein–Price	$F_7(x) = \left[1 + (x_1 + x_2 + 1)^2 \left(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2 \right) \right] \times \left[30 + (2x_1 - 3x_2)^2 \times \left(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2 \right) \right]$	2	$[-2, 2]$
	Hartman	$F_8(x) = -\sum_{i=1}^{4} c_i \exp\left(-\sum_{j=1}^{3} a_{ij} \left(x_j - p_{ij} \right)^2 \right)$	3	$[0, 1]$
	Shekel's foxholes	$F_9(x) = -\sum_{i=1}^{5} \left[(X - a_i)(X - a_i)^{\mathrm{T}} + c_i \right]^{-1}$	4	$[0, 10]$

Table 3: Numerical results.

	Value	PSO	PSOGSA	WOA	GOA	MTDE	SSA	MSSA
F1	Best	5.03e + 03	3.52e − 01	4.42e − 86	6.16e + 00	6.26e − 03	**0.00e + 00**	**0.00e + 00**
	Worst	9.04e + 03	1.00e + 04	1.18e − 68	1.31e + 02	7.22e − 01	2.60e − 69	**0.00e + 00**
	Mean	7.29e + 03	1.43e + 03	2.37e − 70	3.74e + 01	1.06e − 01	5.40e − 71	**0.00e + 00**
	Std	9.60e + 02	3.49e + 03	1.66e − 69	2.64e + 01	1.31e − 01	3.68e − 70	**0.00e + 00**
F2	Best	3.40e + 01	1.87e − 08	7.17e − 57	3.82e + 00	2.76e − 03	**0.00e + 00**	**0.00e + 00**
	Worst	5.83e + 01	8.13e + 01	2.93e − 49	8.62e + 01	3.09e − 01	1.40e − 34	**0.00e + 00**
	Mean	4.31e + 01	1.21e + 01	1.05e − 50	1.41e + 01	4.53e − 02	2.89e − 36	**0.00e + 00**
	Std	5.73e + 00	1.84e + 01	4.75e − 50	1.34e + 01	5.18e − 02	1.98e − 35	**0.00e + 00**
F3	Best	1.09e + 04	2.94e + 03	7.41e + 03	7.58e + 02	1.06e + 02	**0.00e + 00**	**0.00e + 00**
	Worst	3.34e + 04	5.34e + 04	8.57e + 04	7.27e + 03	1.07e + 03	2.38e − 51	**0.00e + 00**
	Mean	1.91e + 04	1.48e + 04	4.30e + 04	2.87e + 03	3.89e + 02	4.76e − 53	**0.00e + 00**
	Std	4.80e + 03	8.81e + 03	1.64e + 04	1.39e + 03	2.10e + 02	3.37e − 52	**0.00e + 00**
F4	Best	2.11e + 02	7.26e + 01	**0.00e + 00**	4.14e + 01	1.37e + 01	**0.00e + 00**	**0.00e + 00**
	Worst	2.70e + 02	2.32e + 02	1.13e − 13	1.76e + 02	4.41e + 01	**0.00e + 00**	**0.00e + 00**
	Mean	2.40e + 02	1.37e + 02	3.41e − 15	9.61e + 01	2.52e + 01	**0.00e + 00**	**0.00e + 00**
	Std	1.30e + 01	3.39e + 01	1.76e − 14	2.80e + 01	7.14e + 00	**0.00e + 00**	**0.00e + 00**
F5	Best	1.28e + 01	7.04e + 00	**8.88e − 16**	2.99e + 00	5.54e − 02	**8.88e − 16**	**8.88e − 16**
	Worst	1.51e + 01	1.92e + 01	7.99e − 15	8.99e + 00	2.50e + 00	**8.88e − 16**	**8.88e − 16**
	Mean	1.43e + 01	1.57e + 01	4.79e − 15	5.33e + 00	1.44e + 00	**8.88e − 16**	**8.88e − 16**
	Std	4.52e − 01	3.36e + 00	2.48e − 15	1.38e + 00	5.53e − 01	**0.00e + 00**	**0.00e + 00**
F6	Best	4.47e + 01	5.66e − 01	0.00e + 00	8.69e − 01	2.26e − 02	**0.00e + 00**	**0.00e + 00**
	Worst	8.84e + 01	9.19e + 01	1.60e − 01	1.40e + 00	6.12e − 01	**0.00e + 00**	**0.00e + 00**
	Mean	6.65e + 01	2.13e + 01	7.85e − 03	1.11e + 00	1.27e − 01	**0.00e + 00**	**0.00e + 00**
	Std	8.99e + 00	3.74e + 01	3.26e − 02	1.01e − 01	1.15e − 01	**0.00e + 00**	**0.00e + 00**
F7	Best	**3.00e + 00**	**3.00e + 00**	**3.00e + 00**	**3.00e + 00**	**3.00e + 00**	**3.00e + 00**	**3.00e + 00**
	Worst	3.05e + 00	8.40e + 01	**3.00e + 00**	8.40e + 01	**3.00e + 00**	3.00e + 01	3.00e + 01
	Mean	3.01e + 00	6.24e + 00	3.54e + 00	7.86e + 00	**3.00e + 00**	5.16e + 00	4.62e + 00
	Std	8.78e − 03	1.60e + 01	3.78e + 00	1.92e + 01	**2.29e − 15**	7.32e + 00	6.48e + 00
F8	Best	**−3.86e + 0**	**−3.86e + 0**	**−3.86e + 0**	**−3.86e + 0**	**−3.86e + 0**	**−3.86e + 0**	**−3.86e + 0**
	Worst	−3.85e + 0	**−3.86e + 0**	−3.09e + 0	−3.03e + 0	**−3.86e + 0**	−3.08e + 0	**−3.86e + 0**
	Mean	**−3.86e + 0**	**−3.86e + 0**	−3.83e + 0	−3.73e + 0	**−3.86e + 0**	−3.85e + 0	**−3.86e + 0**
	Std	1.52e − 03	2.90e − 15	1.09e − 01	2.53e − 01	3.13e − 15	1.08e − 01	**2.83e − 15**
F9	Best	−9.51e + 0	**−1.02e + 1**	**−1.02e + 1**	**−1.02e + 1**	**−1.02e + 1**	**−1.02e + 1**	**−1.02e + 1**
	Worst	−2.14e + 0	−2.63e + 0	−2.63e + 0	−2.63e + 0	**−5.06e + 0**	**−5.06e + 0**	**−5.06e + 0**
	Mean	−6.31e + 0	−6.58e + 0	−7.49e + 0	−6.44e + 0	−9.23e + 0	−8.62e + 0	**−9.54e + 0**
	Std	2.32e + 00	3.32e + 00	3.04e + 00	3.29e + 00	1.98e + 00	2.33e + 00	**1.67e + 00**

4. Experiments on Maximum SLL Reduction of LAA

In order to test the feasibility of optimizing EM problems based on the MSSA, this section designed the simulation experiments of optimizing the element spacings and excitation amplitudes of LAA through MATLAB. Firstly, the system model is presented and the fitness function is formulated. Secondly, different design scenarios of maximum SLL reduction are simulated by MSSA, and the obtained results are compared with those of PSO, PSOGSA, WOA, GOA, MTDE, and SSA. Then, the stability of the algorithms is

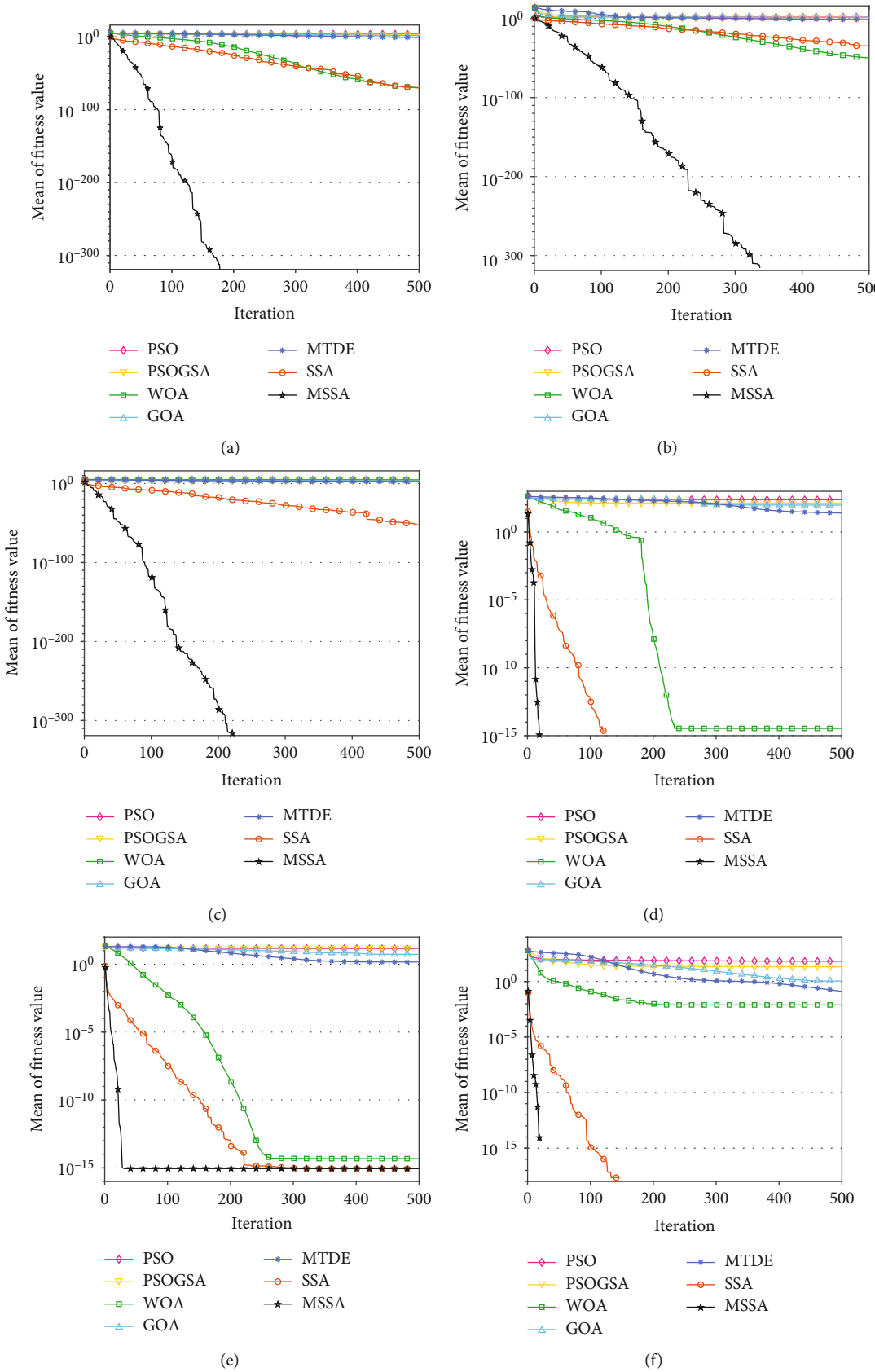


Figure 3: Continued.

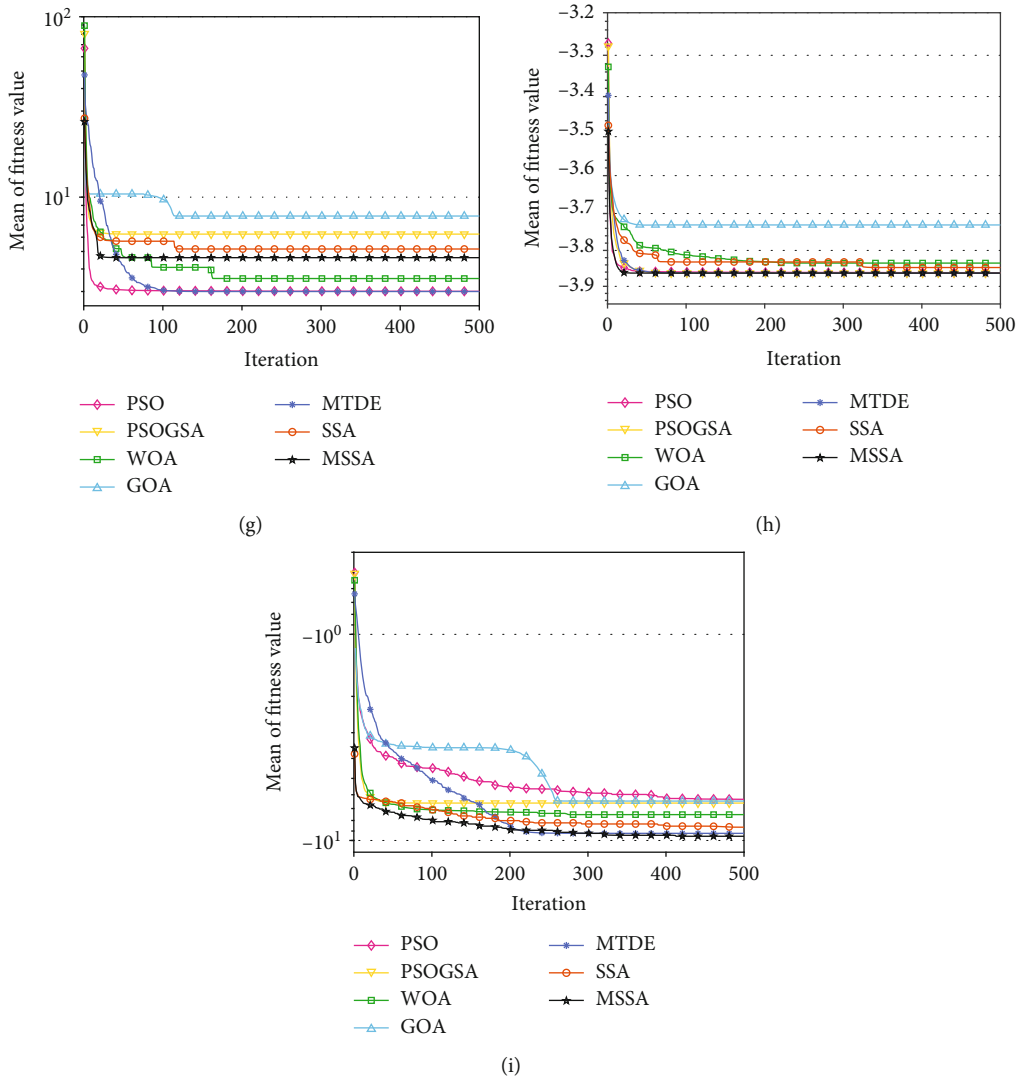


FIGURE 3: Convergence curves of the 7 algorithms on the benchmark test functions: (a) F1; (b) F2; (c) F3; (d) F4; (e) F5; (f) F6; (g) F7; (h) F8; (i) F9.

analyzed. Finally, the optimization performance is verified by FEKO EM simulation under practical conditions.

4.1. Problem Formulation. Considering the symmetry of $2N$ isotropic element LAA in Figure 6, the array factor can be expressed as follows:

$$\mathrm{AF}(\varphi) = 2\sum_{n=1}^{N} I_n \cos\left(kx_n \cos\left(\varphi\right) + \psi_n\right), \tag{10}$$

where $k = 2\pi/\lambda$ is the wave number and I_n, x_n, and ψ_n, respectively, represent the excitation amplitude, position, and phase of the nth element. φ symbolizes the azimuth angle, which is defined as the angle with the positive x-axis.

In this paper, the element spacing and excitation amplitude are taken as optimization variables, and the optimization objective is to suppress the maximum SLL. Therefore, the fitness function can be formulated as

$$\min \ \text{fitness} = \max\left\{20\log_{10}\frac{|\mathrm{AF}(\varphi_{\mathrm{SL}})|}{|\mathrm{AF}(\varphi_{\mathrm{ML}})|}\right\}, \tag{11}$$

$$\text{s.t.} \quad \varphi_{\mathrm{ML}} = \arg\max|\mathrm{AF}(\varphi)|, \quad \varphi \in [0,\pi], \tag{12}$$

$$\varphi_{\mathrm{SL}} \in [0,\varphi_{\mathrm{FN1}}] \cup [\varphi_{\mathrm{FN2}},\pi], \tag{13}$$

$$0 < d_i < \lambda, \quad \forall i \in N, \tag{14}$$

$$0 < I_i < 1, \quad \forall i \in N, \tag{15}$$

where φ_{SL} and φ_{ML} represent the region of the side lobe and main lobe, respectively. φ_{FN1} and φ_{FN2} are the first nulls of the pattern. Constraints (14) and (15) define the range in optimizing the element spacing and excitation amplitude. Based on the HF vertical antenna array, the number of elements is chosen as 8, 16, and 24 to test the optimization ability of the algorithm in different dimensions. In order to maintain

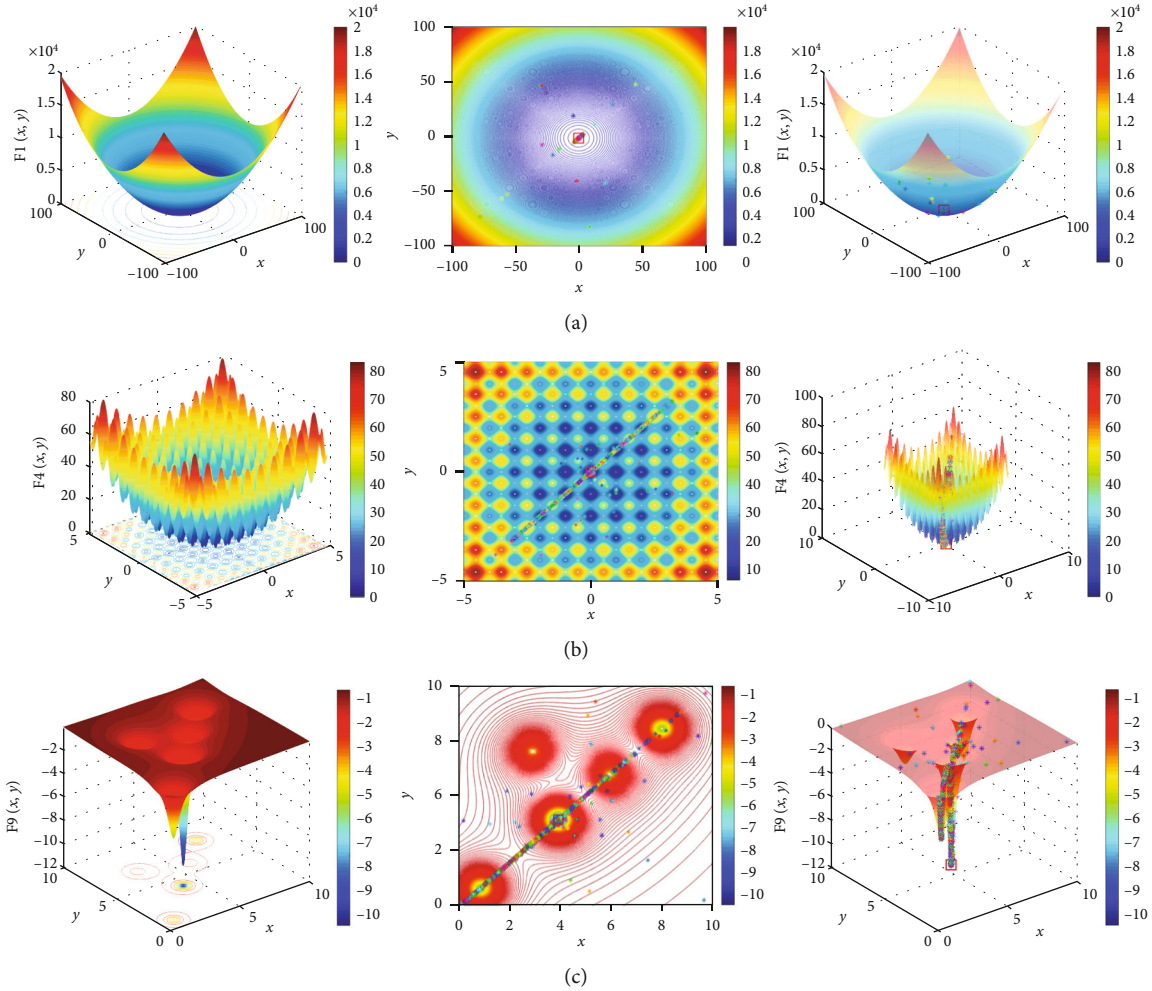


FIGURE 4: Iterative trajectories of MSSA on the benchmark test functions: (a) unimodal F1; (b) multimodal F4; (c) fixed-dimension multimodal F9.

the first null beam width to the maximum extent, for LAA with $2N$ of 8, 16, and 24, φ_{FN1} is limited to 75°, 82°, and 85° and φ_{FN2} is 105°, 98°, and 95°, respectively.

4.2. Experiments on Maximum SLL Reduction with Optimized Element Spacing. In this section, seven algorithms are used to optimize the positions of 8-element, 16-element, and 24-element LAA, respectively. Because the median is not affected by small or large data, it can better reflect the average optimization ability of the algorithms. Therefore, the median value of 25 independent repeated experiments is presented.

4.2.1. 8-Element LAA. Since this is a symmetric array, the spatial dimension of solution is 4. Figure 7 shows the 3D radiation patterns of 8-element LAA before and after MSSA optimization. It can be seen intuitively from the figure that MSSA can effectively reduce the maximum SLL.

Figure 8(a) shows the beam patterns obtained by different algorithms, and Figure 8(b) shows the convergence curves of different algorithms, in which conventional LAA refers to the unoptimized LAA while maintaining uniform element position distribution and uniform excitation amplitude. It can be seen from the figure that the convergence speed and convergence accuracy of MSSA have been improved. The maximum SLL optimized by different algorithms for 8-element LAA is shown in Table 5. As shown, the maximum SLL obtained by MSSA is -20.2186 dB, which is the lowermost among all algorithms. There is an improvement of 7.4214 dB using the proposed approach as compared to the conventional technique. Table 5 also gives the optimized positions of the elements obtained by different algorithms.

4.2.2. 16-Element LAA. In order to maintain the main lobe shape and beam width, it is necessary to impose additional constraint on the total length of the antenna array, as shown in the following formula:

$$\begin{cases} x_1 = 0.25\lambda, \\ x_N = \dfrac{(2N-1)d}{2}. \end{cases} \tag{16}$$

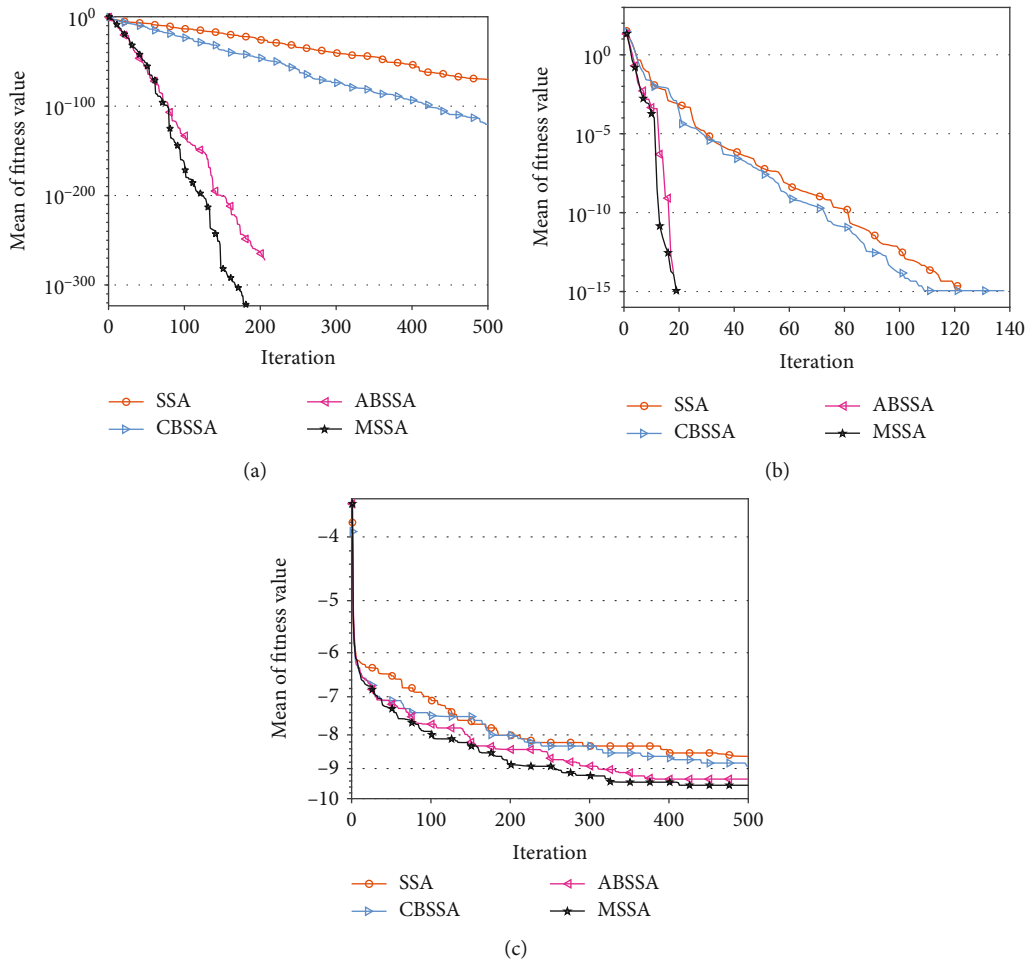


Figure 5: Convergence curves of the improved factors on the benchmark test functions: (a) unimodal F1; (b) multimodal F4; (c) fixed-dimension multimodal F9.

Table 4: p value of the nonparametric test between MSSA and other algorithms.

	PSO	PSOGSA	WOA	GOA	MTDE	SSA
p value	$1.57e-06$	$2.20e-06$	$1.90e-03$	$4.54e-06$	$9.32e-05$	$2.38e-01$

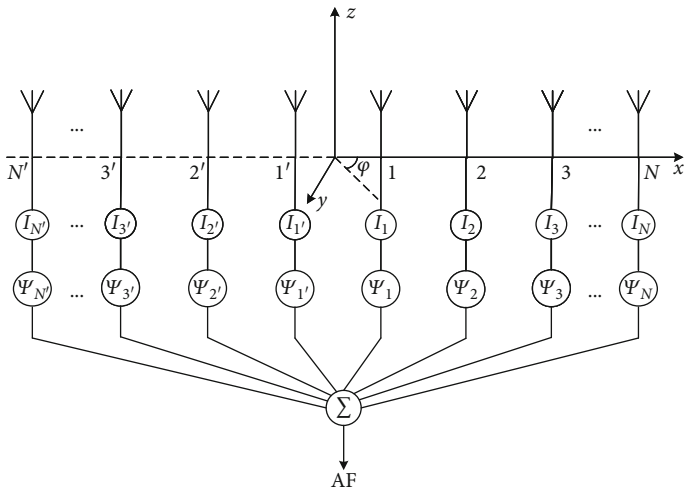


Figure 6: Geometric distribution of $2N$-element LAA.

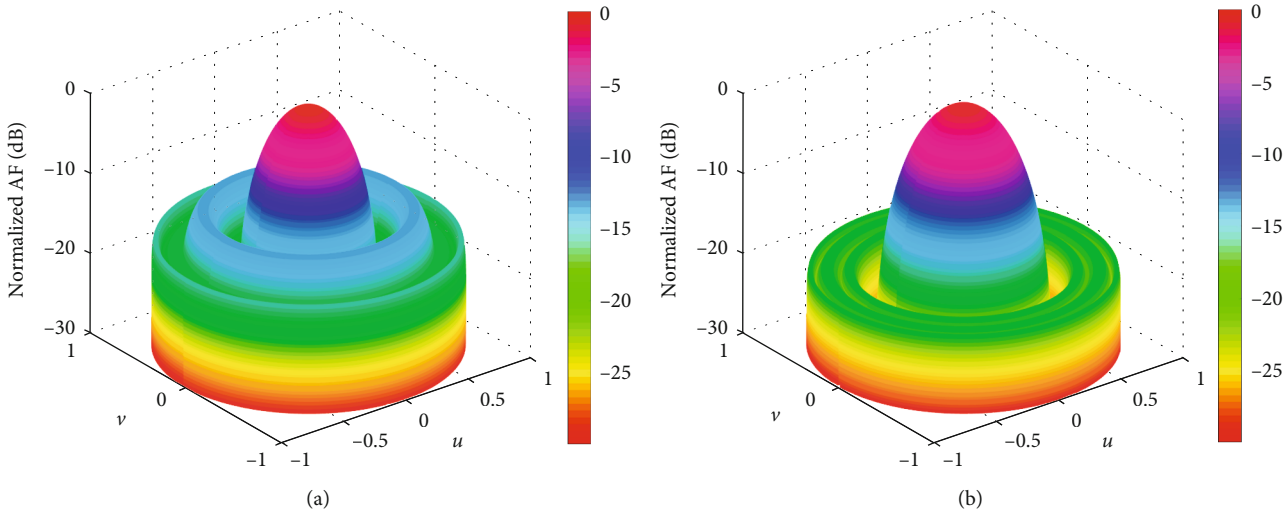


Figure 7: 3D radiation patterns of 8-element LAA before and after MSSA optimization: (a) before optimization; (b) after optimization.

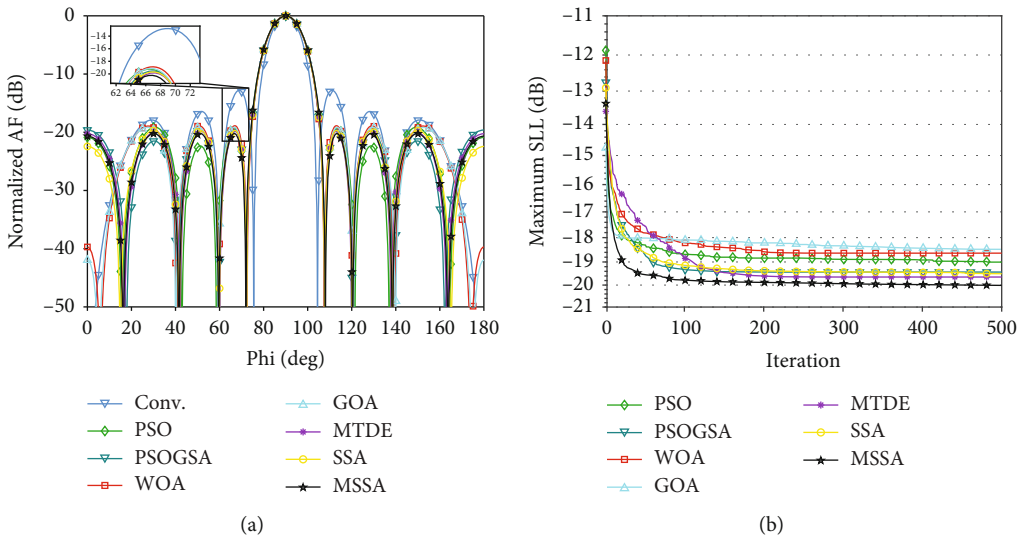


Figure 8: Beam patterns and convergence curves of 8-element LAA optimized by different algorithms: (a) beam patterns; (b) convergence curves.

Table 5: Optimized element positions and maximum SLL of 8-element LAA by different algorithms.

Algorithm	Optimized element positions (λ)	Maximum SLL (dB)
Conv.	0.2500, 0.7500, 1.2500, 1.7500	-12.7972
PSO	0.1243, 0.6381, 0.9773, 1.6283	-19.1209
PSOGSA	0.1120, 0.6269, 1.0036, 1.6384	-19.5447
WOA	0.1625, 0.6044, 1.0286, 1.6326	-18.8768
GOA	0.1529, 0.6016, 1.0115, 1.6185	-19.1405
MTDE	0.1274, 0.6228, 1.0082, 1.6497	-19.8331
SSA	0.1340, 0.6179, 1.0087, 1.6427	-19.6800
MSSA	0.1067, 0.6216, 0.9832, 1.6298	-20.2186

The spacing between the first two elements on both sides of the y-axis is fixed at 0.5λ, so as to ensure that one antenna pair in the antenna array meets Nyquist spatial sampling. The Nth element is fixed at $x_n = (2N-1)d/2$, where $d = 0.5\lambda$ represents the default spacing of the uniform LAA, thereby suppressing the distortion of the main lobe [44]. Since the positions of the first and last elements on both sides of the antenna array are fixed, the positions of other elements are variable. Therefore, the optimized dimension is simplified to $2N-2$.

Figure 9 depicts 3D radiation patterns of 16-element LAA before and after optimization by MSSA. Figure 10 depicts the beam patterns and convergence curves of 16-element LAA optimized by different algorithms. It can be seen that MSSA outperforms other algorithms. Both the positions of 16-element LAA optimized by different algorithms and corresponding maximum SLL are listed in

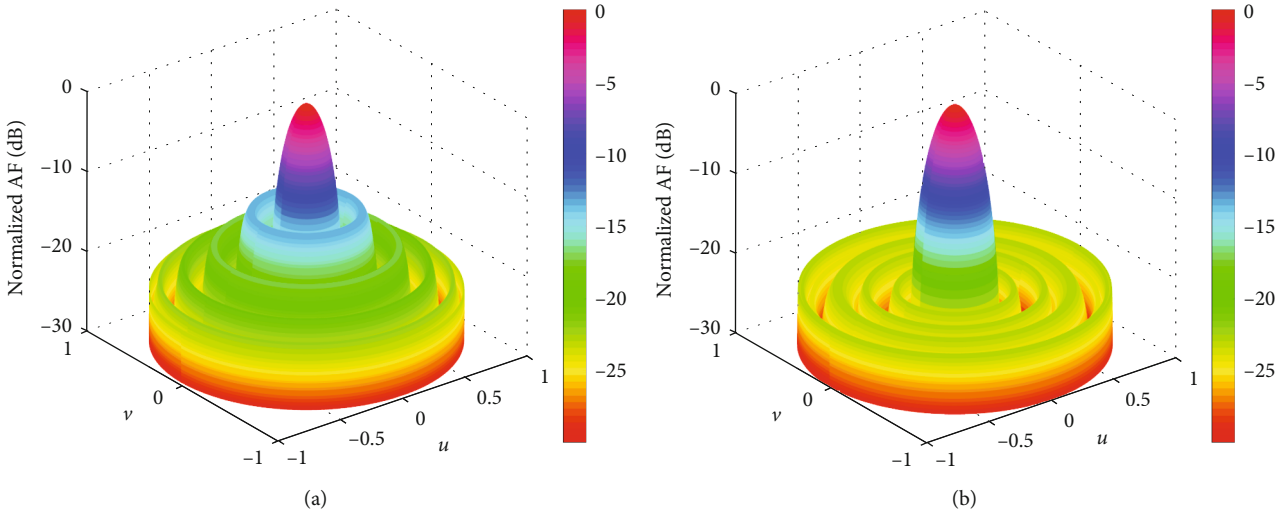


Figure 9: 3D radiation patterns of 16-element LAA before and after MSSA optimization: (a) before optimization; (b) after optimization.

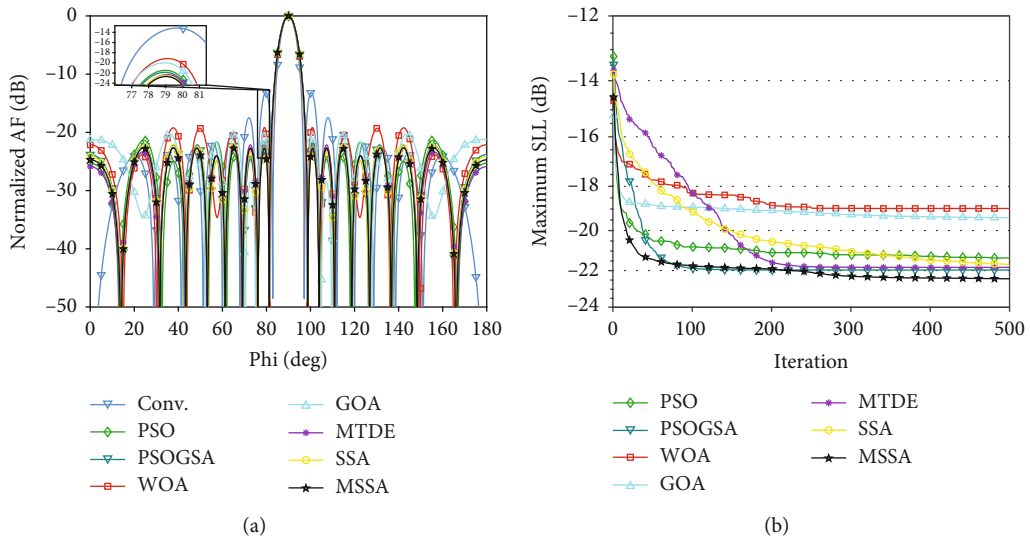


Figure 10: Beam patterns and convergence curves of 16-element LAA optimized by different algorithms: (a) beam pattern; (b) convergence curves.

Table 6. The maximum SLL obtained by MSSA is 9.5292 dB lower than that of the conventional array.

4.2.3. 24-Element LAA. The position constraints of the 24-element LAA are the same as above; that is, the first element is fixed at 0.25λ and the 12th element is fixed at 5.75λ. Therefore, the spatial dimension of solution is simplified to 10.

The 3D radiation patterns of 24-element LAA before and after MSSA optimization are given in Figure 11. Figure 12 shows the beam patterns and convergence curves obtained by different algorithms. Table 7 displays the optimized results and the maximum SLL of 24-element LAA with different algorithms simultaneously. Compared with the conventional method, PSO, PSOGSA, WOA, GOA, MTDE, and SSA, MSSA has better optimization effect, and the maximum SLL is reduced by 10.8158 dB, 3.1391 dB, 1.2248 dB, 6.2655 dB, 5.0487 dB, 0.7132 dB, and 0.8840 dB, respectively.

4.3. Experiments on Maximum SLL Reduction with Optimized Excitation Amplitude. This section conducts and analyzes the optimization performance of MSSA on LAA excitation amplitude. Corresponding to the previous section, different algorithms are used to optimize the 8-element, 16-element, and 24-element LAA.

4.3.1. 8-Element LAA. It can be seen from the 3D radiation patterns of the antenna array in Figure 13 that the maximum SLL of the optimized LAA is significantly reduced. Figure 14 shows the beam patterns and convergence curves of different algorithms. The element excitation amplitude is normalized, and the obtained amplitude is the largest at the center of

Table 6: Optimized element positions and maximum SLL of 16-element LAA by different algorithms.

Algorithm	Optimized element positions (λ)	Maximum SLL (dB)
Conv.	0.2500, 0.7500, 1.2500, 1.7500, 2.2500, 2.7500, 3.2500, 3.7500	-13.1476
PSO	0.2500, 0.5311, 1.0128, 1.3930, 1.8738, 2.3329, 2.9893, 3.7500	-21.3693
PSOGSA	0.2500, 0.5495, 1.0230, 1.3560, 1.8561, 2.3358, 2.9783, 3.7500	-21.8484
WOA	0.2500, 0.6485, 1.0456, 1.3751, 1.9467, 2.4634, 3.0076, 3.7500	-19.1546
GOA	0.2500, 0.5802, 1.1274, 1.3493, 1.9119, 2.3129, 3.0208, 3.7500	-19.9808
MTDE	0.2500, 0.5244, 1.0082, 1.3555, 1.8647, 2.3620, 3.0093, 3.7500	-22.1498
SSA	0.2500, 0.5331, 1.0118, 1.3453, 1.8495, 2.3404, 2.9835, 3.7500	-22.0177
MSSA	0.2500, 0.5226, 1.0038, 1.3486, 1.8518, 2.3447, 2.9948, 3.7500	-22.6768

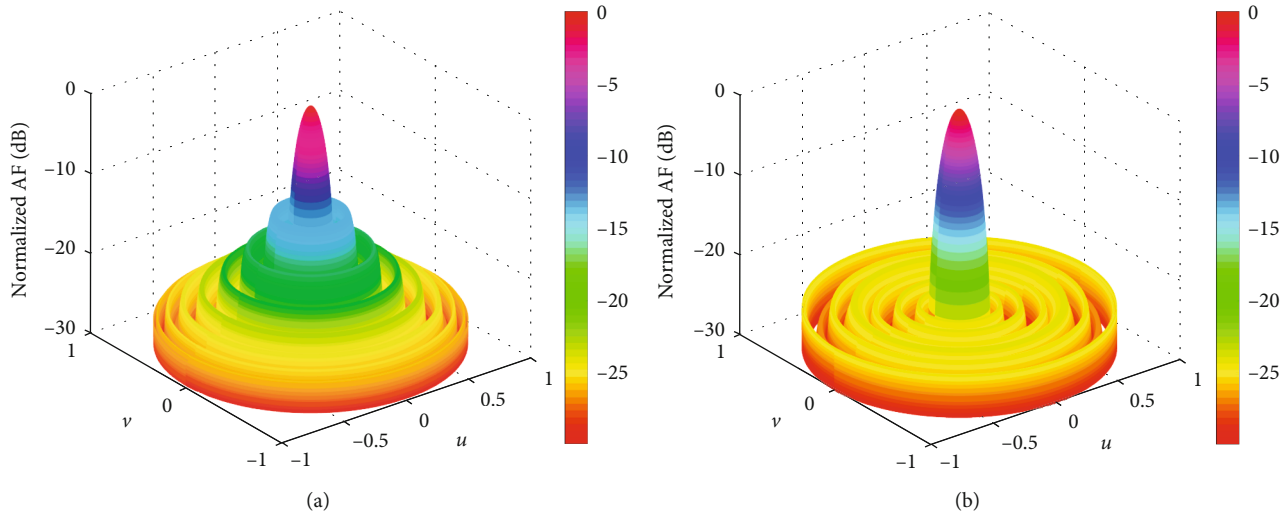


Figure 11: 3D radiation patterns of 24-element LAA before and after MSSA optimization: (a) before optimization; (b) after optimization.

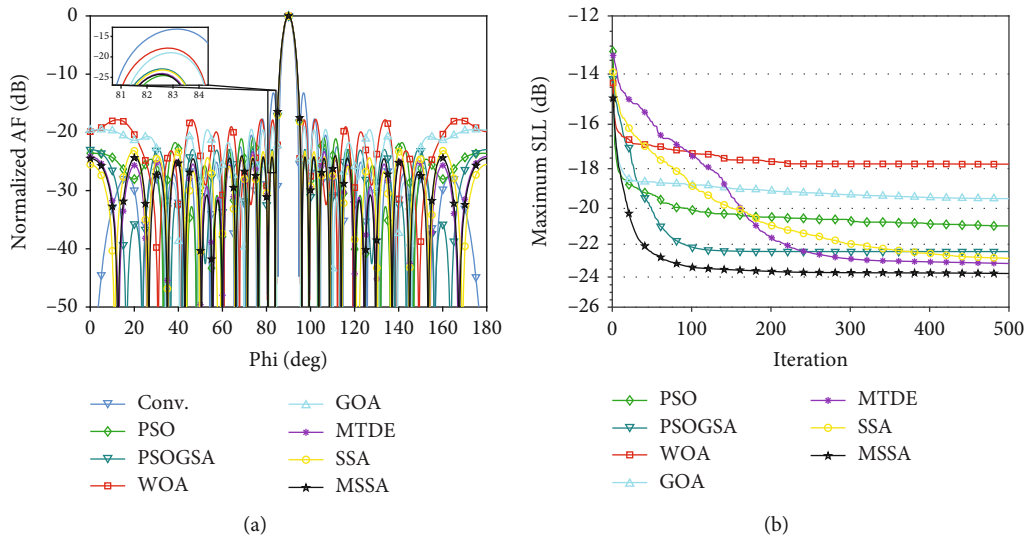


Figure 12: Beam patterns and convergence curves of 24-element LAA optimized by different algorithms: (a) beam patterns; (b) convergence curves.

Table 7: Optimized element positions and maximum SLL of 24-element LAA by different algorithms.

Algorithm	Optimized element positions (λ)	Maximum SLL (dB)
Conv.	0.2500, 0.7500, 1.2500, 1.7500, 2.2500, 2.7500, 3.2500, 3.7500, 4.2500, 4.7500, 5.2500, 5.7500	-13.2091
PSO	0.2500, 0.5328, 0.9972, 1.2560, 1.7894, 2.1072, 2.5783, 3.0281, 3.5374, 4.2919, 5.0668, 5.7500	-20.8858
PSOGSA	0.2500, 0.4890, 1.0882, 1.2757, 1.7539, 2.0904, 2.6058, 3.1024, 3.5855, 4.2269, 4.9847, 5.7500	-22.8001
WOA	0.2500, 0.4275, 1.0262, 1.3683, 1.7913, 2.4037, 3.1937, 3.2078, 3.9114, 4.2607, 5.0305, 5.7500	-17.7594
GOA	0.2500, 0.6940, 1.1051, 1.2948, 1.8599, 2.3254, 2.8115, 3.2905, 3.9984, 4.4875, 5.1074, 5.7500	-18.9762
MTDE	0.2500, 0.5035, 0.9885, 1.2737, 1.7304, 2.1064, 2.5789, 3.0315, 3.5594, 4.2334, 4.9549, 5.7500	-23.3117
SSA	0.2500, 0.5314, 0.9701, 1.2658, 1.7411, 2.1524, 2.6184, 3.0450, 3.5985, 4.2726, 4.9465, 5.7500	-23.1409
MSSA	0.2500, 0.4950, 0.9690, 1.2760, 1.7106, 2.1075, 2.5880, 3.0307, 3.5357, 4.2088, 4.9494, 5.7500	-24.0249

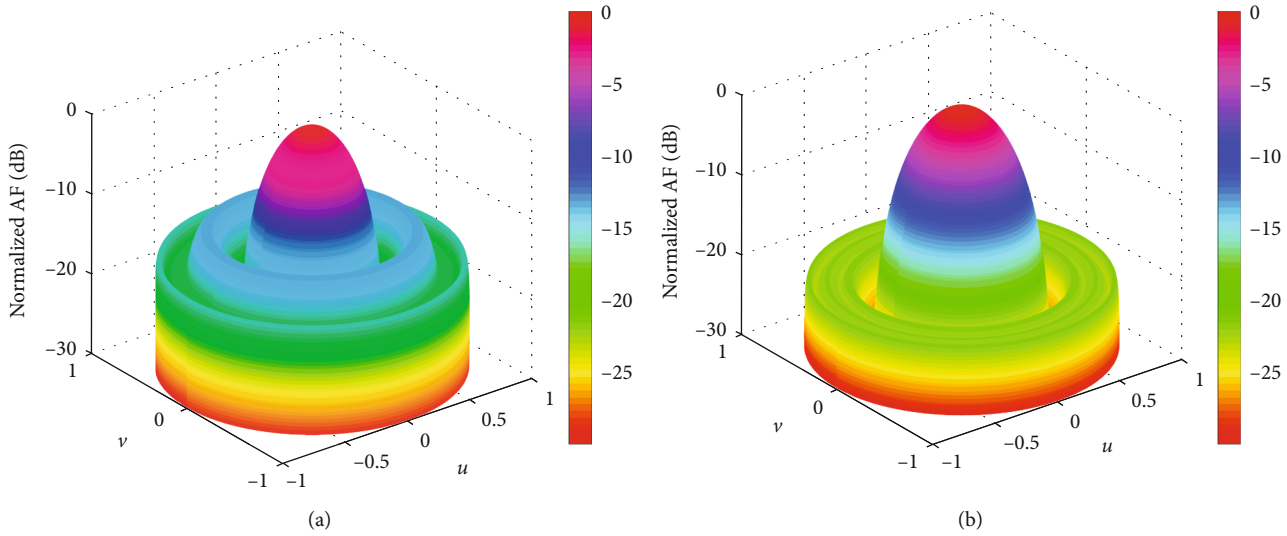


Figure 13: 3D radiation patterns of 8-element LAA before and after MSSA optimization: (a) before optimization; (b) after optimization.

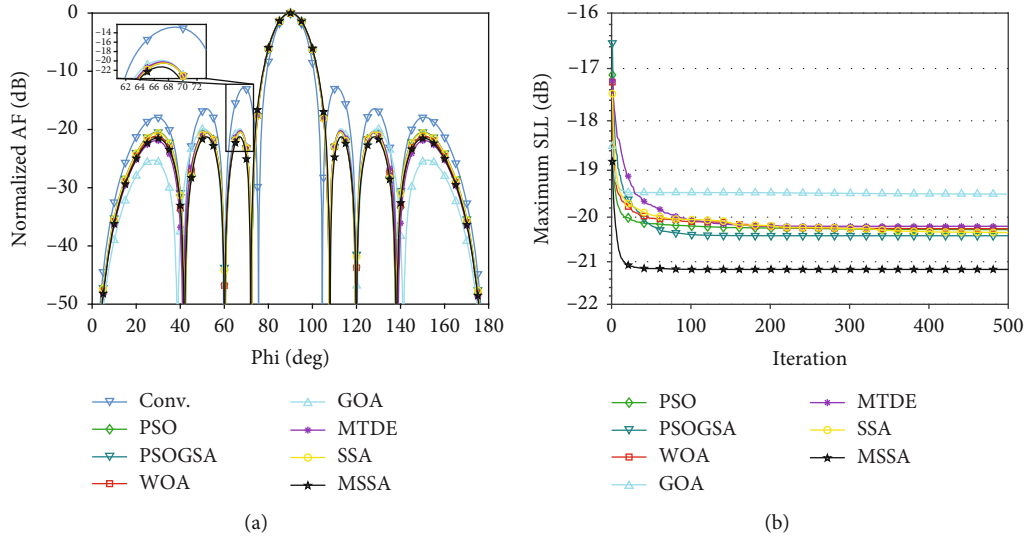


Figure 14: Beam patterns and convergence curves of 8-element LAA optimized by different algorithms: (a) beam patterns; (b) convergence curves.

Table 8: Optimized excitation amplitudes and maximum SLL of 8-element LAA by different algorithms.

Algorithm	Optimized excitation amplitudes (normalized)	Maximum SLL (dB)
Conv.	1.0000, 1.0000, 1.0000, 1.0000	-12.7972
PSO	1.0000, 0.8602, 0.6634, 0.5467	-20.2869
PSOGSA	1.0000, 0.8720, 0.6528, 0.5560	-20.4517
WOA	1.0000, 0.8618, 0.6618, 0.5462	-20.3507
GOA	1.0000, 0.8193, 0.7051, 0.4993	-19.8836
MTDE	1.0000, 0.8584, 0.6735, 0.5349	-20.1902
SSA	1.0000, 0.8702, 0.6544, 0.5542	-20.4375
MSSA	1.0000, 0.8636, 0.6424, 0.5128	-21.2569

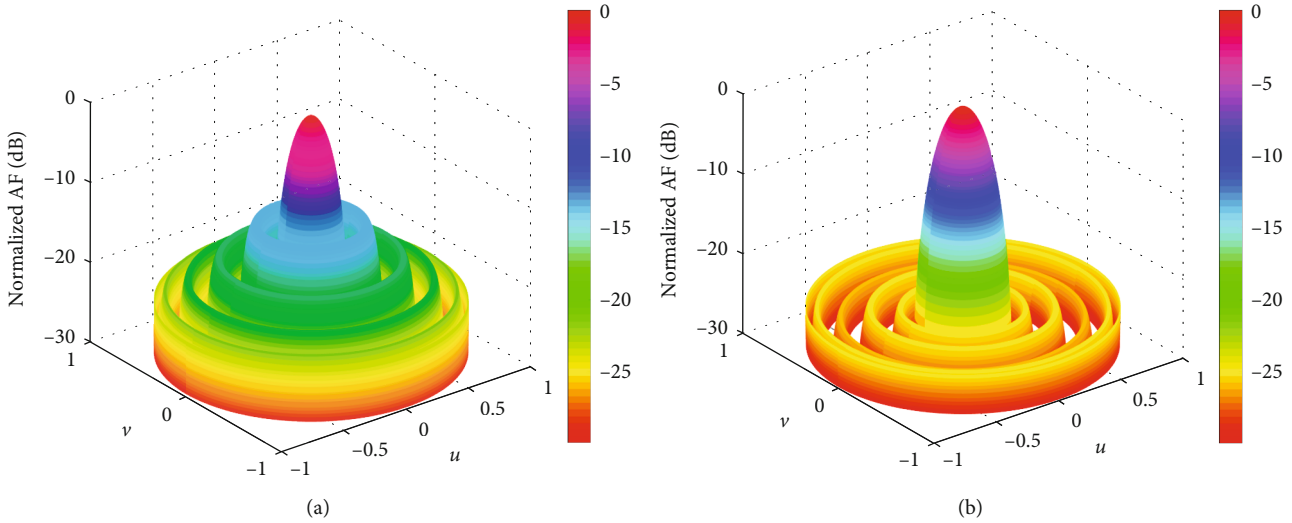


Figure 15: 3D radiation patterns of 16-element LAA before and after MSSA optimization: (a) before optimization; (b) after optimization.

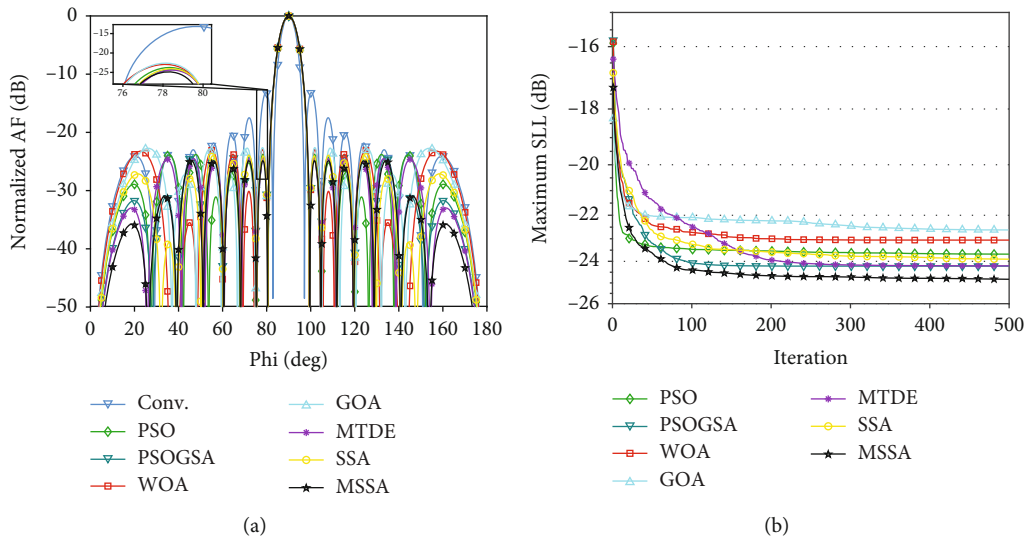


Figure 16: Beam patterns and convergence curves of 16-element LAA optimized by different algorithms: (a) beam patterns; (b) convergence curves.

Table 9: Optimized excitation amplitudes and maximum SLL of 16-element LAA by different algorithms.

Algorithm	Optimized excitation amplitudes (normalized)	Maximum SLL (dB)
Conv.	1.0000, 1.0000, 1.0000, 1.0000, 1.0000, 1.0000, 1.0000, 1.0000	-13.1476
PSO	1.0000, 1.0000, 0.9285, 0.7355, 0.7425, 0.5670, 0.4059, 0.4509	-23.7487
PSOGSA	1.0000, 0.9999, 0.8741, 0.8064, 0.6889, 0.4930, 0.5033, 0.4395	-24.2080
WOA	1.0000, 0.9809, 0.9023, 0.9761, 0.5847, 0.5656, 0.4604, 0.4002	-22.9487
GOA	1.0000, 0.8419, 0.8619, 0.7136, 0.7013, 0.4207, 0.5957, 0.2324	-22.6180
MTDE	1.0000, 0.9478, 0.9071, 0.7483, 0.7115, 0.4904, 0.4592, 0.4567	-24.4013
SSA	1.0000, 0.9652, 0.8346, 0.8395, 0.6237, 0.5275, 0.4877, 0.4115	-23.9233
MSSA	1.0000, 0.9711, 0.8581, 0.7839, 0.6875, 0.4687, 0.4784, 0.3895	-24.8687

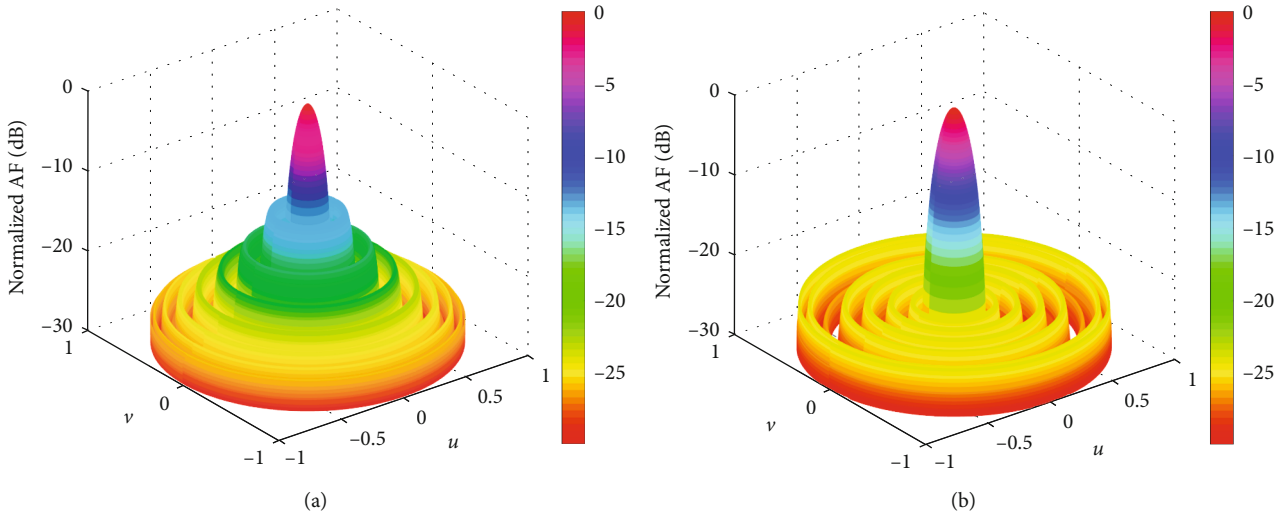


Figure 17: 3D radiation patterns of 24-element LAA before and after MSSA optimization: (a) before optimization; (b) after optimization.

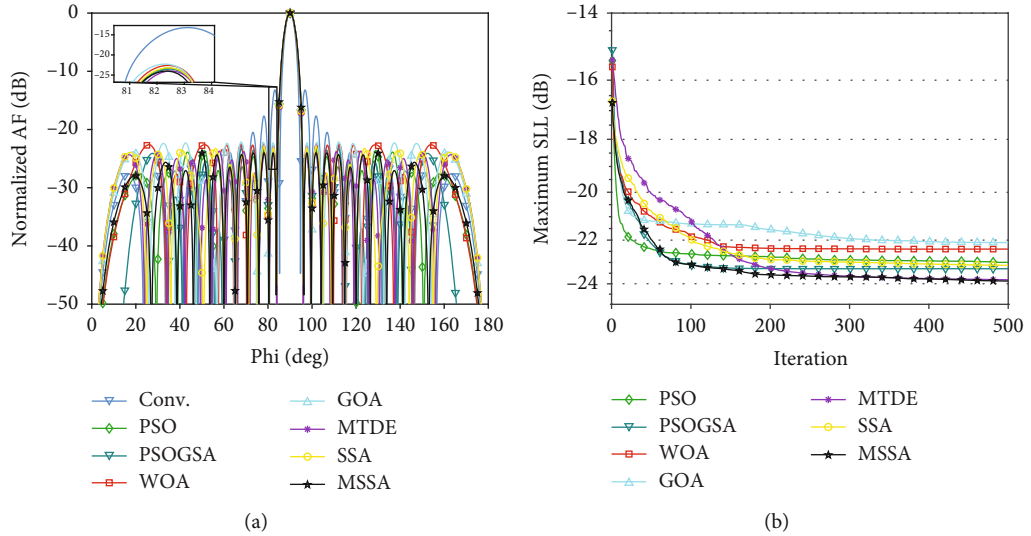


Figure 18: Beam patterns and convergence curves of 24-element LAA optimized by different algorithms: (a) beam patterns; (b) convergence curves.

Table 10: Optimized excitation amplitudes and maximum SLL of 24-element LAA by different algorithms.

Algorithm	Optimized excitation amplitudes (normalized)	Maximum SLL (dB)
Conv.	1.0000, 1.0000, 1.0000, 1.0000, 1.0000, 1.0000, 1.0000, 1.0000, 1.0000, 1.0000, 1.0000, 1.0000	-13.2091
PSO	1.0000, 1.0000, 1.0000, 0.8662, 0.8594, 0.7821, 0.6890, 0.5893, 0.6749, 0.3137, 0.6220, 0.4351	-23.1375
PSOGSA	1.0000, 0.9935, 0.9955, 0.7775, 0.9685, 0.7646, 0.7109, 0.5132, 0.5868, 0.4886, 0.5056, 0.4786	-23.3338
WOA	1.0000, 0.7820, 0.9905, 0.8429, 0.7500, 0.6529, 0.8972, 0.3259, 0.4892, 0.6179, 0.5064, 0.3145	-22.4755
GOA	1.0000, 1.0000, 0.9309, 0.7734, 0.7215, 0.9042, 0.7314, 0.4799, 0.6497, 0.3279, 0.6973, 0.2235	-22.1484
MTDE	1.0000, 0.8970, 0.8728, 0.8554, 0.7775, 0.7191, 0.7041, 0.6090, 0.4789, 0.4453, 0.3361, 0.6153	-23.7130
SSA	1.0000, 0.9598, 0.8152, 0.8919, 0.8345, 0.6096, 0.7762, 0.6822, 0.3127, 0.6130, 0.4044, 0.4461	-22.9933
MSSA	1.0000, 0.9979, 0.9066, 0.9165, 0.8683, 0.7488, 0.6761, 0.5852, 0.5815, 0.3026, 0.6685, 0.3467	-24.0121

Table 11: Statistical results of optimizing element position of 8-element LAA by different algorithms.

Maximum SLL (dB)	PSO	PSOGSA	WOA	GOA	MTDE	SSA	MSSA
Best	-19.7822	-19.5874	-19.6367	-19.8891	-19.8807	-19.8872	-20.2818
Worst	-18.1097	-16.6997	-16.4415	-15.4566	-17.2225	-18.1751	-18.4730
Mean	-19.0057	-19.4304	-18.6314	-18.4730	-19.6761	-19.5019	-20.0155
Std	0.3162	0.5932	0.8600	1.3805	0.5222	0.5069	0.4640

Table 12: Statistical results of optimizing element position of 16-element LAA by different algorithms.

Maximum SLL (dB)	PSO	PSOGSA	WOA	GOA	MTDE	SSA	MSSA
Best	-22.4235	-22.1479	-21.0826	-20.9566	-22.6257	-22.7315	-22.9611
Worst	-20.5435	-18.1108	-14.7938	-16.4117	-20.0713	-19.5287	-20.9210
Mean	-21.3471	-21.9683	-18.9752	-19.3923	-21.9622	-21.7445	-22.4318
Std	0.4310	0.7876	1.4892	1.5365	0.6582	0.8296	0.5563

Table 13: Statistical results of optimizing element position of 24-element LAA by different algorithms.

Maximum SLL (dB)	PSO	PSOGSA	WOA	GOA	MTDE	SSA	MSSA
Best	-22.0353	-22.9531	-19.8425	-22.9147	-24.1282	-24.1570	-24.2459
Worst	-19.6358	-20.4674	-15.9886	-16.2440	-21.3044	-21.1238	-22.1925
Mean	-20.9536	-22.4317	-17.7858	-19.4895	-23.1922	-22.8371	-23.7757
Std	0.5281	0.7872	1.0889	1.5849	0.8052	0.8865	0.6002

Table 14: Statistical results of optimizing excitation amplitude of 8-element LAA by different algorithms.

Maximum SLL (dB)	PSO	PSOGSA	WOA	GOA	MTDE	SSA	MSSA
Best	-20.4030	-20.4472	-20.4466	-20.4460	-20.4313	-20.4472	-21.2873
Worst	-19.5514	-19.5644	-19.6004	-17.5333	-19.9362	-19.6465	-20.5552
Mean	-20.2750	-20.4107	-20.2540	-19.5017	-20.2008	-20.3396	-21.1768
Std	0.1555	0.1729	0.2151	0.8734	0.0974	0.2032	0.1694

array and decreases successively to both sides. Such an amplitude distribution is feasible to the use of power dividers [14]. As can be observed in Table 8, the maximum SLL of conventional LAA is -12.7972 dB, and PSO, PSOGSA, WOA, GOA, MTDE, SSA, and MSSA are -20.2869 dB, -20.4517 dB, -20.3507 dB, -19.8836 dB, -20.1902 dB, -20.4375 dB, and -21.2569 dB, respectively. MSSA achieves the lowest SLL among these seven algorithms. The corresponding excitation amplitudes are also shown in Table 8.

4.3.2. 16-Element LAA. Figure 15 shows the 3D radiation patterns of 16-element LAA before and after MSSA

Table 15: Statistical results of optimizing excitation amplitude of 16-element LAA by different algorithms.

Maximum SLL (dB)	PSO	PSOGSA	WOA	GOA	MTDE	SSA	MSSA
Best	-24.0247	-24.5171	-23.8726	-24.3514	-24.4831	-24.4016	-25.4034
Worst	-23.2618	-23.8922	-21.9349	-21.0661	-23.9823	-22.8781	-24.1036
Mean	-23.6772	-24.2185	-23.0579	-22.6194	-24.2072	-23.9039	-24.8280
Std	0.2276	0.2007	0.5489	0.9146	0.1401	0.4476	0.3704

Table 16: Statistical results of optimizing excitation amplitude of 24-element LAA by different algorithms.

Maximum SLL (dB)	PSO	PSOGSA	WOA	GOA	MTDE	SSA	MSSA
Best	-23.3469	-23.8890	-23.4673	-23.7657	-24.0280	-23.9715	-24.5309
Worst	-22.3166	-21.8494	-20.8848	-20.2394	-23.4616	-22.0058	-22.8864
Mean	-22.9946	-23.2909	-22.3975	-22.1077	-23.8073	-23.1328	-23.8688
Std	0.2637	0.4853	0.6693	0.8942	0.1337	0.4787	0.4439

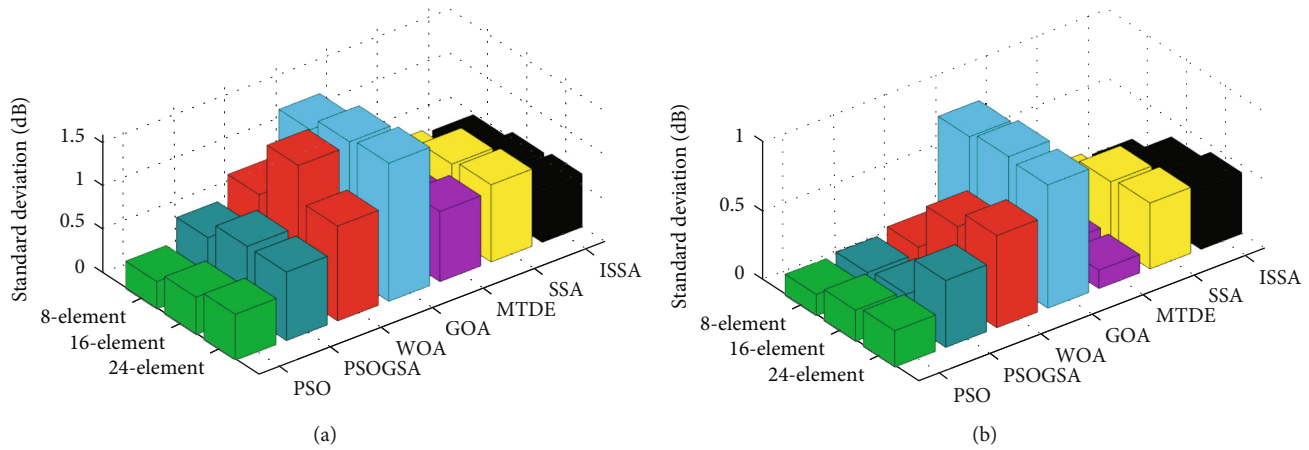


Figure 19: Standard deviation obtained by different algorithms: (a) element position optimization; (b) excitation amplitude optimization.

optimization. The beam patterns and convergence curves obtained by different algorithms are shown in Figure 16. It can be seen from Table 9 that the maximum SLL of PSO, PSOGSA, WOA, GOA, MTDE, SSA, and MSSA are reduced by 10.6011 dB, 11.0604 dB, 9.8011 dB, 9.4704 dB, 11.2537 dB, 10.7757 dB, and 11.7211 dB, respectively. Compared with conventional LAA, MSSA has the largest reduction. In addition, Table 9 also gives the optimized excitation amplitudes obtained by different algorithms.

4.3.3. 24-Element LAA. The 3D radiation patterns of 24-element LAA before and after MSSA optimization are depicted in Figure 17. Figure 18 is the beam patterns and convergence curves of 24-element LAA optimized by different algorithms. Table 10 shows the optimized element excitation amplitudes and maximum SLL. From the experimental results, the results of MSSA optimization are obviously better than those of the other six algorithms. This example further verifies the advantages of the modified algorithm in convergence speed as well as convergence accuracy.

4.4. Stability Test. Due to the randomness of the intelligent optimization algorithm, the results of each run are likely to be different, so it is necessary to discuss and analyze the stability of the algorithms. Similar to the performance analysis of the modified algorithm, Tables 11–16, respectively, give the best value, the worst value, the mean value, and the standard deviation of the maximum SLL obtained by different algorithms in optimizing the element position and excitation amplitude of the 8-element, 16-element, and 24-element LAA.

In order to visually display the fluctuation of data, the standard deviation diagrams of the element position and excitation amplitude optimized by different algorithms are drawn, as shown in Figure 19. According to the results of each run, the maximum SLL obtained by different algorithms are depicted in Figure 20. It is clear that, compared with the other six algorithms, the mean value of MSSA is the lowest, which indicates that the proposed algorithm has higher solution accuracy. The best value and the worst value are smaller than those of PSO, PSOGSA, WOA, GOA, MTDE, and SSA, which shows that the algorithm has strong optimization

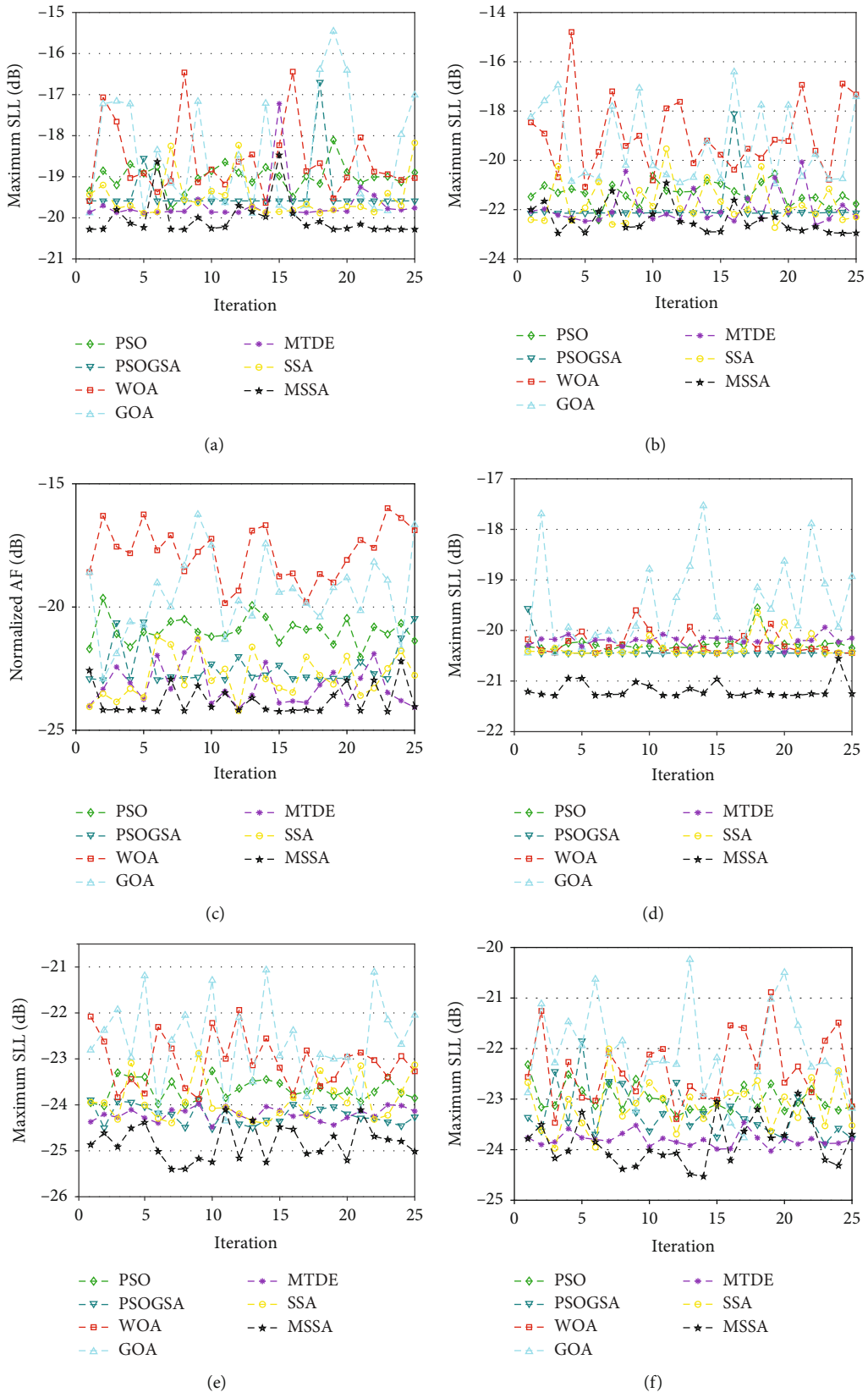


Figure 20: Result curves obtained by different algorithms: (a) 8-element position optimization; (b) 16-element position optimization; (c) 24-element position optimization; (d) 8-element excitation amplitude optimization; (e) 16-element excitation amplitude optimization; (f) 24-element excitation amplitude optimization.

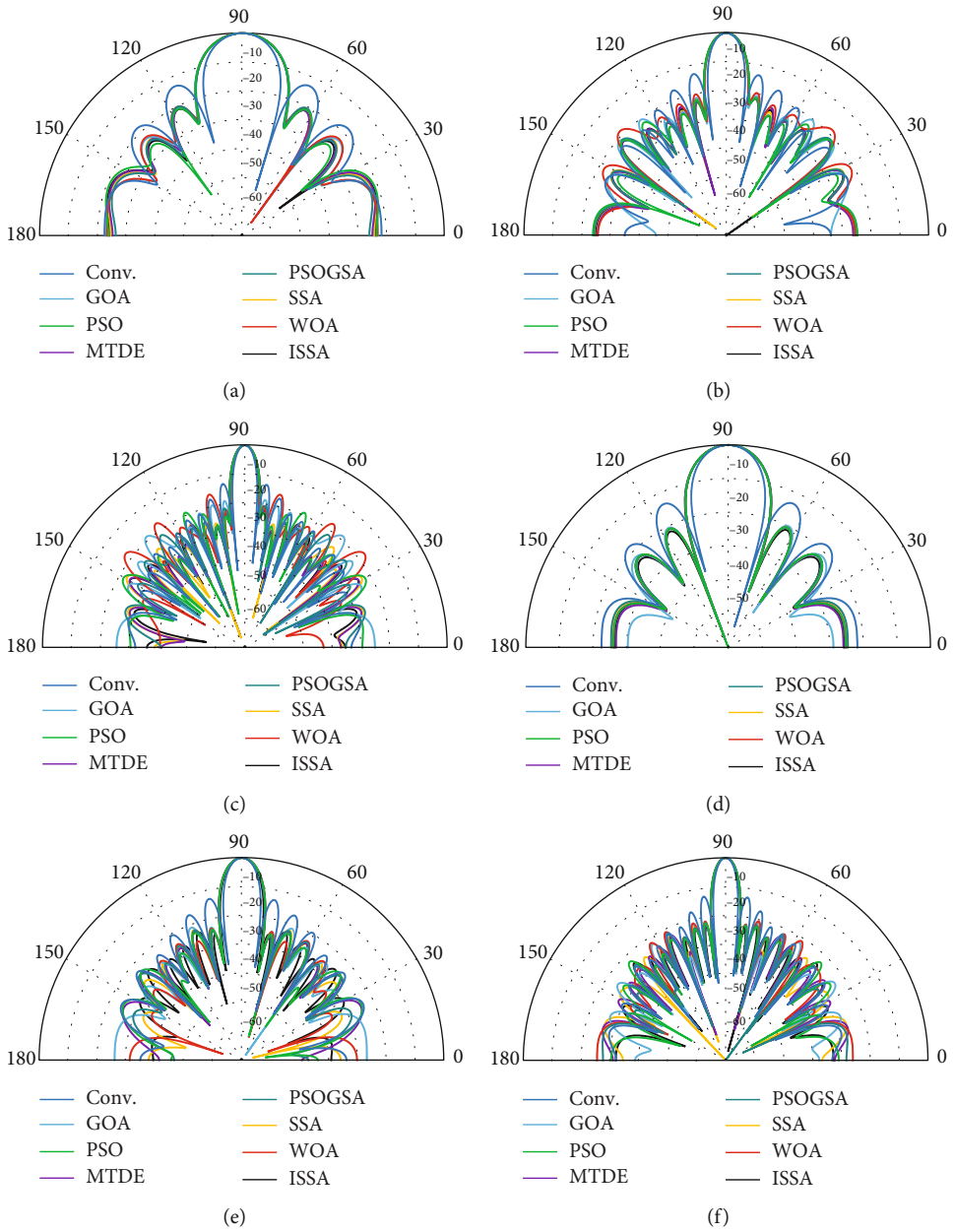


Figure 21: Polar coordinate patterns obtained by different algorithms: (a) 8-element position optimization; (b) 16-element position optimization; (c) 24-element position optimization; (d) 8-element excitation amplitude optimization; (e) 16-element excitation amplitude optimization; (f) 24-element excitation amplitude optimization.

Table 17: Maximum SLL of EM simulation in different scenarios.

	Conv.	PSO	PSOGSA	WOA	GOA	MTDE	SSA	MSSA
(a)	-13.2094	-20.8365	-20.2227	-20.1765	-20.3820	-20.9978	-20.9050	-21.0660
(b)	-13.0542	-21.2888	-21.0354	-18.2653	-20.2464	-21.4397	-21.3786	-21.6052
(c)	-12.9059	-18.9502	-20.4974	-15.6828	-18.4361	-22.0438	-21.6763	-22.1540
(d)	-13.2094	-20.3256	-20.4676	-20.3610	-19.8126	-20.1549	-20.4489	-21.1083
(e)	-13.0542	-23.0645	-22.2065	-22.9711	-20.9268	-22.5175	-21.9086	-23.2999
(f)	-12.9059	-21.9991	-22.1827	-20.6191	-21.1633	-22.6291	-21.6314	-22.8699

ability. Although MSSA stability is slightly inferior to that of the classical PSO algorithm and MTDE in excitation amplitude optimization, it performs better than the remaining algorithms.

4.5. EM Simulation. In practical applications, the induced current between elements will inevitably lead to mutual coupling. Conventional simulation, as a common processing mode in the most existing works on design of LAA for SLL reduction, usually assumes that array is in an ideal environment, and the effects of mutual coupling between the antenna elements are often ignored. EM simulation is an effective method to verify and evaluate the performance of different optimization algorithms in the actual environment [45]. With the help of EM simulation software Altair FEKO 2019, the actual LAA model is constructed firstly, and then, the 2D beam patterns of different antenna arrays in polar coordinates are obtained by simulation according to the above optimized element positions and excitation amplitudes. The beam patterns are shown in Figure 21. The numerical results of maximum SLL are listed in Table 17. It can be seen that the results of maximum SLL reduction are basically consistent with the above simulation results, which demonstrates that the algorithm is also effective for the practical conditions.

To sum up, based on the original algorithm, MSSA can give full play to the advantages of chaotic adaptive inertia weight and improved boundary constraint and enhance the global optimization ability. However, more work can be done to further enhance the stability and robustness of this algorithm. The numerical results of this paper indicate that the modified algorithm shows promise as an effective method in solving the global optimization problems.

5. Conclusions

In this paper, it is the first time that SSA and its modification have been introduced and utilized to the field of EM optimization according to the existing published literature and reports. For one thing, in order to improve the shortcomings of SSA, such as being easy to fall into local optimum and slow convergence speed, a novel modified SSA combining a homogeneous chaotic system, adaptive inertia weight, and improved boundary constraint is proposed. The computational results of three types of benchmark test functions verify the effectiveness of the modified algorithm. For the other thing, with the element positions and excitation amplitude as optimization variables and the maximum SLL reduction as optimization objective, a comparison with the classical PSO and PSOGSA, WOA, GOA, and MTDE through several scenarios is conducted. Simulation results show that MSSA achieved high performance in terms of maximum SLL reduction with a greater improvement in convergence accuracy, convergence speed, and stability. Therefore, SSA, as a new beam pattern optimization method for electromagnetic community and antenna design, has a large research space. In our future work, we will continue to study the algorithm and explore its real-world applications in the field of antenna array design, such as the multiobjective antenna array design problem aimed at minimizing at least two conflicting objectives at the same time, placing deep nulls of the beam pattern in desired directions to avoid the effect of jamming and interference, and other geometric configurations of the antenna array that work in different occasions.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

References

[1] H. Pradhan, B. B. Mangaraj, and S. K. Behera, "Antenna array optimization for smart antenna technology using whale optimization algorithm," in *2019 IEEE Indian Conference on Antennas and Propogation (InCAP)*, pp. 1–4, Ahmedabad, India, December 2019.

[2] J. Wang, Y. Nong, and Z. He, *Antenna Array Theory and Engineering Applications*, Publishing House of Electronics Industry, Beijing, 2015.

[3] G. Ram, D. Mandal, S. P. Ghoshal, and R. Kar, "Nature-inspired algorithm-based optimization for beamforming of linear antenna array system," in *Nature-Inspired Computing and Optimization*, pp. 185–215, Springer, 2017.

[4] B. Li, C. Liu, H. Wu, Y. Zhao, and Y. Dong, "Chaotic adaptive butterfly mating optimization and its applications in synthesis and structure optimization of antenna arrays," *International Journal of Antennas and Propagation*, vol. 2019, Article ID 1730868, 14 pages, 2019.

[5] M. A. Panduro, A. L. Mendez, R. Dominguez, and G. Romero, "Design of non-uniform circular antenna arrays for side lobe reduction using the method of genetic algorithms," *AEU-International Journal of Electronics and Communications*, vol. 60, no. 10, pp. 713–717, 2006.

[6] M. M. Khodier and C. G. Christodoulou, "Linear array geometry synthesis with minimum sidelobe level and null control using particle swarm optimization," *IEEE Transactions on Antennas & Propagation*, vol. 53, no. 8, pp. 2674–2679, 2005.

[7] K. Guney and M. Onay, "Bees algorithm for interference suppression of linear antenna arrays by controlling the phase-only and both the amplitude and phase," *Expert Systems with Applications*, vol. 37, no. 4, pp. 3129–3135, 2010.

[8] U. Singh, H. Kumar, and T. S. Kamal, "Linear array synthesis using biogeography based optimization," *Progress In Electromagnetics Research*, vol. 11, pp. 25–36, 2010.

[9] M. A. Zaman and M. Abdul Matin, "Nonuniformly spaced linear antenna array design using firefly algorithm," *International Journal of Microwave Science and Technology*, vol. 2012, Article ID 256759, 8 pages, 2012.

[10] L. Pappula and D. Ghosh, "Linear antenna array synthesis using cat swarm optimization," *AEU-International Journal of Electronics and Communications*, vol. 68, no. 6, pp. 540–549, 2014.

[11] U. Singh and M. Rattan, "Design of linear and circular antenna arrays using cuckoo optimization algorithm," *Progress in Electromagnetics Research*, vol. 46, pp. 1–11, 2014.

[12] K. Guney and A. Durmus, "Pattern nulling of linear antenna arrays using backtracking search optimization algorithm," *International Journal of Antennas and Propagation*, vol. 2015, Article ID 713080, 10 pages, 2015.

[13] N. I. Dib, "Design of linear antenna arrays with low side lobes level using symbiotic organisms search," *Progress In Electromagnetics Research*, vol. 68, pp. 55–71, 2016.

[14] P. Saxena and A. Kothari, "Optimal pattern synthesis of linear antenna array using grey wolf optimization algorithm," *International Journal of Antennas and Propagation*, vol. 2016, Article ID 1205970, 11 pages, 2016.

[15] R. Salgotra, U. Singh, and S. Sharma, "On the improvement in grey wolf optimization," *Neural Computing and Applications*, vol. 32, no. 8, pp. 3709–3748, 2020.

[16] H. Wu, Y. Yan, C. Liu, and J. Zhang, "Pattern synthesis of sparse linear arrays using spider monkey optimization," *IEICE Transactions on Communications*, vol. E100.B, no. 3, pp. 426–432, 2017.

[17] M. Hesari and A. Ebrahimzadeh, "Introducing deeper nulls and reduction of side-lobe level in linear and non-uniform planar antenna arrays using gravitational search algorithm," *Progress In Electromagnetics Research*, vol. 73, pp. 131–145, 2017.

[18] G. Sun, Y. Liu, H. Li, S. Liang, A. Wang, and B. Li, "An antenna array sidelobe level reduction approach through invasive weed optimization," *International Journal of Antennas and Propagation*, vol. 2018, Article ID 4867851, 16 pages, 2018.

[19] S. Mandal, "Linear antenna array pattern synthesis using elephant swarm water search algorithm," *International Journal of Information Engineering & Electronic Business*, vol. 11, no. 2, pp. 10–20, 2019.

[20] H. Wang, C. Liu, H. Wu, B. Li, and X. Xie, "Optimal pattern synthesis of linear array and broadband design of whip antenna using grasshopper optimization algorithm," *International Journal of Antennas and Propagation*, vol. 2020, Article ID 5904018, 14 pages, 2020.

[21] A. Durmus and R. Kurban, "Optimum design of linear and circular antenna arrays using equilibrium optimization algorithm," *International Journal of Microwave and Wireless Technologies*, pp. 1–12, 2021.

[22] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 67–82, 1997.

[23] J. Xue and B. Shen, "A novel swarm intelligence optimization approach: sparrow search algorithm," *Systems Science & Control Engineering*, vol. 8, no. 1, pp. 22–34, 2020.

[24] G. Liu, C. Shu, Z. Liang, B. Peng, and L. Cheng, "A modified sparrow search algorithm with application in 3d route planning for UAV," *Sensors*, vol. 21, no. 4, article 1224, 2021.

[25] H. Wang and J. Xianyu, "Optimal configuration of distributed generation based on sparrow search algorithm," *IOP Conference Series: Earth and Environmental Science*, vol. 647, no. 1, article 012053, 2021.

[26] Y. Wang and J. Tuo, "Blood glucose prediction based on empirical mode decomposition and SSA-KELM," in *2020 Chinese Automation Congress (CAC)*, pp. 4759–4763, Shanghai, China, November 2020.

[27] S. Kumaravel and V. Ponnusamy, "An efficient hybrid technique for power flow management in smart grid with renewable energy resources," *Energy Sources, Part A: Recovery, Utilization, and Environmental Effects*, pp. 1–21, 2020.

[28] J. Yuan, Z. Zhao, Y. Liu et al., "DMPPT control of photovoltaic microgrid based on improved sparrow search algorithm," *IEEE Access*, vol. 9, pp. 16623–16629, 2021.

[29] Y. Zhu and N. Yousefi, "Optimal parameter identification of PEMFC stacks using adaptive sparrow search algorithm," *International Journal of Hydrogen Energy*, vol. 46, no. 14, pp. 9541–9552, 2021.

[30] B. Liu and D. Rodriguez, "Renewable energy systems optimization by a new multi-objective optimization technique: a residential building," *Journal of Building Engineering*, vol. 35, article 102094, 2021.

[31] Y. Li, S. Wang, Q. Chen, and X. Wang, "Comparative study of several new swarm intelligence optimization algorithms," *Computer Engineering and Applications*, vol. 56, no. 22, pp. 1–12, 2020.

[32] C. Zhang and S. Ding, "A stochastic configuration network based on chaotic sparrow search algorithm," *Knowledge-Based Systems*, vol. 220, article 106924, 2021.

[33] S. Li and Y. Li, *Intelligent Optimization Algorithm Theory and Applications*, Publishing House of Harbin Institute of Technology, Harbin, 2012.

[34] A. Jovanović, L. Lazović, and V. Rubežić, "Adaptive array beamforming using a chaotic beamforming algorithm," *International Journal of Antennas and Propagation*, vol. 2016, Article ID 8354204, 8 pages, 2016.

[35] H. Zang, H. Huang, and H. Chai, "Homogenization method for the quadratic polynomial chaotic system," *Journal of Electronics & Information Technology*, vol. 41, no. 7, pp. 1618–1624, 2019.

[36] H. Yang, T. Chen, and N.-j. Huang, "An adaptive bird swarm algorithm with irregular random flight and its application," *Journal of Computational Science*, vol. 35, pp. 57–65, 2019.

[37] S. Cheng, Y. Shi, and Q. Qin, "Experimental study on boundary constraints handling in particle swarm optimization," *International Journal of Swarm Intelligence Research*, vol. 2, no. 3, pp. 43–69, 2011.

[38] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, pp. 1942–1948, Perth, WA, Australia, November-December 1995.

[39] Y. Shi and R. Eberhart, "A modified particle swarm optimizer," in *1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No. 98TH8360)*, pp. 69–73, Anchorage, AK, USA, May 1998.

[40] S. Mirjalili and S. Z. M. Hashim, "A new hybrid PSOGSA algorithm for function optimization," in *2010 International Conference on Computer and Information Application*, pp. 374–377, Tianjin, China, December 2010.

[41] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Advances in Engineering Software*, vol. 95, pp. 51–67, 2016.

[42] S. Saremi, S. Mirjalili, and A. Lewis, "Grasshopper optimisation algorithm: theory and application," *Advances in Engineering Software*, vol. 105, pp. 30–47, 2017.

[43] M. H. Nadimi-Shahraki, S. Taghian, S. Mirjalili, and H. Faris, "MTDE: an effective multi-trial vector-based differential evolution algorithm and its applications for engineering design problems," *Applied Soft Computing*, vol. 97, article 106761, 2020.

[44] F. Gross, *Smart Antennas with MATLAB, Second Edition*, McGraw-Hill, 2015.

[45] G. Sun, Y. Liu, Z. Chen, S. Liang, A. Wang, and Y. Zhang, "Radiation beam pattern synthesis of concentric circular antenna arrays using hybrid approach based on cuckoo search," *IEEE Transactions on Antennas & Propagation*, vol. 66, no. 9, pp. 4563–4576, 2018.

Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 5544435, 9 pages
https://doi.org/10.1155/2021/5544435

Research Article

PackeX: Low-Power High-Performance Packet Classifier Using Memory on FPGAs

Khalid Rehman [1] **and Zahid Ullah** [2]

[1]*Department of Electrical Engineering, CECOS University of IT and Emerging Sciences, Peshawar, Pakistan*
[2]*Department of Electrical and Computer Engineering, Pak-Austria Fachhochschule: Institute of Applied Sciences and Technology, Haripur, Pakistan*

Correspondence should be addressed to Khalid Rehman; khalid@cecos.edu.pk

Received 23 January 2021; Accepted 24 May 2021; Published 7 June 2021

Academic Editor: Sungchang Lee

Networks are continuously growing, and the demand for fast communication is rapidly increasing. With the increase in network bandwidth requirement, efficient packet-classification techniques are required. To achieve the requirements of these future networks at component level, every module such as routers, switches, and gateways needs to be upgraded. Packet classification is one of the main characteristics of a stable network which differentiates the incoming flow into defined streams. Existing packet classifiers have lower throughput to cope with the higher demand of the network. In this work, we propose a novel high-speed packet classifier named as PackeX that enables the network to receive and forward the data packets in a simplest structure. A size of 128-rule 32-bit is successfully implemented on Xilinx Virtex-7 FPGA. Experimental findings show that our proposed packet classifier is versatile and dynamic compared to the current FPGA-based packet classifiers achieving a speed of 119 million packets per second (Mpps), while consuming 53% less power compared with the state-of-the-art architectures.

1. Introduction

Network devices in their early stages lack packet classification because of the limited application, but the use of Internet is now a multiservice ecosystem; a classifier is one of the main components of the whole networking system [1–3]. It enables the network to deploy service classification, i.e., security, quality of service (QoS), multimedia communications, and monitoring, and distinguish different network traffic from each other [4]. Some simple techniques are also applied to a router to decide if packets should be forwarded or dropped in order to prevent network infrastructure from being violated. Generally, rules are kept relatively static in conventional implementations, so fast classification can be accomplished with the algorithms running over the constructed classifier's well-designed data structure [5]. The primary objective of classifier design has been to perform high-speed packet processing in the past, such as content detection, load balancing, and packet filtering. The classifier can be installed offline as the rule update is rare at that moment. These new applications can respond simultaneously to a wide range of requests from different users, so that the classifier must be updated regularly in order to satisfy different requirements. Standard network migration operations modify the topology of the network, and the procedure adjusts the classifier accordingly [5, 6]. This allows the sorting of packets to be carried out electronically with the help of updating the fast dynamic policy, which is undoubtedly a necessary prerequisite for current and future classifiers.

The introduction of software-defined networking (SDN) [7–9] provides immense potential to support innovative functionality and value-added functionality for network growth. This includes traffic engineering support [10], network function virtualization (NFV) [11, 12], and high-performance cloud computing [13, 14]. Field-programmable gate arrays (FPGAs) due to their immense level of hardware parallelism are becoming extremely famous for the implementation of high-speed networks [2, 15, 16].

Software-defined networks (SDN), traffic engineering, and network function virtualization (NFV) as the next-generation networks are providing the most flexible networks. FPGAs fit completely to its requirement of reconfigurability and flexibility [17, 18].

Figure 1 shows a generalized structure of a packet classifier that helps the network to forward the incoming packet into the corresponding node or hop based on its destination address. It uses content-addressable memory (CAM) to store the destination addresses (IP addresses in the packet) and random-access memory (RAM) to store the next node where the packet needs to be transferred. There are basically many types of packet classifiers but the one we cover in this paper is based on the destination address and the next-hop [19]. A next-hop packet classifier consists of RAM and CAM. RAM performs searches using a memory address and then returns the data from the address [20]. The CAM-based search does the opposite. A function calls the CAM by passing a key that consists of a data word structure, and the CAM search returns a memory address. CAM further differentiates itself from different types of memory that it can perform memory searches in a single clock cycle. CAM can be a binary CAM or a ternary CAM depending on the requirement of the application [21]. However, other classifiers use other techniques to classify the packet from the incoming stream.

2. Motivation

Existing packet classifiers on FPGAs have the performance bottlenecks and cannot achieve higher throughput as required by the high-speed networks [17]. Thus, a high-speed (high throughput) packet classifier is needed to achieve the demand of the next-generation networks while not compromising on power. The novelty of our proposed architecture, PackeX, is its minimalism without compromising the high throughput requirement of the network to classify the incoming traffic. The proposed architecture provides higher speed packet classification by not compromising on the throughput of the network and consumes less power as compared to the existing packet-classification architectures.

3. Key Contributions

The key contributions of our proposed packet-classification architecture are as follows:

(i) The proposed packet classifier, PackeX, is state-of-the-art architecture to classify the network packets compared to the existing architectures

(ii) The proposed PackeX consumes half the power as compared to the state-of-the-art architecture

(iii) PackeX process the incoming data packets at a rate of 119 Mega packets per second (Mpps) using the distributed RAM on target FPGA

(iv) The proposed architecture is scalable and dynamically reconfigurable compared to the existing state-of-the-art TCAM-based packet-classification architectures

The rest of the paper is organized in the following way: Section 4 addresses the related work. Section 5 explains the proposed PackeX classification system and the proposed architecture. The pipelining of the architecture proposed is defined in Section 6. The outcomes of the deployment and performance assessment of our proposed architecture are addressed in Section 7. Section 8 concludes the paper.

4. Related Work

Hardware-based packet classifiers can be divided into three main types: decision-tree, exhaustive search, and decomposition. Decision-tree classifiers have large hardware requirements and the time to process the defined rule set. Our proposed architecture has lower hardware resource requirement and is implementable on FPGAs, providing high performance compared to the state-of-the-art packet classifiers which is shown in Table 1.

Zhang and Zhou reported a scheme based on reducing the TCAM memory usage [22]. This code based on split function was proposed in this study. Firstly, the splitting of d-tuple rule set into d-tuple field takes place, followed by obtaining a unique field test for each dimension. Based on matching with the incoming packet, its storage in SRAM memory and indexing in TCAM using a concatenated field occurs. On the other side, our proposed packet classifier uses a novel and simple architecture to develop the packet addresses into a TCAM that is partitioned to store in the distributed memory of the target FPGA. This improves the throughput compared to the available state-of-the-art packet-classification algorithms.

Using a Multimatch Using Discrimination (MUD) approach, Lakshminarayanan et al. utilized the extra bits of TCAM entry for encoding it, followed by keeping the encoded value in the same TCAM entry [23]. Evidently, these multiple lookup cycles are one of the major disadvantages for high-speed networks. This fact has been explained in detail in the packet classification section of this article. Although the multiple lookup cycles consume a considerable amount of power, nevertheless, the individual power values of PackeX remain low owing to its simple structure [24].

Another packet classifier based on TCAM has been proposed in [25]. This packet classifier reduces the width of TCAM to 36 bits, which leads to reduce the space requirement for TCAM. Consequently, the storage of set rules takes place in the SRAM. However, due to field code formation and indexing mechanisms, this packet classifier requires large memory. In fact, the scattered and sparse SRAM array leads to wasting the majority of SRAM entries. PackeX uses the distributed RAM which is partitioned to reduce the RAM usage and provide the high-speed searching to achieve the high-speed classification of the incoming data packets.

Updating of the packet classifier is also an important aspect of the design, which sometimes becomes the bottleneck for dynamic networks. TCAM update modules are developed to speed up the updating of the searching module

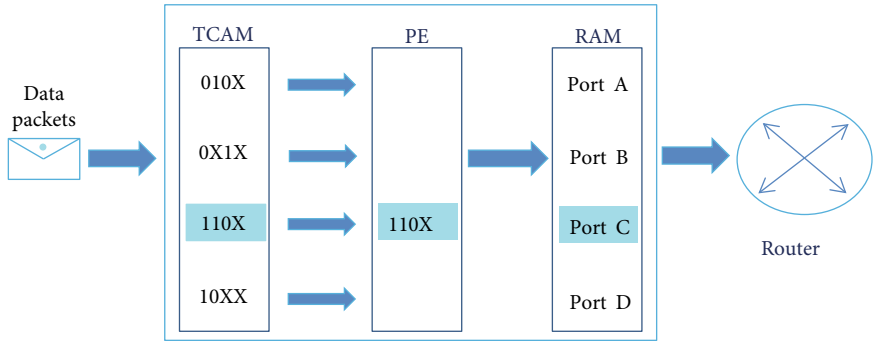


Figure 1: Generalized structure of packet classifier.

Table 1: Performance comparison of PackeX with the related work.

Packet classification engine	Platform	Memory/resource utilization	Power consumption (mW)	Throughput (Mpps)
Zhang et al.	FPGA	Maximum M/R utilization	1120	7 Mpps
Lakshminarayanan et al.	FPGA	Maximum M/R utilization	—	20 Mpps
Wang et al.	FPGA	Maximum M/R utilization	1200	—
Shen et al.	FPGA	Maximum M/R utilization	1900	—
Inayat et al.	FPGA	Average M/R utilization	33.72	20 Mpps
Ullah et al.	FPGA	Average M/R utilization	188	9 Mpps
Irfan et al.	FPGA	Average M/R utilization	—	33 Mpps
PackeX	FPGA	Minimum M/R utilization	17	119 Mpps

of packet classifier which is constantly improving and depends on the type of RAM used in FPGA [26, 27]. BRAM and distributed RAM have different update latencies based on their available depth in the target FPGA. We have chosen the distributed RAM which provides the minimum clock cycle update time which makes our packet classifier unique in its updating process. This makes PackeX dynamic as well as faster to support the modification of the network to mitigate the incoming traffic from time to time.

Lookup tables (LUTs) were used instead of memory blocks by Khatami and Ahmadi Partial reconfiguration in field-programmable gate array (FPGA) is used to shorten the time required to adjust the actions of an architecture. The pipeline system, on the other hand, results in unbalanced memory sharing, resulting in poor throughput and inefficient source allocation [28]. PackeX employs a very basic framework and a pipelining scheme, resulting in fast throughput and optimal use of source allocation.

Aceto et al. proposed a scheme of MIMETIC that can handle capitalized heterogeneity data traffic by using both inter- and intramodalities. The scheme outperforms the single modality-based techniques by supporting more challenging mobile traffic scenarios. MIMETIC uses three datasets to validate the performance improvement over the fusion classifiers, ML-based traffic classifiers, and single modality DL-based schemes [29]. The authors in [30] applied deep learning (DL) algorithm for packet classification on mobile traffic. The algorithm is centered on feature extraction, capable of running with encrypted and complex data traffic. It outperforms the ML state of art technique and previous deep learning-based algorithms by improving F-measure. On the other hand, our proposed packet classifier employs a novel and easy design to convert the packet addresses into a TCAM partitioned for storage in the target FPGA's distributed memory.

The development of binary and ternary CAM also involves the packet classifier because it is the vital component in classifying the address from the stored address in a little time [31]. Thus, Bi-CAMs and TCAMs include the partitioning of the corresponding CAM, binary to ternary conversion of the storage cells, and pipelining of the internal signals to improve the throughput. The proposed packet classifier uses the idea of partitioning taken from HP-TCAM in [15] and pipelining taken from D-TCAM in [16] to develop a novel structure for packet classification in a fastest way. We thus attain a higher speed in terms of millions of packets per second, which is the best and fastest packet classifier to the best of our understanding, while consuming 40% to 53% less power compared to the state-of-the-art architectures.

5. Proposed Architecture

5.1. Terminology. Table 2 shows the notations that are used to describe the proposed packet-classification design (PackeX).

5.1.1. PackeX Internal Structure. PackeX internal structure is based on the distributed RAM available in modern FPGAs which implements the TCAM part of our design to store the network addresses. There are two major components in PackeX:

Table 2: Terminologies or notations.

Notations	Explanation
A_D	Destination address
W	Number of bits to each sub-TCAM. For 6-input, lookup tables (LUTs) w is 6
W	Number of bits in the destination address (A_D)
$D_T \times W_T$	Size of TCAM, where D_T represents depth of TCAM and W_T represents width of TCAM
$D_R \times W_R$	Size of RAM of RAM, where D_R represents depth of TCAM and W_R represents width of TCAM

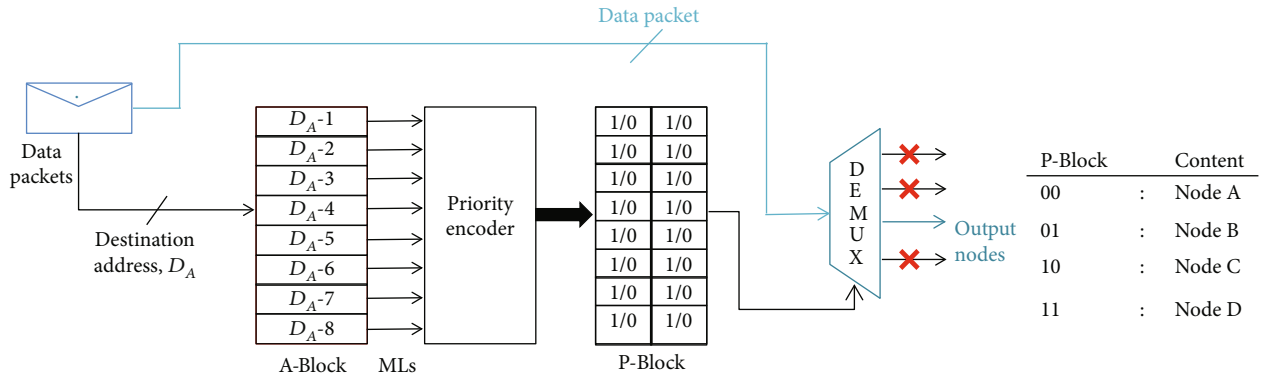


Figure 2: Structure of packet classifier, PackeX.

(i) A-block: a TCAM that stores the addresses of the incoming packets. These addresses in this paper are considered to be destination addresses. However, it can also be a source of addresses by using PackeX as filters for data packets from which they come and determine whether or not to drop or forward the packet. This block is known as A-block (address block)

(ii) P-block: a RAM that stores the information (pointers) for the corresponding nodes. It is known as P-block (pointers block)

Other blocks/components include demultiplexer, priority encoder or a simple encoder, ANDing modules, and connecting circuitry.

Figure 2 shows the internal structure of PackeX. The destination address (D_A) from the incoming packet is searched/-compared with the stored addresses in A-block. One or more of the stored entries are matched with the input D_A and generates the corresponding Match-Lines (MLs). Priority encoder (PE) converts the Match-Lines into an address for P-block which provides the values of the corresponding node. For example, "00" represents node A, "01" represents node B, and so on. The packet is forwarded according to the node specification provided by P-block using a demultiplexer (DEMUX). A-block and P-block are the main blocks storing the address information and node information. The size of demultiplexer (DEMUX) depends on the number of nodes in the network. Here, the nodes are 4 which need a DEMUX of size 1 to 4.

5.2. Classification Procedure. The two basic blocks involved in the classification process are A-block and P-block. Algorithm 1 shows the procedure of the data flow. The destination addresses lead to the Match-Lines which find out the node, and the demultiplexer (DEMUX) activates only one channel out of many channels using the content stored in the P-block. The size of DEMUX is determined by the number of nodes that are supported by PackeX. For instance, 4-node PackeX require 1 : 4 DEMUX while the size of P-block depends on the number of destination addresses stored in the A-block.

5.3. Partitioning of TCAM Module. The TCAM part of PackeX has 6-bit sub-TCAM modules. Multiple sub-TCAM modules combine to form the complete TCAM component of the proposed packet classifier, as shown in Figure 3. The use of 6-bits is due to the built-in structure of distributed RAM inside modern FPGAs that are based on the 6-bit lookup tables (LUTs) also known as LUTRAMs. It holds the destination addresses of the packets that are received by PackeX for classification.

TCAM submodules of 6-bits each forms the TCAM module that stores the destination addresses of the incoming data packets to the packet classifier. The symbol W represents the total size of the stored addresses while the symbol w is 6 because of the 6-input LUTs available in the target FPGA.

The numbers of required sub-TCAM modules are determined from the required size of the addresses that need to be stored in the proposed packet classifier. The ideal addresses are always multiples of 6, i.e., 6, 12, 18, and so on. If the size of addresses/rules is 6 bits, the required sub-TCAM modules

Input: Data packet having destination address (A_D)
Output: Corresponding node where the packet needs to be forwarded (O_P)
MLs < = A-block [A_D]
Node$_n$ < = P-block [MLs]
Output < = DEMUX [Node$_n$]
Procedure (The algorithm is used to forward incoming packet to appropriate output port)
[Apply the incoming data packet to A-block to get destination output port];
for $D_T = 0$ to n **do**
$D_{\text{MUX}} = D_P$
Next D_P
end for
[Apply the incoming destination address to P-block to get destination output port];
for $D_R = 0$ to n **do**
if D_A = RAM (i, j) **then**
$O_P = D_A$
Exit
Next D_P
end for
[Finish];
end procedure
Note: A-block represents the address block where the destination addresses are stored, while P-block represents the pointers block where the pointers to the corresponding node are stored.
A-block is a TCAM and P-block is a RAM.

Algorithm 1: The proposed packet-classification algorithm (PackeX).

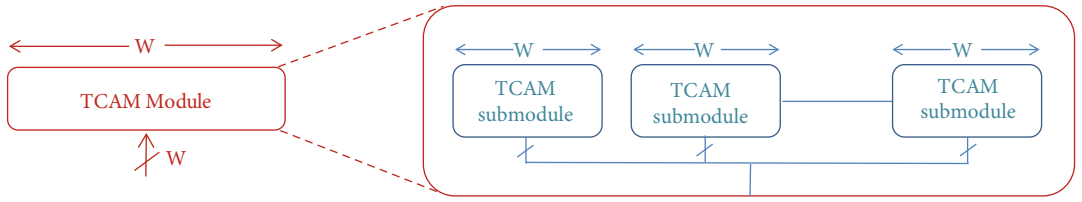


Figure 3: TCAM submodules.

are 1. If the size of addresses/rules is 12 bits, the required sub-TCAM modules are 2, and so on.

6. Pipelining

Performance of digital design degrades when the shortest path from input to output is larger as it reduces the speed of the whole circuit. Pipelined registers are introduced in stages to the digital design in order to reduce the path and improve the throughput. This is done recently by D-TCAM [16] which has shown improvement by incorporating flip-flops (FFs) to each of the distributed RAM that improved the throughput of TCAM on FPGA. We took the pipelining procedure from D-TCAM [16] to enhance the throughput of our system which is degraded by the ANDing circuitry, P-block, encoder, and demultiplexer. The proposed packet classifier achieves a high speed in classifying the incoming packets with 119 Mega packets per second (Mpps), while consuming 40 to 53% less power.

6.1. Example Design. The proposed packet classifier is scalable based on the available hardware resources on FPGA. To understand the basic architecture, we explain the design of 8 destination addresses and 4 nodes. The size of P-block depends on the number of nodes and destination addresses, such that

$$W_R(\text{Width of RAM}) = \log_2(\text{Number of Nodes}),$$
$$D_r(\text{Depth of RAM}) = \text{Number of Destination Addresses}. \quad (1)$$

Thus, the size of RAM required for P-block is 8×2, as shown in Figure 2. Priority encoder (PE) and demultiplexer (DEMUX) are 8 : 3 and 1 : 4, respectively.

6.2. Why an FPGA-Based Packet Classifier? Reconfigurability and high performance can easily put to work. As described in Section 4, packet classifiers are hardware and software-based. Hardware-based packet classifiers are faster but mostly fixed. FPGAs are reconfigurable according to the requirement of the system. Building a packet-classifier on FPGA gives us hardware-like performance (speed) and dynamic nature (reconfigurability) of the whole system. If the network grows in size, so is the packet classifier, because of the reconfigurable blocks available on FPGAs. Thus, our proposed architecture for packet classification, PackeX, has a simple structure

Table 3: Implementation results on Xilinx Virtex-7 FPGA for PackeX.

Approaches	No. of addresses/rules	Platform	No. of SRs	No. of LUTs	Throughput (Mpps)	Power (mW)
PackeX-I	64	Virtex-7	403	616	71	11
PackeX-II	128	Virtex-7	770	1095	119	17

Mpps: Mega packets per second; mW: milliWatt; SR: Slice Register; LUT: lookup table.

achieving improvement in important features, i.e., speed, power reduction, and reconfigurability.

7. Implementation

We have successfully implemented PackeX on Xilinx Virtex-7 FPGA using Xilinx Vivado 2018.2. The FPGA device xc7vx690tffg1157 is used with speed grade -2. Table 3 shows the implementation results for 64-rule 36-bit and 128-rule 32-bit packet classifier which classifies the incoming packets at a speed of 71 and 119 million packets per second (Mpps) and power consumption of 11 mW and 17 mW, respectively, compared to the state-of-the-art packet classifier.

Most importantly, the proposed design is purely a hardware-based design which eliminates the deficiencies of a software-based packet classifier. However, the proposed design is reconfigurable just like software-based design. Thus, PackeX combines the effective and useful properties of both software (i.e., reconfigurability) and hardware (i.e., speed). The experimental results show the feasibility and scalability of PackeX for future software-defined networks by its ability of adapting structure according to the need to the underlying application.

Table 4 shows the hardware utilization of the proposed packet classifier. The utilization of logical resources on the target FPGA is using under 1% of the available hardware except in case of the input/output (I/O) pins. The I/Os can be further reduced to optimize the performance of the system in future work. The lookup tables (LUTs) are 0.25%, LUTRAMs are 0.44%, and Slice Registers (SRs) are 0.09% of the available resources which show the scalability of our proposed architecture. Higher sizes of the packet classifier can be implemented on PackeX.

PackeX receives data from the server, classifies according to the stored addresses/rules, and forwards to the corresponding node. A node can be anything including but not limited to mobile phone, laptop, computer, and another server. Only one of the connected channels is activated to transfer the processed packet.

To implement the P-block, there are many options in modern FPGAs, i.e., single port distributed RAM, single port block RAM, and single port ROM. In our proposed design, the numbers of nodes are fixed to node 1, 2, $N-1$, and N as shown in Figure 4.

We use single port ROM to store the content of P-block which provides the information about the type of node to which the packet needs to be forwarded. It can also be implemented using RAM whenever the system needs dynamic number of nodes in the network.

Table 5 presents the comparative analysis of various approaches in terms of metrics such as dynamic power consumption, latency, and FPGA resource utilization. A comparison of the proposed PackeX-TCAM architecture with the current state-of-the-art FPGA-based TCAMs in terms of power consumption output with the current state-of-the-art FPGA-based TCAMs [36] is given by equation (2).

Table 4: Implementation results of hardware utilization.

S.No.	Resources	Utilization	Available	Utilization (%)
1	LUT	1095	433200	0.25
2	LUTRAM	768	174200	0.44
3	SR	770	866400	0.09

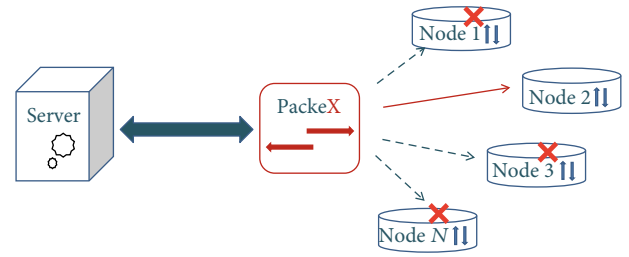


Figure 4: PackeX receives data from the server, classifies according to the stored addresses/rules, and forwards to the corresponding node.

$$\frac{\text{Normalized Power Consumption } [mW/D_R]}{\text{Performance}}. \quad (2)$$

Figure 5 presents the comparative analysis of logical resource utilization of existing state-of-the-art TCAM-based packet-classification architectures implemented on FPGA. As shown, compared with other TCAM-based classifiers, our proposed PackeX classifier is utilizing efficient logical resources.

In addition, Figure 6 represents the power ranking comparison of various TCAM-based architectures. It is envisaged from the figure that the power consumption for G-AETCAM [32], RE-TCAM [33], CLB-TCAM [34], and REST [35] is 36, 28, 22.23, and 161.18, respectively. Conversely, the power consumption for PackeX-I and PackeX-II is 11 and 17 mW, respectively.

Hence, the utilization and less number of resources for PackeX-I and PackeX-II approach lead to lower power consumption ranking without compromising on throughput.

8. Conclusions

Packet classification is a key operation needed in provisioning several important network services. One of the major challenges in designing the next-generation high-speed switches is to deliver high-speed low-power packet

Table 5: Power consumption of PackeX with RE-TCAM [32], G-AETCAM [33], CLB-TCAM [34], and REST [35].

Approaches	Platform	TCAM size ($D \times W$)	No. of SRs	No. of LUTs	Latency	Power (mW)
G-AETCAM [32]	Virtex-6	64×36	4672	3067	2	36
RE-TCAM [33]	Virtex-6	64×36	2337	2513	2	28
CLB-TCAM [34]	Virtex-6	64×36	4750	1472	1	22.23
REST [35]	Kintex-7	72×28	390	130	5	161.18
PackeX-I	Virtex-7	64×36	403	616	2	11
PackeX-II	Virtex-7	128×32	770	1095	2	17

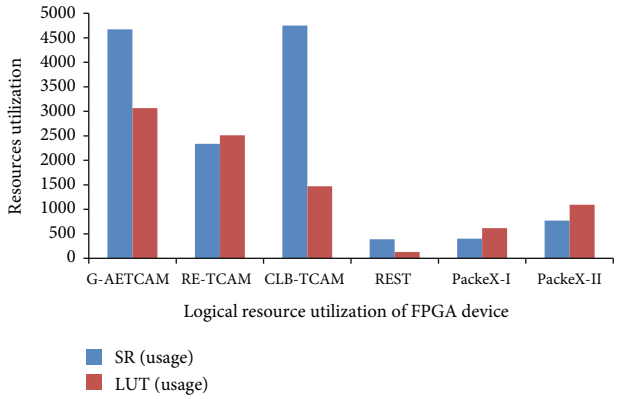


Figure 5: Comparison of resource utilization of FPGAs with PackeX.

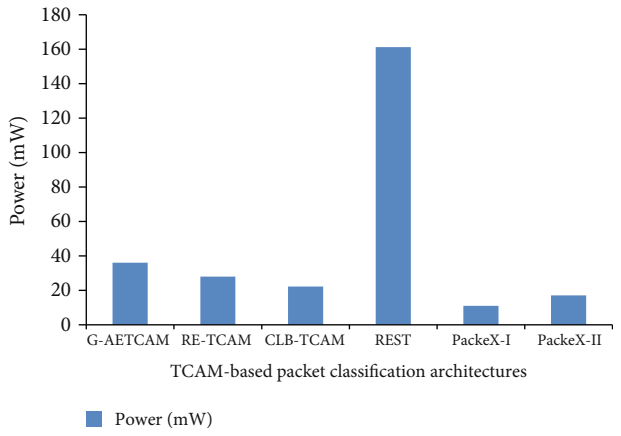


Figure 6: Power consumption ranking of various TCAM-based architectures.

classification. TCAM-based labeling of packets is a de facto standard for the high-performance processing of packets. However, due to inherent parallel structure, the high cost and high energy consumption are the major challenges of its efficient usage/implementation. TCAMs are the dominant industry standard used for multibit classifiers. However, as packet classification policies grow thorough and complex, a fundamental tradeoff arises between the TCAM space and the number of addresses for hierarchical policies.

This paper proposes a novel algorithm based on dynamic programming for solving important problems concerned with packet classification. PackeX presents a high throughput in packet forwarding as compared to the existing algorithms with less power consumption. The proposed packet-classification design classifies the incoming packets with simplicity and provides high throughput and consumed less power compared to the existing algorithms. The PackeX design uses fewer hardware resources. It is reconfigurable when the network needs modification, making PackeX favorable to future defined networks. Our algorithms do not require any change to existing packet classification systems and can be easily deployed. As far as we know, this is the first work to study TCAM speed and memory optimization for packet classification.

In our current work, PackeX is deployed for fast mapping and updating algorithms to eliminate the complication of sequential steps in generating lookup tables (LUTs) and iterative procedure for calculating the required address of the packet. The future works may include deployment of submodules at such locations that can balance the energy consumption and speedup the process. This may be possible by using some statistical distribution for the deployment of modules with respect to the horizontal partitioning and virtual partitioning. In addition, the use of proper pipelining and partitioning can also be investigated in our proposed algorithm in the future.

Data Availability

The data that support the study's outcomes are all briefly introduced, and all information is included in the manuscript.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] H. T. Lin and P. C. Wang, "Fast TCAM-based multi-match packet classification using discriminators," *IEEE Transactions on Multi-Scale Computing Systems*, vol. 4, no. 4, pp. 686–697, 2018.

[2] E. Norige, A. X. Liu, and E. Torng, "A ternary unification framework for optimizing TCAM-based packet classification systems," *IEEE/ACM Transactions on Networking*, vol. 26, no. 2, pp. 657–670, 2018.

[3] K. Pagiamtzis and A. Sheikholeslami, "Content-addressable memory (CAM) circuits and architectures: a tutorial and

survey," *IEEE Journal of Solid-State Circuits*, vol. 41, no. 3, pp. 712–727, 2006.

[4] D. E. Taylor, "Survey and taxonomy of packet classification techniques," *ACM Computing Surveys (CSUR)*, vol. 37, no. 3, pp. 238–275, 2005.

[5] T. Shen, G. Xie, X. Wang et al., "RVH: range-vector hash for fast online packet classification," *arXiv preprint arXiv*, vol. 1909, 2019.

[6] S. Yingchareonthawornchai, J. Daly, A. X. Liu, and E. Torng, "A sorted partitioning approach to high-speed and fast-update OpenFlow classification," in *2016 IEEE 24th International Conference on Network Protocols (ICNP)*, pp. 1–10, IEEE, 2016.

[7] L. Molnár, G. Pongrácz, G. Enyedi et al., "Dataplane specialization for high-performance OpenFlow software switching," in *Proceedings of the 2016 ACM SIGCOMM Conference*, pp. 539–552, Florianopolis, Brazil, 2016.

[8] A. Sivaraman, A. Cheung, M. Budiu et al., "Packet transactions: high-level programming for line-rate switches," in *Proceedings of the 2016 ACM SIGCOMM Conference*, pp. 15–28, Florianopolis, Brazil, 2016.

[9] B. Li, K. Tan, L. Luo et al., "Clicknp: highly flexible and high performance network processing with reconfigurable hardware," in *Proceedings of the 2016 ACM SIGCOMM Conference*, pp. 1–14, Florianopolis, Brazil, 2016.

[10] S. Jain, A. Kumar, S. Mandal et al., "B4," *ACM SIGCOMM Computer Communication Review*, vol. 43, no. 4, pp. 3–14, 2013.

[11] C. Sun, J. Bi, Z. Zheng, H. Yu, and H. Hu, "Nfp: enabling network function parallelism in nfv," in *Proceedings of the Conference of the ACM Special Interest Group on Data Communication*, pp. 43–56, Los Angeles, CA, USA, 2017.

[12] P. Zave, R. A. Ferreira, X. K. Zou, M. Morimoto, and J. Rexford, "Dynamic service chaining with dysco," in *Proceedings of the Conference of the ACM Special Interest Group on Data Communication*, pp. 57–70, Los Angeles, CA, USA, 2017.

[13] D. Firestone, "VFP: a virtual switch platform for host {SDN} in the public cloud," in *14th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 17)*, pp. 315–328, Boston, MA, USA, 2017.

[14] C. Lan, J. Sherry, R. A. Popa, S. Ratnasamy, and Z. Liu, "Embark: securely outsourcing middleboxes to the cloud," in *13th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 16)*, pp. 255–273, Santa Clara, CA, USA, 2016.

[15] Z. Ullah, K. Ilgon, and S. Baeg, "Hybrid partitioned SRAM-based ternary content addressable memory," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 59, no. 12, pp. 2969–2979, 2012.

[16] M. Irfan, Z. Ullah, and R. C. Cheung, "D-TCAM: a high-performance distributed ram based TCAM architecture on FPGAs," *IEEE Access*, vol. 7, pp. 96060–96069, 2019.

[17] C. Li, T. Li, J. Li, D. Li, H. Yang, and B. Wang, "Memory optimization for bit-vector-based packet classification on FPGA," *Electronics*, vol. 8, no. 10, p. 1159, 2019.

[18] A. Ullah, A. Zahir, N. A. Khan, W. Ahmad, A. Ramos, and P. Reviriego, "BPR-TCAM—block and partial reconfiguration based TCAM on Xilinx FPGAs," *Electronics*, vol. 9, no. 2, p. 353, 2020.

[19] F. Baboescu, S. Singh, and G. Varghese, "Packet classification for core routers: is there an alternative to CAMs? IEEE INFOCOM 2003," in *Twenty-second Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE Cat. No. 03CH37428)*, vol. 1, pp. 53–63, IEEE, 2003.

[20] J. Van Lunteren and A. Engbersen, "Multi-field packet classification using ternary CAM," *Electronics Letters*, vol. 38, no. 1, pp. 21–23, 2002.

[21] Y. D. Kim, H. S. Ahn, S. Kim, and D. K. Jeong, "A high-speed range-matching TCAM for storage-efficient packet classification," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 56, pp. 1221–1230, 2008.

[22] Z. Zhang and M. Zhou, *A code-based multi-match packet classification with TCAM*, In Advances in Web and Network Technologies, and Information Management, Springer, 2007.

[23] K. Lakshminarayanan, A. Rangarajan, and S. Venkatachary, "Algorithms for advanced packet classification with ternary CAMs," *ACM SIGCOMM Computer Communication Review*, vol. 35, no. 4, pp. 193–204, 2005.

[24] M. Abbasi, N. Mousavi, M. Rafiee, M. R. Khosravi, and V. G. Menon, "A CRC-based classifier micro-engine for efficient flow processing in SDN-based internet of things," *Mobile Information Systems*, vol. 2020, 8 pages, 2020.

[25] R. Shen, X. Li, and H. Li, "A hybrid TCAM+ SRAM scheme for multi-match packet classification," in *2012 13th International Conference on Parallel and Distributed Computing, Applications and Technologies*, pp. 685–690, IEEE, 2012.

[26] Z. Wang, H. Che, M. Kumar, and S. K. Das, "CoPTUA: consistent policy table update algorithm for TCAM without locking," *IEEE Transactions on Computers*, vol. 53, pp. 1602–1614, 2004.

[27] T. Banerjee-Mishra and S. Sahni, "Consistent updates for packet classifiers," *IEEE Transactions on Computers*, vol. 61, pp. 1284–1295, 2011.

[28] R. I. Khatami and M. Ahmadi, "High throughput multi pipeline packet classifier on FPGA," in *InThe 17th CSI International Symposium on Computer Architecture & Digital Systems (CADS 2013)*, pp. 137-138, IEEE, 2013.

[29] G. Aceto, D. Ciuonzo, A. Montieri, and A. Pescapè, "MIMETIC: mobile encrypted traffic classification using multimodal deep learning," *Computer Networks*, vol. 165, p. 106944, 2019.

[30] G. Aceto, D. Ciuonzo, A. Montieri, and A. Pescapé, "Toward effective mobile encrypted traffic classification through deep learning," *Neurocomputing*, vol. 409, pp. 306–315, 2020.

[31] I. Ullah, Z. Ullah, and J. A. Lee, "EE-TCAM: an energy-efficient SRAM-based TCAM on FPGA," *Electronics*, vol. 7, no. 9, p. 186, 2018.

[32] M. Irfan and Z. Ullah, "G-AETCAM: gate-based area-efficient ternary content-addressable memory on FPGA," *IEEE Access*, vol. 5, pp. 20785–20790, 2017.

[33] H. Mahmood, Z. Ullah, O. Mujahid, I. Ullah, and A. Hafeez, "Beyond the limits of typical strategies: resources efficient FPGA-based TCAM," *IEEE Embedded Systems Letters*, vol. 11, pp. 89–92, 2018.

[34] I. Ullah, U. Afzaal, Z. Ullah, and J. A. Lee, "High-speed configuration strategy for configurable logic block-based TCAM architecture on FPGA," in *2018 21st Euromicro Conference on Digital System Design (DSD)*, pp. 16–21, IEEE, 2018.

[35] A. Ahmed, K. Park, and S. Baeg, "Resource-efficient SRAM-based ternary content addressable memory," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 25, pp. 1583–1587, 2016.

[36] H. Nakahara, T. Sasao, H. Iwamoto, and M. Matsuura, "LUT cascades based on edge-valued multi-valued decision diagrams: application to packet classification," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 6, no. 1, pp. 73–86, 2016.

Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 5541859, 13 pages
https://doi.org/10.1155/2021/5541859

Review Article

Image-Based Detection of Plant Diseases: From Classical Machine Learning to Deep Learning Journey

Rehan Ullah Khan,[1] **Khalil Khan**,[2] **Waleed Albattah**,[1] **and Ali Mustafa Qamar**[3]

[1]*Department of Information Technology, College of Computer, Qassim University, Buraydah, Saudi Arabia*
[2]*Department of Information Technology and Computer Science, Pak-Austria Fachhochschule, Institute of Applied Sciences and Technology, Pakistan*
[3]*Department of Computer Science, College of Computer, Qassim University, Buraydah, Saudi Arabia*

Correspondence should be addressed to Ali Mustafa Qamar; al.khan@qu.edu.sa

Received 22 February 2021; Accepted 24 May 2021; Published 3 June 2021

Academic Editor: Farman Ullah

Plant disease automation in agriculture science is the primary concern for every country, as the food demand is increasing at a fast rate due to an increase in population. Moreover, the increased use of technology today has increased the efficacy and accuracy of detecting diseases in plants and animals. The detection process marks the beginning of a series of activities to fight the diseases and reduce their spread. Some diseases are also transmitted between animals and human beings, making it hard to fight them. For many years, scientists have researched how to deal with the common diseases that affect humans and plants. However, there are still many parts of the detection and discovery process that have not been completed. The technology used in medical procedures has not been adequate to detect all diseases on time, and that is why some diseases turn out to become pandemics because they are hard to detect on time. Our focus is to clarify the details about the diseases and how to detect them promptly with artificial intelligence. We discuss the use of machine learning and deep learning to detect diseases in plants automatically. Our study also focuses on how machine learning methods have been moved from conventional machine learning to deep learning in the last five years. Furthermore, different data sets related to plant diseases are discussed in detail. The challenges and problems associated with the existing systems are also presented.

1. Introduction

The use of technology in the detection and analysis process increases the accuracy and reliability of these processes. For example, the people who use the latest technology to analyze the diseases that arise unexpectedly are at a higher chance of controlling them than those that do not. In the recent occurrence of coronavirus, the world relied on the latest technology to develop preventive measures that have helped reduce the rate at which the disease is transmitted. Crop diseases are a significant threat to human existence because they are likely to lead to droughts and famines. They also cause substantial losses in cases where farming is done for commercial purposes. The use of computer vision (CV) and machine learning (ML) could improve the detection and fighting of diseases. Computer vision is a form of artificial intelligence (AI) that involves using computers to understand and identify objects. It is primarily applied in testing drivers, parking, and driving of self-driven vehicles and now in medical processes to detect and analyze objects [1]. Computer vision helps increase the accuracy of disease protection in plants, making it easy to have food security.

One of the areas that CV has helped most is the detection of the severity of the diseases. Deep learning (DL), a part of the CV, is useful and promising in determining the severity of diseases in plants and animals [2]. It is also used to classify diseases and avoid the late detection of diseases [1]. Plant diseases are slightly different from those that affect human beings. Many factors make diseases similar as well. However, the diseases that can be transmitted from humans to plants and vice versa are rare. The analysis of the data related to this field helps identify how the use of the latest technology can be

improved. The images of leaves and other parts of the plants can be used to detect diseases in plants [3]. The technology could be applied in analyzing images in human beings that also prove the presence of diseases and determine the extent of their destruction. This research study is aimed at analyzing the way image-based technology can be used in detecting diseases in both plants and animals.

2. Background

ML is the technology that allows machines to communicate with human beings and understand their needs. It also makes machines act like human beings and make the decision on behalf of humans. It is one of the areas that have grown fast over the past few years. ML helps in classifying plant diseases. The use of this technology is seen as a significant beginning and achievement in dealing with plant diseases. It has also increased productivity in the field of cultivation. Visualization techniques have also been included in this technology, and it has been improved over the last three years to the current improved levels [4]. The challenges that face the world today, related to the diseases affecting plants and humans, can be reduced if the diseases are identified before they spread to vast areas. The use of ML is widespread in the world today. Diverse methods used in ML and DL help the experts to analyze the plant diseases and know their source in time [4]. The detection of these diseases is affected mainly by several challenges that affect the effectiveness and accuracy of this technology.

The first challenge is the time complexity associated with the use of ML and DL, whereby some of the technologies used in the detection of these diseases are outdated or based on some information from the past. The other challenge is segmentation sensitivity [5]. It means that the region of interest (RoI) requires a high level of accuracy and sensitivity to acquire the required usage and accuracy. The other challenge is that there is a language barrier that affects the way the technology is applied. Another challenge is the inadequate resources that are required to support the application of this technology. Most of the ML and DL activities need many resources to use and implement. Private and government entities usually fund the institutes that use this technology to detect diseases in humans and plants, which could affect the success of the research and implementation of the technology.

The importance of plants in the world has increased over time. The discoveries about the critical roles that plants could play in medicine, energy production, and the recent concerns about the reduction of global warming have for long been a significant part of science and technology [6]. A reduction in the plant cover in the world increases the risk of higher global warming and an increase in the related challenges. The need to build a state-of-the-art convolutional system that supports the image detection technology and classification of plant diseases has led to many research programs to provide the scientists with the required knowledge [7]. Image detection could be applied when necessary to differentiate healthy leaves from those that are not healthy. The convolutional neural networks (CNNs) provide the differences among plant images that help determine the abnormalities that could exist in the plants in the natural environment [6]. The background study shows that the scanning of the images that show the healthy and unhealthy plants forms a basis for comparison by the scientists in this field.

DL can be used to detect abnormalities in both humans and plants. The pixel-wise operations are used to analyze the leaves collected from sick plants, and this is used to classify the diseases according to their impact on the plants. The visible patterns in these leaves are used to decide the diseases that affect the plants and how they can be dealt with to prevent them from spreading. Research shows that the use of DL technology is up to 98.59% accurate [8]. The field of plant pathology has contributed immensely to the control of diseases and reduced global warming. One of the essential background knowledge that guides the use of image detection technology is that the leaves of the infected plants are different from the healthy ones. The leaves are likely to have dark parts, and some may be dry along the edges. The dried parts are also likely to fold up, and this is easy to detect even with a bare eye. The use of ML is to detect these differences without human intervention.

The ML methods used to make decisions on the detection of diseases include Artificial Neural Networks (ANN), decision trees, support vector machines (SVM), and K-means, among others [9]. The computers may not work directly with the images taken in the fields. The images are converted into data that the machines and computers can interpret. It means that the technology requires the coding of the images into data that can be fed into the computer systems. Therefore, the basic knowledge in computer codes and programming forms an integral part of the background knowledge required in this field.

Apart from the aforementioned application of the knowledge in plants and detecting the diseases on time, the image-based detection and recognition of diseases are applied in humans to detect the diseases that affect different body parts. Some human diseases have significant impacts on the tissues and organs that they affect. Adenocarcinoma of the prostate is one of the most common forms of cancer. It is detected using the image-based detection methods whereby the body is scanned for the abnormalities and the images obtained are used to determine if the patient has the disease. It is the second most diagnosed type of cancer in men, with about one in every nine men diagnosed with it in their lifetime [10]. The use of subjective tissue examination has been the primary way of detecting cancer in men suspected of it. The examination of these body tissues is largely dependent on the Gleason system [11]. The field of AI has offered many solutions to the challenge of inaccuracy in the Gleason scale and how it is used to provide solutions to people who have prostate cancer. The AI solutions are applied in analyzing the prostatectomy specimens to determine the impact of cancer on them. Before and after they have been affected by a disease like cancer, the appearance of the body organs is used to design the image-based detection process that is more accurate than the human eyes. In other words, computer systems can detect the difference in the organs more efficiently than humans. It helps to detect the disease long before they get to a fatal stage.

As for some plants like Cassava, it is possible to provide some protection to them from common diseases through image-based detection technology. Agriculture depends on some drought-resistant crops like Cassava to regulate the food supply and ensure an adequate supply of nutrients [12]. However, diseases threaten some of these crops' survival, which makes it hard for the agriculture departments to meet their targets. The background knowledge about the diseases could help avoid such instances. The use of CNN generates a reliable platform where the diseases are analyzed in detail [13]. The accuracy of this method is high and reliable. Further research by experts on the diseases that affect plants like Cassava has classified the diseases in terms of their impact on the leaves and other parts of the plants [14]. For example, many diseases affect the part of the leaves, some affect the entire leaf, and others attack the edges or the stalk. The images of the leaves can be analyzed using the image-based detection system to determine the classification that suits the disease.

In places like sub-Saharan Africa, Cassava is one of the essential foods because it provides people with many carbohydrates. However, due to its vulnerability to viral diseases, it has not effectively sustained the nutritional value it could achieve in this region. In 2014, about 145 million tons of Cassava was harvested in Africa [13]. The food control methods applied in the world today mainly focus on increasing production. The application of the latest technology to detect and control the disease that affects production is a reliable way of reducing the challenges of food inadequacy. Most of the plants are also used as raw materials in many industries. If such plants are of low quality, it means that they will lead to the production of low-standard products.

3. Literature Survey on Types of Plant Diseases

Fungi usually cause diseases that affect the plants, and they typically attack the leaves. Viral and bacterial pathogens cause many others. Precision in agriculture has improved with the increased use of ML and its related features [15]. The reduced production quantity in agriculture hurts many people and animals, which requires modern technology to solve. The extraction and detection of diseases are easier when the image-based detection system is used because of its high accuracy and reduced complications and duplication of data. In some plants like tomatoes, the use of the images to determine the diseases that affect them and the extent of the damages cannot be achieved unless there is a high accuracy rate [16]. The survey on plant diseases shows that many diverse factors determine how technology-based image detection is applied. In other words, the diseases that cause visible dents and changes on the plants are the ones that can be detected using this technology as opposed to the ones that cause damages that cannot be detected from the plants' images [17]. The analysis in this research shows that plant diseases are usually detected when they start showing an impact on the physical appearance of the plants.

The main challenge affecting the field of agriculture is the reduction in production and poor-quality production in plants. The challenge is a result of the poor detection and management of the diseases that affect the plants. The challenge is also extended to affect human beings in several ways. The reduced plant cover due to plant diseases means that global warming, famine, and reduced air purification ensue. Hyperspectral imaging has become a reliable way of detecting crop diseases on time [18]. It is hard to determine the factors that lead to the diseases unless they are detected on time. In other words, if a disease is detected on time, it is easy to relate it to the possible factors that lead to its occurrence. For example, scientists could determine if there was a change in weather or climate that could have led to the occurrence of the disease.

Further research by [6] shows an inadequate database that could be used to provide background knowledge for comparing the images taken. The other challenge is that the symptoms and characteristics of the diseases are diverse and could be similar to a certain degree [19]. For example, many diseases could lead to the wilting of leaves. The challenge is yet to be resolved because more and new images are uploaded progressively by experts.

The other challenge is the lack of suitable instruments for use in the work of image detection. Most of the experts in the field do not have the equipment they require to analyze the images they get from the field, and this makes it hard for them to acquire accurate data and identify the diseases [20]. The other one is that there is a low rate of implementation in some areas due to the regulations put in place to ensure the credibility and reliability of the data from these analyses. For example, after the 4th and 6th International Conference on Machine Learning and Soft Computing, there have been many regulations that may derail the use of ML in some parts [21]. The rules discourage some of the results from the ML functions from being applied in practice because they do not meet the required parameters.

The technology has been in existence for several years now. However, there are still many issues that have not been clarified about its application. The other challenge is related to this fact. Some of the important images that could help determine if disease exists have not been captured. The other one is that the future perspectives of the research are not clear, and this is because of the increased diversity in the diseases that affect both humans and animals [19]. The application of image-based detection is also affected by the increased diversity in the way the diseases appear. Some of the diseases that used to affect the plants a few years ago have evolved into new forms, and they have different impacts and outcomes. It is difficult for the images to be used alone to conclude the diseases and choose a solution. Some of the solutions used in the past have also become ineffective, reducing the effectiveness of the technology.

The aforementioned challenges show that there are many possible ways in which the image-based detection could be applied, but the challenges reduce its usability. The first solution is to provide adequate data that can be used to identify the diseases accurately without confusing the ones that are closely related. The changes in weather, global warming, and other impacts have led to many diseases that have not been documented. The solution is to increase scientists' coverage and promote a better way of collecting information

[22]. The other solution would be to offer training to the scientists in this field to ensure that they are equipped to collect the data. Yet another solution is to create better ways of capturing the data collected about the diseases. The challenge of inadequate information about the diseases can be solved if there is an improved data-captioning process that involves fine details of the images taken and the differences that define them [6]. The images should be analyzed keenly to determine the ones that are affected or infected.

Another solution would be to focus on using the latest technology that is reliable and valid. The confusion that comes with the inadequate database for use in detecting the diseases results from inferior technology and low storage abilities of the existing systems. Most of the images are not stored correctly, which affects the accessibility of the information. It could be solved by the use of modern methods of storing information. For example, the use of cloud computing could help increase the accuracy of storage and accessibility. The other solution is training the people in charge of the research and analysis of the information. A trained DL algorithm increases the accuracy of the technology [23]. The other solution would be to understand the phenotypes used in detecting diseases [24]. The phenotype used in the detection of diseases is usually a result of climate and weather [23, 25]. The other solution would be to update the systems to ensure the data captured is up to date. The high level of uncertainty in the detection of diseases affects the way the technology is implemented. For example, the use of Bayesian DL is associated with several uncertainties [26]. It means that this method is unreliable if used alone.

A combination of several methods could reduce the inefficiencies of the processes. For example, the application of DL meta-architectures provides a solution to some of the errors that are experienced when using other methods of identifying diseases [27]. The other solution would be the application of deep convolutional generative adversarial networks that help in the identification and analysis of the images [28]. The contribution of the adversarial networks increases the accuracy of the detection process. The use of CNN methods could also be effective in dealing with the inaccuracy and slow identification processes of the diseases [29]. The methods have been used to detect the diseases that affect rice and have many benefits [28]. Image-based detection requires many resources, and the authorities should ensure they are available to ensure the activities are smooth.

4. Plant Disease Image Data Sets

The data sets used in the research include the descriptions of the leaves before and after the diseases affect them. The data comprises tables and images of the leaves that are taken in the fields. The data is analyzed and classified in a way that is easy for the readers to understand. For example, Wallelign et al. [9] show the leaves used to determine the soybean plants affected by the diseases. The data set shows healthy leaves and the ones that had dents due to the attack by septorial leaf blight, others by frogeye leaf spot, and those affected by downy mildew as shown in Figure 1.

The images in Figure 1 show that there were visible differences between the leaves affected by the disease and those that had not. The data set was clear and easy to understand. Another form of data was the table that showed the number of leaves that were classified under each disease. The data set is clear and indicates the total number of leaves analyzed and classified into four categories. The other set of data is by [4], which shows the graphical representation of the captured information in the tables.

The data set used in the research can be shown in tables, texts, graphs, and other forms. However, an essential aspect of all of them is the ease of analysis and ease of understanding. Some of the data are also grouped as per the required levels. For example, data can be grouped in terms of the diseases they represent, the time they were collected, or the analysis method [30]. The data captured in some of the research outcomes also show the use of technology and its effectiveness. For instance, the data sets captured using technology allow for a controlled environment; the data sets show the type of control used and its effectiveness. For example, computer-aided diagnosis (CAD) systems were captured in the data analyzed by [31]. The data sets are provided to help understand the usage of this technology and its impact on the quality of the research. The same data shows the classification techniques used and gives reasons for the choice.

The other feature is that the data sets for leaves' analysis are based on the primary data collected in the fields. The reliability of the data is high because it is based on the observable features of the leaves. The data sets are also divided into sections that are easy to understand. For example, the work of [32] shows the divisions of the work in terms of the diseases like Rice Blast (RB), Bacterial leaf Blight (BLB), and Sheath Blight (SB). The use of a PlantVillage data set was also applied in the research by [32]. The data set consists of 54,306 images of 14 different crops representing 26 plant diseases. The images that were included in the data set included leaves having different colors. Figure 2 shows some samples of the PlantVillage data set. The colors indicate the parts of the leaves affected by the diseases that were under investigation. The authors also used the augmented data set proposed by Geetharamani and Pandian [33]. The ImageNet data set was also used in the research, and this led to high-quality research outcomes because of the synergy of combining various methods [32].

The data sets used in the research studies are dependent on the type of information contained. For example, the research by [14] focused on the effects of uncontrolled pests in China and the impacts on the total food produced. The research shows that poor control of pests in China leads to a loss of about 30% of the total foods produced [35]. The data sets in the research are large and show the different production levels and how the pests affect them. The use of a public data set also made it possible to understand the different ways in which the research was done and verified using data that are available in the public domain [36]. The PlantVillage data set was also used in [37] where 14 different types of leaves of cucumber plant are analyzed for seven different diseases. The data sets were mainly combined to provide a good presentation of the data that was collected. The use of the PlantVillage

(a) (b) (c) (d) (e)

Figure 1: Data set used for classification by Wallelign et al. [9].

data set by [38] was used to show how the collected information was helpful in understanding plant diseases.

The data sets used in most references were from the data collected by different researchers and combined in one set. The reliability of the research could be compromised if there is no control over what is to be included in the data sets [39]. The nature of the research also determines how the data sets are used. For example, the collection of leaves and combining them in a table usually involves using the PlantVillage data set. The use of coffee leaf data sets in the research by [40] is backed by the need to show the diseases that affect the coffee leaves and tomatoes and how they can be detected using

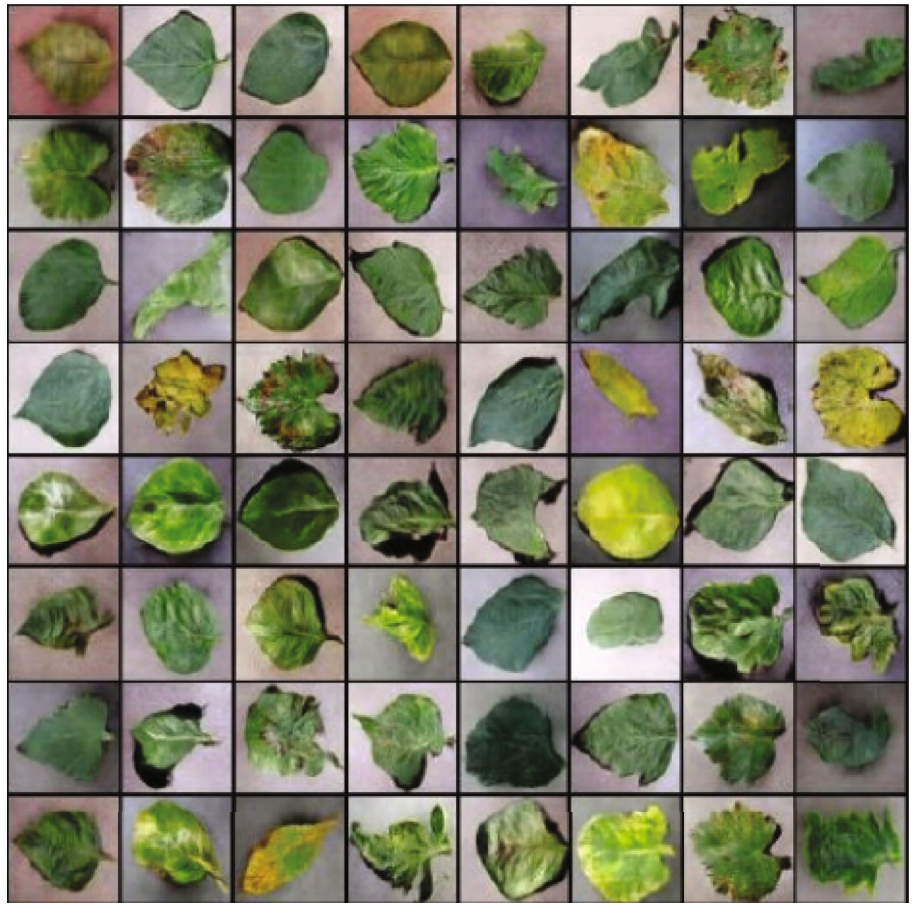


Figure 2: PlantVillage data set—examples of different phenotypes of various plants [34].

image-based detection methods [36]. In the research by [41], the authors used an information-rich color data set and large numerical data sets to display the data collected. The machine learning process also employs the training data set to predict and analyze unseen data. The data sets also include the expected data, making it easy to rely on the data in the research and determine if the data is valid. The use of data sets related to the nature of the research helps achieve the goals of the data collection and analysis [42]. The aforementioned data sets are affected by some challenges and limitations that reduce their applicability.

Another well-known data set used by the research community is Northern Leaf Blight (NLB), which contains infected maize leaves [1]. Some sample images from the NLB data set are shown in Figure 3. NLB consists of 1787 images having 7669 lesions. The images were obtained from maize plants in the field while using a handheld camera. The images in NLB were captured in uncontrolled conditions as opposed to the PlantVillage data set.

4.1. Challenges and Problems with the Data Sets. The explored data sets have several challenges and problems that affect their application. One of the challenges is how the data sets are organized to show the data collected from different fields. For example, when using the PlantVillage data set, the data collected from different fields could be hard to present in the same way. Most of the data may be diverse. The missing uniformity could lead to a challenge in how the data is shown in the tables and other exhibits [43]. The other problem is that the data sets may be hard to convert into graphical representations. The information in the data sets could make more sense if the information is shown in graphical formats [44]. The data could also make more sense to the readers when presented in graphs. However, some of the data sets cannot be transformed into graphs directly and may require redoing the graphs entirely. For the data sets that include tables, it is possible to convert the data into graphs without having to draw manually. The other challenge is that some of the data sets do not summarize the data contained [45]. For example, when data is presented in the PlantVillage, a summary of the total value is given at the end of the table. However, in some data sets, this information is missing.

The other challenge with the data sets is that they may not capture all the information about the factors that lead to the values shown in them. For example, the data presented in work by [27] does not show the factors that affect the classification and detection process. Such missing details could affect the way the data is applied in making decisions about the diseases [46]. The data sets could also be complex for people to use if they do not have some prior training and education related to the statistical analysis of the data. The data sets also contain some details that may be inconsistent with the

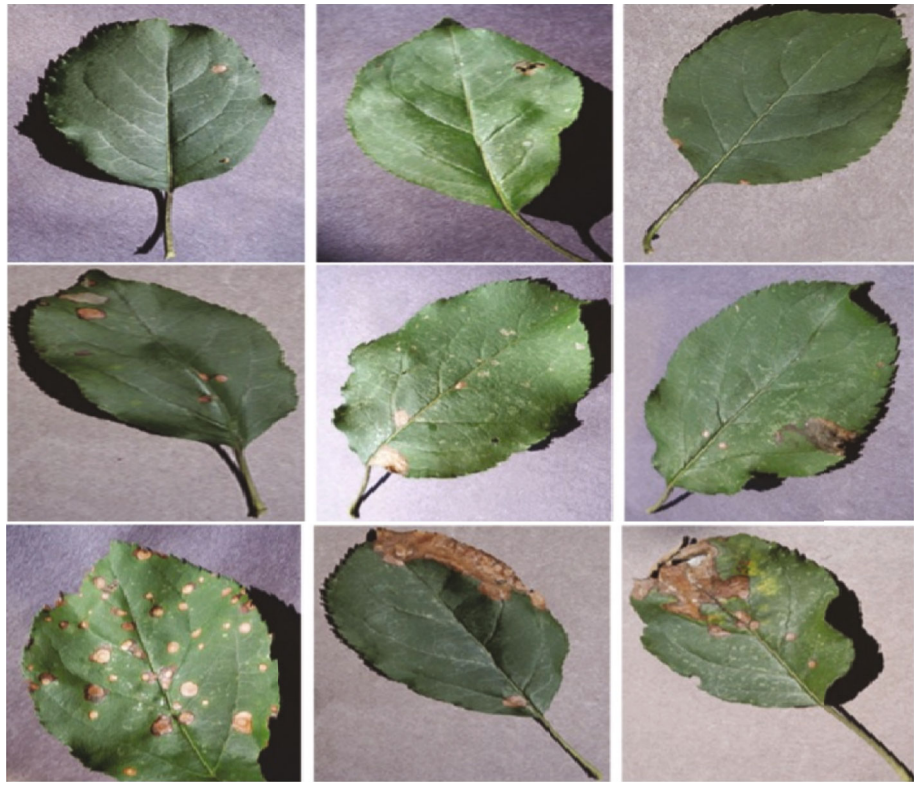


Figure 3: Sample leaf images from the Northern Leaf Blight data set [1].

research. For example, the data sets could contain information about the classification of the diseases and fail to capture the prevention measures that could be used to reduce the chances of having the diseases affecting the plants [47]. In some instances, the data sets are unclear, which could confuse the people who interpret them. Some of the data could show information about the diseases that affect coffee or rice and fail to contain the exact impact on production [20]. The exact impact on the quantity and quality of the product should be captured in such research. The other challenge is that some of the data sets may not allow the use of mobile gadgets [13]. Their complications make it hard to analyze their contents using mobile devices because of a higher screen resolution requirement.

5. Feature Extraction for Disease Identification

The images of the plants have three key features, namely, color, shape, and texture. Compared to color and texture, the shape feature cannot help find the plant's diseases [48]. Hlaing and Zaw [48] classified tomato plant disease using a combination of texture and color features. They used the Scale Invariant Feature Transform (SIFT) to find the texture information, containing details about the shape, location, and scale. Similarly, they gathered the color details from the RGB channel.

Dandawate and Kokare [49] developed an approach for the automatic detection of diseases in soybean plants. They converted the image from RGB to HSV (Hue Saturation Value) color space. Color and cluster-based methods were employed for segmentation. The SIFT method was used to detect the type of plant, based on the shape of the leaf.

Pydipati et al. [50] identified the citrus disease using color texture features along with discriminant analysis. They also employed the color cooccurrence method (CCM) to determine if hue, saturation, and intensity (HSI) color features and the statistical classification algorithms could help identify the diseased leaves. Their method achieved an accuracy of more than 0.95.

Al-bayati and Üstündağ [51] extracted only the area of the leaf affected by the disease. Furthermore, they used feature fusion which helped in feature reduction.

Image-based detection requires many resources, and the authorities should ensure they are available so that the activities are smooth. DL in general and CNN in particular have been developed to analyze multidimensional data such as images. The underlying model is based on the multilayer ANN. Nevertheless, a convolutional layer performs kernel operations over various areas of the provided image. The obtained representation is independent of the operations such as translation or rotation. These kinds of features have been proved to work better as compared to the traditional features earlier used in the detection of plant diseases [2].

6. Comparison of Performance and Result Discussion

Our current research work covers state-of-the-art plant disease recognition using AI in the last five years. We summarize a series of observations that emerge from this work in the following paragraphs:

(i) *Available Databases and Size Issue*. It is difficult to obtain leaf images for specific plant infections. Due to this fact, the sizes of the available plant data sets are very small. Only limited works have reported thousands of images for research purposes [22, 52–56]. Due to the small database size problem, a large portion of the data set is used for the training phase in most of the deep learning methods. However, very few exceptions are there, for example [57–59]. Furthermore, the available database images are collected in very constrained environmental conditions. We believe that images must be gathered in real-world conditions to make the algorithms more practical. Efficient image acquisition of leaf images is the need of the hour. If these images are captured in real-time scenarios, such databases would be warmly welcomed in the research community. In most of the recently reported works, the images captured with smart mobile devices are gaining popularity. Some single-click image systems are also introduced, but much more is supposed to be done by the researchers to automate plant disease identification algorithms. The transition of image-capturing systems to smart devices may help overcome serious issues related to database size.

(ii) *Issues with Available Feature Extraction Methods*. Performing the tasks of preprocessing, feature extraction, and segmentation plays a key role in developing a machine learning-based algorithm. Selecting the most suitable method for preprocessing and segmentation further depends on the nature of the data set. Among many techniques, one that is most suitable for a specific acquisition usually serves the purpose. We observe variability span in the reported algorithms so far under different modules. We observe somehow similar observations for various feature extraction techniques. In a nutshell, the standardization of the reported methods is yet to be fixed and achieved.

(iii) *Difficulties in Classification Module*. Plant disease automation and detection is an active area of research for a long time. Considering very few images for training and testing, highly acceptable results are reported by researchers. Many classifiers are explored by researchers in this domain. This study concluded that backpropagation neural network, SVM, and discriminant analysis (particularly linear) perform much better than others. These are then followed by Naïve Bayes, random forest, k-nearest neighbor, and multilayer perceptron. However, state-of-the-art results are much improved with recently introduced optimized deep neural networks. More proper utilization of deep convolutional neural networks can help in improving the results for large data sets.

(iv) *Limitations of Available Systems*. We argue that image analysis methods are comparatively better than the techniques that visually rate the severity of a particular disease. However, systems which are designed using these imaging techniques are not perfect. The performance of a system highly depends on the quality of the training data. In plant disease automation, it is the training images and certain extracted features, which significantly affect the performance of a system. A system trained with good quality data is trained well. However, most of the existing systems have a specific set of requirements needed to be fulfilled for a system to perform accurately. If some of these constraints are not fulfilled, the system may give inaccurate results, ultimately leading to wrong disease detection. For example, most of the DL-based methods particularly and conventional machine learning methods generally are faced with the problem of overfitting. Researchers must think of adaptive systems which are designed with more flexible requirements. Additionally, some generalized methods should be adapted which work in heterogeneous environments. For improving efficiency, in-depth knowledge of the methods and proper usage of the tools are also necessary.

(v) *Evaluation Measures*. Many measures are available to compare different models for disease classification. These measures are based on four statuses: true-positive (TP), indicating the number of infected samples correctly identified; true-negative (TN), describing the correctly identified healthy images. Similarly, false-positives (FPs) showing the number of healthy samples that have been incorrectly classified as infected ones. Lastly, false-negative (FN) represents the infected samples wrongly categorized as healthy ones. Accuracy is the ratio of the correct classifications (TP + FP) to the total number of classifications (TP + FP + TN + FN). Precision represents the ratio of the correctly identified samples as infected (TP) to the total samples identified as infected (sum of TP and FP). Similarly, recall is the ratio of TP to the actual number of infected samples (sum of TP and FN). Lastly, F-measure represents the harmonic mean of precision and recall.

(vi) *Comparison of Results*. State-of-the-art results for plants disease detection are compared and summarized for various data sets and methods in Table 1. We present a summary of the observations in the following paragraphs:

(a) Dandawate and Kokare [49] used SVM, a linear supervised learning classifier, for the classification of soybean plant diseases. Their proposed system got an average accuracy of 0.938. Al-bayati and Üstündağ [51] used SVM and ANN for the detection of plant diseases. They tested their approaches on the PlantVillage data set. Among various fruits and vegetables, strawberry got the highest precision of

Table 1: Image-based detection and recognition of diseases (year wise from 2016 to 2021).

Paper	Year	Database used	Images used	Accuracy
Atila et al. [32]	2021	PlantVillage	61,486	0.984
Reddy et al. [61]	2021	PlantVillage	54,305	0.900
Sembiring et al. [62]	2021	PlantVillage	17,641	0.972
Adit et al. [12]	2020	PlantVillage	76,000	0.980
Guo et al. [46]	2020	PlantVillage	61,486	0.992
Khan et al. [60]	2020	PlantVillage	7,733	0.978
Saleem et al. [27]	2020	PlantVillage	61,486	0.730
Ashqar and Abu-Naser [20]	2019	PlantVillage	9,000	0.994
Goncharov et al. [38]	2019	PlantVillage	54,306	0.870
Lin et al. [5]	2019	Powdery mildew data set	20	0.961
Nagasubramanian et al. [18]	2019	Hyperspectral image data set	111	0.957
Ozguven et al. [39]	2019	Sugar beet data set	155	0.955
Ramcharan et al. [7]	2019	Cassava disease data set	720	0.790
Shrivastava et al. [31]	2019	Oryza sativa database	619	0.914
Stewart et al. [25]	2019	Aerial image data set	3,000	0.780
Tian et al. [45]	2019	Apple disease image data set	640	0.994
Turkoglo et al. [15]	2019	Turkey plant data set	1,965	0.956
Verma et al. [24]	2019	IPPN phenotyping data set	345	0.904
Barbedo [6]	2018	PlantVillage augmented	50,000	0.960
Barbedo [19]	2018	PlantVillage augmented	50,000	0.980
Cap et al. [37]	2018	Saitama research center data set	60,000	0.780
Dhakal and Shakya [8]	2018	PlantVillage	54,000	0.985
Ferentinos [3]	2018	Open plant database	87,848	0.995
Rangarajan et al. [23]	2018	PlantVillage	13,262	0.950
Ubbens et al. [14]	2018	PlantVillage	18,160	0.962
Wallelign et al. [9]	2018	Soybean image data set	12,673	0.993
Wiesner-Hanks et al. [10]	2018	Cornell University data set	18,222	0.850
Amara et al. [35]	2017	PlantVillage	3700	0.997
Brahimi et al. [36]	2017	PlantVillage	14,828	0.992
Cruz et al. [30]	2017	PlantVillage	299	0.986
Durmus et al. [21]	2017	Korean tomato disease data set	5,000	0.972
Fuentes et al. [16]	2017	Tomato disease image data set	5,000	0.870
Ramcharan et al. [13]	2017	Cassava disease image data set	11,670	0.930
Wang et al. [1]	2017	PlantVillage	2,265	0.904
Ale et al. [43]	2016	Real-world plant disease data set	21,917	0.980
Sladojevic et al. [22]	2016	Images collected from Internet	33,469	0.982

0.991. Moreover, the results improved with the fusion of classifiers. In terms of recall, the best results were also obtained for strawberry (0.959). It was observed that the fusion improved the results for recall, just like the case for precision. As expected, the best F-measure was observed for strawberry (0.975). Furthermore, the F-measure results were better while using the fusion of classifiers

(b) Saleem et al. [27] developed three DL architectures to detect various plant diseases, namely, Single Shot Multibox Detector (SSD), Faster Region-based CNN (Faster-RCNN), and Region-based Fully Convolutional Networks (RFCN). SSD performed all computations in one network and used smaller convolution filters, such as 4×4 and 8×8. Similarly, in Faster-RCNN, object detection is carried out at two different stages in contrast to SSD. In the first step, also known as the region proposal network stage, the images are processed to come up with the region proposals using feature extractors. Consequently, the features help in determining the class-specific proposals for each of the intermediate convolutional layers. Later, during the second step, the characteristics of the same intermediate layer of an image are detected. RFCN is very similar to Faster-RCNN but does not include fully convolutional layers after the region of interest (ROI) pooling

(c) Hernandez and Lopez [26] used Stochastic Gradient Descent (SGD) to train a softmax layer. The convergence was achieved after 60 epochs. SGD got an accuracy of 0.96 and an F1-score of 0.96 as well, while fine-tuning VGG16, a state-of-the-art image classifier, using Bayesian learning algorithms. Monte Carlo (MC) dropout was able to get both accuracy and an F1-score of 0.94. Lastly, Stochastic gradient Langevin dynamics (SGLD) achieved an accuracy of 0.89, whereas an F1-score of 0.88. SGD achieved an accuracy of 0.49 for out-of-sample classification. On the other hand, Monte Carlo (MC) dropout was able to achieve an accuracy of 0.55. It is pertinent to note that MC dropout can be used both during training as well as testing. SGLD got an accuracy of just 0.54, whereas an F1-score of 0.15 for out-of-sample classification. Kamilaris and Prenafeta-Boldu [2] report that many papers related to plant and leave disease detection and employing DL techniques report excellent results, i.e., an accuracy greater than 0.95 or F1-score greater than 0.92. The reason is probably the peculiar characteristics of (sick) leaves/-plants and fruits in the images

(d) Brahimi et al. [36] used AlexNet and GoogLeNet on the PlantVillage data set and obtained an accuracy of 0.99. Similarly, Dhakal and Shakya [8] also used DL for the PlantVillage data set. They considered four class labels: Bacterial Spot, Yellow Leaf Curl Virus, Late Blight, and Healthy Leaf. They were able to get a test accuracy of 0.956. Khan et al. [60] proposed a CNN-based model and worked on five different diseases found in the PlantVillage data set. The method got an accuracy of 0.978. Wallelign et al. [9] used CNN to identify the diseases found in the soybean plant. They used 12,673 images from the PlantVillage data set. The data set was unbalanced and contained 6234 instances of healthy leaf, 3565 images of septorial leaf blight, 2023 images of frogeye leaf spot, whereas just 851 images of downy mildew. They used three convolutional layers, each followed by a max-pooling layer. ReLU activation function was employed in each of the convolutional layers and the fully connected (FC) layer. Their results showed that it is much better to use color images than grayscale or segmented ones. Their approach was able to get a test accuracy of 0.993 after dropout. Nevertheless, the best model got an F1-score of 0.99

(e) Reddy et al. [61] used CNN to identify the plant species while using color images of leaves. They used five data sets, including PlantVillage, Leaf Snap, UCI leaf, Flavia, and Swedish leaf. Their model uses four convolutional layers, two FC layers, and a softmax layer. The proposed method gets perfect accuracy for three data sets, Flavia, Swedish leaf, and UCI leaf. Similarly, an accuracy of 0.980 is obtained for Leaf Snap, whereas 0.900 is obtained for the PlantVillage data set. Sembiring et al. [62] developed a concise CNN for detecting the diseases in the tomato plant. They also used the PlantVillage data set and got an accuracy of 0.972. The accuracy is slightly less than the one achieved by VGG16 (0.983). Nevertheless, the proposed method required less than one-fourth of the time needed by VGG16. Table 1 provides a comparison of different approaches

7. Conclusion and Future Works

In this paper, we have discussed how ML in general and DL in particular have helped identify the diseases in plants. If the diseases are not correctly identified, they affect the crop yield and ultimately result in long-term issues, such as global warming and even famine. The proposed work summarizes multiple studies regarding plant disease automation and identification through different ML methods. The proposed manuscript also shows well-acceptance of a range of CV methods in this domain, making it a wide area of research to be explored in the near future. Some points are summarized in the following paragraphs, which may help to improve and enhance the current state-of-the-art and give researchers some potential ideas to further explore the field in the future.

(i) *Disease Stage Identification*. Disease stage identification is one of the main areas to be explored regarding plant disease identification. Each disease has several stages. Most of the researchers only focused their work on the type of disease identification, but none of these works target a particular disease stage identification. Additionally, such systems must possess the capability to suggest some measures depending particularly on specific disease stages. Identification of disease forecasting will help agriculturists to take proper actions and precautions to reduce damage percentage.

(ii) *Quantification of a Disease*. Another exciting area to be explored is the quantification of a particular disease. Although there has been much work done in this field, very few researchers have identified the extent of the damage caused by the disease. They can help a lot since remedial actions could be taken according to the severity of the disease. Such kind of quantification will detect the infected proportion of a particular culture with some disease. This research perspective is vital as the number of pesticides can be controlled. Normally, the farmers apply chemicals to cure diseases without any prior analysis or quantification. Such kind of practice is extremely harmful to human health. Developing an efficient image processing application will help to determine if specific chemicals are needed or not.

(iii) *Mobile and Online Applications*. Literature reported several solutions regarding applications of disease identification. However, few of the portals and mobile applications are publicly available and

accessible online. Some of these applications are Assess Software and Leaf Doctor, which are publicly available for use. However, these applications work on leave images with a flat and only black background. Therefore, such kinds of online systems and applications are highly needed for plant diseases' recognition. The availability of these types of software will help farmers to identify a particular kind of disease. Such kinds of software may be used to get analysis reports which can be sent to a disease expert for getting some suggestions.

(iv) *Exploring Transfer Learning to Increase Data Size*. Similarly, noting the current trends of the developments in CV, which are moving very fast towards DL methods, for plant disease detection is not very satisfactory. Given the difficulty of the data, particularly the training stage, the best option to be explored is transfer learning. For investigating knowledge transfer, a heterogeneous domain strategy may be adapted. Considering automatic plant disease identification, the keywords which may be explored are LSTMs, optical flow frames, temporal pooling, and 3D convolution. The last point that must be remembered is that better and carefully engineered methods are needed for further exploring this area. For example, the case of data augmentation may be further investigated.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

The authors gratefully acknowledge Qassim University, represented by the Deanship of Scientific Research, on the financial support for this research under the number (10269-coc-2020-1-3-I) during the academic year 1441 AH/2020 AD.

References

[1] G. Wang, Y. Sun, and J. Wang, "Automatic image-based plant disease severity estimation using deep learning," *Computational Intelligence and Neuroscience*, vol. 2017, Article ID 2917536, 8 pages, 2017.

[2] A. Kamilaris and F. X. Prenafeta-Boldú, "Deep learning in agriculture: a survey," *Computers and Electronics in Agriculture*, vol. 147, pp. 70–90, 2018.

[3] K. P. Ferentinos, "Deep learning models for plant disease detection and diagnosis," *Computers and Electronics in Agriculture*, vol. 145, pp. 311–318, 2018.

[4] R. I. Hasan, S. M. Yusuf, and L. Alzubaidi, "Review of the state of the art of deep learning for plant diseases: a broad analysis and discussion," *Plants*, vol. 9, no. 10, article 1302, 2020.

[5] K. Lin, L. Gong, Y. Huang, C. Liu, and J. Pan, "Deep learning-based segmentation and quantification of cucumber powdery mildew using convolutional neural network," *Frontiers in Plant Science*, vol. 10, p. 155, 2019.

[6] J. G. A. Barbedo, "Factors influencing the use of deep learning for plant disease recognition," *Biosystems Engineering*, vol. 172, pp. 84–91, 2018.

[7] A. Ramcharan, P. McCloskey, K. Baranowski et al., "A mobile-based deep learning model for cassava disease diagnosis," *Frontiers in Plant Science*, vol. 10, p. 272, 2019.

[8] A. Dhakal and S. Shakya, "Image-based plant disease detection with deep learning," *International Journal of Computer Trends and Technology*, vol. 61, no. 1, pp. 26–29, 2018.

[9] S. Wallelign, M. Polceanu, and C. Buche, "Soybean plant disease identification using convolutional neural network," in *Proc. Thirty-First International Florida Artificial Intelligence Research Society Conference (FLAIRS-31)*, pp. 146–151, Melbourne, FL, USA, 2018.

[10] T. Wiesner-Hanks, E. L. Stewart, N. Kaczmar et al., "Image set for deep learning: field images of maize annotated with disease symptoms," *BMC Research Notes*, vol. 11, no. 1, p. 440, 2018.

[11] K. Nagpal, D. Foote, Y. Liu et al., "Publisher correction: development and validation of a deep learning algorithm for improving Gleason scoring of prostate cancer," *NPJ Digital Medicine*, vol. 2, no. 1, p. 113, 2019.

[12] V. V. Adit, C. V. Rubesh, S. S. Bharathi, G. Santhiya, and R. Anuradha, "A Comparison of Deep Learning Algorithms for Plant Disease Classification," in *Advances in Cybernetics, Cognition, and Machine Learning for Communication Technologies, Lecture Notes in Electrical Engineering*, vol. 643, pp. 153–161, Springer, Singapore, 2020.

[13] A. Ramcharan, K. Baranowski, P. McCloskey, B. Ahmed, J. Legg, and D. P. Hughes, "Deep learning for image-based cassava disease detection," *Frontiers in Plant Science*, vol. 8, article 1852, 2017.

[14] S. Verma, A. Chug, A. P. Singh, S. Sharma, and P. Rajvanshi, "Deep Learning-Based Mobile Application for Plant Disease Diagnosis: A Proof of Concept With a Case Study on Tomato Plant," in *Applications of Image Processing and Soft Computing Systems in Agriculture*, pp. 242–271, IGI Global, 2019.

[15] M. Türkoğlu and D. Hanbay, "Plant disease and pest detection using deep learning-based features," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 27, no. 3, pp. 1636–1651, 2019.

[16] A. Fuentes, S. Yoon, S. Kim, and D. Park, "A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition," *Sensors*, vol. 17, no. 9, article 2022, 2017.

[17] M. Loey, A. ElSawy, and M. Afify, "Deep learning in plant diseases detection for agricultural crops," *International Journal of Service Science, Management, Engineering, and Technology*, vol. 11, no. 2, pp. 41–58, 2020.

[18] K. Nagasubramanian, S. Jones, A. K. Singh, S. Sarkar, A. Singh, and B. Ganapathysubramanian, "Plant disease identification using explainable 3D deep learning on hyperspectral images," *Plant Methods*, vol. 15, no. 1, p. 98, 2019.

[19] J. Barbedo, "Impact of dataset size and variety on the effectiveness of deep learning and transfer learning for plant disease classification," *Computers and Electronics in Agriculture*, vol. 153, pp. 46–53, 2018.

[20] B. Ashqar and S. Abu-Naser, "Image-based tomato leaves diseases detection using deep learning," *International Journal of Engineering Research*, vol. 2, no. 12, pp. 10–16, 2019.

[21] H. Durmus, E. O. Gunes, and M. Kirci, "Disease detection on the leaves of the tomato plants by using deep learning," in *Proc.*

6th International Conference on Agro-Geoinformatics, pp. 1–5, Fairfax, VA, 2017.

[22] S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk, and D. Stefanovic, "Deep neural networks based recognition of plant diseases by leaf image classification," *Computational Intelligence and Neuroscience*, vol. 2016, Article ID 3289801, 11 pages, 2016.

[23] A. Rangarajan, R. Purushothaman, and A. Ramesh, "Tomato crop disease classification using pre-trained deep learning algorithm," *Procedia Computer Science*, vol. 133, pp. 1040–1047, 2018.

[24] J. Ubbens and I. Stavness, "Corrigendum: deep plant phenomics: a deep learning platform for complex plant phenotyping tasks," *Frontiers in Plant Science*, vol. 8, article 2245, 2018.

[25] E. L. Stewart, T. Wiesner-Hanks, N. Kaczmar et al., "Quantitative phenotyping of Northern Leaf Blight in UAV images using deep learning," *Remote Sensing*, vol. 11, no. 19, article 2209, 2019.

[26] S. Hernández and J. L. López, "Uncertainty quantification for plant disease detection using Bayesian deep learning," *Applied Soft Computing*, vol. 96, article 106597, 2020.

[27] M. H. Saleem, S. Khanchi, J. Potgieter, and K. M. Arif, "Image-based plant disease identification by deep learning meta-architectures," *Plants*, vol. 9, no. 11, article 1451, 2020.

[28] J. Li, J. Jia, and D. Xu, "Unsupervised representation learning of image-based plant disease with deep convolutional generative adversarial networks," in *Proc. 37th Chinese Control Conference (CCC)*, pp. 9159–9163, Wuhan, China, 2018.

[29] A. Singh, B. Ganapathysubramanian, S. Sarkar, and A. Singh, "Deep learning for plant stress phenotyping: trends and future perspectives," *Trends in Plant Science*, vol. 23, no. 10, pp. 883–898, 2018.

[30] A. Cruz, A. Luvisi, L. De Bellis, and Y. Ampatzidis, "Vision-based plant disease detection system using transfer and deep learning," in *Proc. 2017 ASABE Annual International Meeting*, Spokane, WA, USA, 2017.

[31] V. Shrivastava, M. Pradhan, S. Minz, and M. Thakur, "Rice Plant disease classification using transfer learning of deep convolution neural network," *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XLII-3/W6, pp. 631–635, 2019.

[32] Ü. Atila, M. Uçar, K. Akyol, and E. Uçar, "Plant leaf disease classification using EfficientNet deep learning model," *Ecological Informatics*, vol. 61, article 101182, 2021.

[33] G. Geetharamani and J. A. Pandian, "Identification of plant leaf diseases using a nine-layer deep convolutional neural network," *Computers & Electrical Engineering*, vol. 76, pp. 323–338, 2019.

[34] D. P. Hughes and M. Salathe, "An open access repository of images on plant health to enable the development of mobile disease diagnostics through machine learning and crowdsourcing," 2015, https://arxiv.org/abs/1511.08060.

[35] J. Amara, B. Bouaziz, and A. Algergawy, "A deep learning-based approach for banana leaf diseases classification," in *Datenbanksysteme für Business*, Technologie und Web, Stuttgart, 2017.

[36] M. Brahimi, K. Boukhalfa, and A. Moussaoui, "Deep learning for tomato diseases: classification and symptoms visualization," *Applied Artificial Intelligence*, vol. 31, no. 4, pp. 299–315, 2017.

[37] H. Q. Cap, K. Suwa, E. Fujita, S. Kagiwada, H. Uga, and H. Iyatomi, "A deep learning approach for on-site plant leaf detection," in *Proc. IEEE 14th International Colloquium on Signal Processing & Its Applications (CSPA)*, pp. 118–122, Batu Feringghi, 2018.

[38] P. Goncharov, G. Ososkov, A. Nechaevskiy, and I. Nestsiarenia, "Disease detection on the plant leaves by deep learning," in *Selected Papers from the XX International Conference on Neuroinformatics, in Advances in Neural Computation, Machine Learning, and Cognitive Research II*, pp. 151–159, Moscow, Russia, 2019.

[39] M. M. Ozguven and K. Adem, "Automatic detection and classification of leaf spot disease in sugar beet using deep learning algorithms," *Physica A: Statistical Mechanics and its Applications*, vol. 535, article 122537, 2019.

[40] J. G. M. Esgario, R. A. Krohling, and J. A. Ventura, "Deep learning for classification and severity estimation of coffee leaf biotic stress," *Computers and Electronics in Agriculture*, vol. 169, article 105162, 2020.

[41] A. Lowe, N. Harrison, and A. P. French, "Hyperspectral image analysis techniques for the detection and classification of the early onset of plant disease and stress," *Plant Methods*, vol. 13, no. 1, p. 80, 2017.

[42] A. Mahlein, "Plant disease detection by imaging sensors – parallels and specific demands for precision agriculture and plant phenotyping," *Plant Disease*, vol. 100, no. 2, pp. 241–251, 2016.

[43] L. Ale, A. Sheta, L. Li, Y. Wang, and N. Zhang, "Deep learning based plant disease detection for smart agriculture," in *Proc. 2019 IEEE Globecom Workshops*, pp. 1–6, Waikoloa, HI, USA, 2019.

[44] R. Chalapathy and S. Chawla, "Deep learning for anomaly detection: a survey," 2019, 2020, https://arxiv.org/abs/1901.03407.

[45] Y. Tian, G. Yang, Z. Wang, E. Li, and Z. Liang, "Detection of apple lesions in orchards based on deep learning methods of cyclegan and yolov3-dense," *Journal of Sensors*, vol. 2019, Article ID 7630926, 13 pages, 2019.

[46] Y. Guo, J. Zhang, C. Yin et al., "Plant disease identification based on deep learning algorithm in smart farming," *Discrete Dynamics in Nature and Society*, vol. 2020, Article ID 2479172, 11 pages, 2020.

[47] M. Arsenovic, M. Karanovic, S. Sladojevic, A. Anderla, and D. Stefanovic, "Solving current limitations of deep learning based approaches for plant disease detection," *Symmetry*, vol. 11, no. 7, p. 939, 2019.

[48] C. S. Hlaing and S. M. M. Zaw, "Tomato plant diseases classification using statistical texture feature and color feature," in *Proc. IEEE/ACIS 17th International Conference on Computer and Information Science*, Singapore, 2018.

[49] Y. Dandawate and R. Kokare, "An automated approach for classification of plant diseases towards development of futuristic Decision Support System in Indian perspective," in *Proc. IEEE International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Kochi, India, 2015.

[50] R. Pydipati, T. F. Burks, and W. S. Lee, "Identification of citrus disease using color texture features and discriminant analysis," *Computers and Electronics in Agriculture*, vol. 52, no. 1–2, pp. 49–59, 2006.

[51] J. S. H. Al-bayati and B. B. Üstündağ, "Evolutionary feature optimization for plant leaf disease detection by deep neural networks," *International Journal of Computational Intelligence Systems*, vol. 13, no. 1, pp. 12–23, 2020.

[52] A. Meunkaewjinda, P. Kumsawat, K. Attakitmongcol, and A. Srikaew, "Grape leaf disease detection from color imagery using hybrid intelligent system," in *Proc. IEEE 5th International Conference on Electrical engineering/electronics, Computer, Telecommunications and Information technology ECTI-CON*, pp. 513–516, Krabi, 2008.

[53] R. D. L. Pires, D. N. Gonçalves, J. P. M. Oruê et al., "Local descriptors for soybean disease recognition," *Computers and Electronics in Agriculture*, vol. 125, pp. 48–55, 2016.

[54] S. Shrivastava, S. K. Singh, and D. S. Hooda, "Color sensing and image processing-based automatic soybean plant foliar disease severity detection and estimation," *Multimedia Tools and Applications*, vol. 74, no. 24, pp. 11467–11484, 2015.

[55] M. Schikora, B. Neupane, S. Madhogaria et al., "An image classification approach to analyze the suppression of plant immunity by the human pathogen Salmonella Typhimurium," *BMC Bioinformatics*, vol. 13, no. 1, p. 171, 2012.

[56] J. G. A. Barbedo, L. V. Koenigkan, and T. T. Santos, "Identifying multiple plant diseases using digital image processing," *Biosystems Engineering*, vol. 147, pp. 104–116, 2016.

[57] D. Majumdar, D. K. Kole, A. Chakraborty, and D. D. Majumder, "An integrated digital image analysis system for detection, recognition and diagnosis of disease in wheat leaves," in *Proc. 3rd international ACM Symposium on Women in Computing and Informatics*, pp. 400–405, New York, USA, 2015.

[58] S. Kai, L. Zhikun, S. Hang, and G. Chunhong, "A research of maize disease image recognition of corn based on BP networks," in *Proc. of 3rd IEEE International conference on measuring technology and mechatronics automation (ICMTMA)*, pp. 246–249, Shanghai, China, 2011.

[59] S. N. Ghaiwat and P. Arora, "Cotton Leaf Disease Detection by Feature Extraction," in *Research Advances in the Integration of Big Data and Smart Computing*, pp. 89–104, IGI Global, 2016.

[60] M. A. Khan, T. Akram, M. Sharif, and T. Saba, "Fruits diseases classification: exploiting a hierarchical framework for deep features fusion and selection," *Multimedia Tools and Applications*, vol. 79, no. 35-36, pp. 25763–25783, 2020.

[61] S. R. G. Reddy, G. P. S. Varma, and R. L. Davuluri, "Optimized convolutional neural network model for plant species identification from leaf images using computer vision," *International Journal of Speech Technology*, 2021.

[62] A. Sembiring, Y. Away, F. Arnia, and R. Muharar, "Development of concise convolutional neural network for tomato plant disease classification based on leaf images," *Journal of Physics: Conference Series*, vol. 1845, article 012009, 2021.

Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 9912626, 17 pages
https://doi.org/10.1155/2021/9912626

Research Article

C^2S^2-LOOP: Circular Chessboard-Based Secure Source Location Privacy Model Using ECC-ALO in WSN

Naveed Jan,[1] **Sarmadullah Khan**,[2] **Ali H. Al-Bayatti**,[2] **Madini O. Alassafi**,[3] **Ahmed S. Alfakeeh**,[3] **and Mohammad A. Alqarni**[4]

[1]*Department of Information Engineering Technology, Shuhada-e-APS University of Technology Nowshera, Pakistan*
[2]*School of Computer Science and Informatics, De Montfort University, The Gateway, Leicester LE1 9BH, UK*
[3]*Faculty of Computing and Information Technology, King Abdul Aziz University, Jeddah 21589, Saudi Arabia*
[4]*College of Computer Science and Engineering, University of Jeddah, Jeddah 23218, Saudi Arabia*

Correspondence should be addressed to Naveed Jan; jannaveed@gmail.com

Received 11 March 2021; Accepted 7 May 2021; Published 25 May 2021

Academic Editor: Sungchang Lee

Source location privacy (SLP) is a serious issue in wireless sensor networks (WSN) since Eavesdroppers tries to determine the source location. Hunting Animals in Forest is considered as an example for SLP. Many conventional schemes have been proposed for SLP in WSN, namely, Random Walk Routing, and Fake Messages Transmission, which cause critical issues (less safety period, packet delivery latency, and high energy consumption). Furthermore, the security analysis is not properly investigated in any previous work. In this paper, we propose a new model called the circular chessboard-based secure source location privacy model (C^2S^2-LOOP) with the following tasks: key generation, network topology management (clustering), intercluster routing (travel plan), and data packets encryption. All sensor nodes are deployed in a circular chessboard (Circular Field) and the key generation ($\boldsymbol{PU}_K, \boldsymbol{SE}_K$) is invoked using elliptic curve cryptography (ECC) with Ant Lion Optimization algorithm, which mitigate the issues of conventional ECC. Then, the network topology is managed using clustering where residual energy of the nodes is used for Cluster Head (CH) selection. Intercluster routing is implemented using packet traversing using clockwise and anticlockwise directions, which are mainly concerned with establishing a secure route between the source to the destination node. To ensure data security, we present the Chaotic Artificial Neural Network (C-ANN) in which encryption is executed. Assume that CH near to the source node has a high trust value, then it traverses (clock-wise) real packets towards sink node and similarly in the left side region (anticlockwise), fake packets are transmitted. Network simulations (OMNeT++) are evaluated and compared with the previous approaches, and finally, our proposed scheme concludes that it maintains not only source node location privacy (large safety period) and also reduces energy consumption by more than 40% and latency by more than 35%.

1. Introduction

WSN is a kind of wireless network that is comprised by a grouping of sensor nodes that can sense, collect, and broadcast information to adjacent nodes through direct communication or relay nodes [1, 2]. Source location privacy is an emerging area in event-driven applications such as military and wildlife tracking applications. At any time, the sensor node becomes a source node, and thus when event is detected by a node, the message for a specific event is transmitted to the sink node [3–5]. In many cases, the source node can be easily located through the shared wireless transmission medium. Hence, the objective of this source location privacy is to protect the source node location [6]. The safety period is the main concern in SLP, i.e., how long until the node is discovered [7]. To address this concern, several methods have been proposed for SLP in the sensor network. SLP can be classified into the following: flooding-based approach, phantom routing, and fake/dummy messaging approach. Most of the authors have used Random Walk Routing for source location in which nodes are selected in random and adversary/attacker cannot find the source location [8, 9]. Previously,

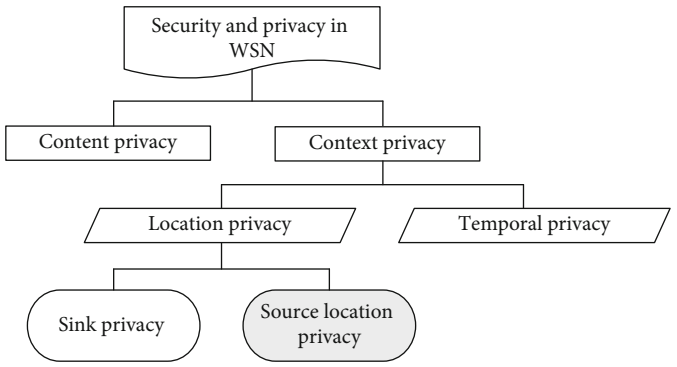


Figure 1: Security and privacy in WSNs.

dummy message injection method is presented which improves the location privacy but it causes more overhead among sensor nodes [10].

Other approaches are geographic routing, cyclic entrapment, in-network location anonymization, cross-layer routing, separate path routing, network coding, and limit node detectability. These approaches are consuming more energy, large delay, and decreased safety period [11–13]. Source node location privacy becomes important for message transmission, but confidentiality of data packets must be ensured through encryption strategy, which is much more necessary to adequately address the problem of source location privacy. Then, we discussed the concepts [14] include safety period, unlikability, identity privacy, timing privacy, route privacy, and contextual privacy. However, context privacy is used for source location identification. Figure 1 indicates the security and privacy of WSN, while Figure 2 shows an example of animal tracking providing a good example of SLP in the sensor network.

1.1. Source Location Privacy. SLP is an important issue and enormous works have been undertaken in this field. A number of attacks related to SLP in WSN. Privacy of source node location is ensured with the use of context protection. Adversaries may use packet tracing to determine the source node location, while the data is being sent from the source to the sink node. However, two adversaries are involved in SLP that is as follows:

(i) *Local Attacker.* This type of attacker can only eavesdrop within range and back-trace the routing path until it reaches the source node

(ii) *Global Attacker.* This type of attacker learns about the whole network traffic and who can make use of temporal consistency and the correlation between packets. Hence, it makes use of the following:

(a) Received signal strength indicator

(b) Direction of the packets

(c) Inter-packets time

(d) Packet's occurrence time

Furthermore, the global attacker is near to the sink node of the sensor network and tries to find out the location of the

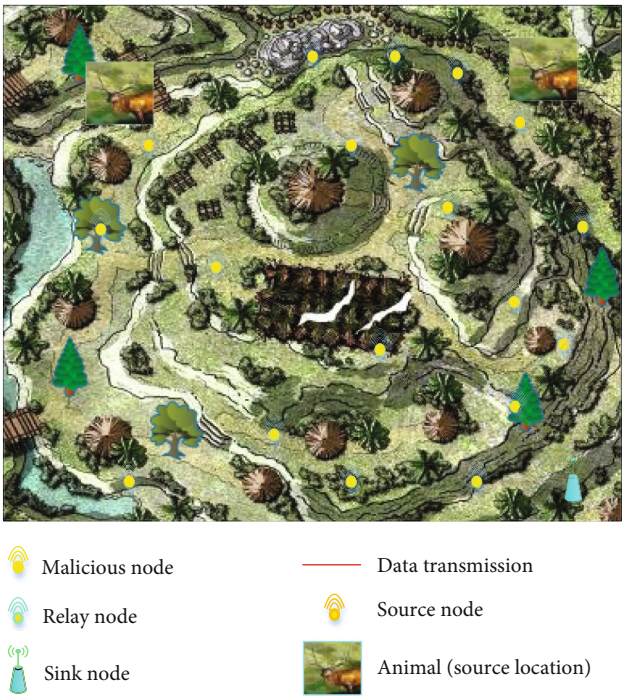


Figure 2: Example for SLP in sensor network.

source node of the message [15]. Security is a challenging operation in SLP. Obtaining some parameters such as safety period may improve location privacy, but it may lead to severe issue on the other side such as residual energy. Hence, a balanced trade-off between location privacy and energy consumption is more desirable to obtain high QoS performance and fewer computations in SLP. However, modeling a secure and robust scheme to address these parameters is a difficult process. Very popular routing protocols are required and we need to be augmented these routing protocols with a novel scheme. Energy consumption is still an important issue in accomplishing SLP. To address these issues and show better performance (minimum energy consumption, minimum latency, and maximum safety period) for the source location privacy and easy and flexible network topology management, C^2S^2-LOOP provides an efficient solution.

1.2. Contributions and Organizations. The main contributions of this paper are as follows:

(i) We propose a circular chessboard based secure source routing (C^2S^2-LOOP) to protect a source's location privacy against an attacker

(ii) We propose an elliptic curve cryptography-based ant lion optimization algorithm for key generation in which public and secret keys are generated. This solves issues of the traditional ECC algorithm

(iii) Equal number of nodes are deployed in each cluster and CH is selected based on residual energy

(iv) Transmission of packets from source to sink is done using clockwise and anticlockwise directions for real and fake packets depending on the value of random number R_n

(v) Before packets transmission from the source node, it will be encrypted using a chaotic artificial neural network and then forwarded to adjacent nodes

(vi) We evaluate the performance by a comprehensive set of experimentations for the proposed vs. previous approaches, and our proposed scheme demonstrates the effectiveness and improvement to source location privacy preserving in terms of energy consumption, latency, safety period, and network lifetime

The remainder of this paper is as follows: in Section II, we present the state of the art in the field of SLP in WSN. In Section III, we describe the problems and effects of analysis for protecting source location privacy. In Section IV, we present our proposed system model where we elaborate the proposed approach in further detail by presenting subsections. An experiment of the proposed approach is concentrated in Section V, and also, the simulation results obtained by the proposed as well as the previous approaches are described. Section VI concludes our proposed approach, and future directions of this proposed approach are highlighted in Section VII.

2. State of the Art

Over the last few decades, several vulnerabilities have been determined in terms of SLP in WSN. Protecting source location privacy is not only abundant for preserving source location but also some other metrics have been presented in the current literature. In this section, we present this state of the art.

The authors in [16] presented a dynamic SLP algorithm to achieve a high level of SLP with a high number of messages. The proposed dynamic hybrid approach uses a random walk procedure for packet transmission to the base station and a fake source allocation strategy for energy consumption reduction. A dynamic SLP approach is more energy-efficient, but it is suitable for the deployment of WSN environments. It increases communication overhead since the whole network is dynamic in nature, and also, the hybrid approach is the combination of fake source nodes and random walks, which also increases network overhead and delay. In [17], a differentially private framework-based differential privacy was presented to protect the location of event occurrence. Privacy of event occurrence location is required that requires that an adversary must not able to distinguish between real and dummy traffic messages. The proposed approach produces reduced sensitivity to a single node transmission, which is achieved through real dummy and cumulative reporting to the same event (fake/real events). The first component mechanism generates dummy events to protect source location privacy in which the attacker can easily find the source location by receiving more dummy messages. The authors in [18] proposed a ring-loop routing for a source location privacy scheme in IoT-assisted WSN. It increases the backtracking time of the adversary; a confounding time-domain transmission was introduced to transmit real packets. In addition to Phantom nodes and fake packets transmission, a confounding ring was also utilized to protect the source location privacy. Firstly, real packets are forwarded in the basic ring using confounding time-domain transmission, and secondly, phantom nodes forward using confounding with the shortest path. The proposed scheme is suitable for a local adversary, but it is not suitable for global attackers and it follows the basic ring for packet transmission to the sink node.

A multiobjective optimization model was proposed for SLP [19], which is based on two criterions such as pareto efficiency (selection either) and confirm efficiency. A genetic algorithm (GA) was employed to generate Pare-to-Optimal schedules. In addition, two functions were derived in which the main purpose is to minimize total slot usage and also reduces coupled slot usage, which follows the time division multiple access (TDMA) procedure. It directs the predicted attacker path away from the source node. Its advantage is that near-optimal capture radio coupled with path creation that causes no extra messages overhead. This approach was led to high energy consumption and latency. The authors in [20] designed a new protocol called redundancy-based convergence based preserved source location privacy (RBCPSLP) technique in energy harvesting WSN. It improves the privacy preservation by creates as many interference sources. A branch convergence-based SLP preserving strategy was proposed, and hence, routes merge into few routes before arrival at the sink node. In this work, the route is not effective to preserve SLP.

A trace cost-based SLP protection (TCSLP) scheme was proposed in WSN [21]. Initially, authors built a phantom area, which consists of phantom nodes far away from the source node. Then, two routing schemes were integrated (shortest path routing and random walking) for packets transmission, which improves source node location privacy. Phantom node may not be at the right place of the network, which results in the reduction of privacy levels as well as an increment in energy consumption. A new source location privacy protection scheme was proposed in [22], which is implemented in a more practical adversarial model, i.e., smart adversary. This scheme was implemented under both local and global attackers. In order to defend against these adversaries, the authors proposed a lightweight message sharing approach, and then, each message is mapped to a set of shares. Message sharing is not more reliable and scalable for large-scale networks. Energy-efficient and optimum

routing is a critical and general issue in WSN. An improved version of routing for WSN was proposed over cluster-based WSN [23]. Initially, a network is divided into number cells, and then, GA is applied to find an optimum number of nodes. To increase the speed of clustering, the *k*-means algorithm was proposed. Both GA and *k*-means clustering are time-consuming processes. In [24], the authors presented a dynamic routing scheme for source node location protection. It aims at improving paths for data transmission, and the proposed dynamic routing scheme aims to choose maximum paths for data transmission. Firstly, it selects an initial node from the boundary of the network and every packet traverse by a greedy route and a directed route to the sink node. It causes high complexity and high energy consumption. The authors in [25] focused on SLP by introduced suitable modifications to sensor routing protocols that make it hard for an adversary to backtrack the origin of the source node. This paper jointly considers the issues of energy consumption and location privacy in WSN. The drawbacks of this approach are to lack of security and privacy leakage.

In [26], all-direction routing algorithm (ARR) was proposed for SLP which against parasitic WSN. In ARR, agent nodes are randomly selected in directions by the source nodes based on local decisions rather than knowing the whole topology of the networks. In this work, proper agent node selection is difficult, and it is difficult for agent to transmit packets from the source node to the destination agent node. In [27], the authors proposed two novel methods such as angle protocol for source location (APS) and EAPS (enhanced protocol for source location). These methods adjust dynamically emission radius in routing. APS protocol provides geographically dispersed phantom source nodes and utilizes the energy from energy-abundant regions to create the path. In this work, the dynamical adjustment of node emission radius is complex. Koh et al. [28] have proposed multipath routing named optimal privacy-enhancing routing algorithm (OPERA). The authors have considered global adversary for both lossy and lossless observations that used Bayesian Maximum Posteriori (MAP) estimation scheme. The biggest strength of OPERA is that it provides a good trade-off between energy consumption and privacy but leads to high computational complexity. A novel tree-based diversionary routing scheme was proposed [29] to preserve SLP based on hide-and-seek strategy. The proposed tree-based routing provides strong resistance to reverse trace of the adversary and also resistance to direction-oriented attack. The proposed tree-based routing scheme utilizes remaining energy only at different remote regions and creates diversionary routes as more and then sent to one route in regions near to sink node. A list of nomenclature is shown in Table 1

A confused arc-based SLP scheme was proposed in [30], which is implemented in WSN assisted Internet of Things. The main scope of this paper is to finish packets transmission via random walk based on the node direction. In this scheme, nodes select their next relay nodes in a particular communication range. The proposed scheme is compared with the ring routing, which reduces energy consumption and delays at the expense of safety time for WSN. In this work, the safety period is very less for packet transmission. On the other hand, a *K*-means clustering scheme based on SLP was proposed in WSN enabled IoT [31]. To protect the source location, fake source nodes are deployed in the network for sensing dummy messages. To increase the safety period, *k* -means clustering was proposed, which forms clusters. Fake packets are transmitted via fake source nodes to reach the destination node. This scheme is failed to increase the safety period and reduce the latency at minor energy consumption, but lack of security.

Table 1: List of nomenclature.

Symbol	Meaning
$C_1, C_2, C_3 \cdots$	No. of circular fields
$C_{ID1}, C_{ID2}, C_{ID3} \cdots$	Circular identifier
a	Area of circular field
$N(SN) \in C$	No. of nodes in each circular field
SN_i	Sensor nodes
S_{INK}	Sink node
HL_p	Hello packet
$c_1, c_2, c_3 \cdots$	No. of clusters
CH	Cluster head
CH_C	Current CH
CH_l	New CH
c_M	Cluster member
S_T	Sensing time
A_T	Aggregation time
S_{ps}	Packet size for sensor node
E_R	Sensor residual energy
P_T	Plain text
C_T	Ciphertext
T_S	Time stamp
PSRN	Pseudorandom number
x, y	Sensor position
S	Source node
SE_k	Private key
PU_k	Public key
X, Y, a, b	Domain parameters
BP	Base point
p	Prime number
MAX_E_R	Node with maximum residual energy

3. Problem Description

We firstly evaluate the effect of overheads (routing and communication between nodes) during SLP using different methods (random walk, fake message transmission, dummy message injection, and so on). Then, we investigate the effect of energy consumption and also considered some optimal ethics for introducing a new scheme for protecting source location information [32]. In [33], a novel chess-board

alternation (CBA) was proposed for SLP against global attacker. Initially, the network is partitioned into two sets and each set operates in an alternating fashion. This CBA has obtained perfect privacy, but it decreases energy cost up to 50% and transmission delay by more than 40%. This method follows a random walk procedure, which is a simple method that increases delay when the probability of adjacent random walk nodes is high. Global attacker can easily capture source node location due to random walk routing (node direction and RSSI). In [34], grid-based cluster methodology was proposed in which the network is divided into a number of clusters. In a grid-based clustered environment, three techniques were proposed such as Dynamic Tree (DT), Dynamic Shortest Path Tree (DSP), and Hybrid Scheme. The proposed hybrid scheme used dummy messages to mislead the adversary since it tracks the source node location, which wastes the energy of sensor nodes. A large packet delay is due to the process of three schemes. Cluster head selection is not effective since it only considers the residual energy of a node. It leads to cluster-based SLP is not effective. A two-phase routing was proposed for SLP [35]. The basic idea behind this routing is to consume less amount of energy with the use of random walks and escape angle (total sequitur angle $2\alpha_1$ and escape angle α_1). Energy consumption of a node is high since it uses multiple virtual source nodes, and also, this routing scheme delivers the packet to the base station through random-multipaths, and thus, it utilizes an excess amount of energy for transmission to the sink node. A dynamic routing scheme was proposed in [36], which firstly chosen an initial node from the network boundary and every packet is traverse using greedy routing and then used directed route before reaching to the sink node. Event packet is encrypted using a secret key, which is shared between the sink node and the source node. Packets are forwarded through boundary nodes, which consumes more time (number of hops) for packet transmission. In this work, a strong security framework was required which is used for packet encryption and secret key generation. A new scheme called strategic location-based random routing was presented. Here, routing paths are not optimal since it selects random routes which results large delay for packets delivery at the sink node, and also, random nodes are selected by node position, which does not always result in accurate location protection. Energy consumption of the whole network is high due to the packet's transmission through different paths.

4. C^2S^2-Loop System Model

In this section, we describe our proposed model of circular chessboard-based secure routing for SLP. The proposed system architecture is shown in Figure 3.

4.1. System Overview. C^2S^2-LOOP is comprised of four entities such as Sensor Nodes (SNs), Cluster Heads (CHs), Global Attacker, and Sink Node (SN). Global attacker used to track the location of the source node, which tends to leak the privacy and also leakage of information. We implement the following operations: (i) key generation, (ii) network topology management, (iii) travel plan, and (iv) data packets encryption. The initial step is held once because it does not need to repeat the process until the transmission range of the network is changed.

4.2. Assumptions for C^2S^2-LOOP

(1) All sensor nodes are deployed randomly over the circular region with equal initial energy

(2) All sensor nodes are homogeneous (nodes having similar functionalities of processing and communication)

(3) All sensor nodes are moving randomly at constant speed and location are aware

(4) All sensor nodes have a frame header of 104 bits and transmit packet based on time interval

(5) Sink is placed at the center of region, which is stationary

(6) Adversary (Global Attacker) is near to the sink node of network, which tries to find out the source location

(7) It has a global view, and thus, it can view the whole communication patterns of the network

4.3. C^2S^2-LOOP Network Setup. In C^2S^2-LOOP, the circular region is divided into M circular fields. We consider $M = 3$, which means the sensor network is divided into three circular fields c_1, c_2, c_3. The radius of circular fields are denoted as r_1, r_2, r_3, and the ratio is 1 : 2 : 3, respectively. We deployed sensor nodes in a circular chessboard c_1, c_2, c_3. However, there are two shapes are identified in chessboard (white and gray). In each gray and white space, we can see a fair distribution of sensor nodes and does not deploy densely. In a circular chessboard, nodes are deployed in the ratio of 1 : 2 : 3. Area a is determined for each circular field is a critical concern due to deployment and location determination. The sensing area for circular field 1 is depicted in Figure 4.

The area computation for all c_1, c_2, c_3 is as follows:

$$\begin{aligned} a(c_1) &= \prod R^2, \\ a(c_2) &= \prod (2R)^2 - \prod R^2 = 4\prod R^2 - \prod R^2 = 3\prod R^2, \\ a(c_3) &= \prod (3R)^2 - \prod (2R)^2 = 9\prod R^2 - 4\prod R^2 5\prod R^2, \end{aligned} \tag{1}$$

where R is the radius of the circular fields. After sensor nodes deployment, SN_i is responsible for executing sensing S_T and transmitting data packets D_T at time T. The sink node S_{INK} issues public and private keys for all sensor nodes. S_{INK} aggregated all data packets with the packet length S_{ps} through CHs while reducing energy consumption. Other sensor nodes transmit the sensed packets to CH. The process of key generation, network topology management, travel plan, and packet encryption are implemented in many works, but C^2S^2-LOOP incurs a large safety period, minimum packet delay, and minimum energy consumption. Improvement of these QoS

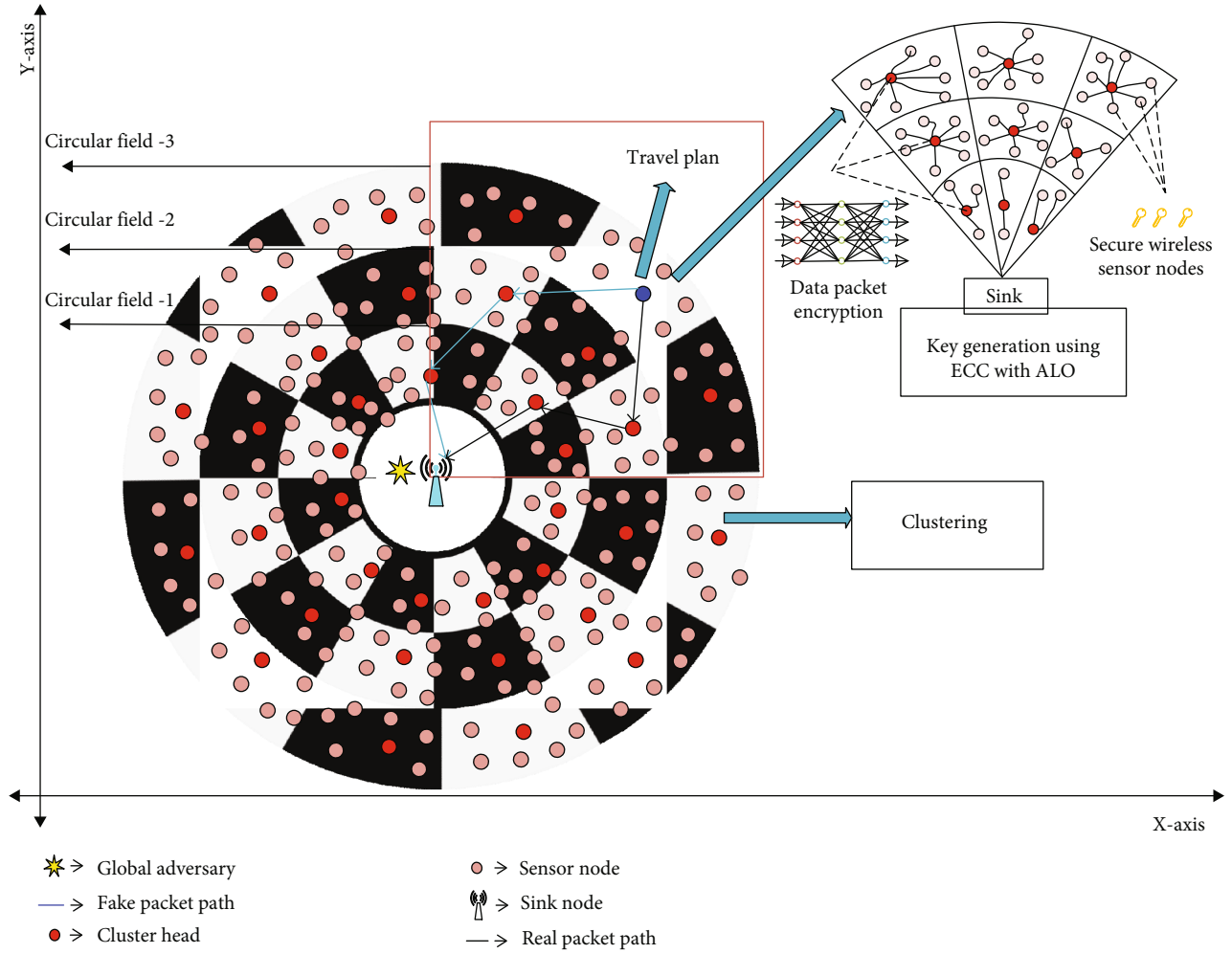


Figure 3: System architecture.

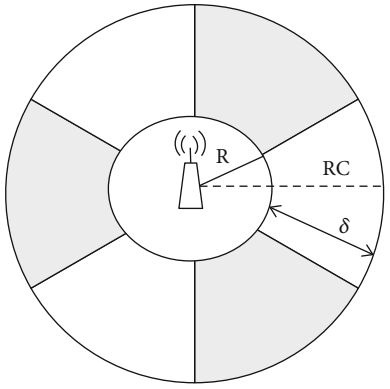


Figure 4: Sensing area for circular field 1.

metrics in the circular chessboard and protecting source location privacy is the main focus of this paper.

4.4. Key Generation. Firstly, registration is performed using ECC with ALO algorithm in which private key and public keys are generated, which is highly secured than the conventional ECC algorithm. ECC is a public-key cryptography system that performs computation by elliptic curve instead of polynomial arithmetic or integer. It provides strong security compared to RSA. Elliptic curve is expressed by an equation of two variables with coefficients. Elliptic curve over real numbers satisfies the set of points, which must satisfy the following equation:

$$Y^2 = X^3 + aX + b, \tag{2}$$

where X, Y, a, b are real numbers, applying different sets of values for a and b.

Definition 1. Elliptic curve.

The elliptic curve over Z_p, $p > 3$ is the set of all pairs $(X, Y) \in Z_p$ which satisfy

$$Y^2 \equiv X^3 + a \times X + b \mod p, \tag{3}$$

where $a, b \in Z_p$ $(0, 1, \cdots p-1)$, i.e., set of integer values with

modulo p arithmetic, and the condition is the following

$$4a^3 + 27b^2 \neq 0 \mod p. \tag{4}$$

For example, $a = -4, b = 0.67$ gives the elliptic curve with the following equation:

$$Y^2 = X^3 - 0.4x + 0.67, \tag{5}$$

However, ECC is demonstrated by two fields such as prime field and binary field. In order to deal with the cryptographic operations, a suitable finite field is selected with contains a large number of points. Some of the benefits of using ECC are as follows: (1) it uses a smaller key size for ciphertexts and signatures, (2) it supports very fast key generation, (3) fast encryption and decryption process than RSA, and (4) for computations, ECC required less memory and CPU cycles than RSA. Therefore, it is suitable for all applications.

Key generation is a significant process where we generate both public key and private key PU_K, SE_K, respectively. This SE_K is generated using the ALO algorithm. ECC key generator description is as follows [37]:

$$\begin{aligned} SE_K &= I \times \text{Selected from the interval } (1, p-1), \\ PU_K &= \quad SE_K \times BP, \end{aligned} \tag{6}$$

where BP is the base point taken from the elliptic group. However, ECC selects private key in a random manner, which leads to the incorrect generation of ciphertext. To mitigate such issues, ALO is proposed for private key generation, and then, the private key and base point are used to generate the public key, written in equation. The process of ALO is described underneath: ALO is a new metaheuristic algorithm released in 2015 by Seyedali Mirjalili [38]. It is based on the behaviors of Ant Lions Hunting mechanism. The major steps involving in ALO for hunting prey are (1) Ants random walk, (2) Building traps, (3) Entrapment of Ants in Traps, and (4) Catching Preys, and rebuilding traps are executed. In order to determine the private key (optimal solution), ALO follows the following process:

(i) *Initialization.* Firstly, initialize the population of Ants and Antlions in a random way

(ii) *Fitness Computation.* Compute the fitness of Ants and Antlions and determine the best antlion is the optimum solution (Elite)

(iii) *Update Ants Position.* Choose an antlion by Roulette Wheel. Then updates Ants position using random walk around choose Antlion and Elite. In addition, we compute fitness value for all Ants

(iv) *Replace an Antlion.* In this step, replace an antlion with its corresponding ant (only when it becomes fitter)

(v) *Update Elite.* When an action becomes fitter compared to Elite

(vi) *Termination Condition.* When termination condition is reached, return the Elite as the optimum solution for the given problem. Else, update ant's position and continue until the termination condition is reached

In ALO, fitness is computed based on SN_i coordinates (x, y), Identity, i.e., PSRN, and current time stamp T_S. Hence, it is defined by the following:

$$f = \{SN_i(x, y), PSRN, T_S\}, \tag{7}$$

where f is the fitness value for private key selection.

Pseudocode 1: SE_K key generation using ALO.1) Begin

2) Initialize population of Ants and Antlions

3) Compute f for Ants and Antlions

4) Determine the Best_Antlions

5) Consider the current Best_AntLion is the Elite//Optimum Solution

6) While (End Criterion) is fulfilled

7) For every Ant

8) Select an Antlion via Roulette Wheel strategy

9) Update C and D using following

10) $C^t = C^t/I$ & $D^t = D^t/I$(11)

11) Make random walk and normalize it using eqn. (11)

12) $x(t) = [0, \text{cumsum}\,(2r(t_1) - 1) \cdots cumsum(2r(t_n) - 1)]$

$$X_i^t = \frac{(X_i^t - w_i) \times (D_i - C_i^t)}{(D_i^t - w_i)}$$

13) Update ant position using $Ant_i^t = R_a^t + R_e^t/2$

14) End for

15) Compute f for all Ant_i

16) Substitute an Antlion by corresponding ant becomes fitter using eqn

17) $\text{Antlion}_j^t = \text{Ant}_i^t$ if $f(\text{Ant}_i^t) > f(\text{Antlion}_j^t)$

18) Update e^t when Antlion fitter than the Elite

19) End while

20) Return Elite

Pseudocode of the proposed ALO is illustrated as follows: in the ALO algorithm, Ants and Antlions are initialized randomly. In pseudocode, w_i denotes minimum random walk of the ith variable. C_i^t denotes the maximum of the random walk at ith iteration. R_a^t and R_e^t are the random walk around the Antlion chosen by the Roulette Wheel, and elite at tth

1) Begin
2) Initialize population of Ants and Antlions
3) Compute f for Ants and Antlions
4) Determine the Best_Antlions
5) Consider the current Best_AntLion is the Elite//Optimum Solution
6) While (End Criterion) is fulfilled
7) For every Ant
8) Select an Antlion via Roulette Wheel strategy
9) Update C and D using following
10) $C^t = C^t/I$ & $D^t = D^t/I$(11)
11) Make random walk and normalize it using eqn. (11)
12) $x(t) = [0, \text{cumsum}\,(2r(t_1) - 1) \cdots cumsum(2r(t_n) - 1)]$

$$X_i^t = \frac{(X_i^t - w_i) \times (D_i - C_i^t)}{(D_i^t - w_i)}$$

13) Update ant position using $Ant_i^t = R_a^t + R_e^t/2$
14) End for
15) Compute f for all Ant_i
16) Substitute an Antlion by corresponding ant becomes fitter using eqn
17) $\text{Antlion}_j^t = \text{Ant}_i^t$ if $f(\text{Ant}_i^t) > f(\text{Antlion}_j^t)$
18) Update e^t when Antlion fitter than the Elite
19) End while
20) Return Elite

Pseudocode 1: SE_K key generation using ALO.

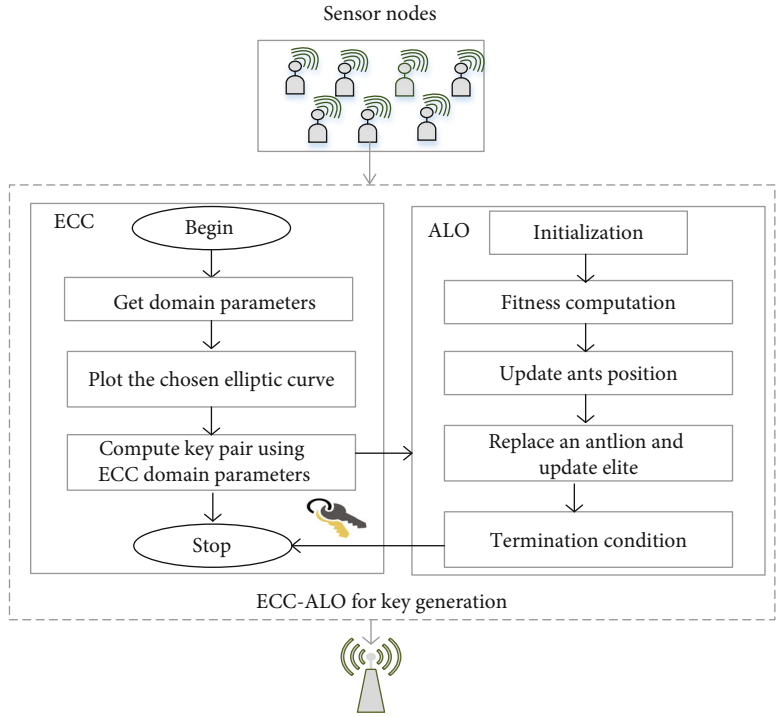


Figure 5: Key generation using ECC with ALO algorithm.

iteration variable, Ant_i^t indicates the current position of ith ant at tth iteration. Figure 5 depicts the flow diagram for ECC with ALO for key generation.

4.5. Network Topology Management. Network topology is managed by the selection of optimal CH. For CH selection, node residual energy E_R is considered.where E_I is the initial energy of a node and E_i is the energy left after the process of round 1 [39].

Definition 2. Node residual energy.

Node residual energy is the energy left after the process of data transmission. It is denoted as E_R, which is computed as follows:

$$E_R = E_I - E_{i=1}, \quad (8)$$

4.5.1. Cluster Formation. To form clusters, SN_i broadcasts hello packet HL_P to collect adjacent node information. In each circular field area, SN_i presented within the gray and white shape is considered as a cluster.

4.5.2. Cluster Head Selection. With the passage of time or in other words as the network lifetime increases, the node energy decreases. If CH remains static, it will soon die out of the battery power and will not remain part of the network. Therefore, dynamic CH selection is used. The node with maximum residual energy is elected in each round (SN_i having MAX_E_R) as CH. Each individual gray and shape area of nodes involved in the CH election process. This helps in improving the overall life of the network by equally utilizing the energy of all nodes and minimizes the energy hole problem in the network.

4.6. Travel Plan. Before packet transmission, we determine the travel plan between the source node to the sink node, which is not virtual or uses fake source nodes. In this work, we called routing is a travel plan. However, routing is the most fundamental issue in WSN since routing schemes play an important role in WSN. Attackers use the backtracking technique to catch the source node. In backtracking at some node, if the attacker is not able to hear any packet, then it requires to turn back and traverse to other directions. The communication range of the attacker and sensor node is the same. A random number R_n is generated by the source node which is compared with the already selected number P_n. If the $R_n \geq P_n$, then the real packet is sent in a clockwise direction and a fake packet in anticlockwise direction and vice versa. With the use of a fake path, an attacker cannot find the source location. It requires a greater number of backtracks, and finally, it receives fake packets and we also encrypt the fake messages and forwarded to the fake path. Real packets and fake packets are encrypted in the same manner so that the attacker cannot differentiate between them. It helps to increase the safety period of the network and unable to find the source location.

4.7. Data Packets Encryption. Data packets are encrypted and forwarded using a chaotic artificial neural network where we obtain chaotic sequences as a ciphertext. C-ANN is a combination of two algorithms such as Chaos Theory and Artificial Neural Network. There are several reasons to combine Chaos Theory with ANN: (1) it provides promising and efficient results for encryption and decryption, (2) Imitate function and structure of the human brain to be better, and (3) Chaos theory time series are considered as learning rate in ANN, which provide efficient results. In this work, we proposed a Hop Field Neural Network in ANN. With the use of keys generated using ECC with ALO, packets are encrypted using C-ANN.

Pseudocode 2 demonstrates the encryption process of packets using C-ANN. We generate encrypted packets by the following pseudocode.

Pseudocode 2: C-ANN Algorithm.1) Begin

2) Compute the length of key for encryption

3) Split the key into subsequences of 8 bytes

4) Initialize parameters μ and the initial point $X(0)$

5) For input message M (P_T)

6) Chaotic Sequence is follows: $X(1), X(2), X(3)\ldots$

7) $X(n+1) = \mu \mathrm{X(n)}(1 - \mathrm{X(n)})$

//Transforms binary representation

8) Binary Sequence is follows: $X(1), X(2), X(3)\ldots$

$B = (8n-8)b(8n-7)\cdots B(n-2)B(8n-1)$

9) $WF = B$

10) $WF + I/P$ function $\rightarrow C_T$

Few other biological-inspired routing protocols are also mentioned in [40].

5. Performance Evaluation

In this section, we evaluate the performance of C^2S^2LOOP by conducting various experiments. Firstly, we demonstrate how the simulations will impact the network performance. Then, we define the parameter description in terms of safety period, latency, energy consumption, total energy consumption, and network lifetime. Lastly, we illustrate the comparison of C^2S^2LOOP with previous works.

5.1. Simulation Model. Experiments conducted using the OMNeT++ model which consists of the following four parts: (1) network topology definition (NED) describes the module structure which consists of gates and parameters. It is saved in .ned, (2) message definitions (.msg) in which we can define various message types and insert more data fields; (3) it will convert message definitions into full-fledged C++ classes (.ini); and (4) it consists of simple module sources. They are C++ files with an extension of .h and .cc. The configuration files (omnet.ini, NED) for simulation are depicted in Figures 6 and 7.

Figure 8 shows the simulation environment and experiments conducted over Windows 7 (Ultimate-x86) Operating System. The sink node is located at the center of circular region X, Y = 3000m, 3000m. The number of sensor nodes deployed is 200, and nodes are uniformly and randomly distributed over the circular field. Nodes' initial energy level is equipped with 100 J. Some of the parameters used for simulation are given in Table 2.

5.2. Case Study: Panda Hunter Game. This paper provided a case study of SLP for the Panda-Hunter Game model. Sensor nodes are deployed in a large environment to monitor the behavior of Panda. Panda is a source here, which sends messages to the sink node and Hunter is an adversary. Figure 9 pictorial representation of Panda Game is implemented using our proposed model.

The Hunter observes data packets and traces panda location from the sink node same as mentioned in [41, 42]. Our

1) Begin
2) Compute the length of key for encryption
3) Split the key into subsequences of 8 bytes
4) Initialize parameters μ and the initial point $X(0)$
5) For input message M (P_T)
6) Chaotic Sequence is follows: $X(1), X(2), X(3)\ldots$
7) $X(n+1) = \mu \mathrm{X(n)}(1 - \mathrm{X(n)})$

//Transforms binary representation

8) Binary Sequence is follows: $X(1), X(2), X(3)\ldots$
   $B = (8n-8)b(8n-7)\cdots B(n-2)B(8n-1)$
9) $WF = B$
10) $WF + I/P$ function $\rightarrow$ C_T

Pseudocode 2: C-ANN Algorithm.

Circular_Chessboard_Based.ned | omnetpp.ini | ECC_ALO.cc

```
package Circular_Chessboard_Based;
import inet.linklayer.common.GPA;
import inet.networklayer.ECC_ALO;
import inet.networklayer.configurator.ipv4.IPv4NetworkConfigurator;
import inet.node.inet.AdhocHost;
import inet.node.inet.Sensors;
import inet.node.wireless.Sink;
import inet.physicallayer.ieee802154.packetlevel.Ieee802154NarrowbandScalarRadioMedium;
import inet.physicallayer.ieee802154.packetlevel.Statplot;
import inet.transportlayer.contract.CANN;
network Circular_Chessboard_Based_NW{
    parameters:
        int numSensors;
        @display("bgi=device/bg,s");
    submodules:
        Sensor[numSensors]: AdhocHost {@display("is=l;i=device/Sensor");}
        Sink: Sink {@display("p=1500,1500;i=device/sink,#FF0080;t=Sink,t,Yellow;is=vl;b=,,oval");}
        configurator: IPv4NetworkConfigurator {config = xml("<config><interface hosts='*' address='145.236.x.x' netmask='255.255.0.
            @display("p=1,1");}
        radioMedium: Ieee802154NarrowbandScalarRadioMedium {@display("p=1,1");}
        statplot: Statplot {@display("p=1,1;is=vs");}
        ecc_alo: ECC_ALO {@display("p=16,2");}
    connections allowunconnected:}
```

Figure 6: NED file for simulation.

SLP goal here is to increase the time required for Hunter to capture the location of the Panda (Safety Period). While sending messages to the sink node, energy usage is an important concern. Some of the functionalities of the Hunter are as follows: (1) it does not interfere with the sensor network, (2) it contains devices to measure Angle of Arriving message, (3) it moves at any rate and unlimited power, and (4) it has a global view of how the network works.

5.3. Definition of Simulation Metrics. In order to estimate the performance of C^2S^2LOOP, we consider the following QoS parameters.

5.3.1. Safety Period. It is the time duration taken to deliver a number of data packets from S to the S_{INK} before global attacker reaches the source location. Hence, this parameter is gauged from the number of packets successfully received at the S_{INK} before tracing of the attacker. It is computed by the following:

$$S_P = \sum_{i=1}^{n} T\left(P\left(S\right), P(S_{\mathrm{INK}})\right) + R(P\left(S\right), P(S_{\mathrm{INK}})), \quad (9)$$

where S_P is the safety period, $T(P\left(S\right), P(S_{\mathrm{INK}}))$ is the transmission time of packet P sending from S to S_{INK} node, and $R(P\left(S\right), P(S_{\mathrm{INK}}))$ is the packet received time at S_{INK} node.

5.3.2. Latency. It is defined as the amount of time duration is taken by a packet to reach its destination from the source to

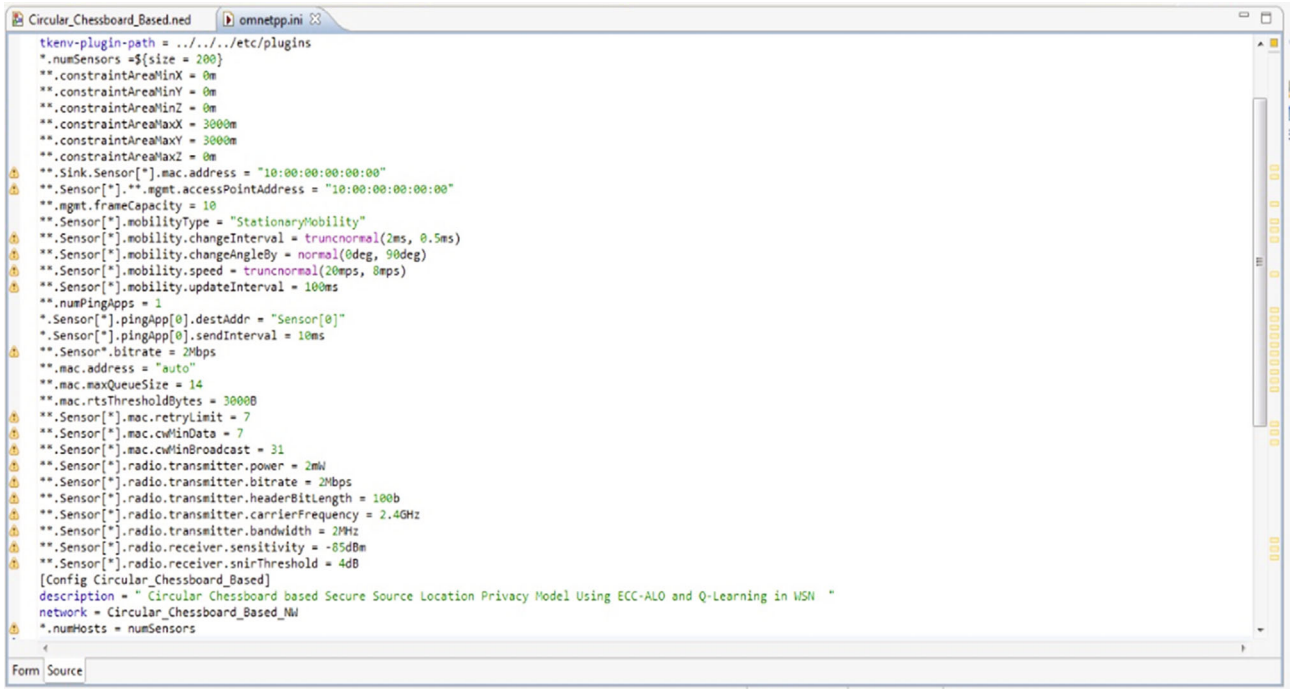


Figure 7: OMNETPP file used in simulation.

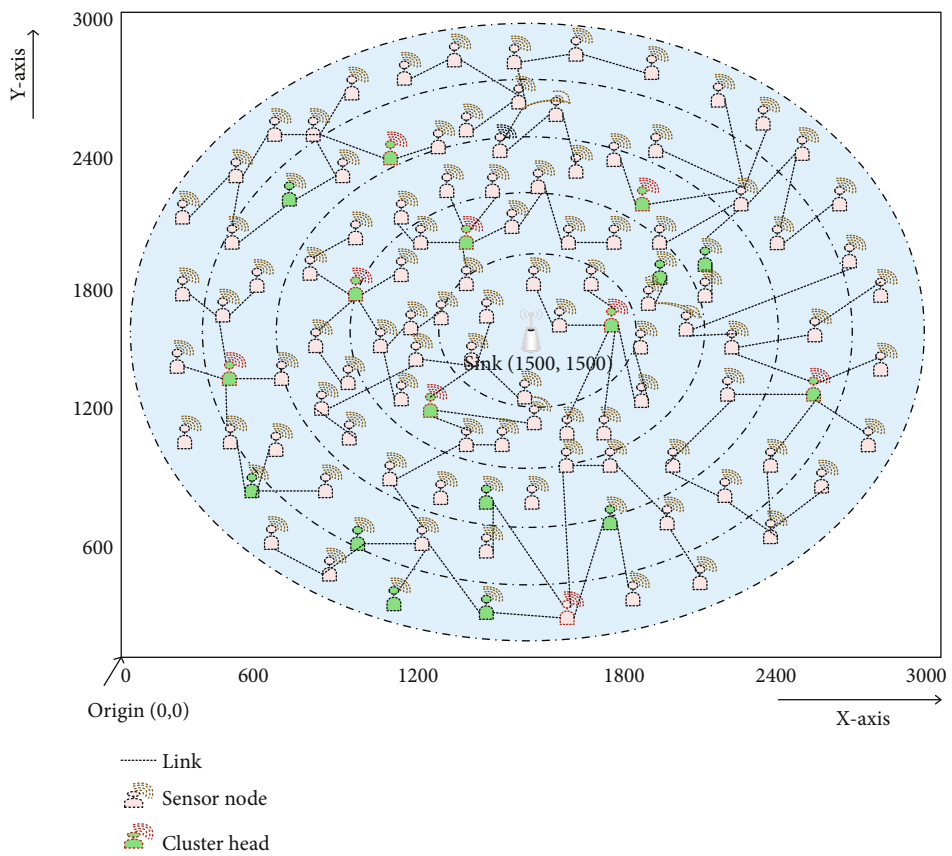


Figure 8: Simulation environment.

Table 2: Simulation parameters.

Parameter	Values
Simulation tool	OMNeT++4.6
Number of nodes	200
Number of circular fields	5
Simulation area	3000 m × 3000 m
MaxQueueSize	14
Sink mobility model	Random waypoint
Packet interval	10 ms
Number of packets	400
Packet size	1280 bytes
Path loss model	Free space path loss
Frequency	2.4 GHz
Channel bandwidth	20 MHZ
Transmission range	500 m
Header length	20B
Network layer protocol	IPv4
Key size	512 bytes
Simulation time	1000 ms
Bit rate	2 Mbps
Sensor mobility change angle	0-90 degrees
MAC protocol	IEEE 802.15.4
ALO	
No. of ants (population N)	50
Max iterations (t)	500
No of variables	6
Random number	(0,1)

the sink node. The computation of latency is based on the latency of queuing, processing, propagation, and transmission. It is computed by the following:

$$L = \mathrm{TR}_{S-\mathrm{SINK}} - \mathrm{TS}_{S-\mathrm{SINK}} \quad (10)$$

Where L is the latency, $\mathrm{TR}_{S-\mathrm{SINK}}$ is the time taken for packets transmission from the source node to the sink node, and $\mathrm{TS}_{S-\mathrm{SINK}}$ is sending time of packets transmitted from the source node to the sink node.

5.3.3. Energy Consumption. Let consider SN be the set of all nodes present in the network.

$$SN_i = \{SN_1, SN_2, SN_3 \cdots SN_{n-1}, SN_n\}. \quad (11)$$

Energy consumption E_C of SN_i is computed by the following:

$$E_C = E_I - E_R. \quad (12)$$

5.4. Comparative Study. In this section, a comparative study is conducted for the proposed and previous works such as CBA scheme [33], grid-based clustering [34], SLP-R [32], and two-phase routing [35]. Table 3 shows the comparison.

5.4.1. Safety Period Analysis. In a lot of previous works, source location privacy level is estimated in terms of safety period. It must be higher for better network performance. However, it increases when the distance between S and S_{INK} increases. Previous approaches are failed that achieved poor performance in terms of privacy level.

Figure 10 shows safety period analysis for hop count and distance between source node to the sink node. When the hop count increases, the safety period increases. CBA is designed with more privacy criteria for preserving source location privacy since it keeps only homogeneous sensor nodes for network deployment.

The safety period of CBA is closer to our C^2S^2LOOP which works well for global attacker and dummy message transmitted through random walk routing. In SLP-R, the source node can transmit a greater number of data packets before reaches the adversary to source origin. Furthermore, two-phase routing and grid-based clusters increase privacy level (safety period), but it decreases while the distance to S and S_{INK} increases. The average safety period regarding of it shows 156.66 s, 159.83 s, 87.5 s, 188.66 s, and 245.16 s for two-phase routing, SLP-R, grid-based clusters, CBA, and the proposed scheme, respectively.

Figure 11 indicates the performance of the safety period with respect to the distance between S to S_{INK}. Distance between S to S_{INK} increases gradually for all four previous works considered in this paper. In this case, the position of the source node is varied between 20 and 120 min intervals of 20 m. From the graph, it can be observed that the safety period of the proposed scheme increases as the distance increase compared to previous works. This is due to the fact that we determine the optimum route instead of random walk routing, phantom routing, and fake source routing. If a number of circular field increases, the safety period also increases with the help of distance between S to S_{INK}. The average safety period with respect to distance is 192.85 s, 194.14 s, 105 s, 214.28 s, and 290.42 s for two-phase routing, SLP-R, grid-based clusters, CBA, and the proposed scheme, respectively.

5.4.2. Latency Analysis. The latency in terms of the number of hop counts taken by the packets transmitted from the S to reach the S_{INK} is depicted in Figure 12. Our proposed approach shows minimal latency while all four previous works show maximal. The minimal delay in the proposed approach can be attributed to the fact that random routing paths are longer and selecting next-hop is not an optimal way. When the degree of randomness of the network is high, hop count is also high. However, we remark that the proposed scheme is best and it can adopt for Delay Tolerant Source Location Privacy Applications where privacy level and latency are sacrificed than previous works. In CBA, packet transmission is delayed due to alternating fashion of 50% nodes in the active stage and 50% of nodes in the sleep stage. Alternating fashion required large computations, and hence, packet delay is high.

Figure 13 depicts the performance of latency with respect to distance between S to S_{INK} node. From the plots, it can be seen that the proposed approach causes minimal latency for

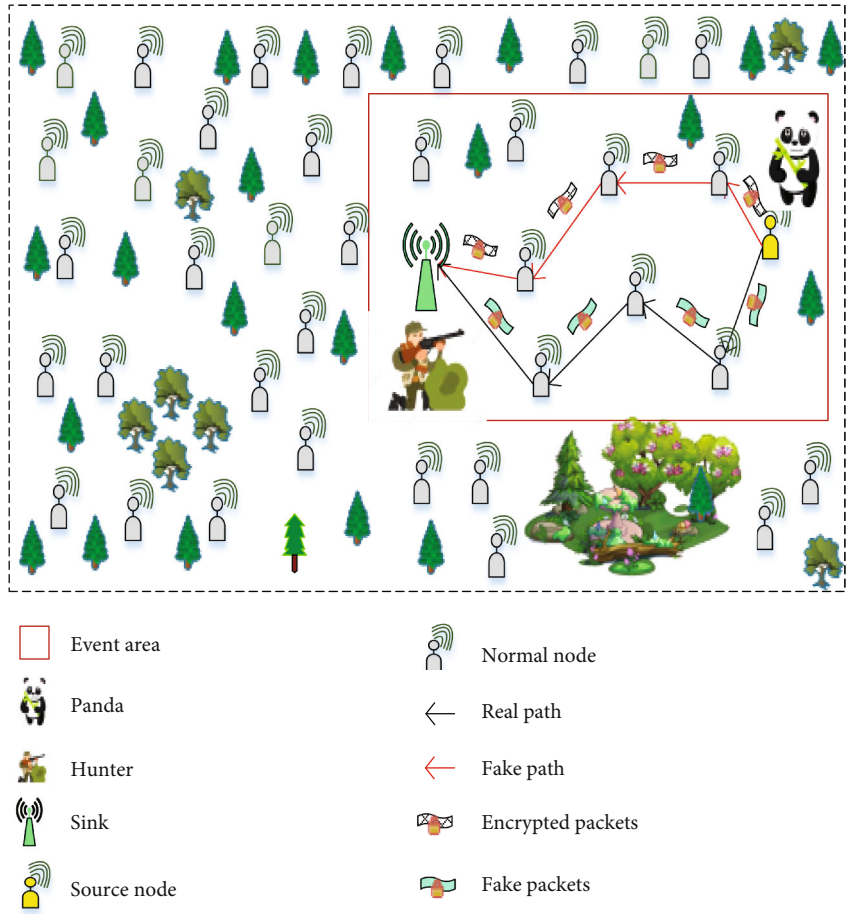


Figure 9: Panda game (proposed model use case).

Table 3: Comparison of proposed and previous approaches for different QoS parameters.

Method	Description	Parameters S_p	L	E_c	Demerits
CBA scheme	Source location privacy protection scheme is proposed against global attacker	→	↑	↑	(i) High energy consumption (ii) High delay
Grid-based clusters	The network is divided into number of clusters	↓	↑	↑	(i) Wastage of energy (ii) Large packet delay
SLP-R	Source location privacy using random routing	→	↑	↑	(i) Increases energy consumption due to random paths (ii) Privacy leakage
Two-phase routing	Escape angles based random walk routing	→	→	↑	(i) It utilized an excess amount of energy (ii) Network lifetime suffers due to multiple virtual source nodes

↑ = High, ↓ = Low, → = Medium.

packet transmission. Two-phase routing, SLP-R, grid-based clusters, and CBA require maximal delay. We proposed a circular chessboard-based SLP scheme, which main intention is to reduce packet delay and energy consumption. Among previous works, CBA and SLP-R require large latency. This can be attributed by random routing paths. Our solution here is the optimum formation of clusters in a circular chessboard and construct travel plan for routing packets. Thus, we obtained minimal latency when the distance between sensor nodes to the S_{INK} node is high.

The average latency with respect to hop count is 0.25 s, 0.49 s, and 0.131 s for the proposed approach and previous approaches. Similarly, the average latency with respect to distance is 0.25 s, 0.491 s, 0.465 s, 0.475 s, and 0.131 s for two-phase routing, SLP-R, grid-based clusters, CBA, and the proposed approach, respectively.

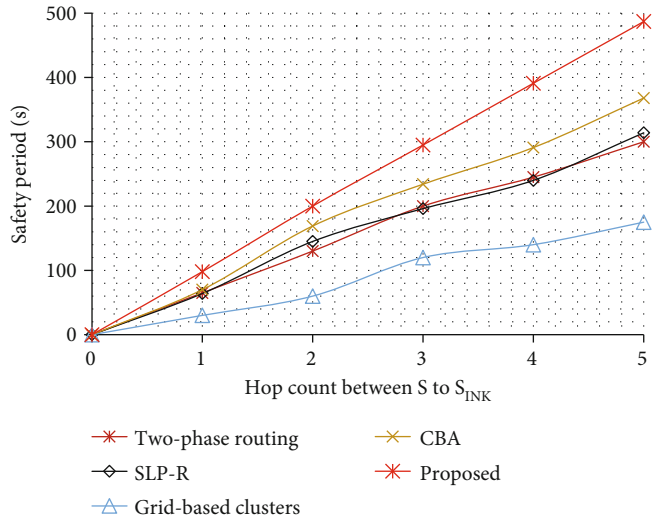


Figure 10: Safety period vs. hop count between SN to sink node.

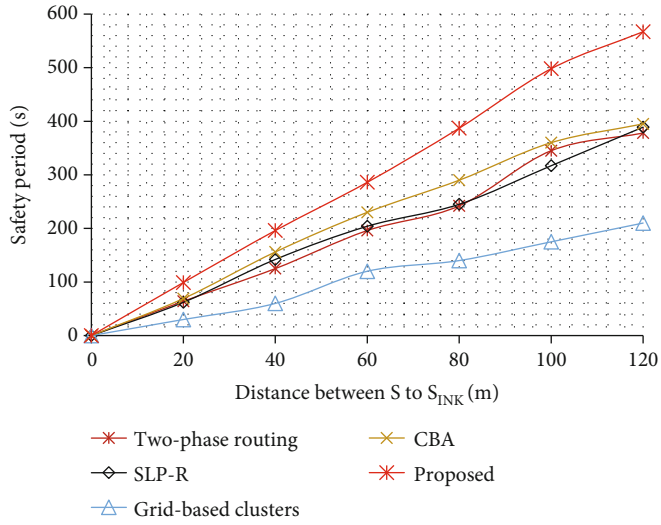


Figure 11: Safety period vs. distance between S to SINK.

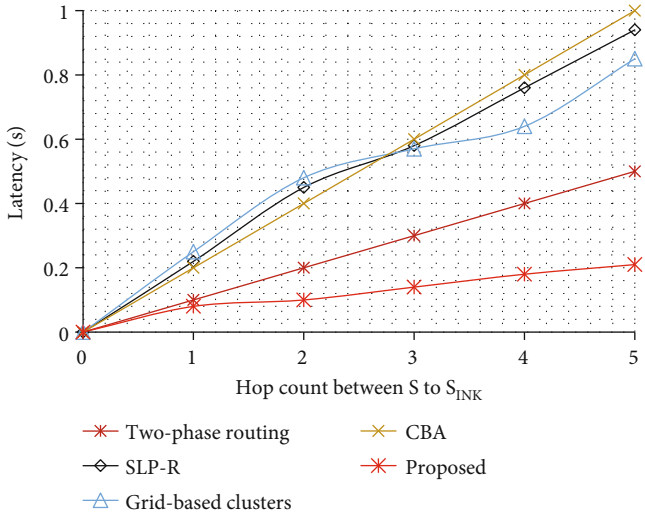


Figure 12: Latency vs. hop count between S to SINK.

5.4.3. Energy Consumption Analysis. Energy consumption to transmit a single bit of data packet is equal to processing 800 instructions. The energy consumption due to transmission of data packets from S to S_{INK} node is depicted in Figure 14. A large energy consumption primarily depends on the hop count the packets take to reach the S_{INK} node from source node S.

The energy consumption of the previous works is the maximum, and we obtained the least for the proposed approach. The reason for this least energy consumption by our proposed approach is as follows: (i) hop count between S to S_{INK} is less in the proposed approach compared to two-phase routing, SLP-R, grid-based clusters, and CBA. (ii) Our proposed approach forward aggregated packets through CH, which reduces large energy consumption than previous works. The average energy consumption for node-sending "n" packets is 935 J, 1010 J, 1250 J, 1041.66 J, and 524 J for two-phase routing, SLP-R, grid-based clusters, CBA, and the proposed approach.

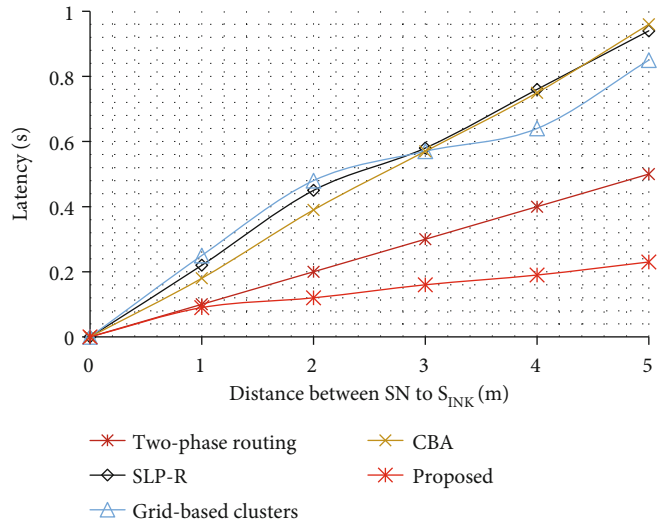


Figure 13: Latency vs. distance between SN to sink node.

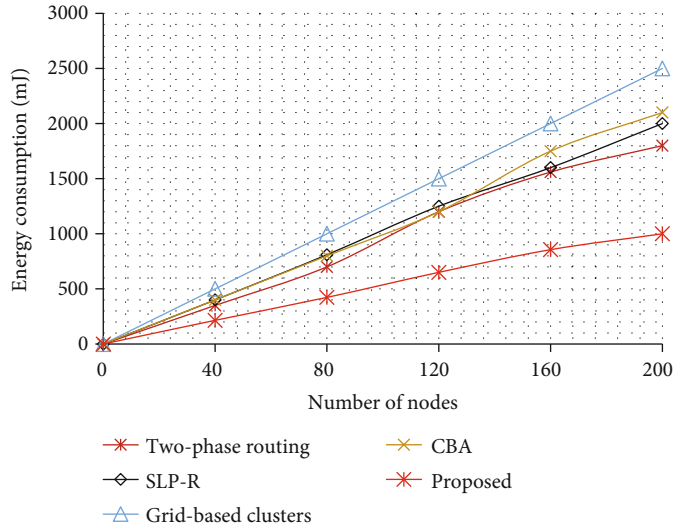


Figure 14: Energy consumption vs. number of nodes.

Table 4: Comparison between proposed vs. previous works.

Metric		Two-phase routing	SLP-R	Grid-based clusters	CBA	Proposed
Average safety period (s)	Hop count	156.66	159.83	87.5	188.66	245.16
	Distance	192.85	194.14	105	214.28	290.42
Average latency (s)	Hop count	0.25	0.49	0.465	0.5	0.11
	Distance	0.25	0.491	0.465	0.475	0.131
Energy consumption (mJ)		935	1010	1250	1041.667	524

6. Results Discussion

The efficiency of SLP in WSN is based on QoS requirements. Depends upon routing and network configuration (sensor node deployment and simulation values of each parameter), performance is varied. Energy is one of the important metrics, because sensor nodes are not replaceable and rechargeable. Hence, harvesting in WSN is considered for preserving source location privacy. Our proposed scheme does not harvest since we perform energy-efficient clustering for network management.

Table 4 illustrates the comparison between proposed and previous works in terms of average safety period, and latency based on hop count and distance, average energy

consumption, average total energy consumption, and average network lifetime.

The advantages of C^2S^2-LOOP are as follows: (i) a large safety period is obtained if the number of circular fields increases, safety period also increases with the aid of our proposed model (key generation, network topology management, travel plan, and packet encryption). (2) Low packet delivery delay from source to the sink node. This is due to our proposed algorithms such as ECC with ALO and chaotic-based ANN algorithm. These algorithms perform speedily and produce accurate and better performance than the earlier works. (3) C^2S^2-LOOP ensures full security and privacy of source node location. (4) In the case of data packet redundancy, energy consumption is minimized using clustering and hence the whole network is managed and also minimizes load at each sensor node. In addition, the individual transmission of packets to sink node is inefficient so a global attacker can be easily tracing packets through adjacent sensors.

7. Conclusion

In this paper, a novel C^2S^2-LOOP scheme is designed to protect the source location privacy in WSN. In SLP, QoS requirements are important such as safety period, packet delay, and energy consumption. Previous works in SLP do not consider all these parameters, which affects the network performance. These parameters are achieved with the aid of our C^2S^2-LOOP scheme. C^2S^2-LOOP invoked with the following operations: key generation, network topology management, travel plan, and data packets encryption. Sensor nodes authenticated using the elliptic curve ALO algorithm, which selects optimum secret keys, which are verified at the sink node. Next, the network topology is managed by the formation of clusters. In clusters, node residual energy is used for CH election. Intracluster communication is initiated by single-hop and intercluster routing (from source to the sink node) is implemented using clockwise and anticlockwise directions. To ensure data confidentiality, chaotic ANN is used in which data packets are encrypted and transmitted to the next-hop. Extensive experimentation is conducted using the OMNeT ++ network simulator that shows the that proposed C^2S^2-LOOP maximizes the safety period while minimizing energy consumption and latency as compared to previous works such as two-phase routing, SLP-R, grid-based clusters, and CBA.

8. Future Work

In the future, we have planned to extend our research in the following aspects:

(i) We deploy IoT sensors for monitoring SLP in real-time applications (disaster monitoring and control). Hence, we combine IoT and WSN environments

(ii) We prevent the network from security attacks (spoofing attacks, DoS attacks, and more)

(iii) We also protect sink location privacy against both global and local attackers

Data Availability

The data that support the findings of this study are all briefly introduced and all information are available in the manuscript.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This project was funded by the Deanship of Scientific Research (DSR), King Abdulaziz University, Jeddah, Saudi Arabia under Grant No. (KEP-19-611-38). The authors acknowledge with thanks the Deanship of Scientific Research (DSR), King Abdulaziz University, Jeddah, Saudi Arabia, for technical and financial support.

References

[1] Z. Qian, Q. Xiaolin, and D. Youwei, "Intelligent silent zone for source-location privacy based on context-awareness in WSNs," *Transactions of Nanjing University of Aeronautics and Astronautics*, vol. 35, no. 1, pp. 203–218, 2018.

[2] W. Chen, M. Zhang, G. Hu, X. Tang, and A. K. Sangaiah, "Constrained random routing mechanism for source privacy protection in WSNs," *IEEE Access*, vol. 5, pp. 23171–23181, 2017.

[3] M. Bradbury and A. Jhumka, "A near-optimal source location privacy scheme for wireless sensor networks," in *2017 IEEE Trustcom/BigDataSE/ICESS*, Sydney, NSW, USA, August 2017.

[4] A. Bushnag, A. Abuzneid, and A. Mahmood, "Source anonymity in WSNs against global adversary utilizing low transmission rates with delay constraints," *Sensors*, vol. 16, no. 7, p. 957, 2016.

[5] B. Chakraborty, S. Verma, and K. P. Singh, "Staircase based differential privacy with branching mechanism for location privacy preservation in wireless sensor networks," *Computers & Security*, vol. 77, pp. 36–48, 2018.

[6] C. Gu, M. Bradbury, J. Kirton, and A. Jhumka, "A decision theoretic framework for selecting source location privacy aware routing protocols in wireless sensor networks," *Future Generation Computer Systems*, vol. 87, pp. 514–526, 2018.

[7] Y. He, G. Han, H. Wang, J. Adu Ansere, and W. Zhang, "A sector-based random routing scheme for protecting the source location privacy in WSNs for the Internet of Things," *Future Generation Computer Systems*, vol. 96, pp. 438–448, 2019.

[8] G. Han, X. Miao, H. Wang, M. Guizani, and W. Zhang, "CPSLP: a cloud-based scheme for protecting source-location privacy in wireless sensor networks using multi-sinks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 3, pp. 2739–2750, 2019.

[9] L. Mutalemwa and S. Shin, "Achieving source location privacy protection in monitoring wireless sensor networks through proxy node routing," *Sensors*, vol. 19, no. 5, p. 1037, 2019.

[10] Q. Wang, J. Zhan, X. Ouyang, and Y. Ren, "SPS and DPS: two new grid-based source location privacy protection schemes in wireless sensor networks," *Sensors*, vol. 19, no. 9, pp. 2074–2093, 2019.

[11] A. Jhumka, M. Bradbury, and M. Leeke, "Fake source-based source location privacy in wireless sensor networks," *Concurrency and Computation: Practice and Experience*, vol. 27, no. 12, pp. 2999–3020, 2015.

[12] J. Wang, R. Zhu, S. Liu, and Z. Cai, "Node location privacy protection based on differentially private grids in industrial wireless sensor networks," *Sensors*, vol. 18, no. 2, pp. 410–425, 2018.

[13] S. Li, Y. Xiao, Q. Lin, and Z. Qi, "A novel routing strategy to provide source location privacy in wireless sensor networks," *Wuhan University Journal of Natural Sciences*, vol. 21, no. 4, pp. 298–306, 2016.

[14] A. Proano, L. Lazos, and M. Krunz, "Traffic decorrelation techniques for countering a global eavesdropper in WSNs," *IEEE Transactions on Mobile Computing*, vol. 16, no. 3, pp. 857–871, 2017.

[15] Y. Wang, L. Liu, and W. Gao, "An efficient source location privacy protection algorithm based on circular trap for wireless sensor networks," *Symmetry*, vol. 11, no. 5, pp. 632–646, 2019.

[16] M. Bradbury, A. Jhumka, and M. Leeke, "Hybrid online protocols for source location privacy in wireless sensor networks," *Journal of Parallel and Distributed Computing*, vol. 115, pp. 67–81, 2018.

[17] B. Chakraborty, S. Verma, and K. P. Singh, "Differentially private location privacy preservation in wireless sensor networks," *Wireless Personal Communications*, vol. 104, no. 1, pp. 387–406, 2019.

[18] H. Wang, G. Han, L. Zhou, J. A. Ansere, and W. Zhang, "A source location privacy protection scheme based on ring-loop routing for the IoT," *Computer Networks*, vol. 148, pp. 142–150, 2019.

[19] J. Kirton, M. Bradbury, and A. Jhumka, "Towards optimal source location privacy-aware TDMA schedules in wireless sensor networks," *Computer Networks*, vol. 146, pp. 125–137, 2018.

[20] C. Huang, M. Ma, Y. Liu, and A. Liu, "Preserving source location privacy for energy harvesting WSNs," *Sensors*, vol. 17, no. 4, 2017.

[21] H. Wang, G. Han, C. Zhu, S. Chan, and W. Zhang, "TCSLP: a trace cost based source location privacy protection scheme in WSNs for smart cities," *Future Generation Computer Systems*, vol. 107, pp. 965–974, 2020.

[22] N. Wang, J. Fu, J. Zeng, and B. K. Bhargava, "Source-location privacy full protection in wireless sensor networks," *Information Sciences*, vol. 444, pp. 105–121, 2018.

[23] S. Dehghani, B. Barekatain, and M. Pourzaferani, "An enhanced energy-aware cluster-based routing algorithm in wireless sensor networks," *Wireless Personal Communications*, vol. 98, no. 1, pp. 1605–1635, 2018.

[24] G. Han, L. Zhou, H. Wang, W. Zhang, and S. Chan, "A source location protection protocol based on dynamic routing in WSNs for the Social Internet of Things," *Future Generation Computer Systems*, vol. 82, pp. 689–697, 2018.

[25] C. Ozturk, Y. Zhang, and W. Trappe, "Source-location privacy in energy-constrained sensor network routing," in *SASN '04: Proceedings of the 2nd ACM workshop on Security of ad hoc and sensor networks*, Washington, DC, USA, 2004.

[26] N. Wang and J. Zeng, "All-direction random routing for source-location privacy protecting against parasitic sensor networks," *Sensors*, vol. 17, no. 3, 2017.

[27] Z. Jia, X. Wei, H. Guo, W. Peng, and C. Song, "A privacy protection strategy for source location in WSN based on angle and dynamical adjustment of node emission radius," *Chinese Journal of Electronics*, vol. 26, no. 5, pp. 1064–1072, 2017.

[28] J. Y. Koh, D. Leong, G. W. Peters, I. Nevat, and W.-C. Wong, "Optimal privacy-preserving probabilistic routing for wireless networks," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 9, pp. 2105–2114, 2017.

[29] J. Long, M. Dong, K. Ota, and A. Liu, "Achieving source location privacy and network lifetime maximization through tree-based diversionary routing in wireless sensor networks," *IEEE Access*, vol. 2, pp. 633–651, 2014.

[30] G. Han, H. Wang, J. Jiang, W. Zhang, and S. Chan, "CASLP: a confused Arc-based source location privacy protection scheme in WSNs for IoT," *IEEE Communications Magazine*, vol. 56, no. 9, pp. 42–47, 2018.

[31] G. Han, H. Wang, M. Guizani, S. Chan, and W. Zhang, "KCLP: a k-means cluster-based location privacy protection scheme in WSNs for IoT," *IEEE Wireless Communications*, vol. 25, no. 6, pp. 84–90, 2018.

[32] M. Raja and R. Datta, "An enhanced source location privacy protection technique for wireless sensor networks using randomized routes," *IETE Journal of Research*, vol. 64, no. 6, pp. 764–776, 2018.

[33] Q. Zhou, X. Qin, and X. Xie, "Hiding contextual information for defending a global attacker," *IEEE Access*, vol. 6, pp. 51735–51747, 2018.

[34] M. F. Al-Mistarihi, I. M. Tanash, F. S. Yaseen, and K. A. Darabkh, "Protecting source location privacy in a clustered wireless sensor networks against local eavesdroppers," *Mobile Networks and Applications*, vol. 25, no. 1, pp. 42–54, 2020.

[35] R. Manjula and R. Datta, "A novel source location privacy preservation technique to achieve enhanced privacy and network lifetime in WSNs," *Pervasive and Mobile Computing*, vol. 44, pp. 58–73, 2018.

[36] L. Mutalemwa and S. Shin, "Strategic location-based random routing for source location privacy in wireless sensor networks," *Sensors*, vol. 18, no. 7, p. 2291, 2018.

[37] O. P. Verma, N. Jain, and S. K. Pal, "Design and analysis of an optimal ECC algorithm with effective access control mechanism for big data," *Multimedia Tools and Applications*, vol. 79, no. 15-16, pp. 9757–9783, 2020.

[38] S. Mirjalili, "The ant lion optimizer," *Advances in Engineering Software*, vol. 83, pp. 80–98, 2015.

[39] M. Lemos, R. Filho, R. Rabêlo, C. de Carvalho, D. Mendes, and V. Costa, "An energy-efficient approach to enhance virtual sensors provisioning in sensor clouds environments," *Sensors*, vol. 18, no. 3, 2018.

[40] S. Ahmed, "Nature Inspired Optimization Techniques, a review for FANETs," *Sukkur IBA Journal of Emerging Technologies*, vol. 3, no. 2, pp. 40–58, 2020.

[41] N. Jan, A. Al-Bayatti, N. Alalwan, and A. I. Alzahrani, "An enhanced source location privacy based on data dissemination in wireless sensor networks (DeLP)," *Sensors*, vol. 19, no. 9, article 2050, 2019.

[42] N. Jan and S. Khan, "Energy-efficient source location privacy protection for network lifetime maximization against local eavesdropper in wireless sensor network (EeSP)," *Transactions on Emerging Telecommunications Technologies*, no. article e3703, 2019.

Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 5580963, 15 pages
https://doi.org/10.1155/2021/5580963

Research Article

QoS-Oriented Optimal Relay Selection in Cognitive Radio Networks

Shakeel A. Alvi,[1] **Riaz Hussain**,[2] **Atif Shakeel**,[2] **Muhammad Awais Javed**,[2] **Qadeer Ul Hasan**,[2] **Byung Moo Lee**,[3] **and Shahzad A. Malik**[2]

[1]*Department of Electrical Engineering, National University of Sciences and Technology (NUST), Islamabad, Pakistan*
[2]*Department of Electrical and Computer Engineering, COMSATS University, Islamabad, Pakistan*
[3]*Department of Intelligent Mechatronics Engineering and Convergence Engineering for Intelligent Drone, Sejong University, Seoul 05006, Republic of Korea*

Correspondence should be addressed to Byung Moo Lee; blee@sejong.ac.kr

Received 25 February 2021; Revised 31 March 2021; Accepted 14 April 2021; Published 24 April 2021

Academic Editor: Sungchang Lee

A cognitive radio network can be employed in any wireless communication systems, including military communications, public safety, emergency networks, aeronautical communications, and wireless-based Internet of Things, to enhance spectral efficiency. The performance of a cognitive radio network (CRN) can be enhanced through the use of cooperative relays with buffers; however, this incurs additional delays which can be reduced by using virtual duplex relaying that requires selection of a suitable relay pair. In a virtual duplex mode, we mimic full-duplex links by using simultaneous two half-duplex links, one transmitting and the other one receiving, in such a way that the overall effect of duplex mode is achieved. The relays are generally selected based on signal-to-interference-plus-noise ratio (SINR). However, other factors such as power consumption and buffer capacity can also have a significant impact on relay selection. In this work, a multiobjective relay selection scheme is proposed that simultaneously takes into account throughput, delay performance, battery power, and buffer status (i.e., both occupied and available) at the relay nodes while maintaining the required SINR. The proposed scheme involves the formulation of four objective functions to, respectively, maximize throughput and buffer space availability while minimizing the delay and battery power consumption. The weighted sum approach is then used to combine these objective functions to form the multiobjective optimization problem and an optimal solution is obtained. The assignments of weights to objectives have been done using the rank sum (RS) method, and several quality-of-service (QoS) profiles have been considered by varying the assignment of weights. The results gathered through simulations demonstrate that the proposed scheme efficiently determines the optimal solution for each application scenario and selects the best relay for the respective QoS profile. The results are further verified by using the genetic algorithm (GA) and particle swarm optimization (PSO) techniques. Both techniques gave identical solutions, thus validating our claim.

1. Introduction

Facilitated by continuously evolving mobile broadband technologies such as 5G and the proliferation of powerful smart wireless devices such as phones, tablets, and laptops, the emergence of a plethora of social media-centric mobile applications with predominant video/audio content has enormously increased the data traffic in mobile wireless networks. This trend is evidenced in the International Telecommunication Union (ITU) report, according to which it is expected to have 17 billion wireless devices and 97 billion machine-to-machine (M2M) devices in the year 2030 [1]. This huge number of devices will generate a tremendous amount of data to be transmitted. To satisfy this demand, the mobile operator will require an additional spectrum or use the existing spectrum more efficiently. However, spectrum availability is limited as most of the bands suitable for mobile systems have already been allocated or licensed [2–4]. Concerted efforts are being made to free some bands to enable newer mobile services [5]. On the other hand, several

studies have indicated that the use of licensed bands is highly inefficient [6, 7]. It has been observed that the large portion of the licensed bands remains underutilized in various geographical locations and the spectrum utilization varies; some bands are utilized as low as 15% while even those highly utilized are up to 85% busy in time [8]. This inefficient use of spectrum by the users leaves spectrum holes, also called the white spaces [9]. In this backdrop, cognitive radio (CR) concept has emerged to exploit these spectrum holes or white spaces by allowing another set/group of users to opportunistically discover transmission opportunities and utilize them for communication [10]. In a cognitive radio network (CRN), a set of users called the secondary users (SUs) opportunistically use spectrum resources allocated to licensed network users, also called the primary users (PUs). The SUs sense the licensed band to discover holes and then utilize these opportunities for their data transmission with the condition that the communication of the primary user of the spectrum will not be affected. The key functionalities of a cognitive radio network include spectrum sensing, analysis, assignment, and management [11]. The spectrum access in a CRN can be grouped into two main categories that are overlay and underlay [12]. In the overlay, the channel is opportunistically used by the secondary user when not in use by the primary network, whereas, in the underlay, the channel is simultaneously utilized by both, but the power of the secondary user remains below the threshold level of the primary. Thus, simultaneous transmissions by both primary and secondary networks are possible. By using relays in an underlay CRN, the distance between the nodes of the secondary network can be reduced and spatial diversity for the communication can also be provided [13]. Further, the use of relay buffers improves throughput performance [14]. The introduction of cooperative relaying with buffers can enhance performance in terms of reduced outage probability and power while increasing throughput but the cost of increased packet delays [15, 16]. However, this delay can be mitigated through a virtual duplex relaying scheme which involves the use of two relays or relay pairs to form two links, one from source-to-relay and the other from relay-to-destination [17]. In virtual duplex mode, we mimic the full-duplex link by using simultaneously two half-duplex links. The main concept is to select a pair of relays, each operating in a half-duplex mode, one transmitting and the other one receiving, in such a way that the overall effect of duplex mode is achieved.

A critical factor in a CRN with buffer-aided cooperative relaying is the selection of a suitable relay pair that can enhance its performance in terms of outage, throughput, and delay but improves spectral efficiency as well. The selection of relays is based on meeting quality-of-service metrics such as an outage, throughput, delay, and fairness as well as the efficient use of resources, for example, bandwidth, available power, and relay node buffer capacity. Most of the existing work on relay selection relies on meeting an SINR threshold while optimizing one of the performance measures, i.e., throughput, delay, or outage [18–20]. This approach is limited in the context of varying application-specific quality-of-service requirements where, for example, one application may need delay performance optimization, whereas another application may require maximizing throughput or minimizing power consumption. The work in this paper is aimed at developing an optimal relay selection scheme that caters to the needs of the diverse application QoS requirements. The proposed scheme is based on a weighted sum multiobjective optimization approach and considers several scenarios with different combinations of application QoS needs in terms of throughput, delay, battery power, and the relay node buffer space. Thus, solution obtained provides the most suitable relay for each of these scenarios and thus allows the CRN performance to be optimized dynamically for a diverse set of user applications. To the best of our knowledge, the multiobjective problem formulated in this work is unique in the sense that it simultaneously considers four objectives, i.e., throughput, delay, battery power, and relay buffer state to optimally select the relay pair for a virtual duplex mode of communication.

The rest of the paper is organized as follows: Related work on relay selection and multiobjective optimization in wireless communications are covered in Section 2. The system model, the description of objective functions, and the formulation of the optimization problem are presented in Section 3. The proposed scheme for relay selection is introduced in Section 4. The results and analysis are presented in Section 5, and the conclusions are drawn in Section 6.

2. Related Work

Relay selection has received considerable attention in the literature as the adoption of relays in wireless networks has improved the overall system performance through improved diversity, reduced interference, and shadow mitigation [21]. The introduction of cooperative relaying, wherein the nodes between the transmitter and receiver coordinate to ascertain a suitable relay node, has resulted in gains in throughput, coverage, and energy efficiency [22, 23]. Recently, the incorporation of buffers at relay nodes has allowed further improvements in the performance of cooperative relaying [24]. A cognitive radio network (CRN) is aimed at improving the spectral efficiency of the network, and by adding buffer-aided relaying to the network, the performance of the network can be further enhanced [25]. The selection of suitable relays is fundamental to the performance of relaying, and the incorporation of relay buffer results in enhancement of important parameters like an outage, power consumption, and throughput, but the downside is additional packet delays [26]. The authors in [27] incorporated buffer size in their relay selection scheme; their objective was to reduce the packet delay and improve diversity as well. But if the size of the buffer is decreased, then the probability of packet drop increases. In [28], the authors discuss max-max relaying scheme, which works in simplex mode. It first selects the best source-to-relay link and stores this packet in its buffer. Then, in the next time slot, the best relay-to-destination link is selected to transmit this packet from relay-to-destination. Another relaying scheme known as the max-link scheme [29] gives priority to the best link quality. That is whichever link source-to-relay or relay-to-destination has the best

quality is selected first for reception or transmission, respectively. Both of these schemes have the downside that delay increases with the increase of several relays or an increase in buffer size.

In another work [30], the authors have proposed a buffer-aided relay selection scheme, which they claim results in better delay performance. In this scheme, a higher priority is given to the selection of the relay-to-destination links: In this scheme, $S \longrightarrow R$ was selected only once, and no $R \longrightarrow D$ link can be selected. This results minimum queue length at the relay nodes; thus, the average packet delay reduces as well. In all these schemes, relay operates in simplex mode. In [31], the authors argue that by given priority to the only relay-to-destination links, the end-to-end delay may still be high due to packets jumbling up at the source and may even result in packet drop. To overcome the problem of added delay, a virtual model of operation was proposed. Furthermore, the main criteria of a relay or relay pair selection have been the signal-to-noise ratio. Other factors may be as important for an efficient communication network, such as efficiency, battery life, and buffer size, and thus should be part of selection criteria.

Multiobjective optimization (MOO) has been extensively used in wireless networks to determine optimal solutions in cases where there is more than one desired goal. In [32], the authors conduct a comprehensive survey of the latest techniques involved in the modeling of multiobjective optimization concerning relaying with conflicting objectives. They also highlighted the advantages and disadvantages of each technique. The authors also classified each of the existing approaches based on the types of objectives and investigated main problem domains, critical trade-offs, and key techniques used in each class. A survey of the use of MOO in wireless sensor network (WSN) is presented in [33], which includes in-depth discussions on the fundamental approaches, metrics, and relevant algorithms. In [34], the authors analyzed various objectives to classify them as conflicting, supporting, or design dependent. They then presented a MOO problem relating to wireless sensor networks (WSNs) which consist of parameters like the required inputs, the desired outputs, and the constraints. They also put forward various constraints that must be considered while formulating MOO problems in WSN.

The use of MOO in cognitive radio networks (CRNs) is presented in [35]. In this work, the authors conducted a comparison of various types of optimization techniques. It was also studied how to combine different objectives to obtain an optimal solution in CRN. In [36], the performance of an underlay cognitive sensor network (CSN) with the simultaneous wireless information and power transfer- (SWIPT-) enabled relay node is investigated. This work focuses on maximizing achieving rates in CSN, with the condition of minimizing interference to the PUs. This objective tried to achieve by optimizing the transmit power. The sensing time and desired SNR are key to energy-efficient CRN. Thus, optimizing these parameters would result in energy-efficient communication in CRN. With this aim, the authors in [37] studied joint optimization of these important performance parameters. The authors argued that by optimizing the sensing time with the desired SNR, while remaining within the required detection probability, energy efficiency can be achieved in a network. The throughput-delay trade-off problem for cooperative spectrum sensing (CSS) is investigated in [38]. They claimed that in their proposed algorithm while constraining the delay to a certain value, the throughput is maximized since simultaneously it is not possible to maximize throughput and minimize delays.

In [39], it has been argued that network efficiency can be maximized only when the greenhouse gas emission (GHGE) is kept to a minimum in green cooperative cognitive radio networks (GCCRNs). Since dealing with multiple conflicting objectives, so MOO was employed to maximize the rate while keeping GHGE to a minimum, in their relay selection process. Applying different optimization techniques, nonconvex problems were transformed into a convex one and throughput maximization with GHGE minimization was achieved. Finally, by employing the zero-norm principle, an optimal relay node was selected. In another work [40], authors applied cross-entropy optimization (CEO), while optimizing two conflicting objectives which are maximizing the total rate and minimizing the greenhouse gas emissions in GCCRN.

In [41], the authors considered the problem of maintaining the desired level of QoS for the PUs and optimizing the performance of a cooperative cognitive radio user. The scheme was based on an overlay CRN in which the admission of PUs packets was probabilistically controlled in the secondary user (SU) relaying queue. Two queues with policies of work-conserving and non-work-conserving are made, with aims of optimization SU throughput or delay, respectively. In [42], a dual-hop, full-duplex underlay CRN is considered. Here, the aim was to minimize the outage probability of the SU, by an adaptive power allocation scheme.

When the SU nodes are to be used for relaying PU data, then some sort of incentive or payoff is to be offered. Usually, such schemes are used in overlay CRN. And if these are harvesting energy out of PUs network, then such SU nodes should also be charged. The work in [43] investigated such payoff mechanisms intending to maximize gains to both networks. They proposed a greedy-based algorithm to solve this problem. It was claimed that their scheme is quite effective and very close to the optimal solution. With the understanding that cognitive radio networks are used to enhance spectral efficiency, the authors in [44] worked on improving the performance matrices such as throughput and delay. They proposed a distributed algorithm, which they implemented in MATLAB. They compared their scheme with uniform allocation and max-min bandwidth allocations schemes and claimed 12 percent and 20 percent improvement with respect to compared the schemes, respectively. A delay constraint multihop network was considered in [45]. In this work, the rates and the power are adaptively adjusted in such a way to ensure average power constraint at each node. It was claimed that this adaptive power scheme outperforms in comparison with the constant power scheme. In [46], the authors investigated the performance of cache-assisted SWIPT cooperative systems. Multiple relays capable of caching and energy harvesting are to facilitate communication between a source and destination. Here, the aim is to

maximize the throughput and energy storage, subject to a QoS requirement. Now, the bases of a relay selection could be a relay with maximum throughput or maximum stored energy. In [47], two things were addressed that is when the relay node should cooperate and cooperation should be on what bases. They proposed a Nash bargaining-based strategy to resolve these issues. Multiobjective optimization is increasingly being used in optimizing performance in several emerging domains such as the Internet of Things (IoT), mm-wave communications, and D2D. For example, in [48], the authors discussed network optimization techniques for IoT and reviewed recent work done in this area. The review is concluded with open issues and challenges for network optimization in IoT. In [49], a TDMA-based MWN is studied with the aim of minimizing the end-to-end delays. A cross-layer optimization technique, while using multipath routing, was used to allocate time slots such that the average end-to-end delay is minimized.

Authors in [50] argued that besides the link SRN, it is equally important to take into account the status of buffers at the relay node. They advocated a relay node with maximum available space for reception and a node with maximum occupied space for transmission. Another scheme known as combined relay selection (CRS) [29] is based on the concept of shortest-in longest-out (SILO). The scheme relaxes the requirement of the best link quality; it is only required to be qualified. More emphasis is given to available buffer space (ABS) or occupied buffer space (OBS) while assigning weights. Relay nodes are then selected based on weight assignment. The problem of rate and routing is addressed [51] in multihop self-backhaul millimeter-wave (mm-wave) networks. This paper proposes the use of multiple antennas for diversity and traffic splitting techniques for throughput enhancement. The authors in [52] studied the problem of cochannel interference in an underlay cooperative relaying scheme, in a D2D communication system. As energy efficiency is of significant importance, so the aim here was to maximize energy efficiency. The survey in [34] established that Pareto optimal (PO) and weighted sum (WSUM) are the leading optimization techniques. Few relay selection schemes use the iterative method of relay selection and consider buffer status while selecting a relay. Authors in [29, 50, 53, 54] also considered buffer state in their relay selection schemes; all of these schemes are half-duplex and the link quality needs to be only qualified. In wireless sensor network, there is an astringent constraint on power consumption, so in [55], the authors studied trade-off to be made between delay and energy consumption while determining the route towards the BS. They proposed a MOO routing protocol based on ant colony optimization technique, with energy consumption and cost and end-to-end delay as objectives. Improvements in terms of energy efficiency and delay reduction were claimed. Energy efficiency (EE) and the spectral efficiency (SE) performance of multihop full-duplex cognitive relay networks was investigated in [56]. First multiobjective optimization problems were transformed into a single function, and then, the nonconvex problem was successfully transformed into convex form. The authors claimed that the proposed algorithm efficiently solved the considered problem and the best trade-off among EE and SE can be achieved by proper selection of priority factor.

3. Proposed Scheme

In CRN, the secondary users (SUs) collectively discover the vacant resources through cognition and share the resources for their communication. In the underlay mode of communication, the SUs are particularly restricted in transmission power, which limits the range of communication. In our model, by allowing relaying the range is extended. To increase the communication speed, i.e., data rate, full-duplex mode is used, which theoretically doubles throughput as compared to half-duplex mode since in a one-time slot, both transmission and reception take place. However, in our system, if full duplexing is deployed, i.e., if the same relay is used for transmission and reception, then the system suffers from loop interference. To overcome this loop interference yet keeping the system throughput, double the virtual duplexing, which allows transmission as well as the reception in the same slot but using two distinct relays. The use of two distinct relays significantly reduces the interference. Now, these relays can be fixed or mobile. In our work, we have not restricted it to one type, so it can work in either case. In this section, the system model of a multihop CRN with buffer-aided relaying is described followed by the formulation of the relay selection optimization problem based on particular QoS requirements.

3.1. System Model. We consider a secondary network consisting of one source "S," one destination "D," and a group of "N " buffer-aided relay nodes in between, operating in an underlay mode alongside a primary network. Direct communication between the source and the destination is not possible. Also, there is a power constraint on the secondary transmitters and the relays to ensure that communication of the primary network is not affected. The limited battery life of the relay nodes is considered, so it becomes pertinent with the efficiency in transmission. Each relay is equipped with a buffer of finite size. Moreover, we need to be cognizant of buffer space availability as well as the assignment of priority to different packets to meet the QoS requirement of nondelay tolerant communications. Similarly, the battery power or residual energy of the node will ensure the requisite SINR at the receiving end. The energy required to receive, decode, and retransmit to the destination needs to be ascertained and should be included in the relay selection process. Thus, buffer size and battery life are important determining factors for relay selection. In our scheme, the requirements are determined based on the type of application: voice, video, data, real-time or non-real-time, etc. The system model of our scheme is shown in Figure 1. It is assumed that the relays will first share their channel state, buffer state, and battery status information as depicted in Figure 2 [24, 57].

3.2. Objective Functions. The focus of this work is to devise a method to select the best relay, with due consideration to four important performance parameters, i.e., throughput, delay, relay buffer space, and battery power at the relay nodes, while

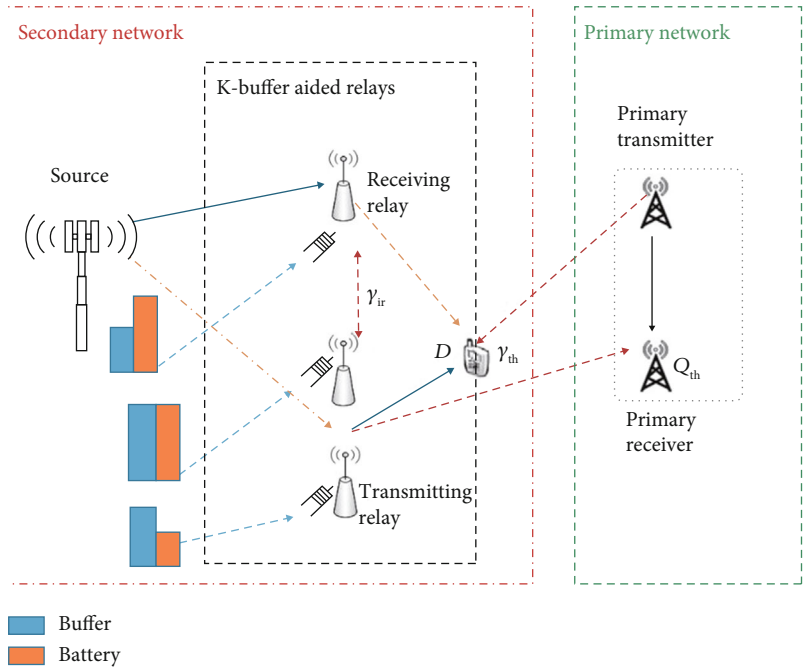


Figure 1: System model.

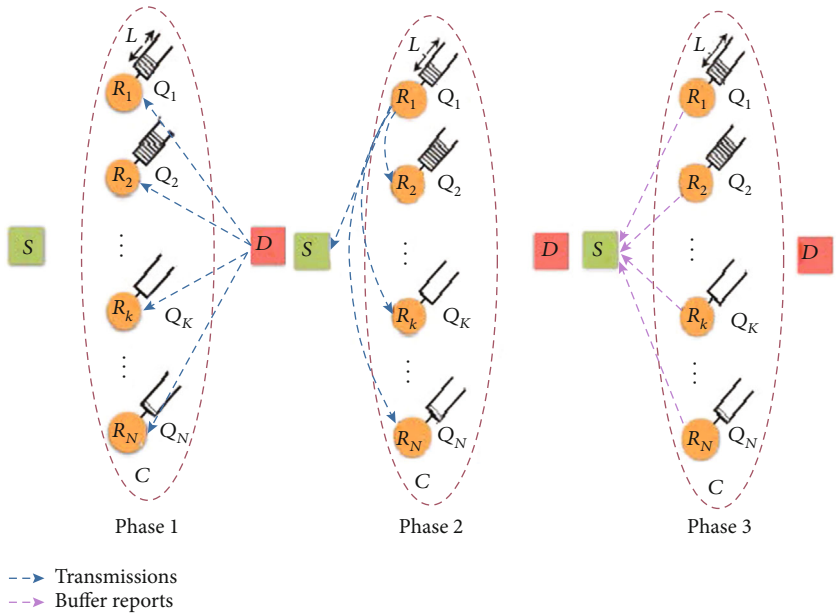


Figure 2: CSI and other information exchange procedure.

taking into account the maximum allowable interference constraints imposed by the primary network. The emphasis is on choosing the suitable relays that satisfy a set of certain QoS profiles corresponding to application requirements. A QoS profile would generally include multiple objectives in terms of the performance parameters such as maximizing throughput and buffer space availability while minimizing the delay and battery power consumption. In this section, we first derive expressions for the aforementioned objectives in light of the constraints imposed by the system.

3.2.1. Throughput. The objective function that maximizes the throughput of a network is represented by f_1. Since system capacity is given by the Shannon capacity theorem and is closely related to throughput, in general, average throughput is given as follows:

$$f_1 = \eta = R \times (1 - P_{\text{out}}), \tag{1}$$

where R represents the capacity of a link, which is dependent on SINR and is governed by Shannon's theorem, and P_{out} is

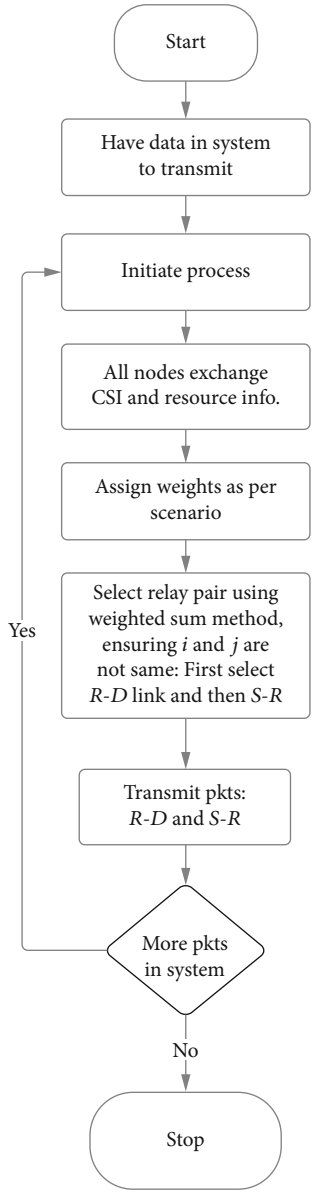


Figure 3: Algorithm flow diagram.

the probability that none of the relay's incoming and outgoing links is available. In a multihop system, the end-to-end capacity is limited by the weakest link; however, by enabling virtual duplexing, the capacity becomes the average of the capacities of the two links, i.e., the selected incoming link and the selected outgoing link in a time slot. Thus, the instantaneous capacities of the incoming link ($R_{S\longrightarrow R}$) and outgoing link ($R_{R\longrightarrow D}$) are as follows:

$$R_{S\longrightarrow R} = \log_2(1+\gamma_{S\longrightarrow R}), \tag{2a}$$

$$R_{R\longrightarrow D} = \log_2(1+\gamma_{R\longrightarrow D}), \tag{2b}$$

respectively, where $\gamma_{S\longrightarrow R}$ and $\gamma_{R\longrightarrow D}$ are obtained as follows:

$$\gamma_{S\longrightarrow R} = \frac{\left|g_{S\longrightarrow R_i}\right|^2 \times P_{SS}}{\left|I_{P\longrightarrow D_j}\right|^2 P_{PT} + \sum_{l=1}^{N}\left|I_{TR_j\longrightarrow SR}\right|^2 P_{TR_j} + n}, \tag{3a}$$

$$\gamma_{R\longrightarrow D} = \frac{\left|g_{R_j\longrightarrow D}\right|^2 \times P_{TR}}{\left|I_{P\longrightarrow D_j}\right|^2 P_{PT} + \sum_{l=1}^{N}\left|I_{TR_j\longrightarrow SR}\right|^2 P_{TR_j} + n}, \tag{3b}$$

where i and j represent the i^{th} source-to-relay and j^{th} relay-to-destination links, respectively, selected and simultaneously active in a time slot, enabling the virtual duplexing. $|g_{S\longrightarrow R_i}|^2$ and $|g_{R_j\longrightarrow D}|^2$ are the channel gains from source-to-relay and relay-to-destination, $|I_{P\longrightarrow D_j}|^2$ and $|I_{TR_j\longrightarrow SR}|^2$ are the interference, and P_{PT}, P_{TRj}, and P_{SS} are the transmitter powers of the primary source, the transmitting relay, and secondary source, respectively. The average throughput of the incoming link, i.e., $S\longrightarrow R$, is $R_{S\longrightarrow R}\times(1-P_{\text{out},S\longrightarrow R})$, and the outgoing link, i.e., $R\longrightarrow D$, is $R_{R\longrightarrow D}\times(1-P_{\text{out},R\longrightarrow D})$.

So, the end-to-end throughput is the average of the two and the long-term average gives the following:

$$R_{ss} = R_{S\longrightarrow R}\times(1P_{\text{out},S\longrightarrow R}), \bar{R}_{R\longrightarrow D}\times(1P_{\text{out},R\longrightarrow D}). \tag{4}$$

To calculate throughput from Equation (1), we need to find the outage probability of the system. The secondary network will be in outage if (i) all links, i.e., $S\longrightarrow R$ and $R\longrightarrow D$, are in outage, or (ii) buffers of all cooperating relays are full and all $R\longrightarrow D$ links are in outage, or (iii) buffers of all cooperating relay are empty and all $S\longrightarrow R$ links are in outage.

$$P_{\text{out},S_l} = P_{\text{out},S\longrightarrow R}\times P_{\text{out},R\longrightarrow D}, \tag{5}$$

$$P_{\text{out},S\longrightarrow R} = \left(1-e^{(-\delta/\gamma_{S\longrightarrow R})}\right)^{K_{S_l,S\longrightarrow R}}, \tag{6}$$

$$P_{\text{out},R\longrightarrow D} = \left(1-e^{(-\delta/\gamma_{R\longrightarrow D})}\right)^{K_{S_l,R\longrightarrow D}}, \tag{7}$$

where $\delta = 2^{r_t}-1$ and r_t is the target data rate in bps/Hz and K is the number of links which is $\leq N$ and S_l is the buffer state. We denote Q_{th} as the interference threshold at the primary receiver. To incorporate this constraint in our selection scheme, it is ensured that the transmission from the secondary source and relay network must satisfy the following condition:

$$\left|I_{(SS\longrightarrow PR)}\right|^2 P_{SS} + \sum_{l=1}^{N}\left|I_{TR_j\longrightarrow SR}\right|^2 P_{TR_j} \leq Q_{\text{th}}. \tag{8}$$

The acceptable level of interference at the receiving relay of interest is as follows:

$$\left|g_{P\longrightarrow R_i}\right|^2 P_p + \sum_{l=1}^{N}\left|I_{TR_j\longrightarrow SR}\right|^2 P_{TR_j} \leq \gamma_{\text{th}}, \tag{9}$$

```
N = Number of Relays;
NumberOfObjectives = 4; // i.e. 1: Throughput, 2: Delay, 3: Battery and 4: Buffer
Exchange CSI and Resource Information
//Discover the available incoming links, i.e. S ⟶ R links
ListOfAvailableIncomingLinks[] = [] //List contains indices of available relays for Rx, initially the list is empty
NumberOfAvailableIncomingLinks = 0;
for i=1:N do
    if γ_sr^i > γ_th AND // SNR of i^th S ⟶ R link
    ABS_i > 0 AND // Available buffer at i^th relay
    Bat_i > E_th // Batery power at i^th relay
    then
        NumberOfAvailableIncomingLinks++;
        ListOfAvailableIncomingLinks[NumberOfAvailableIncomingLinks] = i;
    end if
end for
//Discover the available outgoing links, i.e. R ⟶ D links
ListOfAvailableOutgoingLinks[] = [] //List contains indices of available relays for Tx, initially the list is empty
NumberOfAvailableOutgoingLinks = 0;
for i=1:N do
    if γ_rd^i > γ_th AND // SNR of i^th R ⟶ D
    OBS_i > 0 AND // Occupied buffer at i^th relay
    Bat_i > E_th// Batery power at i^th relay
    then
        NumberOfAvailableOutgoingLinks++;
        ListOfAvailableOutgoingLinks[NumberOfAvailableOutgoingLinks] = i;
    end if
end for
// Now we select the best outgoing link out of the available outgoing links as per application QoS requirements.
selectedOutgoingRelayIndex = -1; // initially none selected
bestValue = 0; // initialized
for i=1: NumberOfAvailableOutgoingLinks do
    Y[ListOfAvailableOutgoingLinks[i]]=∑_{j=1}^{NumberOfObjectives} w_j × f_j(ListOfAvailableOutgoingLinks[i])
    if Y[ListOfAvailableOutgoingLinks[i]] ¿ bestValue then
        bestValue = Y[ListOfAvailableOutgoingLinks[i]];
        selectedOutgoingRelayIndex = ListOfAvailableOutgoingLinks[i];
    end if
end for
// Now select the best incoming link out of the available incoming links as per application QoS requirements and also it is not the
selected outgoing link.
selectedIncomingRelayIndex = -1; // initially none selected
bestValue = 0; // initialized
for i=1:NumberOfAvailableIncomingLinks do
    Y [ListOfAvailableIncomingLinks[i]] = ∑_{j=1}^{NumberOfObjectives} w_j × f_j(ListOfAvailableOutgoingLinks[i])
    if Y[ListOfAvailableIncomingLinks[i]] ¿ bestValue AND
ListOfAvailableIncomingLinks[i] != selectedOutgoingRelayIndex then
        bestValue = Y[ListOfAvailableIncominggLinks[i]];
        selectedIncomingRelayIndex = ListOfAvailableIncomingLinks[i];
    end if
end for
```

Algorithm 1: Relay selection algorithm.

where P_{PT} is the transmitter power of the primary source, P_{TRj} and P_{SS} are the powers of the transmitting relay and the secondary source, respectively. $\sum_{l=1}^{N} |I_{TR_j \longrightarrow SR}|^2 P_{TR_j}$ is the sum of interferences from all transmitting relays of secondary network, and γ_{th} is the SINR for the required data rate.

3.2.2. Delay. The objective function that minimizes the delay of a network is represented by f_2. Delays depend on the number of time slots that a packet remains unattended in the buffer, whether in the source buffer or the buffer of the transmitting relay. We consider the end-to-end delay, so it is the sum of delays encountered in transmission of a data packet from source-to-destination and also that at the relaying

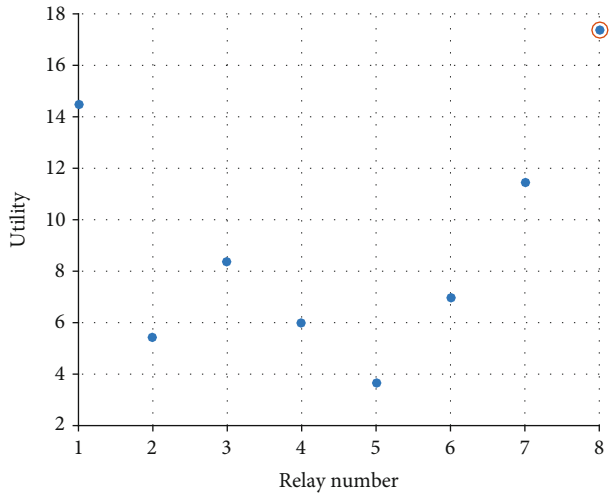


Figure 4: Scenario I at t_0: throughput.

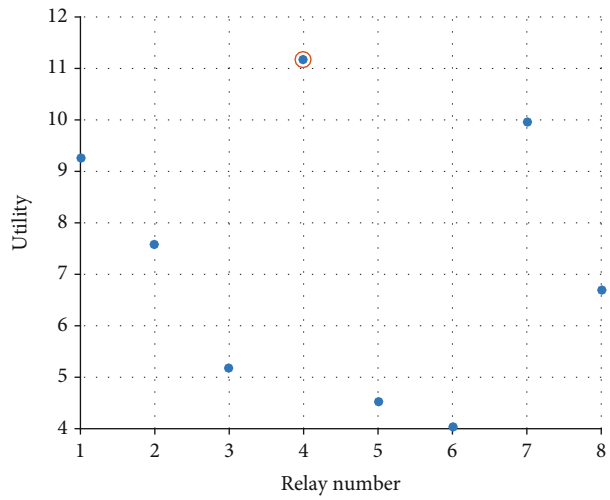


Figure 5: Scenario II at t_0: delay.

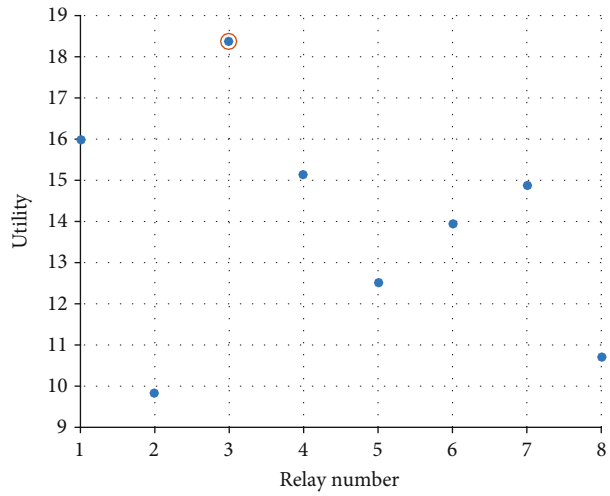


Figure 6: Scenario III at t_0: battery.

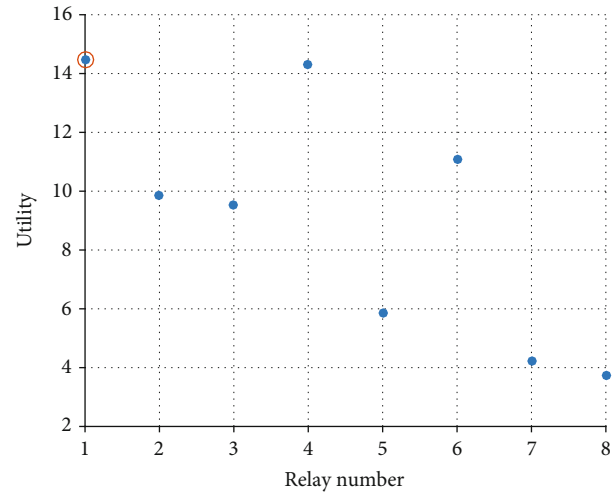


Figure 7: Scenario IV at t_0: buffer.

buffer. By applying Little's law, we get the average packet delay through the system as follows:

$$f_2 = \mathbb{D}_{\text{Total}} = \mathbb{D}_T = \mathbb{D}_S + \mathbb{D}_R = \frac{L_s}{\eta_s} + \frac{L_k}{\eta_k}, \tag{10}$$

where L_s and L_k are the average queue lengths at the source and the relay, respectively, and η_s(source $\longrightarrow$ relay) and η_K (relay $\longrightarrow$ destination) are the average throughput. As we enable the virtual duplexing (VD), different relays are selected for transmission and reception, which eliminates the loop interference (LI). Just as in full duplexing (FD), in VD also, a new packet is forwarded to the relay in each time slot. The delay is approximated as the mean of the two delays, that is, $\mathbb{D}_{\text{average}} = (\mathbb{D}_S + \mathbb{D}_R)/2$. The mean queue at any buffer is obtained by the following:

$$L_K = \sum_{l=1}^{(L+1)^N} \pi_l R_l(B_K), \tag{11}$$

where $R_l(B_K)$ gives the number of packets (or the buffer length) of buffer B_K at state S_l, and π_l is the stationary probability of that state. Assuming that the probability selecting any relay is same, the average throughput at relay (R_k) is given by the following:

$$\eta_k^* = \frac{\eta^*}{N} = \frac{(\text{Average throughput of whole system})}{(\text{Number of relays})}, \tag{12}$$

where η^* is the average throughput of the considered two-hop network and is obtained by the following:

$$\eta_k^* = R \times (1 - P_{\text{out}}), \tag{13}$$

where R is the link data rate with no outage in the system. We assume one time slot per data packet per hop. So, in our proposed virtue of virtual duplexing relaying system, we have $R = 1$, and thus, $\eta_k^* = 1 - P_{\text{out}}/(1 \times N)$, and finally, the total

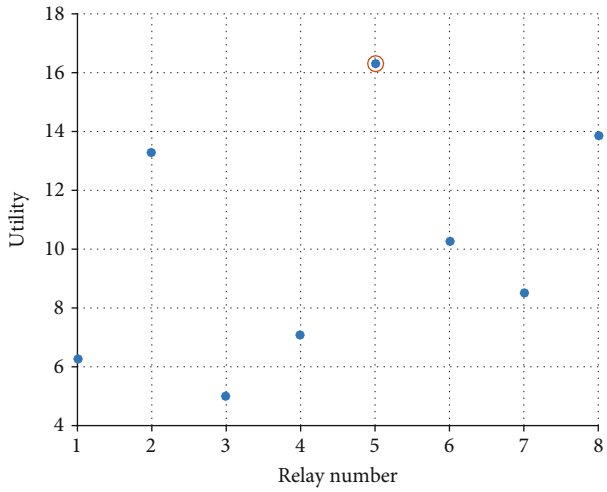


Figure 8: Scenario I at t_1: throughput.

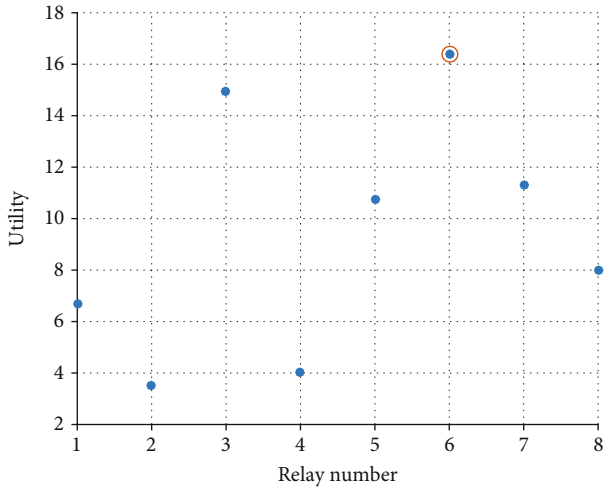


Figure 9: Scenario II at t_1: delay.

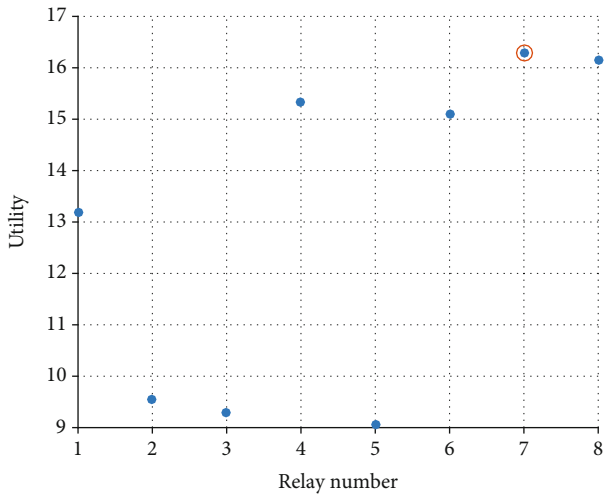


Figure 10: Scenario III at t_1: battery.

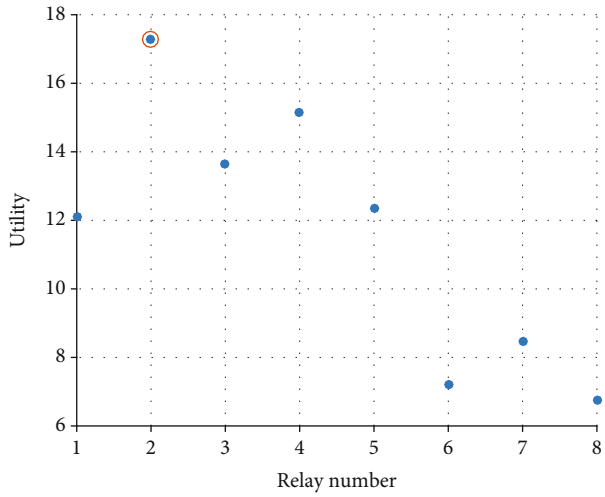


Figure 11: Scenario IV at t_1: buffer.

delay is obtained as follows.

$$\mathbb{D}^* = \frac{N\sum_{l=1}^{(L+1)^N} \pi_l R_l(B_K)}{(1 - P_{\text{out}})}. \tag{14}$$

3.2.3. Buffer Size. In a multiobjective relay selection approach, the buffer size at the relay node is considered an important performance parameter. We formulate this function based on buffer status that is occupied buffer space (OBS) and available buffer space (ABS); if a relay node has more OBS and less ABS, then the policy of SILO (shortest-in longest-out) is adopted. Besides, this link quality is also considered as through adaptive modulation, higher data rate can be achieved. Depending upon the application, buffer size requirement is calculated.

$$L = rtt_i \times \frac{C}{\sqrt{n}}, \tag{15}$$

where L is the buffer size, rtt_i is the average round trip time for relay i, C is the link capacity, and n is the number of flows sharing the link. Here, we consider $n = 1$. As capacity is linked to SNR, so a common factor between the different objective functions is SNR [58]. Another method of determining the required buffer size is given in [59] as follows:

$$f_3 = L = \frac{(rtt_{\max} - rtt_{\min}) \times C}{S}, \tag{16}$$

where S is the packet size; again, the above equation is indirectly dependent upon SINR. Since

$$C = B \times \log_2(1 + \gamma). \tag{17}$$

In our model, each relay R_k is equipped with a finite buffer of size L packets. L_k^o is the occupied buffer space (OBS), and $L_k^a = L - L_k^o$ is the available buffer space (ABS). When applying in relay selection, even when our scenario is

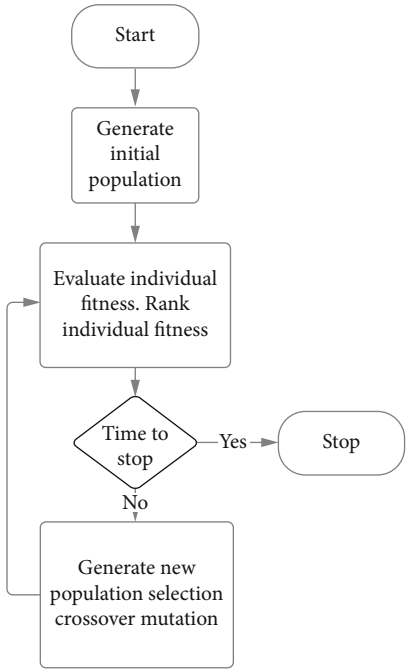


Figure 12: Flow diagram of GA.

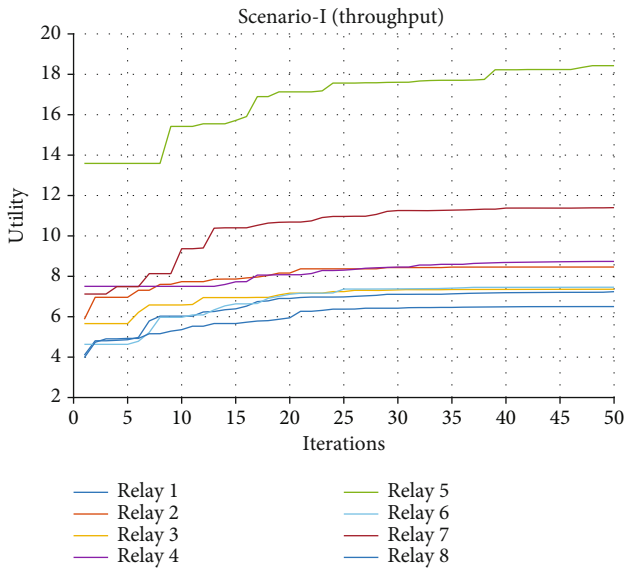


Figure 13: Optimal relay selection with GA.

based on buffer size, as subdivision in this will be made on weight assignment depending upon whether $S \longrightarrow R$ or $R \longrightarrow D$ link is to be selected. When the buffer size is not of concern, then equal weights can be assigned to these links. The constraint here will be that the relay receiving should have buffer space more than our data packet and the relay selected for transmission does not have empty buffer. Moreover, the policy of SILO will be applied; this will cater for delays as well.

3.2.4. Battery Power. Usually, the relay nodes are running on stringent energy budget and the aim would be to transmit a

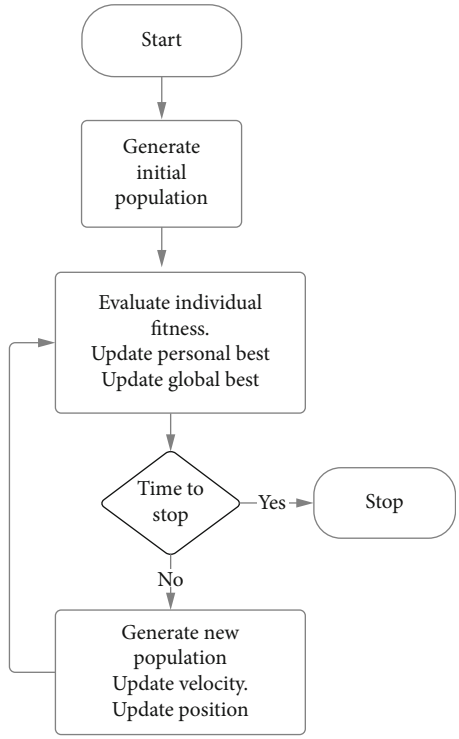


Figure 14: Particle swarm optimization flow diagram.

required amount of packets to its destination, under certain channel conditions, in the most energy efficient way. Thus, it would be important to calculate the minimum energy required to reliably transmit these packets. This energy is usually denoted by MEC, that is minimum energy consumption. While selecting, we need to ascertain that at least this much energy is available at the relay node. This MEC is calculated as follows:

$$f_4 = E_{\min}(D_L) = P_t D_L \left(B \log_2 \left(1 + \frac{P_t g}{N_o B} \right) \right), \quad (18)$$

where D_L is the amount of data, B is the channel bandwidth, g is the channel gain, and P_t is the transmitter power.

In our scheme, the transmitter has the capability to maintain a constant power while adjusting the transmission rate depending upon channel conditions. Moreover, it can also adopt a different policy in which the power is varied to maintain a constant transmission rate. Thus, depending upon the channel condition, a constant received SNR is maintained, which is denoted by γ_c, under the peak power constraint $P_{\max}$ [60]. Depending upon QoS requirements, the power adaptation will be preferred as it can achieve better energy efficiency than rate adaptation, but at the cost of a certain probability of transmission outage.

3.3. Problem Formulation. The relay selection problem being considered here involves the four objective functions as in Equations (1), (10), (16), and (18), which correspond to the throughput, delay, buffer space availability, and battery power consumption, respectively. The objectives in Equations (1) and (16) are required to be maximized, while the

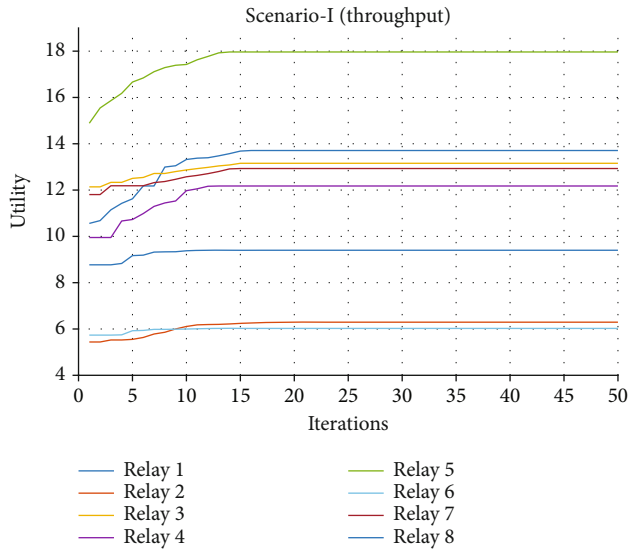


Figure 15: Result verification with PSO (scenario I).

other two in Equations (10) and (18) need to be minimized. Further, these objective functions conflict as well; therefore, a MOO approach can be adopted. This MOO problem can be solved using the weighted sum approach (also known as the scalarization method) [61, 62] and [53].

Thus, using the weighted sum approach, the multiobjective optimization problem for relay selection is formulated as follows:

$$Y_i = \left(w_1 f_1^i(X) + w_2\left(1 - f_2^i(X)\right) + w_3 f_3^i + w_4\left(1 - f_4^i\right)\right), \tag{19}$$

where $\sum_{l=1}^{4} = 1$,

$$Y_{\max} = \underset{(i\in\mathbb{I})}{\text{Maximum}}\, Y_i, \tag{20}$$

where $\mathbb{I} = (1, 2, 3, \cdots, N)$.

The assignment of weights can be done in three ways: equal weights, rank order centroid (ROC) weights, and rank sum (RS) weights [63]. We have set weights using the RS method [64]; however, ranks of each weight for respective objective are application dependent.

A variety of quality-of-service (QoS) requirement profiles involving a mix of real-time (RT) traffic and non-real-time (RT) traffic along with power consumption and buffer space limitations have been considered through assignment of ranks. For this work, a set of four QoS profiles has been considered which typically corresponds to varying QoS requirements of typical application/user traffic. Each QoS profile is characterized by assigning ranks to the objectives in order of priority. For instance, in one case, throughput objective, being accorded as the highest priority, is assigned 1st rank, followed by the delay, power consumption, and buffer space objectives with 2nd, 3rd, and 4th ranks, respectively. For other cases, ranks of the objectives are varied with the highest rank assigned to a different objective, thus emphasizing a particular QoS profile.

The four QoS profiles with corresponding ranks are represented by the four scenarios as listed below:

(1) Scenario I: throughput (1st), delay (2nd), battery power (3rd), and buffer size (4th)

(2) Scenario II: delay (1st), throughput (2nd), battery power (3rd), and buffer size (4th)

(3) Scenario III: battery power (1st), delay (2nd), throughput (3rd), and buffer size (4th)

(4) Scenario IV: buffer size (1st), delay (2nd), throughput (3rd), and battery power (4th)

For the above scenarios, appropriate weights are allocated to each of the objectives according to their respective ranks and relay selection is carried out to determine the most suitable relay corresponding to the particular QoS profile.

4. Relay Selection Method

The proposed relay selection scheme is application dependent. For example, there may be an application for which delay is of less concern than throughput, i.e., it requires that its data is passed on to the relays at a faster rate yet it is not concerned how much delay is incurred in relaying the data. Likewise, there can be an application for which buffer size is more important than battery life. With due consideration of all the factors, the best relay from the cluster of nodes can be selected as follows:

$$R_{\text{Best}}^{R\longrightarrow D} = \arg\max_{R_j\in\{1,2\cdots k\}} \left\{ \bigcup_{\substack{L_k\neq 0, Bat\geq E_{\text{th}} \\ \gamma_{R\longrightarrow D} > \gamma_{\text{th}}}} \left|h_{R_k D}\right|^2 \right\}, \tag{21}$$

$$R_{\text{Best}}^{S\longrightarrow R} = \arg\max_{R_i\in\{1,2\cdots k\}} \left\{ \bigcup_{\substack{L_k\geq a, \\ \gamma_{S\longrightarrow R} > \gamma_{\text{th}}, \\ i\neq j}} \left|h_{SR_k}\right|^2 \right\}, \tag{22}$$

$$R_{\text{Best}}^{\text{Pair}} = \left(R_{\text{Best}}^{R\longrightarrow D}, R_{\text{Best}}^{S\longrightarrow R}\right). \tag{23}$$

4.1. Relay Selection Algorithm. Figure 3 shows that the flow diagram of the algorithm and the pseudocode are given as Algorithm 1. As shown, the relay selection process starts with the announcement of requirements by the secondary source, to which all available relays respond with the exchange of CSI and resource information. In our scheme first, relay-to-destination CSI, buffer, and battery state are obtained from all available relays. Depending upon application requirements, i.e., the QoS profile, weights are assigned to different objectives. The network runs a weighted sum optimization algorithm to select the best relay from the available relays

and stores its index. First, selecting best $S \longrightarrow D$ is inline with [30]; this minimizes the number of packets in the relay buffer, thus resulting in minimization of delay. Next, we look for the best available $S \longrightarrow R$ link, as we have already selected relay-to-destination link so it will not be in available links, and automatically, our transmitting and receiving relays will not be the same. Depending upon application requirements, i.e., the QoS profile, weights are assigned to different objectives. The network runs a weighted sum optimization algorithm to select the best relay from the available relays. Our scheme is a bit different from [65], in which if $R_{r1} = R_{t1}$, the proposed protocol selects the relay with the second best $S \longrightarrow R$ channel R_{r2} and the relay with the best $R \longrightarrow D$ channel R_{t1} for reception and transmission, respectively.

5. Results and Analysis

The evaluation of the proposed relay selection scheme for a variety of simulation scenarios is carried out, using MATLAB. The validation of results, through simulations in MATLAB using genetic algorithm (GA) and particle swarm optimization (PSO), was also performed. The system model depicted in Figure 1 is the basis for the simulation setup, and evaluation of the proposed relay selection scheme for a variety of simulation scenarios is carried out. The number of nodes in the relay cluster is variable, and the proposed relay selection scheme chooses the best relay from the cluster. The simulation scenarios correspond to a diverse set of application traffic QoS requirement situations that reflect the varying level of importance of the multiple objectives being considered in this work. As mentioned earlier, using the weighted sum method to find the optimal relay pair requires selecting scalar weights w_i and maximizing the composite objective function. Changing the weights as per application requirements will give the optimal relay corresponding to that particular QoS profile.

Each of the four scenarios was considered in two different time slots, t_0 and t_1. The results for the above mentioned four scenarios are reported in Figures 4–11. It can be seen that for each scenario, a different relay is chosen corresponding to the objective with the highest rank (i.e., with the largest weight).

For example, at time slot t_0, for scenario I where throughput has precedence, the relay index 8 is selected as depicted in Figure 4. Similarly, for scenario II where delay performance is emphasized for time-critical applications, relay index 4 is the most suitable (Figure 5). For scenario III and scenario IV, where battery power and buffer size are of prime importance, then the optimal relays are relay index 3 and relay index 1, respectively, as shown in Figures 6 and 7. The process is repeated for time t_1. Again, it was found that by changing the weights, the optimal relay changes. The above results demonstrate the efficient working of the relay selection scheme in terms of adaptability to varying application QoS requirements, thus enabling to dynamically select the most suitable relay for communication.

5.1. Result Validation with GA and PSO. The results were further verified by performing optimization using two different metaheuristic algorithms that are GA and PSO. The reason for using these is that these algorithms have similarities. For the time being, we just validated the first scenario that is throughput, and the unified objective function after assigning more weight to throughput was tested with GA and PSO for time slot t_1. It was found that the same relay which was selected with our weighted sum iterative method that is relay no. 5 was declared optimal by both GA and PSO. The main idea of GA [66] is to mimic the natural selection and the survival of the fittest.

The flow diagram of the algorithm is given in Figure 12, which further elaborates the concept.

For our simulation, a population size of 100 and 50 iterations was selected. Figure 13 shows the results of the simulations done using GA.

Particle swarm optimization (PSO) was developed by Kennedy and Eberhart in 1995 [67]. It is based on the concept of swarm intelligence. Here, individual knowledge is inferior, but collective knowledge and sharing of information makes it much superior. The mathematical model consists of large number of particles with two basic properties, which are velocity and position. In each iteration, these two are modified/improved towards the optimal solution. Figure 14 gives the flow diagram of particle swarm algorithm.

Again, we selected a swarm size of 100 and 50 iterations. When the simulations were carried out using PSO, for selecting best out of 8 nodes, based on QoS requirements (scenario I: throughput), the results of simulations are shown in Figure 15, which were similar to GA. The results also showed that PSO under these conditions performed better than others giving an optimal value in less iteration. That is, GA took little more iterations to converge. This is a very preliminary analysis and comparison with two different algorithms which are GA and PSO. More detailed analysis and comparisons are left for future work.

6. Conclusions

The performance of a cognitive radio network (CRN) can be significantly enhanced by integrating buffer-aided relaying in it. Relay selection is a critical component in achieving the levels of performance desired, for a relaying in the network. Generally, the main criteria for a relay selection have been SINR; however, other factors may be equally important for an efficient wireless network, such as throughput, delay, battery life, and buffer status, and thus should be given due consideration. More specifically, in today's communication scenarios with multimedia-rich applications, having diverse QoS needs, relay selection has to be optimized for the various applications and traffic profiles. In this context, the focus of this work has been to devise a relay selection scheme that enables us to choose the best relay for the specific QoS profile meeting application requirements. A multiobjective optimization problem is formulated with throughput, delay, battery power, and buffer status as the main objectives under the SINR constraint. As these objectives contradict each other, multiobjective optimization with the weighted sum approach is employed. Several QoS profiles are defined by setting the ranks for the multiple objectives based on a certain order of priorities using the rank sum (RS) method. Appropriate

weights are assigned to objectives in the QoS profiles that reflect varying QoS requirements of the application traffic scenarios. The proposed relay selection scheme provides an optimal solution corresponding to each of the various application scenarios, which then consequently determines the optimal relay for the particular QoS profile corresponding to each application scenario. The results gathered through the relay selection mechanism are validated using genetic algorithm (GA) and particle swarm optimization (PSO) as well. The comprehensive analysis carried out in this work clearly indicates that the proposed relay selection scheme is well suited to ascertain the most suitable relays for the enhanced performance of emerging applications with a diverse range of QoS requirements. In this work, we assumed that there is minimal effect of a slight increase in overhead for communicating its resources. But a thorough cost/benefit analysis of overhead and effect of inaccurate or delayed CSI and resource information warrants thorough investigation. We intend to take up the effect of inaccurate or delayed CSI and also include buffer and battery state-based Markov modeling in our future research work.

Data Availability

Data is available on request from the corresponding author.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This work was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Korea Government (MSIT) (Grant No.: NRF-2020R1F1A1048470) (Grant No.: NRF-2019R1A4A1023746).

References

[1] ITU-R Report M 2370-0, "IMT traffic estimates for the years 2020 to 2030," 2015.

[2] Office of Spectrum Management, National Telecommunications and Information Administration, United States Department of Commerce, "United States frequency allocations - the radio spectrum," in *United States Frequency Allocations: The Radio Spectrum Chart*, NTIA, United States Department of Commerce, 2016, https://www.ntia.doc.gov/.

[3] A. M. Wyglinski, M. Nekovee, and T. Hou, *Cognitive radio communications and networks: principles and practice*, Academic Press, 2009.

[4] Pakistan Telecommunication Authority, "Pakistan table of frequency allocations (9 KHz-1000 GHz)," in *Frequency Allocation Board*, Government of Pakistan, 2005.

[5] FCC, "Before the Federal Communications Commission Washington DC," 2013, https://www.cs.ucdavis.edu/liu/289I/Material/FCC-03-322A1.pdf.

[6] M. A. McHenry, P. A. Tenhula, D. McCloskey, D. A. Roberson, and C. S. Hood, "Chicago spectrum occupancy measurements and analysis and a long-term studies proposal," in *Proceeding of TAPAS '06 Proceedings of the First International Workshop on Technology and Policy for Accessing Spectrum*, Boston, Massachusetts, USA, August 2006.

[7] S. Ali, Z. Chen, and F. Yin, "Spectrum occupancy of cellular networks in Pakistan for cognitive radio-measurements using spectrum analyzer," *International Journal of Information and Electronics Engineering*, vol. 6, no. 1, 2016.

[8] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty, "Next generation/dynamic spectrum access/cognitive radio wireless networks: a survey," *Computer Networks*, vol. 50, no. 13, pp. 2127–2159, 2006.

[9] R. Tandra, A. Sahai, and S. M. Mishra, "What is a spectrum hole and what does it take to recognize one?," *Proceedings of the IEEE*, vol. 97, no. 5, pp. 824–848, 2009.

[10] J. Mitola and G. Q. Maguire, "Cognitive radio: making software radios more personal," *IEEE Personal Communications*, vol. 6, no. 4, pp. 13–18, 1999.

[11] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty, "A survey on spectrum management in cognitive radio networks," *IEEE Communications Magazine*, vol. 46, no. 4, pp. 40–48, 2008.

[12] L. Giupponi and C. Ibars, "Distributed cooperation in cognitive radio networks: overlay versus underlay paradigm," in *IEEE 69th Vehicular Technology Conference (VTC) Spring*, Barcelona, Spain, 2009.

[13] Z. Shu, Y. Qian, and R. Q. Hu, "Delay based channel allocations in multi-hop cognitive radio networks," in *International Wireless Communications and Mobile Computing Conference (IWCMC)*, Dubrovnik, Croatia, 2015.

[14] N. B. Mehta, V. Sharma, and G. Bansal, "Performance analysis of a cooperative system with rateless codes and buffered relays," *IEEE Transactions on Wireless Communications*, vol. 10, no. 4, pp. 1069–1081, 2011.

[15] R. A. Sultan, A. K. Sultan, and M. Youssef, "Buffered-relay selection in an underlay cognitive radio network," in *Proceedings of International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt)*, pp. 201–207, Tsukuba, Japan, 2013.

[16] N. Zlatanov, R. Schober, and P. Popovski, "Throughput and diversity gain of buffer-aided relaying," in *Proceedings of IEEE Global Communication Conference (GlobeCom)*, pp. 1–6, Houston, TX, USA, 2011.

[17] S.-N. Hong and G. Caire, "Virtual full-duplex relaying with half-duplex relays," *IEEE Transactions on Information Theory*, vol. 61, no. 9, pp. 4700–4720, 2015.

[18] H. Kartlak, N. Odabasioglu, and A. Akan, "Adaptive multiple relay selection and power optimization for cognitive radio networks," in *9th International Conference on Communications (COMM)*, Bucharest, Romania, 2012.

[19] H. Xing and X. Kang, "Optimization for DF relaying cognitive radio networks with multiple energy access points," in *IEEE Global Communications Conference, (GLOBECOM)*, Washington, DC, USA, 2016.

[20] H. Kartlak, N. Odabasioglu, and A. Akan, "Cooperation methods for cognitive radio systems," *Wireless Personal Communications*, vol. 94, no. 4, pp. 3217–3232, 2017.

[21] J. N. Laneman, D. N. C. Tse, and G. W. Wornell, "Cooperative diversity in wireless networks: efficient protocols and outage behavior," *IEEE Transactions on Information Theory*, vol. 50, no. 12, pp. 3062–3080, 2004.

[22] A. Bletsas, A. Khisti, D. P. Reed, and A. Lippman, "A simple cooperative diversity method based on network path

selection," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 3, pp. 659–672, 2006.

[23] A. Bletsas, H. Shin, and M. Win, "Cooperative communications with outage-optimal opportunistic relaying," *IEEE Transactions on Wireless Communications*, vol. 6, no. 9, pp. 3450–3460, 2007.

[24] N. Nomikos, T. Charalambous, I. Krikidis, D. N. Skoutas, D. Vouyioukas, and M. Johansson, "A buffer-aided successive opportunistic relay selection scheme with power adaptation and inter-relay interference cancellation for cooperative diversity systems," *IEEE Transactions on Communications*, vol. 63, no. 5, pp. 1623–1634, 2015.

[25] G. Chen, Z. Tian, Y. Gong, and J. Chambers, "Decode-and-forward buffer-aided relay selection in cognitive relay networks," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 9, pp. 4723–4728, 2014.

[26] M. Darabi, B. Maham, X. Zhou, and W. Saad, "Buffer-aided relay selection with interference cancellation and secondary power minimization for cognitive radio networks," in *IEEE International Symposium on Dynamic Spectrum Access Networks (DYSPAN)*, pp. 137–140, McLean, VA, USA, 2014.

[27] D. Poulimeneas, T. Charalambous, N. Nomikos, I. Krikidis, D. Vouyioukas, and M. Johansson, "Delay-and diversity-aware buffer-aided relay selection policies in cooperative networks," in *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1–6, Doha, Qatar, 2016.

[28] F. Benkhelifa, A. S. Salem, and M.-S. Alouini, "Rate maximization in MIMO decode-and-forward communications with an EH relay and possibly imperfect CSI," *IEEE Transactions on Communications*, vol. 64, no. 11, pp. 4534–4549, 2016.

[29] I. Krikidis, T. Charalambous, and J. S. Thompson, "Buffer-aided relay selection for cooperative diversity systems without delay constraints," *IEEE Transactions on Wireless Communications*, vol. 11, no. 5, pp. 1957–1967, 2012.

[30] Z. Tian, Y. Gong, G. Chen, and J. A. Chambers, "Buffer-aided relay selection with reduced packet delay in cooperative networks," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 3, pp. 2567–2575, 2017.

[31] S. Alvi, R. Hussain, Q. Hasan, and S. Malik, "Improved buffer-aided multi-hop relaying with reduced outage and packet delay in cognitive radio networks," *Electronics*, vol. 8, no. 8, p. 895, 2019.

[32] J.-H. Cho, Y. Wang, I.-R. Chen, K. S. Chan, and A. Swami, "A survey on modeling and optimizing multi-objective systems," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1867–1901, 2017.

[33] Z. Fei, B. Li, S. Yang, C. Xing, H. Chen, and L. Hanzo, "A survey of multi-objective optimization in wireless sensor networks: metrics, algorithms, and open problems," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 1, pp. 550–586, 2017.

[34] M. Iqbal, M. Naeem, A. Anpalagan, A. Ahmed, and M. Azam, "Wireless sensor network optimization: multi-objective paradigm," *Sensors*, vol. 15, no. 7, pp. 17572–17620, 2015.

[35] M. R. Ramzan, N. Nawaz, A. Ahmed, M. Naeem, M. Iqbal, and A. Anpalagan, "Multi-objective optimization for spectrum sharing in cognitive radio networks: a review," *Pervasive and Mobile Computing*, vol. 41, pp. 106–131, 2017.

[36] W. Lu, Y. Lin, H. Peng, T. Nan, and X. Liu, "Joint resource optimization for cognitive sensor networks with SWIPT-enabled relay," *Sensors*, vol. 17, no. 9, article 2093, 2017.

[37] S. Yaolian, Z. Fan, S. Yubin, and L. Hua, "Energy efficiency optimization of cognitive relay network based on cooperative spectrum sensing," *The Journal of China Universities of Posts and Telecommunications*, vol. 22, no. 3, pp. 26–34, 2015.

[38] H. Hu, H. Zhang, and H. Yu, "Throughput-delay trade-off for cognitive radio networks: a convex optimization perspective," *Abstract and Applied Analysis*, vol. 2014, Article ID 430696, 8 pages, 2014.

[39] M. Soleimanpour-moghadam, M. Askarizadeh, S. Talebi, and S. Esmaeili, "Low complexity green cooperative cognitive radio network with superior performance," *IEEE Systems Journal*, vol. 13, no. 1, pp. 345–356, 2019.

[40] M. Naeem, A. S. Khwaja, A. Anpalagan, and M. Jaseemuddin, "Green cooperative cognitive radio: a multi-objective optimization paradigm," *IEEE Systems Journal*, vol. 10, no. 1, pp. 240–250, 2016.

[41] A. M. Elmahdy, A. El-Keyi, T. ElBatt, and K. G. Seddik, "Optimizing cooperative cognitive radio networks performance with primary QoS provisioning," *IEEE Transactions on Communications*, vol. 65, no. 4, pp. 1451–1463, 2017.

[42] H. Thampy and A. V. Babu, "Outage probability analysis and optimization of cognitive full-duplex relay networks," *Wireless Personal Communications*, vol. 105, no. 4, pp. 1329–1352, 2019.

[43] J. Zhang, L. Dai, Z. Wang, and D. W. Ng, "Optimization for DF relaying cognitive radio networks with multiple energy access points," in *IEEE Global Communications Conference, (GLOBECOM)*, Washington, DC, USA, 2016.

[44] M. Minupriya, "An efficient algorithm for throughput maximization and delay minimization in cognitive radio wireless mesh network," *International Journal of Engineering Research and Technology (IJERT)*, vol. 5, no. 13, 2017.

[45] M. Taki, T. Svensson, and M. B. Nezafati, "Delay constrained throughput optimization in multi-hop AF relay networks, using limited quantized CSI," *EURASIP Journal on Wireless Communications and Networking*, vol. 2019, no. 1, 2019.

[46] S. Gautam, T. X. Vu, S. Chatzinotas, and B. Ottersten, "Cache-aided simultaneous wireless information and power transfer (SWIPT) with relay selection," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 1, pp. 187–201, 2019.

[47] Z. Zhang, J. Shi, H.-H. Chen, M. Guizani, and P. Qiu, "A cooperation strategy based on Nash bargaining solution in cooperative relay networks," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 4, pp. 2570–2577, 2008.

[48] N. N. Srinidhi, S. M. D. Kumar, and K. R. Venugopal, "Network optimizations in the Internet of Things: a review," *Engineering Science and Technology, an International Journal*, vol. 22, no. 1, pp. 1–21, 2019.

[49] M. Alvandi, M. Mehmet-Ali, and J. F. Hayes, "Delay optimization and cross-layer design in multihop wireless networks with network coding and successive interference cancelation," *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 2, pp. 295–308, 2015.

[50] P. Xu, Z. Ding, I. Krikidis, and X. Dai, "Achieving optimal diversity gain in buffer-aided relay networks with small buffer size," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 10, pp. 8788–8794, 2016.

[51] T. K. Vu, M. Bennis, M. Debbah, and M. Latva-Aho, "Joint path selection and rate allocation framework for 5G self-backhauled mm-wave networks," *IEEE Transactions on Wireless Communications*, vol. 18, no. 4, pp. 2431–2445, 2019.

[52] R. Zhang, C. Qi, Y. Li, Y. Ruan, C.-X. Wang, and H. Zhang, "Towards energy-efficient underlaid device-to-device communications: a joint resource management approach," *IEEE Access*, vol. 7, pp. 31385–31396, 2019.

[53] W. Raza, H. Nasir, N. Javaid, M. Imran, and M. Guizani, "Buffer size and link quality based cooperative relay selection in wireless networks," in *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC)*, pp. 1489–1494, Valencia, 2017.

[54] S. Ullah, M. H. Malik, M. F. Tuysuz, M. Hasnain, and M. E. Aydin, "Max-gain relay selection scheme for wireless networks," *Engineering Science and Technology, an International Journal*, vol. 24, no. 1, pp. 183–191, 2021.

[55] T. Kaur and D. Kumar, "MACO-QCR: multi-objective ACO-based QoS-aware cross-layer routing protocols in WSN," *IEEE Sensors Journal*, vol. 21, no. 5, pp. 6775–6783, 2021.

[56] S. Poornima and A. V. Babu, "Power adaptation for enhancing spectral efficiency and energy efficiency in multi-hop full duplex cognitive wireless relay networks," *IEEE Transactions on Mobile Computing*, p. 1, 2020.

[57] S.-L. Lin and K.-H. Liu, "Relay selection for cooperative relaying networks with small buffers," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 8, pp. 6562–6572, 2016.

[58] G. Appenzeller, I. Kaslassy, and N. McKeown, "Sizing router buffers," *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 4, 2004.

[59] S. C. F. Chan, K. M. Chan, K. Liu, and J. Y. B. Lee, "On queue length and link buffer size estimation in 3G/4G mobile data networks," *IEEE Transactions on Mobile Computing*, vol. 13, pp. 1298–1311, 2014.

[60] A. H. Sodhro, L. Chen, A. Sekhari, Y. Ouzrout, and W. Wu, "Energy efficiency comparison between data rate control and transmission power control algorithms for wireless body sensor networks," *International Journal of Distributed Sensor Networks*, vol. 14, no. 1, 2018.

[61] T. Murata, H. Ishibuchi, and H. Tanaka, "Multi-objective genetic algorithm and its applications to flowshop scheduling," *Computers & Industrial Engineering*, vol. 30, no. 4, pp. 957–968, 1996.

[62] R. T. Marler and J. S. Arora, "Survey of multi-objective optimization methods for engineering," *Structural and Multidisciplinary Optimization*, vol. 26, no. 6, pp. 369–395, 2004.

[63] J. Jia, G. W. Fischer, and J. S. Dyer, "Attribute weighting methods and decision quality in the presence of response error: a simulation study," *Journal of Behavioral Decision Making*, vol. 11, no. 2, pp. 85–105, 1998.

[64] H. J. Einhorn and W. McCoach, "A simple multi-attribute utility procedure for evaluation," *Behavioral Science*, vol. 22, no. 4, pp. 270–282, 1977.

[65] A. Ikhlef, J. Kim, and R. Schober, "Mimicking full-duplex relaying using half-duplex relays with buffers," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 7, pp. 3025–3037, 2012.

[66] Y. Huang and S. Chan, "Optimizing the performance of genetic algorithms for finding the optimal value of a given function," in *[1991] Proceedings of the 34th Midwest Symposium on Circuits and Systems*, vol. 2, pp. 819–822, Monterey, CA, USA, 1991.

[67] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95- International Conference on Neural Networks*, vol. 4, pp. 1942–1948, Perth, WA, Australia, 1995.