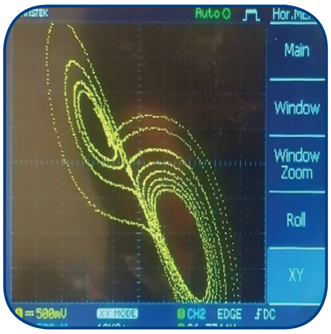
Volume 14, Number 1, 2023

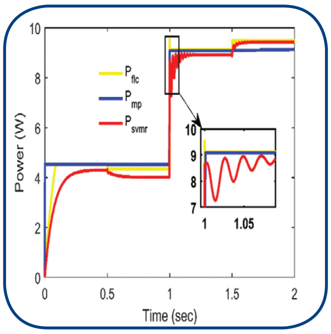
ISSN 1847-6996

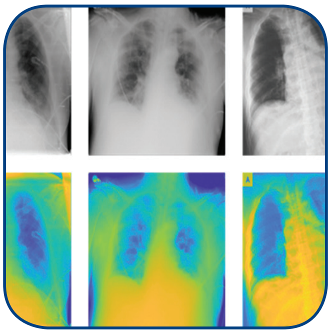
FERIT
FACULTY OF ELECTRICAL ENGINEERING, COMPUTER
SCIENCE AND INFORMATION TECHNOLOGY **OSIJEK**

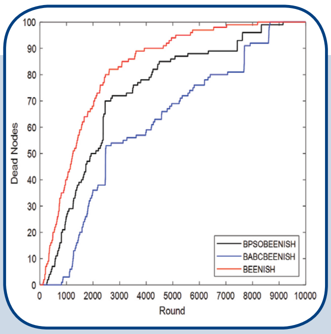
IJECES
International Journal
of Electrical and Computer
Engineering Systems

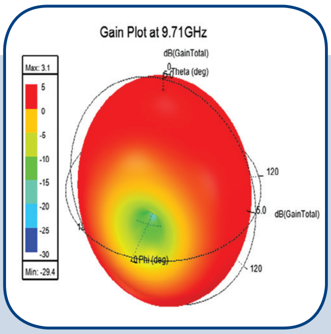
# International Journal of Electrical and Computer Engineering Systems











ISSN: 1847-6996

# INTERNATIONAL JOURNAL OF ELECTRICAL AND COMPUTER ENGINEERING SYSTEMS

Published by Faculty of Electrical Engineering, Computer Science and Information Technology Osijek,
Josip Juraj Strossmayer University of Osijek, Croatia

Osijek, Croatia | Volume 14, Number 1, 2023 | Pages 1 - 118

The International Journal of Electrical and Computer Engineering Systems is published with the financial support of the Ministry of Science and Education of the Republic of Croatia

**CONTACT**

**International Journal of Electrical**
**and Computer Engineering Systems**
**(IJECES)**

Faculty of Electrical Engineering, Computer
Science and Information Technology Osijek,
Josip Juraj Strossmayer University of Osijek, Croatia
Kneza Trpimira 2b, 31000 Osijek, Croatia
Phone: +38531224600, Fax: +38531224605
e-mail: ijeces@ferit.hr

**Subscription Information**

The annual subscription rate is 50€ for individuals, 25€ for students and 150€ for libraries.
Giro account: 2390001 - 1100016777,
Croatian Postal Bank

**Proofreader**

**Ivanka Ferčec**
J.J. Strossmayer University of Osijek, Croatia

**Editing and technical assistence**

**Davor Vrandečić**
J.J. Strossmayer University of Osijek, Croatia

**Stephen Ward**
J.J. Strossmayer University of Osijek, Croatia

**Dražen Bajer**
J.J. Strossmayer University of Osijek, Croatia

**Journal is referred in:**

- Scopus
- Web of Science Core Collection (Emerging Sources Citation Index - ESCI)
- Google Scholar
- CiteFactor
- Genamics
- Hrčak
- Ulrichweb
- Reaxys
- Embase
- Engineering Village

**Bibliographic Information**

Commenced in 2010.
ISSN: 1847-6996
e-ISSN: 1847-7003
Published: quarterly
Circulation: 300

**IJECES online**

https://ijeces.ferit.hr

# TABLE OF CONTENTS

# Enhancing Heterogeneous Wireless Sensor Networks Using Swarm Intelligence –Based Routing Protocols

Original Scientific Paper

**Salima Nebti**
Emir Abdelkader University, Department of Media and Communication
Street Filali, Constantine, Algeria
snebti@live.fr

**Mohamed Redjimi**
20 Aout 1955 University, Faculty of Sciences, Department of Computer Science
Route El-Hadaiek, Skikda, Algeria
medredjimi@gmail.com

***Abstract*** – *The design of efficient communication protocols for wireless sensor networks has aroused great interest in the research community, especially in the face of the limited energy of sensor nodes and the frequent change in network topology. Routing remains a challenging problem in wireless communications, as deploying or replacing sensor nodes in hazardous environments is difficult. Many studies have been devoted to alleviate certain limitations, such as clustering to maintain network connectivity, injecting heterogeneity to avoid the rapid death of nodes, or incorporating evolution-based optimization methods to find the best network configuration. This work combined heterogeneity and swarm-based optimization to efficiently balance energy consumption between nodes to increase network reliability. Specifically, this work employed the binary particle swarm optimizer and the binary artificial bees colony optimizer to find approximately the optimal set of cluster heads (CHs) with their optimal number. Based on the probabilistic principle of the heterogeneous protocols: SEP, EDEEC, and BEENISH, a new refined formulation of CHs selection using swarm optimization is proposed. The swarm flight is guided towards the best CHs with an objective function representing a good balance between the initial and residual energy of nodes. Compared to the standard heterogeneous protocols SEP, EDEEC, and BEENISH, the developed protocols significantly perform better in terms of stability (FND), the round of half nodes' death (HND), the network lifetime (LND), and energy saving. Indeed, the BABC-SEP was found 31,66% better than SEP in terms of remaining energy percentage, and CHs selection in EDEEC and BEENISH using BABC improved them by more than 20% in the percentage of remaining energy.*

***Keywords***: *Wireless sensor networks, SEP, EDEEC, BEENISH, Binary PSO, Binary ABC, energy efficiency*

## 1. INTRODUCTION

Wireless sensor networks (WSNs) are an increasingly attractive area of research due to their simplicity, adaptability, scalability, fault tolerance, and ability to remotely monitor hostile environments. This new technology is now being investigated in various domains, such as medicine, industry, agriculture, ecology, military domain, etc. A communication protocol is a fundamental function of wireless communications, which aims to discover the best route that saves energy and ensures rapid data delivery. In ad-hoc networks, routing is performed by specific nodes, called routers, which are often physically protected. Whereas in a wireless network, the routing is performed by sensors themselves [1], this is why a sensor failure can generate a significant loss of information, and deteriorate dramatically the network reliability. Moreover, it is well-known that limited sensor power is the main cause of node failure, and has long imposed a great challenge on the research community [2]. Furthermore, maintaining network connectivity, self-reconfiguration, reliability, and latency are great challenges in designing wireless networks [3], [4].

The clustering-based protocols represent an effective solution to some of these problems. In clustering approaches, nodes are divided into groups, each joining the nearest cluster head based on its signal strength. Usually, the cluster head is a node with higher energy capacity and is responsible for processing and aggregating data collected from its member nodes to reduce data redundancy and hence the network latency [5]. Furthermore, multi-hop clustering approaches can help transmit data packets within the communication range of sensor nodes and thus maintain network connectivity and data reliability [6].

Most of the clustering-based protocols focus on extending the network lifetime without considering network stability or the first node death period, which is a fundamental factor for many real-world applications of WSNs. Heterogeneous networks have been introduced recently to extend the network lifetime and its stability period. In heterogeneous protocols, some nodes are powered with higher energy capacity to perform additional tasks. Typically, these nodes act as cluster heads for more data reliability and longer network stability.

Several heterogeneous communication protocols have been proposed, such as SEP, EDFCM, and ZREECR, which are more stable than energy-efficient, and the DEEC-based protocols, such as EDEEC, DDEEC, which are much more energy-efficient than stable [7].

Despite their variety, the proposed solutions remain limited since the CHs selection process is probabilistic in their setup phase. Finding the optimal set of cluster heads is a Non-deterministic Polynomial (NP)-hard problem, which involves searching in a vast space for potential solutions [8]. Swarm-based methods have been proven effective in solving NP-hard complex problems.

In this work, an effort is made to improve the standard heterogeneous routing protocols, namely, SEP, EDEEC, and BEENISH, based on swarm optimization in their setup phases to select the most powerful cluster heads (CHs). More specifically, the binary particle swarm optimizer (PSO) and the binary artificial bees colony optimizer (ABC) are used to select the best CHs in terms of their initial and remaining energy; the main purpose is to prevent the quick death of nodes to extend the network lifetime and to refine data reliability. Compared to SEP, EDEEC, and BEENISH the obtained results were improved in terms of stability (FND), the round of half-node death (HND), the network lifetime (LND), the number of packets delivered to the base station and energy saving.

The rest of the article is organized as follows:

In the second section, some of the closely related works are briefed. Section 3 describes the principle of the used heterogeneous protocols, namely, SEP, EDEEC, and BEENISH protocols. Section 4 presents the introduced optimization techniques in the mentioned communication protocols and their adaptation to select the best cluster heads (CHs). This work ends with a conclusion and some perspectives.

## 2. RELATED WORKS

Swarm intelligence optimization methods are robust and concurrent optimization techniques without centralized control, which mimic the natural collective behavior of animal groups to solve complex problems that have a vast search space for potential solutions [9].

Selecting the most powerful cluster heads, the shortest routes between nodes, or enhancing latency and reliability constitute the focus of swarm-based communication protocols. This section presents some of these broad swarm-based contributions.

The idea explored in [10] is to use Refined Bacterial Foraging Optimization (RBFO) and Hybrid BFO-BSO (Bee swarm Optimization) to select the Cluster Heads in WSNs. The considered objective function is a weighted sum of the Packet Loss Ratio and the minimum remaining energy divided by the initial energy of a node. Results proved that RBFO and the Hybrid BFO-BSO provided better performance in terms of power conservation, the packet loss rate, and the end-to-end delay with respect to KBFO and LEACH.

In order to extend the network lifetime, D. Karaboga et al [11], introduced the ABC optimizer in the setup phase of LEACH to efficiently select the cluster head nodes. The considered objective function is the sum of the distances between nodes and their CHs and the distances between the CHs and the base station.

The work presented by M. A. Latiff et al is another centralized PSO-based protocol. The optimized objective function is a weighted sum of Euclidean distances between nodes and their CHs, and the network's remaining energy. Results were better than LEACH, LEACH-C, GA, and K-means [12].

An ACO and ABC-based approach for route construction is presented by J.C. Blandón et al [13], where node selection is relayed on their energy and their distances to the base station. Results were better in terms of energy conservation compared to a non-bio-Inspired algorithm.

Another route establishment approach based on a cooperative PSO to find the best path from a source node to the nearest mobile sink is developed by Y.F. Hu et al [14]. In this work, each node represents a particle, and the set of particles with the best Fitness is selected for data routing to the sink node. The optimized function is the sum of particles' remaining energy divided by a weighted function taking into account the distance, the consumed energy, and the communication delay between nodes. The obtained results were superior to IAR and TTDD protocols in terms of delay and energy.

In this paper [15], the selection of the best cluster heads is improved by the ABC optimization method. The work also used the polling control based on busy/idle nodes in the steady state phase to improve energy conservation.

Wang et al [16], used the CGTABC algorithm for cluster-head selection in the setup phase of LEACH and used an ACO-based routing algorithm to find the best routes between CHs and the base station.

A PSO-based approach for path discovery from sender nodes to the Sink is presented in [17]; the considered objective function is only based on the sum of distances between nodes building the path to the Sink. The PSO-based path discovery performs better than GA based algorithm in terms of energy efficiency.

In [18], an improved artificial bees colony algorithm is used to generate routing paths in a multi-hop clustering-based approach. The optimized function is based on the average energy of the routing path, its minimum energy, and the length of the shortest path. The cluster head selection is based on their remaining energy and the average energy of their member nodes. This approach extended the network lifetime compared to LEACH, EEUC, and MSDG protocols.

In [19], ABC and ALO optimization algorithms were used for CHs and their vicinity CHs (VCHs) selection in the setup phase of the LEACH protocol. The ALO-LEACH protocol outperformed ABC-LEACH in terms of energy consumption, throughput, and the number of alive nodes.

Despite the rich literature on swarm intelligence-based protocols to prolong the lifetime of WSNs, the quick death of some nodes cannot be avoided. To overcome this drawback, heterogeneous protocols have been introduced and improved based on swarm optimization. Examples of heterogeneous swarm-based protocols include the work presented in [20], in which a ring clustering-based approach whose cluster head selection is performed by the PSO method in heterogeneous sensor networks.

Another heterogeneous fault-tolerant and energy-efficient protocol to solve the hotspot problem is presented in [21]. This approach allowed better allocation of time transmission slots in the TDMA protocol using the PSO method and provided a longer network lifetime compared to other heterogeneous protocols such as CEEC and E-BEENISH. A PSO-based approach for the CHs selection in a three-level heterogeneous network is presented in [22]. This approach resulted in better performance in terms of network lifetime, stability period, throughput, and scalability compared to SEP and LEACH.

As exhibited above, most of the swarm intelligence-based protocols incorporate swarm optimization techniques in relatively old homogeneous protocols such as LEACH. The main drawback of LEACH-based protocols is the rapid death of CH nodes, which deteriorates the data reliability, and shortens the network lifetime. The focus of this paper is to study the effectiveness of swarm optimization in heterogeneous protocols. To this end, a new formulation of cluster head selection based on BPSO or BABC in two, three, and four-level heterogeneous networks is proposed. The achieved protocols enabled better results in terms of stability (FND), the round of half nodes' death (HND), and the network lifetime (LND) compared to SEP, EDEEC, and BEENISH protocols.

### 2.1 SEP PROTOCOL (STABLE ELECTION PROTOCOL)

SEP is a heterogeneous protocol designed for the routing of two energy level networks consisting of normal nodes with initial energy Eo and advanced nodes with more energy: Eo×(1+a); "a" is a positive real value. Being selected based on their initial energy; the advanced nodes are more likely to become CHs using the probabilistic equations below [23], [24]:

$$P_{norm} = \frac{P}{1 + a.m} \tag{1}$$

$$P_{advan} = \frac{P}{1 + a.m} * (1 + a) \tag{2}$$

$m$: is the fraction of advanced nodes

In SEP, each node generates a random number between 0 and 1. If this number is less than a threshold that takes into account its initial energy and the number of rounds in which it is not elected as a CH, this node will take the role of a cluster head.

The threshold is defined for each type of node (normal $S_n$ or advanced $S_a$) as per the equations below [24]:

$$T(S_n) = \begin{cases} \dfrac{P_n}{1 - P_n * \left(r \bmod \dfrac{1}{P_n}\right)}, if\ S_n \in G' \\ 0 \qquad , otherwise \end{cases} \tag{3}$$

$$T(S_a) = \begin{cases} \dfrac{P_a}{1 - P_a * \left(r \bmod \dfrac{1}{P_a}\right)}, if\ S_a \in G'' \\ 0 \qquad , otherwise \end{cases} \tag{4}$$

$G'$ and $G''$ are, respectively, the set of normal nodes and the set of advanced nodes which have not been elected as CHs in the last $1/P_n$ and $1/P_a$ rounds.

After cluster head identification, each node joins the group of its closest cluster head, and the communication within each group is planned according to the TDMA protocol, where each cluster head establishes a transmission schedule between its member nodes to avoid cohesion and to conserve the node energy in its waiting or idle states. The cluster heads communicate with the base station according to CSMA protocol to verify the channel availability and ensure data delivery.

### 2.2. EDEEC PROTOCOL

EDEEC is designed for the routing of three-levels heterogeneous networks consisting of normal nodes, advanced nodes and super nodes according to their initial energy: $E_0$, ($E_0.a$) and ($E_0.b$) respectively, with $a$ >1 and $b$>a. The selection of nodes as CHs is based on their types, their residual energy $E_i(r)$ and the r$^{th}$ network average energy $\bar{E}_i(r)$ as formulated by the following equations [25]:

$$P_{i\,norm} = \frac{P.E_i(r)}{(1 + a + b).\bar{E}(r)}$$

$$P_{i\,advan} = \frac{P.E_i(r).a}{(1 + a + b).\bar{E}(r)} \tag{5}$$

$$P_{i\,super} = \frac{P.E_i(r).b}{(1 + a + b).\bar{E}(r)}$$

$E_i(r)$ is the residual energy of node "I" in round "r" $\bar{E}_i(r)$ represents the r$^{th}$ network average energy at round r, which is calculated as below:

$$\bar{E}(r) = \frac{1}{N}.E_{total}\left(1 - \frac{r}{R}\right) \quad (6)$$

$R$ is the estimated network lifetime and is calculated as:

$$R = \frac{E_{total}}{E_{round}} \quad (7)$$

$$E_{round} = k\begin{pmatrix} 2NEelec + NEda \\ +S.Emp.d^4_{CH\ to\ BS} + N.Efsd^2_{N\ to\ CH} \end{pmatrix} \quad (8)$$

$k$ is the packet size.

$S$ is the optimal number of cluster heads and calculated as

$$S = \sqrt{\frac{N}{2\pi}}.\frac{xm}{d_{to\ BS}}.d_o$$

$d_{to\ CH}$ and $d_{to\ BS}$ Are respectively the average distance between CH and member nodes, and the average distance between a cluster head and the base station.

$$d_{to\ CH} = \frac{xm}{\sqrt{2\pi k}}, d_{to\ BS} = 0.765\frac{xm}{2},$$

In EDEEC, each node generates a random value between 0 and 1 if this value is lower than the threshold $T(S_i)$ calculated as below, then this node becomes a cluster head [25]:

$$\begin{cases} T(S_i) = \dfrac{p}{1 - p * r\ mod(\frac{1}{p})}, & S_i \in G \\ T(S_i) = 0\ , & otherwise \end{cases} \quad (9)$$

$p$ is calculated by equation (5) and represents the related selection probability of a node type.

### 2.3. BEENISH PROTOCOL

BEENISH is designed for routing heterogeneous wireless networks constituted of four types of nodes, called respectively: normal, advanced, super, and ultra-super nodes, according to their initial energy: $E_0$, $(E_0.a)$,$(E_0.b)$ and $(E_0.u)$, with $u > b > a$. The nodes' selection as CHs is based on their types, their residual energy $E_i(r)$, and the rth network average energy $E(r)$ as formulated by equation (10) below [26]:

$$\begin{aligned} P_{i\ norm} &= \frac{P.E_i(r)}{\left(1+m.\left(a+m_0.(-a+b+m_1.(-b+u))\right)\right).\bar{E}(r)} \\ P_{i\ advan} &= \frac{P.(1+a).E_i(r)}{\left(1+m.\left(a+m_0.(-a+b+m_1.(-b+u))\right)\right).\bar{E}(r)} \\ P_{i\ super} &= \frac{P.(1+b).E_i(r)}{\left(1+m.\left(a+m_0.(-a+b+m_1.(-b+u))\right)\right).\bar{E}(r)} \\ P_{i\ ultra} &= \frac{P.(1+u).E_i(r)}{\left(1+m.\left(a+m_0.(-a+b+m_1.(-b+u))\right)\right).\bar{E}(r)} \end{aligned} \quad (10)$$

As in EDEEC protocol, each node generates a random value between 0 and 1, if this value is less than the threshold $T(S_i)$ calculated by equation (9) on the corresponding $P_i$ of equation (10), then the node becomes a CH.

## 3. THE PROPOSED WORK

In order to save more energy and keep the network running as long as possible, BPSO and BABC have been introduced in the setup stage of SEP, EDEEC, and BEENISH protocols. The objective is to find the most powerful CHs of each round to prevent their rapid death and consequently improve network reliability. In SEP, the role of being a CH is alternated between nodes by probabilistic equations taking into account the type of nodes (advanced or normal), the desired percentage of cluster heads, the set of unelected nodes as CHs, and the number of completed rounds [27]. In EDEEC and BEENISH protocols, the CH role alternation between nodes is based on probabilistic equations considering the node types, their residual energy, the $r^{th}$ network average energy, and the set of unelected nodes as CHs for a number of rounds.

In this work, the CHs selection is based on an optimization process guided by BPSO or BABC towards the best ones in terms of their number and energy. The objective is to find the approximate optimal set of CHs in terms of both their initial and residual energy, combining in such a way the principle of CHs selection in SEP, EDEEC, and BEENISH protocols.

The proposed approach is clustering-based, where each cluster head receives internal messages from its cluster members, aggregates similar packets, and acts as a gateway with the other cluster heads, which helps to reduce redundancy and therefore improves latency.

The clustering process is commonly performed in two main phases: the setup phase and the communication phase. In the setup phase, CHs identification and cluster formation are performed. While in the communication phase, the sensed data are forwarded from nodes to CHs via the TDMA protocol and then from CHs to the base station via the CSMA protocol. These steps are addressed in the next subsections

### 3.1. THE SETUP PHASE

In this phase, the cluster-head selection is performed centrally by the base station based on two powerful swarm intelligence-based methods. The list of found CHs is then broadcast to all nodes, where each of them joins the nearest cluster head (CH) according to the strength of its radio signal (RSSI). Then, each CH defines a transmission schedule with its member nodes based on the time division multiple access (TDMA) protocol.

To find an optimal network configuration, BPSO and BABC are suggested to solve the CHs selection problem. To do this, the structure of each solution, whether a particle or a bee, is a vector of binary values indicating whether the associated node is selected as a cluster head or not.

The trajectory of particles (bees) in the search space is guided by an objective function whose maximization favors the CHs with the greatest ratio of the sum of CHs' residual energies to the sum of their initial energies as formulated below:

$$F = \frac{\sum_{i=1}^{nb\ CHs} Er_i}{\sum_{i=1}^{nb\ CHs} Ei_i} \tag{11}$$

$Er_i$ is the residual energy of node i

$Ei_i$ is its initial energy

$nb\ CHs$ is the number of nodes elected as CH.

In another way, the preferred CHs are those with higher initial and higher residual energy.

3.1.1 THE BINARY ABC FOR CHS SELECTION

In ABC optimization, the artificial colony of bees is organized into three types of bees: Employed bees relating to food sources, Onlooker bees observing the dance of the employed bees to select a food source, and scout bees searching for random food sources [27].

**The ABC steps**

```
1. Bees initialization
2. For each iteration, do
3. Employed bees phase
4. Onlooker bees phase
5. Scout bees phase
6. End for
```

Updating Employed and Onlooker bees in binary ABC is based on the following steps [28], [29]:

- Produce a new bee (NewBee) in the neighborhood of the old "$d$" dimensional bee "$B$" by the equation below.

$$\begin{gathered} NewBee_d = Bee_{B,d} + \alpha.\varphi.\left(Bee_{B,d} - Bee_{K,d}\right) \\ \alpha = 1, \varphi\epsilon[-1,1], K \text{ is different from } B \end{gathered} \tag{12}$$

- Normalize the newfound Bee to binary values based on the sigmoid function; that is, if the normalized position ($NewBee$) by the sigmoid function is less than 0.5, then the $NewBee$ is set to 1 otherwise to 0.

An onlooker bee selects an employed bee "$G$" using the roulette wheel on the bees probabilities "$P(B)$" as below [27]:

$$F(B) = e^{-\frac{F(B)}{mean\ (F)}} \tag{13}$$

*F(B)* is the Fitness of the employed bee "$B$"

*mean (F)*: is the average Fitness of Employed Bees.

$$P(B) = \frac{F(B)}{\sum_{i=1}^{N} F(i)} \tag{14}$$

"$N$" is the number of employed bees

**The BABC-based routing protocol**

```
Input: A sink and a number of sensor nodes randomly positioned in the area of interest.
Output: Cluster heads identification and data routing.
Step 1: Network Initialization
1. Initialize fraction $m$ of $n$ nodes as advanced nodes with initial energy $E_0.(1+a)$ in the SEP-based protocols
2. Initialize fraction $m$ of $n$ nodes as intermediate nodes and fraction $mo$ of $m$ as super nodes with initial energy: $(E_0*a)$ and$(E_0*b)$ in the EDEEC-based protocol
3. Initialize a fraction $m$ of $n$ nodes as intermediate nodes, a fraction $mo$ of $m$ as super nodes and a fraction $m_1$ of $mo$ as ultra-super nodes respectively with$(E_0*a)$, $(E_0*b)$ and $(E_0*u)$ in the BEENISH- based protocol.
4. Initialize the rest of the normal nodes with $E_0$ energy capacity.
5. Initialize the Sink with unlimited energy power.
Step2: Bees initialization
6. Initialize a number of employed bees with random binary values and a size equal to the number of network nodes.
7. For each round, do
Step 3: The employed bees phase
8. For each employed bee B do
9. Produce a New Bee in the neighborhood of B using the equation (12).
10. Replace B with the New Bee if it is better in terms of Fitness (equation (11))
11. Otherwise, increase the bee B inefficiency counter
12. End for
Step 4: The onlooker Bees phase
13. For each onlooker bee, do
14. Select an employed bee "G" using the roulette wheel on the calculated probabilities by equations (13) & (14).
15. Produce a New Bee in the neighborhood of G by equation (12)
16. Normalize into binary the newfound Bee
17. Replace the bee G with the newfound Bee if it is better in terms of Fitness (equation (11))
18. Otherwise, increase the bee G inefficiency counter
```

```
19. End for
Step 5: The Scout bees phase
20. Randomly reset the ineffective solutions (their
    inefficiency counter is upper than a limit value)
21. Calculate the new Fitness of each employed
    Bee
Step 6: cluster heads identification
22. The nodes associated with value 1 in the best
    found employed Bee are the cluster heads of
    the current round.
23. The rest of the nodes join the closest cluster
    heads.
Step 7: The steady-state phase
24. Forward data from nodes to CHs based on
    TDMA protocol and from CHs to BS based on
    CSMA protocol.
25. Until a maximum number of rounds
```

#### 3.1.2 The binary PSO for CH selection

PSO is an optimization method, which attempts to imitate the collective flight of birds. In the basic PSO method, each solution called particle has a position (Pos) in the search space, a random speed (Velocity), a personal best solution (Pbest), and a global or swarm best solution (Gbest) [30].

```
The BPSO steps
1. Particles initialization
2. For each iteration, do
3.     Update Pbest
4.     Update Gbest
5.     Update velocity
6.     Update positions
7. End
```

The *Pbest* is the personal best-found solution of the particle, and *Gbest* is the best-found solution by the group of particles [31].

Particle velocity update in PSO is based on the equation below [18]:

$$V_{pd} = w.V_{pd} + c_1.r_1.\left(Pbest_{pd} - Pos_{pd}\right) + c_2.r_2.\left(Gbest - Pos_{pd}\right) \tag{15}$$

$c_1$, $c_2$ are respectively the cognitive and social factors, $r_1$ and $r_2 \in ]0, 1[$, $w$ is the inertia weight.

In binary optimization, the velocity of each particle is normalized between [0, 1] using the sigmoid function as per the equation below [32]:

$$V_{pd} = \frac{1}{\left(1+e^{-V_{pd}}\right)} \tag{16}$$

Then a random value between 0 and 1 is generated, if the $V_{pd}$ value is upper than the random value, then the normalized position is set to 1, otherwise to 0.

Below is the BPSO-based solution to CHs selection:

```
The BPSO-based routing protocol
Input: A sink, a number of nodes randomly deployed in the area of interest
Output: Cluster heads identification & packets routing
1.  Step 1: Network Initialization
2.  Step2: Particles initialization
3.  For each round, do
4.      Step 3: Particles evaluation using equation (11)
5.      Step 4: Update Pbest and Gbest
6.      Step 5: Update particles' velocity using eq (15)
7.      Step 6: Normalize velocity and update particles' positions
8.      Step 7: cluster heads identification
        The nodes associated with value 1 in the Gbest
        particle are the cluster heads of the current round
9.      Step 8: The communication phase
        Forward data from nodes to CHs and from CHs to
        BS based on TDMA & CSMA protocols. Update the
        network energy based on the first-order energy
        model.
10. Until the maximum number of rounds.
```

### 3.2. THE COMMUNICATION PHASE

In the communication phase (Steady-state phase), which is the same as in SEP, EDEEC, and BEENISH protocols, the CHs receive data from their member nodes and perform their aggregation according to TDMA protocol, and then send the compressed signals to the base station according to the CSMA protocol.

In order to simulate the energy expenditure by the electronic circuits of sensor nodes, the first-order radio energy model is implemented for better comparison as it is the most widely used model in clustering-based protocols [24].

Let $E_{Tx}$ and $E_{Rx}$ be respectively the consumed energy by the transmitter and the receiver circuits of a sensor node.

There are two channel models to transmit a k-bit packet to a receiver $M$ meters away:

The free-space channel model is used when the distance between a source node and the destination node is less than a predefined threshold as formulated below [33]:

$$E_{Tx}\,(k,M) = k \; \times Eelec + k \times efs \; \times M^2 \;, \;\; if \;\; M < d_0 \qquad (17)$$

$Eelec$ is the required energy by the electronic circuit of the transmitter.

$efs$ is the required energy by the amplifier circuit in free space

$d_0 = \sqrt{efs/emp}$ , is a distance threshold.

The multipath fading channel model is used to amplify the signal thus avoiding its degradation when the distance between the source and destination nodes is greater than the predefined threshold [33].

$$E_{Tx}\,(k,M) = \; k \; \times Eelec + k \times emp \; \times M^4 \;, if \;\; M > d_0 \qquad (18)$$

$emp$ : is the required energy by the amplifier circuit in multipath fading space.

The consumed energy by a CH node to receive a k-bit packet is [33]:

$$E_{Rx}(k) \; = k \; (Eelec + \text{EDA}) \qquad (19)$$

EDA: is the required energy for data aggregation.

## 4. RESULTS & DISCUSSION

Experiments were run in Matlab 2018, under Windows 10 with an Intel(R) Core(TM) i5-5300U, 2.30 GHz, and 4GB RAM. Sensors are powered with an initial energy of 0.5 Joules and the Sink is powered with unlimited energy.

**Table 1.** Parameters setting

| The network parameters | |
|---|---|
| The size of the detection area | 250×250 m$^2$ |
| Number of nodes | 100 |
| Initial Energy of each Node | 0.5 Joules |
| Eelec | 50 nano joules |
| Emp (the amplifier energy) | 100 Pico joules |
| EDA (Data Aggregation Energy) | 5 nano joules |
| K(Size of a data packet) | 4000 bits |
| **BPSO parameters** | |
| Number of particles | 20 |
| $C_1$=$C_2$ | 1.49 |
| W( inertia weight) | 0.78 |
| Velocity constriction | [-5, 5] |
| **BABC parameters** | |
| Number of employed bees | 20 |
| Number of Onlooker Bees | 20 |
| Abandonment Limit | 20 |
| $\alpha$ (Acceleration Coefficient) | 1 |
| Upper & Lower bounds | 5 & -5 |

Heterogeneity is injected into each network type according to these percentages: the fraction of ultra-super nodes, super nodes, advanced or intermediate nodes is respectively: m1=0.2, mo=0.3, m=0.5, and their corresponding energy factors are respectively: u=2.75, b=2.5, and a=2.12.

Table 2 presents the related data to residual and dead-node curves of figures 1, 2 and 3.

**Table 2.** Comparison in terms of FND, HND, and LND

| | FND | HND | LND | RES % | Time |
|---|---|---|---|---|---|
| BABC-SEP | 640 | 2471 | 7646 | 36,5 | 0,031 |
| BPSO-SEP | 217 | 1921 | 7493 | 22,21 | 0,047 |
| SEP | 444 | 1619 | 4883 | 4,84 | 0 |
| BABC-BEENISH | 815 | 2469 | 8604 | 34,2 | 0,032 |
| BPSO-BEENISH | 264 | 1929 | 8975 | 23,60 | 0,046 |
| BEENISH | 143 | 1234 | 7981 | 12,05 | 0,003 |
| BABC-EDEEC | 588 | 2468 | 8353 | 33,19 | 0,032 |
| BPSO-EDEEC | 173 | 1473 | 8315 | 18,34 | 0,031 |
| EDEEC | 140 | 1105 | 7844 | 11,47 | 0 |

Figures 1, 2, and 3 show the behavior of the proposed protocols in terms of energy saving, the number of dead nodes, and the number of packets delivered to BS.

A comparison between the studied protocols, in terms of the Round of First Node Dies (FND), the Round of Half Node Dies (HND), the Round of Last Node Dies (LND), and the percentage of remaining energy in the network is shown in the table 2.

The percentage of residual energy in the network is calculated as below [34]:

$$Res = 100 \times \frac{\sum_{i=1}^{n} Er_i(r)}{\sum_{i=1}^{n} Eo_i} \qquad (20)$$

$Er_i(r)$ is the residual energy of node "$I$" in round "$r$"

$Eo_i$ is the initial energy of node "$I$"

Analysis of table 2, shows that the BABC-based protocols perform significantly better than SEP, EDEEC, and BEENISH protocols in terms of FND, HND, LND, and energy saving percentage.

The first death is observed with the EDEEC protocol (in round 140) with slow sensors' death until the total death of the network's in 7844 rounds.

The BPSO-EDEEC protocol delays the first death of nodes until round 173 with a slower sensor death rate than EDEEC (from round 173 until round 8315) because the selection of CHs is based on an optimized process by the binary PSO algorithm.

The BABC-based protocols seem to be the best way to delay sensor death. Indeed, the BABC-SEP, BABC-EDEEC, and BABC-BEENISH protocols record their first death in rounds 640, 588, and 815, respectively; their half nodes death is recorded in 2471, 2468, and 2469, and their total network nodes death (LND) is recorded in 7646, 8353, 8604 rounds respectively.

Moreover, the BABC-based approaches outperform the other algorithms in terms of HND and percent improvement in power saving compared to SEP, EDEEC, and BEENISH protocols. Especially the BABC-SEP approach that has a superior HND with more than 1000 rounds and an energy-saving percentage of more than 31 % compared to SEP protocol. The BABC-BEENISH and the BABC-EDEEC have also a higher HND with more than 1000 rounds compared to BEENISH and EDEEC and saved their power by more than 21%.

The BABC-SEP protocol delays the first network death until round 640 with a slower increase in dead sensors compared to BPSO-SEP (its first death at round 217) and the rest of the protocols. In addition, the BABC-SEP protocol extends the lifetime of the network up to 7646 rounds thanks to its efficient strategy of searching for powerful CHs.

Additionally, the BPSO-based approaches perform slightly better than SEP, EDEEC and BEENISH protocols in terms of HND, LND, and energy-saving percentage. Indeed, the BPSO-based approaches offer a higher HND with more than 300 rounds and save the energy of SEP, and BEENISH protocols by more than 11 %.

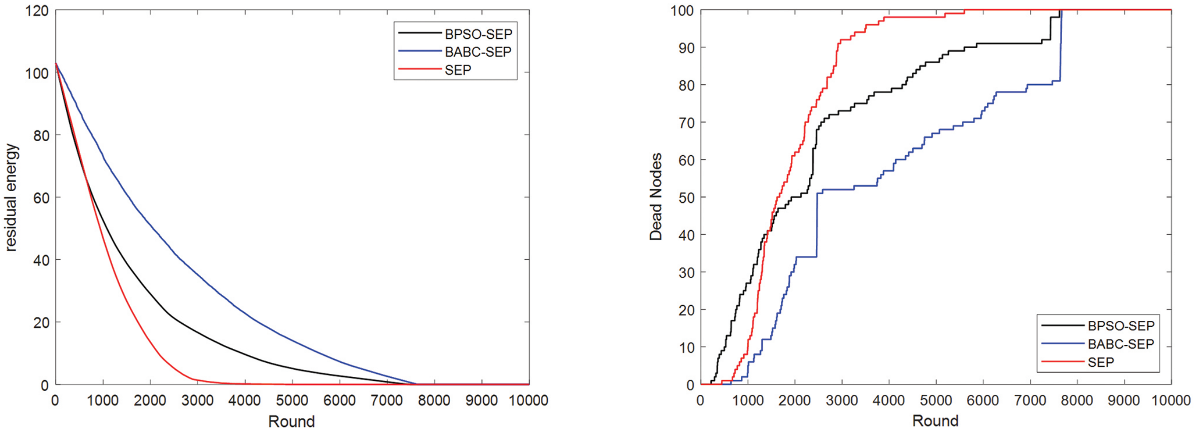


**Fig. 1.** The behavior of SEP-based protocols

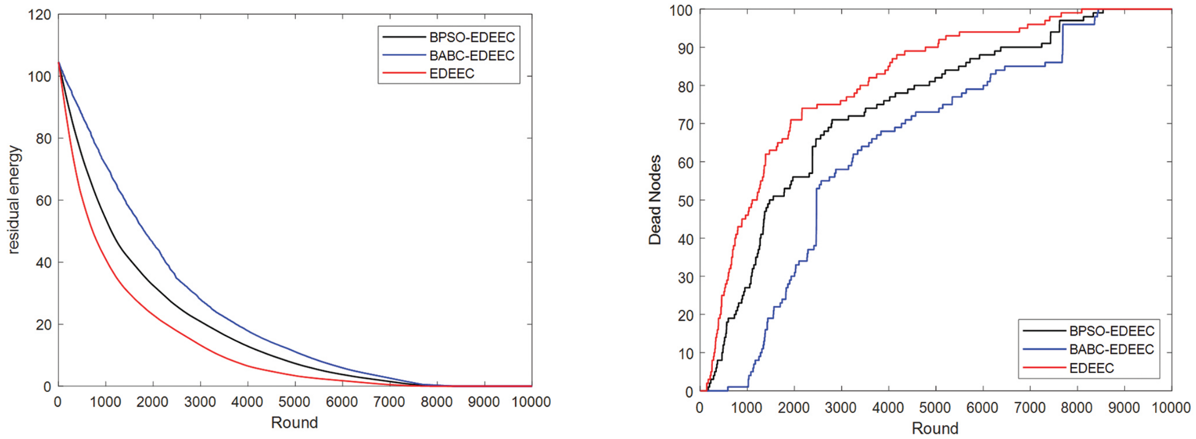


**Fig. 2.** The behavior of EDEEC-based protocols

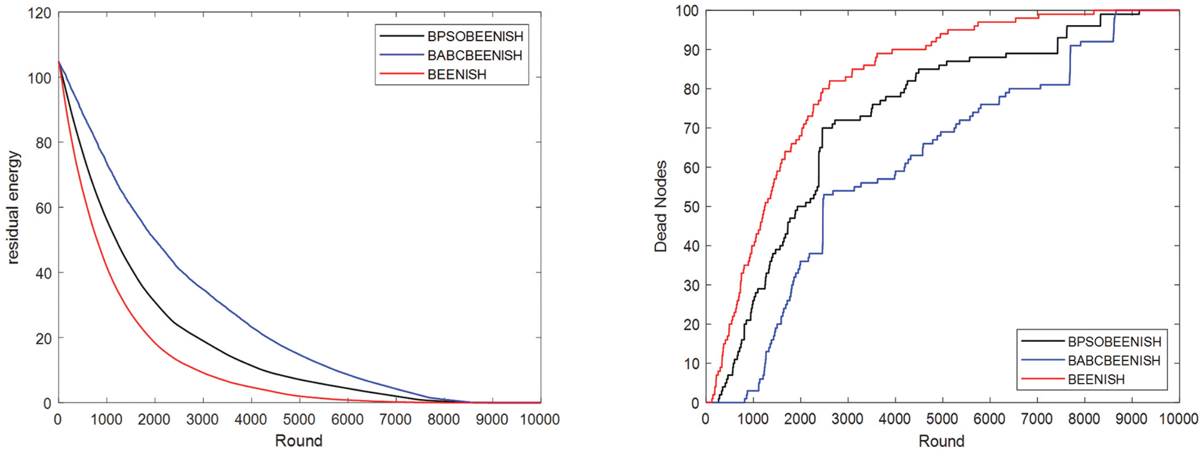


**Fig. 3.** The behavior of BEENISH-based protocols

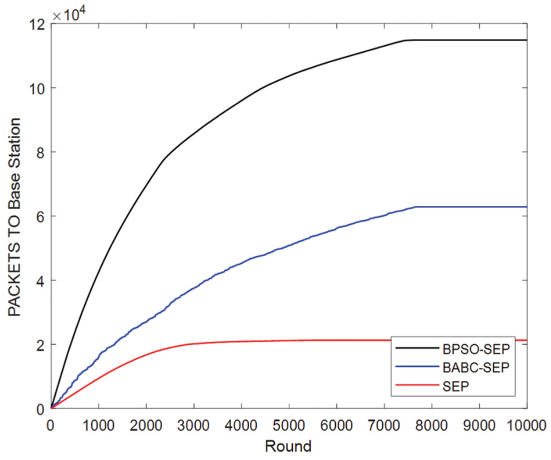


**Fig. 4.** Packets to BS of SEP-based protocols

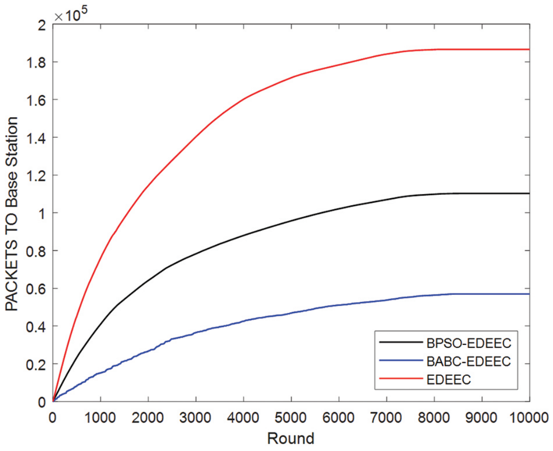


**Fig. 5.** Packets to BS of EDEEC-based protocols

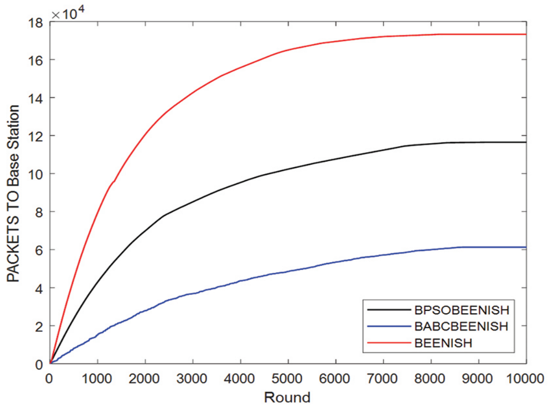


**Fig. 6.** Packets to BS of BEENISH-based protocols

### 4.1. FINDING

BABC-based Approaches, namely: BABC-BEENISH, BABC-EDEEC, and BABC-SEP, are the best in terms of FND, HND, and LND. In particular, the BABC-BEENISH protocol that provided the longest stability period and the maximum network lifetime.

BPSO-based approaches, namely: BPSO-SEP, BPSO-EDEEC, and BPSO-BEENISH, contributed respectively to improving the SEP, EDEEC and BEENISH protocols in terms of HND and LND, and improved EDEEC and BEENISH in terms of FND while the SEP protocol remains better than BPSO-SEP in terms of stability (FND).

From the obtained results, it can be seen that the BABC algorithm has perfectly contributed to improving the three protocols SEP, EDEEC, and BEENISH in terms of FND, HND, LND, and energy-saving percentage.

From Table 2 and figures (1 to 3), the proposed approaches compete with the heterogeneous protocols SEP, EDEEC, and BEENISH in terms of delay and the number of packets delivered to the base station.

From the obtained curves, we observed that EDEEC and BEENISH provided the highest rate of packets delivered to the base station, followed by BPSO-based approaches, then BABC-based approaches, and the SEP protocol comes last, providing the lowest rate of packets delivered to the BS.

BEENISH is better than EDDEC in terms of stability (FND) and network lifetime extension (LND). Whereas, the SEP protocol is better in terms of stability.

### 4.2 DISCUSSION

There is a difference between the initial energy levels of the three protocols: SEP, EDEEC, and BEENISH, since the EDEEC protocol, has a fraction of super nodes with more energy than the advanced nodes of the SEP protocol, and BEENISH has a fraction of ultra-super nodes with more energy than EDEEC-protocol' super nodes. This is why EDEEC provides better results than SEP, and BEENISH provides better results than EDEEC. We can say that these solutions are hardware based rather than software, as it is explained below:

In SEP-based protocols, the number of normal nodes is m×n. Thus, the total network energy=number of normal nodes ×Eo+ number of advanced nodes ×Eo×(1+a) =(1-m)×n× Eo+ m×n× Eo×(1+a)= 103.

In EDEEC-based protocols, the number of normal nodes is n×(1-m), the number of intermediate nodes is n×m×(1-mo), and the number of super-nodes is n×m×mo.

Thus, the total energy = n×(1-m)× Eo+ n×m×(1-mo)× Eo ×(1+a) + n×m×mo× Eo ×(1+b)=107,91.

In BEENISH-based protocols, the number of normal nodes is n×(1-m), the number of intermediate nodes is n×m×(1-mo), the number of super nodes is n×m×mo×(1-m1), and the number of ultra-super nodes is n×m×mo×m1.

Thus, the network energy = n × (1-m) × Eo + n × m × (1-mo) × Eo × (1+a) + n × m × mo × (1-m1) × Eo × (1+b) + n × m × mo × m1× Eo ×(1+u)=110, 035.

The proposed BABC or BPSO-based approaches have contributed to improving the three types of protocols based on the principle of finding the most powerful CHs thus avoiding the rapid exhaustion of nodes' energy and the loss of data packets.

The binary ABC algorithm has solved the routing problem more efficiently than the binary PSO. However, the number of delivered packets to the BS by the EDEEC and BEENISH protocols are the highest, due to their distributed strategy, where neighboring nodes to the BS send their packets directly to the BS without aggregation.

The BPSO optimizer converges faster than the BABC optimizer to the approximate optimal solution without maintaining diversity. Therefore, the selected CHs by BPSO are always the most powerful and can ensure sending the received packets from their member nodes.

## 5. CONCLUSION

The main objective of this work was to improve energy efficiency and lifetime extension in heterogeneous WSNs using swarm optimization methods. To this end, two communication protocols for WSNs have been developed using swarm optimization methods. The first is based on binary PSO, while the second is based on the binary ABC that have been employed to improve the performances of the standard heterogeneous protocols SEP, EDEEC & BEENISH. The proposed protocols were significantly better in terms of energy saving and lifetime extension, especially those based on binary ABC, which displayed an energy-saving percentage of more than 30% compared to the protocols of basis: SEP, EDEEC, and BEENISH. This was made possible through better load balancing and, therefore, a better alternation of the CH's role between the network nodes using the swarm optimization methods.

In future works, the following perspectives can be addressed:

- Implementation of these algorithms in real-world applications, such as environmental monitoring and irrigation systems in agriculture.
- Consider the packet loss rate, the link quality, delay, and reliability to refine the quality of results through multi-objective optimization.
- Explore other more recent swarm intelligence methods, such as the comprehensive learning particle swarm optimization (CLPSO). Salp swarm algorithm, the Rao algorithm, etc.

## 6. REFERENCES

[1] P. K. Kowsalya, R. Harikumar, "Performance analysis of adaptive routing structure for wireless sensor network based on load balancing", Wireless Personal Communications, Vol. 8, 2015, pp. 1-13.

[2] A. Baraa, K. Enan. "A new evolutionary based routing protocol for clustered heterogeneous wireless sensor networks", Applied Soft Computing, Vol. 12, No. 7, 2012, pp. 195-207.

[3] K. Haseeb, K. Abu Bakar, A. H. Abdullah, T. Darwish, "Adaptive energy aware cluster-based routing protocol for wireless sensor networks", Journal of Wireless Networks, Vol. 23, 2017, pp. 1953-1966.

[4] M. Lehsaini, "Diffusion et couverture basées sur le clustering dans les réseaux de capteurs : application à la domotique", PhD thesis, 2009. (in French)

[5] M. Hussain Malik, Varsha, P. Verma, "Simulation of ABC optimization on RZLEACH in WSN", International Journal of Engineering Applied Sciences and Technology, Vol. 3, No. 5, 2018, pp. 80-86.

[6] M. Saleem, G. A. D. Caro, M. Farooq, "Swarm intelligence based routing protocol for wireless sensor networks: Survey and future directions", Information Sciences, Vol. 181, No. 20, 2011, pp. 4597–4624.

[7] V. Katiyar, N. Chand, S. Soni, "A Survey on Clustering Algorithms for Heterogeneous Wireless Sensor Networks", Int. J. Advanced Networking and Applications 745 Vol. 2, No. 4, 2011, pp. 745-754.

[8] D. Li, Q. Liu, X. Hu, X. Jia, "Energy efficient multicast routing in ad hoc wireless networks," Computer Communications, Vol. 30, No. 18, 2007, pp. 3746-3756.

[9] T. Gui, C. Ma, F. Wang, D. E. Wilkins, Survey on Swarm Intelligence based Routing Protocols for Wireless Sensor Networks: An Extensive Study, Proceedings of the IEEE International Conference on Industrial Technology, 2016, pp. 1944-1949.

[10] A. Rajagopal, S. Somasundaram, B. Sowmya, "Performance Analysis for Efficient Cluster Head Selection in Wireless Sensor Network Using RBFO and Hybrid BFO-BSO", International Journal of Wireless Communications and Mobile Computing, Vol. 6, No. 1, 2018, pp. 1-9.

[11] D. Karaboga, S. Okdem, C. Ozturk," Cluster based wireless sensor network routings using artificial bee colony algorithm", Proceedings of the International conference on autonomous and intelligent systems, 2010, p. 15.

[12] M. A. Latiff, C. C. Tsimenidis, B. S. Sharif, "Energy-aware clustering for wireless sensor networks using particle swarm optimization", Proceedings of the IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications, 3-7 September 2007, p. 15.

[13] J. C. Blandón, J. A. López, L. E. Tobón, "Routing in wireless sensor networks using bio-inspired algorithms", Entre Ciencia e Ingeniería, Vol. 12, No. 24, 2018, pp 130-137.

[14] Y. F. Hu, Y. S. Ding, L. H. Ren, K. R. Hao, H. Han, "An endocrine cooperative particle swarm optimization algorithm for routing recovery problem of wireless sensor networks with multiple mobile sinks", Information Sciences, Vol. 300, 2015, pp. 100-113.

[15] Z. Wang, H. Ding, B. Li, L. Bao, Z. Yang, "An Energy Efficient Routing Protocol Based on Improved Artificial Bee Colony Algorithm for Wireless Sensor Networks", IEEE Access, 2020.

[16] Z. Wang, H. Ding, B. Li, L. Bao, And Z. Yang, "An Energy Efficient Routing Protocol Based on Improved Artificial Bee Colony Algorithm for Wireless Sensor Networks", IEEE Access, Vol. 8, 2020.

[17] S. Sarangi, B. Thankchan, "A Novel Routing Algorithm for Wireless Sensor Network Using Particle Swarm Optimization", IOSR Journal of Computer Engineering, Vol. 4, No. 1, 2012, pp. 26-30.

[18] T. Zhang, G. Chen, Q. Zeng, G. Song, C. Li, H. Duan, "Seamless clustering multi-hop routing protocol based on improved artificial bee colony algorithm", EURASIP Journal on Wireless Communications and Networking, Vol. 75, No. 2020, 2020.

[19] G. Devika, D. Ramesh, A. G. Karegowda, "A solution to Energy-Balanced Routing and Data Aggregation based on Artificial Bee Colony and Ant Lion Optimization for LEACH Protocol", SSAHE Journal of Interdisciplinary Research. Vol. 1, No. 1, 2020, pp. 33-47.

[20] M. Dhanasekar, P. Vijayakumar, M. Phil., "Particle Swarm Optimization (PSO) Based Cluster Head Selection for Ring Clustering Routing in Wireless Sensor Networks (WSNs)", International Journal of Research in Advent Technology, Vol. 6, No.12, 2018.

[21] S. Uppalapati, "Energy-Efficient Heterogeneous Optimization Routing Protocol for Wireless Sensor Network", Instrumentation Mesure Métrologie, Vol. 19, No. 5, 2020, pp. 391-397.

[22] C. A. Bhuvaneswari, G. Vairavel, "Optimized Energy Using Centralized Clustering Protocol In Heterogeneous Wireless Sensor Networks", ARPN Journal of Engineering and Applied Sciences, Vol. 16, No. 2, 2021.

[23] P. Giri, A. Potnis, P. Tripathi, "Comparative Study of LEACH, SEP, TEEN, DEEC, AND PEGASIS In Wireless Sensor Network", International Research Journal of Engineering and Technology, Vol. 5, No. 6, 2018.

[24] A. Kaur, S. Saini, "Simulation of Low Energy Adaptive Clustering Hierarchy Protocol for Wireless Sensor Network", International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 3, No. 7, 2013.

[25] A. Yadav, S. Kumar, "An Enhanced Distributed Energy-Efficient Clustering (DEEC) Protocol for Wireless Sensor Networks", International Journal of Future Generation Communication and Networking, Vol. 9, No. 11, 2016, pp. 49-58.

[26] T. N. Qureshi, N. Javaid, A. H. Khan, A. Iqbal, E. Akhtar, M. Ishfaq, "BEENISH: Balanced Energy Efficient Network Integrated Super Heterogeneous Protocol for Wireless Sensor Networks, International Workshop on Body Area Sensor Networks (BASNet-2013)", Procedia Computer Science, Vol. 19, 2013, pp. 920-925.

[27] D. Karaboga, B. Gorkemli, C. Ozturk, N. Karaboga, "A comprehensive survey: artificial bee colony (ABC) algorithm and applications", Artificial Intelligence Review, Vol. 42, 2014, pp. 21-57.

[28] D Karaboga, B Akay, "Proportional–integral–derivative controller design by using artificial bee colony, harmony search, and the bees algorithms", Proceedings of the Institution of Mechanical Engineers, Part I, Vol. 224, 2010.

[29] M. S. Ioná, R. Schirru, "Identifying Nuclear Power Plant Transients Using The Discrete Binary Artificial Bee Colony (Dbabc) Algorithm", Proceedings of the International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering Rio de Janeiro, Brazil, May 8-12, 2011.

[30] J. Kennedy, R. Eberhart, "Particle swarm optimization", Proceedings of the IEEE International Conference on Neural Networks, Vol. 4, Perth, WA, Australia, 1995, pp. 1942-1948.

[31] M. A. El-Shorbagy, A. E. Hassanien, "Particle Swarm Optimization from Theory to Applications", International Journal of Rough Sets and Data Analysis, Vol. 5, No. 2, 2018.

[32] M. F. Taşgetiren, Y. C. Liang, "A Binary Particle Swarm Optimization Algorithm for Lot Sizing Problem", Journal of Economic and Social Research, Vol. 5, No. 2, 2004, pp. 1-20.

[33] W. B. Heinzelman, A. P. Chandrakasan, H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks", IEEE Transactions on Wireless Communications, Vol. 1, No. 4, 2002, pp. 660-670.

[34] A. Rodríguez, M. Pérez-Cisneros, J.C. Rosas-Caro, C. Del-Valle-Soto, J. Gálvez, E. Cuevas, "Robust Clustering Routing Method for Wireless Sensor Networks Considering the Locust Search Scheme", Energies, Vol. 14, No. 11, 2021.

# Cross-Layer Model of Dynamic Distribution of Radio Resources and Data Flow Service in LTE Networks

Original Scientific Paper

**Ulugbek Amirsaidov**
Tashkent University of Information Technologies named after
Muhammad al-Khwarizmi,
Faculty of Telecommunication technologies, Department of Data Communication Networks and Systems
Amir Temur 108, Tashkent, Uzbekistan
u.amirsaidov@mail.ru

**Azamat Qodirov**
Tashkent University of Information Technologies named after
Muhammad al-Khwarizmi,
Faculty of Telecommunication technologies, Department of Data Communication Networks and Systems
Amir Temur 108, Tashkent, Uzbekistan
azamattuit2013@gmail.com

***Abstract*** *– In this article, the results of the development of a mathematical model for the time-frequency resource allocation of the uplink channel and flow service in LTE (Long Term Evolution) networks are given. The proposed model is aimed at ensuring the maximum performance of the radio channel and the guaranteed quality of service for data flows of wireless network users. A comparative analysis of the proposed model with the existing methods of the time-frequency resource allocation of the LTE technology is carried out in terms of ensuring the overall performance of the uplink and allocating the required transmission rate to user stations while maintaining the quality of service. It is shown that the proposed model of dynamic distribution of radio resources and servicing of data streams increases the overall performance of the uplink compared to the Round Robin and Proportional Fair methods, by 1.42 and 1.23 times, respectively.*

***Keywords****: LTE, time-frequency resource, resource block, scheduling block, cross-layer mathematical model, required transmission rate, guaranteed quality of service*

## 1. INTRODUCTION

At present, the implementation of modern infocommunication services in terms of increasing their mobility and accessibility is directly related to the further implementation of wireless telecommunication technologies. However, the limiting factor on this path is the low performance compared to wired solutions, provided by wireless technologies. At the same time, to improve the performance of wireless technologies, all available tools are being used to manage the available network resource: frequency, time, channel, buffer, and information [1,2]. In this regard, more and more attention from scientists, designers of network equipment, and developers of relevant standards are being paid to finding effective solutions for the formation and optimal distribution of the time-frequency resource available at the data link layer, and subsequently at the network layer of the OSI (Open Systems Interconnection Basic Reference Model) model. The basic principles of the hierarchical system are the principles of consistency and coordination adopted at all levels of management. In telecommunication networks, all functions of the functional levels must be coordinated with each other to achieve the specified indicators of quality of service. A characteristic feature of existing solutions for the implementation of functional levels is a statistical strategy for the distribution of network resources. In this regard, an important scientifically applied problem arises, which consists in optimizing the processes of distributing a network resource and servicing flows based on the development of cross-layer models.

In this paper, a cross-layer model is developed in which the distribution of a network resource at the data link layer is formulated as an optimization problem of distributing radio resources between network users according to the criterion of maximum uplink channel performance, and at the network layer - as an optimization problem of distributing the allocated amount of radio resources between different classes of flows according to the criterion of minimum average packet delay.

## 2. BRIEF DESCRIPTION OF THE OBJECT OF STUDY

The main function of the MAC (Media Access Control) protocol of the LTE-Advanced is the dynamic distribution of radio resources between subscribers of the UE (User Equipment) network.

The scheduler is responsible for the allocation (scheduling) of resources for user stations. Such resources primarily include symbols (time resources) and frequency subcarriers (frequency resource). The entire channel resource is divided into the RB (resource blocks) [1,2]. One block consists of 12 adjacent subcarriers occupying a bandwidth of 180 kHz and a one-time slot (6 or 7 OFDM (Orthogonal frequency-division multiplexing) symbols with a total duration of 0.5 ms). Each OFDM symbol on each of the subcarrier's forms a RE (resource element), which is characterized by a pair of values {k, l}, where k is the subcarrier number, and l is the symbol number in the resource block. In a typical configuration (with 7 OFDM symbols in one slot), each resource block includes 12 · 7 = 84 resource elements. Some of the resource elements are used to transmit a pilot signal, which is used for synchronization and radio channel state estimation. The resource allocated to the subscriber is always a multiple in the frequency domain of a 180 kHz bandwidth, and in the time domain, an interval duration is 1 ms, which corresponds to two radio signal slots or one subframe.

The base station distributor block (eNodeB) receives several service signals: the AR (allocation request) from the user equipment, radio channel parameters, the QoS (Quality of Service) requirements (QoS Class Identifier, QCI), etc.

The eNodeB receives information about the radio channel parameters from the UE using the CQI (Channel Quality Indicator). The UE reports the obtained CQI to the eNodeB by comparing the measured SNR (Signal-to-noise ratio) according to a linear function.

Based on the obtained CQI value for each SB (Scheduling Block), the data transmission rate of user stations is adjusted on the allocated time-frequency resource by using adaptive modulation and coding [2]. Then the bandwidth of the subframe allocated to a particular user station directly depends on the MCS (modulation and coding scheme) used and is numerically equal to the number of bits transmitted in a time equal to the duration of the temporary subframe. The choice of the MCS used depends entirely on the characteristics of the signal-interference situation in the area of the user station, including and from its territorial remoteness from the base station.

Thus, the task of scheduling a frequency and time resource in LTE technology should be formulated as a task of distributing SB between network UEs depending on the declared transmission rate and distributing the allocated transmission rate between different queues, considering the requirements for the quality of service of data flows.

In works [3-8], the tasks and functions of the following main known methods of radio resource distribution are characterized:

- cyclic method (Round Robin Scheduler);
- method of maximum carrier power to interference level (Max C/I Ratio, Best CQI scheduling).
- method of proportional fair distribution of service (Proportional Fair Scheduling);

The essence of the cyclic method (Round Robin Scheduler) is that the entire available time-frequency resource is sequentially allocated to each UE [4]. Even though the network resource is allocated to stations, as a rule, for the same time interval, they get access to different channel bandwidth, because the distance to the base station and the signal-to-interference situation in the region of each UE is generally different. This is accompanied by the selection of different MCSs, which will result in different allocated bit rates.

The use of the Max C/I Ratio method helps to maximize the performance of the radio channel because the entire time-frequency resource is allocated to those UEs that have the maximum SNR ratio (CQI) values. At the same time, the QoS requirements of other stations are practically ignored. If several stations have the same SNR values, then the downlink bandwidth is shared equally among them. Therefore, stations with low SNR will receive service only when user stations with high SNR do not communicate with the base station, which is the main disadvantage of this method. The Proportional Fair Scheduling algorithm favors a UE that has a high SNR while providing sufficient frequency and time resources for the UE with the worst SNR [5]. This technique is aimed at providing high network throughput and ensuring a balanced distribution of frequency and time resources between UEs.

The tasks of distributing radio resources are also relevant in 5G (fifth generation) networks [9-12]. In [13], UE's CQI state for each RB is considered simultaneously in LTE MAC layer resource allocation with cross-layer support. In [14], an Adaptive LTE-Advanced cross-layer packet Scheduling to guarantee real-time high-speed packet service for LTE-Advanced is purposed. In [15], the long-term time-average optimization problem is converted into a series of the single-time-slot online problem by using the Lyapunov optimization technique. In [16], the performance of three well-known uplink schedulers namely, Maximum Throughput (MT), First Maximum Expansion (FME), and Round Robin (RR) are compared.

**Table 1.** The comparison of the related works

| Reference | Objective | Application area | Method |
|---|---|---|---|
| [9] | Enhancing the QoS architecture | 5G Network | The agile multi-user scheduling |
| [10] | Implementing the correspondent optimal solution | Mobile Network | User-centric schedular |
| [11] | Scheduler for Public Safety Communications | 5G Network | LTE Scheduling in Downlink/ Uplink |
| [12] | Resource allocation in spectrum-sharing OFDMA femtocells with heterogeneous services | LTE Network | Practical Low-Complexity Algorithm |
| [13] | A smart and flexible scheme for Enhanced Utilization Resource Allocation | LTE Network | Enhanced utilization resource allocation (EURA) scheme |
| [14] | Adaptive LTE-Advanced cross-layer packet Scheduling | LTE Network | Adaptive Reward Priority Scheduling and Dynamic Resource Allocation algorithm |
| [15] | Cross-layer resource management mechanism for an indoor multiuser visible light communication (VLC) access network | VLC access network | VLC Resource Management Algorithm |
| [16] | Executing the scheduling algorithm for an open issue in the Long Term Evolution (LTE) standard. | LTE Network | Scheduling algorithm |

## 3. CROSS-LAYER MODEL OF RADIO RESOURCE ALLOCATION

In the proposed cross-layer model, the methods of queuing theory [17,18] and optimization [19] are used.

When solving the problem of distributing frequency and time resources, it is necessary to consider the configuration of the LTE frame, since uplink subframes alternate with downlink subframes and subframes for transmitting service information [2].

Assume that the number of UEs transmitting a request for allocation of a radio resource is equal to $N$, the number of subframes allocated for transmitting information in the uplink is equal to $M$, and the number of SBs in one subframe is equal to $K$. Each UE has $S$ queues for different types of data flows. It is necessary to allocate the total number ($K*M$)of SBs among $N$ UEs and the allocated transmission rates for the UEs to be distributed among $S$ queues.

It is necessary to calculate the control variables of $x_{n,m,k}$, which determines the order of distribution of SB:

$$x_{n,m,k} = \begin{cases} 1, If\ k-th\ SB\ on\ m-th\ subframe \\ is\ allocated\ to\ n-th\ UE; \\ 0,\quad otherwise. \end{cases} \tag{1}$$

As a result of the calculation of variables (1), subframes are assigned and SBs are distributed by user stations, which will transmit data with an effective rate of $R_{n,m,k}$ by the specified MCS. When calculating the required variables, it is necessary to fulfill two constraint conditions

The first condition is for sticking the k-th SB during the transmission of the m-th subframe for no more than one UE.

$$\sum_{n=1}^{N} x_{n,m.k} = 1,\quad m=\overline{1,M}\,, k=\overline{1,K}. \tag{2}$$

The second condition of allocation for the n-th UE is the number of SBs providing the required transmission rate of ($R_n^t$).

$$\sum_{m=1}^{M}\sum_{k=1}^{K} x_{n,m.k}R_{n,m,k} \geq R_n^t, n=\overline{1,N}. \tag{3}$$

The SB allocation problem can be solved using an optimality criterion aimed at maximizing the overall uplink performance.

$$max\sum_{n=1}^{N}\sum_{m=1}^{M}\sum_{k=1}^{K} x_{n,m,k} \tag{4}$$

considering the constraint conditions (2,3).

Each n-th user station distributes its allocated transmission rate (3) among $S$ queues. It is necessary to calculate the control variable of $y_{n,s}$ ($0 \leq y_{n,s} \leq 1$), showing the share of the allocated transmission rate for servicing the s-th queue (data flow).

$$\sum_{s=1}^{S} y_{n,s} = 1,\quad n=\overline{1,N}. \tag{5}$$

When calculating the required variable $y_{n,s}$, it is necessary to ensure that the average delay s - data flow ($T_{n,s}$) must be less than or equal to the allowable value ($T^t_{n,s}$).

$$T_{n,s} \leq T^t_{n,s},\quad n=\overline{1,N} \tag{6}$$

The problem of distributing the allocated transmission rate between queues can be solved using the optimality criterion aimed at minimizing the total average delay of data flows.

$$min\sum_{s=1}^{S} T_{n,s}\,,\quad n=\overline{1,N}, \tag{7}$$

when considering constraint conditions (5.6). The service rate of the s-th data flow of the n-th UE is defined as:

$$\mu_{n,s} = \frac{1}{y_{n,s}\sum_{m=1}^{M}\sum_{k=1}^{K} x_{n,m.k}R_{n,m,k}},\ n=\overline{1,N}, s=\overline{1,S} \tag{8}$$

Considering the UE as a queuing system of the M/M/1 type, the average delay of the s-th data flow can be determined by the formula [13,14]

$$T_{n,s} = \frac{1}{\mu_{n,s} - \lambda_{n,s}}, \qquad n = \overline{1,N}, s = \overline{1,S} \tag{9}$$

where $\lambda_{n,s}$ is the rate of arrival of the s-data stream in the n-th UE, $\lambda_{n,s} < \mu_{n,s}$.

Expression (8) determines the functional relationship of the variables responsible for the distribution of radio resources at the data link ($x_{n,m,k}$) and network ($y_{n,s}$) layers of LTE.

The problem of distributing radio resources at the data link layer (4) is a linear programming problem. The required variables (1) are boolean. The problem of the distribution of radio resources at the network layer (7) is a non-linear programming problem.

To solve the linear programming problem, we will use the capabilities of the MatLab system [15], represented by the Optimization Toolbox package and the program "intlinprog".

$$[x, fval] = intlinprog(f, intcon, A, B, Aeq, Beq, lb, ub) \tag{10}$$

where *intlinprog* uses this basic strategy to solve mixed-integer linear programs. *intlinprog* can solve the problem in any of the stages. If it solves the problem in a stage, *intlinprog* does not execute the later stages. f is a linear optimization criterion (4), A and B define linear inequality constraints (3), Aeq and Beq define linear equality constraints (2), *intcon* - defines the integer value of the required variables (1), which takes the value 0 (*lb*) or 1 (*ub*).

To solve the problem of nonlinear programming, we will use the program "fmincon".

*[y,fval]=fmincon('myfun',y0,[ ],[ ],Aeq,Beq,lb, ub,'confun')*(11)

where *fmincon* finds a constrained minimum of a scalar function of several variables starting at an initial estimate. This is generally referred to as constrained nonlinear optimization or nonlinear programming. *myfun* - non-linear optimization criterion (7), Aeq and Beq set linear equality constraints (5), *confun* - sets non-linear constraints (6), *y0* - sets the initial values of the required variables, taking values from 0 (*lb*) to 1 (*ub*).

## 4. ANALYSIS OF NUMERICAL RESULTS

To analyze the solutions to problems (10) and (11), we consider an example in which the following were used as initial data:

- number of active UEs – N= [5,10,15];
- the number of subframes for the uplink direction of transmission - M=4 (LTE frame duration is 10 ms, duration of one subframe is 1 ms);
- the number of SBs generated during the transmission of one subframe is K=25 (the number of resource blocks is 50 at a frequency of 10 MHz, the number of resource elements is 84, and the number of symbols is 7).
- effective user information transfer rates by CQI and MCS are given in table 1.
- the number of queues in the UE - S=3.

**Table 2.** Effective UE Information Rates (R) According to CQI and MCS.

| Number of UE | Indices of CQI | MCS | | R bit/s/Hz Total period |
|---|---|---|---|---|
| | | Modulation | Code Rate | |
| 1 | 15 | 64QAM | 948/1024 | 5.5547 |
| 2 | 14 | 64QAM | 873/1024 | 5.1152 |
| 3 | 13 | 64QAM | 772/1024 | 4.5234 |
| 4 | 12 | 64QAM | 666/1024 | 3.9023 |
| 5 | 11 | 64QAM | 657/1024 | 3.3223 |
| 6 | 10 | 64QAM | 466/1024 | 2.7305 |
| 7 | 9 | 16QAM | 616/1024 | 2.4063 |
| 8 | 8 | 16QAM | 490/1024 | 1.9141 |
| 9 | 7 | 16QAM | 378/1024 | 1.4766 |
| 10 | 6 | QPSK | 602/1024 | 1.1758 |
| 11 | 13 | 64QAM | 772/1024 | 4.5234 |
| 12 | 12 | 64QAM | 666/1024 | 3.9023 |
| 13 | 11 | 64QAM | 567/1024 | 3.3223 |
| 14 | 10 | 64QAM | 466/1024 | 2.7305 |
| 15 | 9 | 16QAM | 616/1024 | 2.4063 |

During the experimental process, the following constraints were adopted:

- the incoming packet stream is Poisson [M/M/1];
- packet service time is described by exponential distribution;
- the amount of buffer memory is unlimited.

For example, all user stations were set to the same required transmission rates. In Fig. 1, how the overall uplink performance varies from the required transmission rate for N=5 is shown.

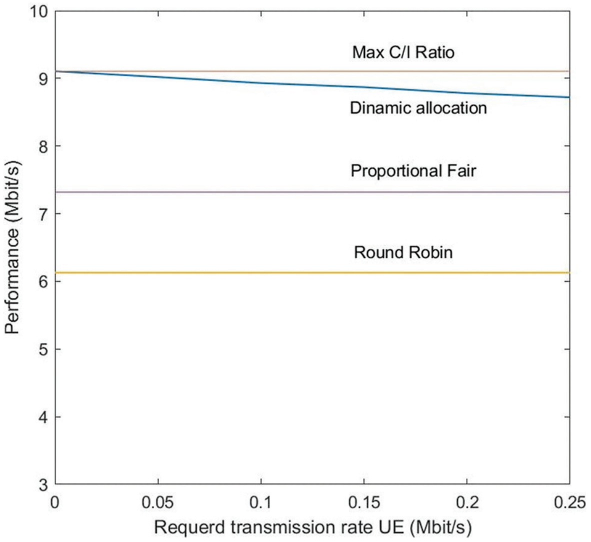


**Fig. 1.** The dependence of the overall performance of the uplink on the required transmission rate.

**Table 3.** The performance of uplink communication

| Required transfer UE, Mbit/s | Performance, Mbit/s | | | |
|---|---|---|---|---|
| | Max C/I Ratio | Round Robin | Proportional Fair | Dynamic Allocation |
| 0.05 | 9.105 | 6.13 | 7.32 | 9.02 |
| 0.1 | 9.105 | 6.13 | 7.32 | 8.93 |
| 0.15 | 9.105 | 6.13 | 7.32 | 8.87 |
| 0.2 | 9.105 | 6.13 | 7.32 | 8.78 |
| 0.25 | 9.105 | 6.13 | 7.32 | 8.72 |

As shown by the simulation results (Fig. 1), the overall performance of the uplink using known methods did not change throughout the entire measurement interval and amounted to 6.13 Mbit/s for the Round Robin method, 7.32 Mbit/s for the Proportional Fair method, and 7.32 Mbit/s for the Max C / I Ratio - 9.11 Mbit/s. The overall performance of the uplink when using the dynamic method (4) in the section of $R_n^{\,t} = 0 \div 0.025\ Mbit/s$ had a maximum value corresponding to the Max C/I Ratio Mbit/s method. On the interval of $R_n^{\,t} = 0.025 \div 0.25$ Mbit/s, the overall performance decreased by 4.2% to 8.72 Mbit/s. $R_n^{\,t} = 0.025 \div 0.25$ Mbit/s

In Table 4, the results of the calculations are shown, and a graph of the total uplink performance versus the required transmission rate using the dynamic distribution method (4) and various values of $N$ is illustrated.

**Table 4.** The performance of uplink communication on dynamic allocation

| Required transfer UE, Mbit/s | Performance, Mbit/s | | |
|---|---|---|---|
| | N=5 | N=10 | N=15 |
| 0.05 | 9.02 | 8.28 | 8.01 |
| 0.1 | 8.93 | 7.63 | 7.18 |
| 0.15 | 8.87 | 7 | 6.51 |
| 0.2 | 8.78 | 6.48 | 5.68 |
| 0.25 | 8.72 | 5.89 | 4.93 |

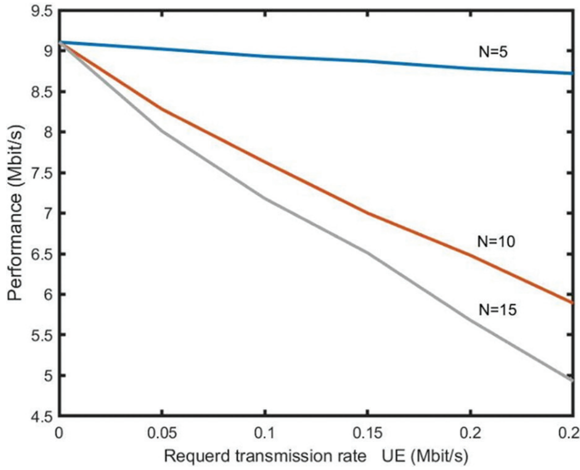


**Fig. 2.** The dependence of the overall uplink performance versus required transmission rate using the dynamic allocation method and different values of N.

Obviously, as the number of active UEs increases, the overall uplink performance decreases because some UEs have a low CQI due to poor SNR (Table 1).

With the number of active UEs N=10 and the dynamic resource allocation method, the first user sees 6.46 Mbit/s. This resource, by (7), is distributed among three queues intended for three data flows. Let the allowable delay time of the first flow be 0.3 ms, the second flow is 0.6 ms, and the third flow is 0.9 ms. In table 5, the results of the calculations are shown, and in Fig. 3, the dependence of the average delay of flows on the intensity of arrival of the first flow at a given intensity of the second (0.03 Mbit/s) and third (0.01 Mbit/s) flows is shown.

**Table 5.** The delay in the data flow

| The intensity of the 1 st stream, Mbit/s | Delay, ms | | |
|---|---|---|---|
| | 1 st flow | 2 nd flow | 3 rd flow |
| 0.01 | 0.3 | 0.6 | 0.707 |
| 0.1 | 0.3 | 0.6 | 0.755 |
| 0.2 | 0.3 | 0.6 | 0.816 |
| 0.4 | 0.305 | 0.601 | 0.9 |
| 0.5 | 0.312 | 0.603 | 0.901 |
| 1 | 0.34 | 0.61 | 0.906 |
| 1.2 | 0.356 | 0.618 | 0.908 |
| 1.4 | 0.368 | 0.624 | 0.911 |
| 1.6 | 0.38 | 0.63 | 0.914 |
| 1.8 | 0.39 | 0.635 | 0.916 |
| 2 | 0.403 | 0.641 | 0.92 |

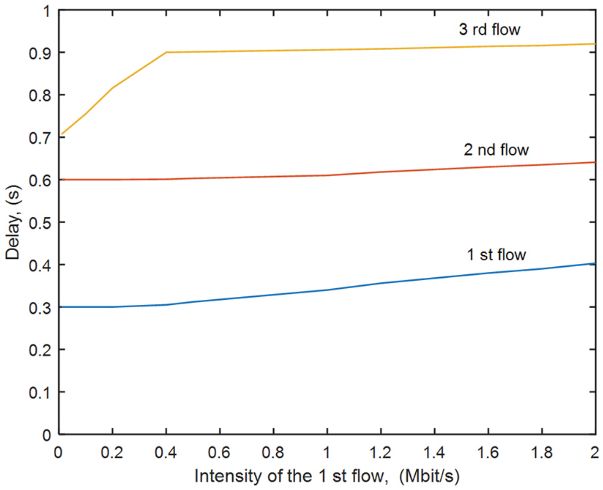


**Fig. 3.** The dependency graph of the average delay of data flows on the intensity of the arrival of data flows.

The obtained results show (Fig. 3) that for the first data flow the allowable delay time is up to $\lambda_{1,1}$=0.3 Mbit/s, for the second flow – up to $\lambda_{1,1}$=0.9 Mbit/s, and for the third flow – up to $\lambda_{1,1}$=1.4 Mbit/s. In the interval $\lambda_{1,1}$=0÷0.4 Mbit/s, the average delay time of the third data flow is less than the allowable value.

Thus, the practical implementation of the proposed cross-layer model allows:

- to improve the efficiency of radio resources, use;
- to provide the required indicators of the quality of service of data flows.

## 5. CONCLUSION

The analysis of existing methods of distribution of radio resources of the uplink communication channel of the LTE network has been carried out. The shortcomings of the known methods are determined and the requirements for promising solutions for the distribution of radio resources between user stations are formulated. A cross-layer model of radio resource distribution between user stations in the data link layer and various data flows in the network layer is proposed. The criterion for the optimal distribution of radio resources in the data link layer is the maximum performance of the uplink. The criterion for the optimal distribution of radio resources in the network layer is to minimize the sum of delays of various data flows with constraints on the allowable delay time for each data flow. The proposed cross-layer model provides a guaranteed transmission rate in compliance with the requirements for the delay time of data flows.

In future works, the influence of the amount of service information on the quality indicators of the transmission of user information, as well as the possibility of using a single optimality criterion in the problems of distributing radio resources at the data link and network layers of the network, will be researched.

## 6. REFERENCES:

[1] 3GPP TS 36.211, 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA); Physical Channels and Modulation (Release 11), Valbonne, France, Sophia Antipolis, 2012.

[2] A. Ghosh, J. Zhang, R. Muhamed, J. Cr. Andrews, "Fundamentals of LTE", Prentice Hall, 2010, p. 464.

[3] Z. Tang, "Traffic Scheduling for LTE Advanced", Linking: Division of Communication Systems, 2010, p. 71.

[4] M. T. Kawser, H. M. A. B. Farid, A. R. Hasin, A. M. J. Sadik, I. K. Razu, "Performance Comparison between Round Robin and Proportional Fair Scheduling Methods for LTE". International Journal of Information and Electronics Engineering, 2012, Vol. 2, No. 5, pp. 678-681.

[5] T. Girici, C. Zhu, J. R. Agre, A. Ephremides, "Proportional Fair Scheduling Algorithm in OFDMA – Based Wireless Systems with QoS Constraints", Journal of Communications and Networks, Vol. 12, No. 1, pp. 30-42, 2010.

[6] S. Hussain, "Dynamic Radio Resource Management in 3GPP LTE", Karlskrona: Blekinge Institute of Technology, 2009, p. 58.

[7] G. Galaviz, D. H. Covarrubias, A. G. Andrade, S. Villarreal, "A resource block organization strategy for scheduling in carrier aggregated systems", EURASIP Journal on Wireless Communications and Networking, 2012, pp. 107-124.

[8] S. V. Garkusha, "Distribution model of scheduling blocks in the downlink of LTE technology" GESJ: Computer Science and Telecommunications, Vol. 39, No. 3, 2013, pp. 76-94.

[9] K. Pedersen, G. Pocovi, J. Steiner, A. Maeder, "Agile 5G Scheduler for Improved E2E Performance and Flexibility for Different Network Implementations", IEEE Communications Magazine, 2018, pp. 210-217.

[10] Q. Liao et al. "Resource Scheduling for Mixed Traffic Types with Scalable TTI in Dynamic TDD Systems", Proceedings of the IEEE Globecom Workshops, Washington, DC, USA, 4-8 December 2016.

[11] K. Gomez, L. Goratti, F. Granelli, T. Rasheed, "A Comparative Study of Scheduling Disciplines in 5G Systems for Emergency Communications". Proceedings of the 1st International Conference on 5G for Ubiquitous Connectivity, Levi, Finland, 2014, pp. 40-45.

[12] H. Zhang, C. Jiang, N. C. Beaulieu, X. Chu, X. Wen, M. Tao, "Resource allocation in spectrum-sharing OFDMA femtocells with heterogeneous services", IEEE Transactions on Communications, Vol. 62, No. 7, 2014, pp. 2366-2377.

[13] Y. T. Mai, C. Hu, "Design of dynamic resource allocation scheme for real-time traffic in the LTE network". Wireless Communication Network, Vol. 14, 2022.

[14] B. J. Chang, Y.H. Liang, P. Y. Chang, "Adaptive Cross-Layer-Based Packet Scheduling and Dynamic Resource Allocation for LTE-Advanced Relaying Cellular Communications", Wireless Personal Communications, Vol. 96, 2017, pp. 939-960.

[15] M. S. Demir, M. Uysal, "A cross-layer design for dynamic resource management of VLC networks", IEEE Transactions on Communications, Vol. 69, No. 3, 2021, pp. 1858-1867.

[16] S. B. Ismail, D. B. M. Ali, N. Ya'acob, "Performance Analysis of Uplink Scheduling Algorithms in LTE Networks", Indonesian Journal of Electrical Engineering and Computer Science, Vol. 9, No. 2, 2018, pp. 373-379.

[17] T. I. Aliev, "Fundamentals of modeling discrete systems", Saint Petersburg, Russia, Saint Petersburg State University ITMO Publishing House, 2016, p. 363. (in Russian)

[18] U. B. Amirsaidov, A. A. Qodirov "A Packet Delay Assessment Model in the Data Link Layer of the LTE", JOIV International Journal on Informatics Visualization, Vol. 5, No. 4, 2021, pp. 402-408.

[19] A. L. Goldstein, "Optimization in the MATLAB environment", The publishing house of PNRPU, 2015, p. 192.

# A Novel Miniaturized Isotropic Patch Antenna for X -Band Radar Applications Using Split Ring Resonators

Original Scientific Paper

**Jyothsna Undrakonda**
Dept. of EECE, GITAM Deemed to be University,
Visakhapatnam, India.
jyothsna.1511@gmail.com

**Ratna Kumari Upadhyayula**
Dept. of EECE, GITAM Deemed to be University,
Visakhapatnam, India.
rupadhya@gitam.edu

***Abstract*** *– A new circular patch antenna with a novel metamaterial structure that achieves high bandwidth and positive gain across the operating band. The proposed antenna was Designed by incorporating three split ring resonators into the patch and fabricating it with 15 ×10 ×1.6 mm3. The use of a metamaterial structure with negative permittivity and permeability reduced mutual coupling in a wideband antenna. The designed antenna shows the isotropic nature at 9.71 GHz in the operating band from 8.80 to 12.89 GHz for X band applications specifically for detecting objects using radars. The optimetrics technique analyzed impedance matching with a good return loss of -30 dB. In comparison to previous works, miniaturization achieved up to 81.94%. The efficiency of 95.6% and isotropic pattern were also achieved at 9.71 GHz using HFSS020R2.*

***Keywords****: Metamaterial, X-band, impedance matching, mutual coupling, isotropic, gain*

## 1. INTRODUCTION

In wireless communication systems, especially military radar systems, transmitting and receiving antennas should possess a good gain with an anisotropic pattern. It can receive and send the signal without delay, and loss transfers data in all directions. But practically, it is not possible to design an antenna with broadband properties like an Isotropic pattern, i.e., it should radiate in all directions equally with positive gain for wideband applications like X-band. Devices with fast communication and a high data rate are required in the new communication era, with no data loss. The antenna should have high signal strength and equally transmit signals in all directions with less return loss. The literature survey clearly explains the contribution of different researchers to achieve this challenge.

For wide bandwidth communication applications, designing antennas play a crucial role. Antenna size should be small enough to be compatible with all wearable intelligent device applications quickly, so miniaturization of the antenna is another challenge for wideband applications with good signal strength.

Many methods are introduced by designing a miniaturized antenna for Wideband applications with good signal strength, i.e., high gain with low return loss. One is by taking another artificial material on the antenna. There are remarkable changes observed performance characteristics of the antenna. The maximum of wideband application antennas in the literature shows the band. Still, with low signal strength and high return loss, those are not suitable for radar and satellite applications with high data rates and loss in data.

The key objective of the proposed work is to design a miniaturized antenna for wideband application within good gain and minimum return loss. A metamaterial structure achieves a compact antenna with a positive gain throughout the band. The proposed antenna is also easy to fabricate and compatible with all planar structured device applications, especially for satellite and radar applications with highly rigid surfaces.

Many researchers worked to increase the bandwidth by using substrates [1-3]. Typically, available materials have properties of positive permittivity & permeability. Metamaterials are created by introducing new materials with negative permittivity and permeability [4-6].

Researchers who utilized these new artificial materials to design antennas observed a significant improvement in antenna parameters.

O. Borazjani [7] used a 4x9 array E.B.G. layer to improve bandwidth for X band applications and achieved improvement up to 1.6G Hz with a simple design. Lin Peng [8] used Mushroom-Type E.B.G. Complementary ring resonator-type meta structures for dual/triple band applications. Ahmad A. Gheethan [9] used a 2x2 collection split ring resonator M.N.G. Deepa Pattar [10] to reduce the mutual coupling of linearly and circularly polar antenna arrays SRR&CSRR to design an antenna. With this, they achieved the X band with a good gain of about 10dbi. They returned a loss of 17db. It reduced a mutual coupling up to 6db.N. A. Estep [11] used a complementary split-ring resonator with negative index material between (permittivity and permeability) from 4.2 to 4.6GHz used to X Band applications. Prince Jain [12] used an I-shaped meta-structure for X-band applications that resonates between 5 to 15 GHz. They observed the FOM for meta-structure, which performs the Structure.

Bhaskar Reddy [13] used a complementary loaded octagonal split ring resonator at operating frequency band 3. 33 GHz.they achieved five bands used for Satellite, Wi-MAX, and X -Band Applications. Mohamed Lashab [14] explained electrically small antennas with different meta structures explained clearly. Ampavathina Sowjanya [15], Microstrip bandpass filters designed using split-ring resonators for X band Applications.

Researchers added artificial material to the antenna to improve parameters like gain, bandwidth, and efficiency. But not able to distribute signals in all directions with good strength in a wide bandwidth range with positive gain used for radar applications. Work has been investigated, analyzed, and validated. The proposed antenna CP-TCSSR structure attains an excellent wide band, size reduction, positive gain over a frequency band, and isotropic pattern at a particular frequency 9.71GHz. The unique feature of the proposed antenna is most appropriate for defense tracking and weather monitoring applications. The S.R.R. (split-ring resonator) is excited by the impedance matching the 50ohm transmission line with dimensions of 15 mm and a width of 10 mm in the proposed antenna. The proposed antenna operated at a wideband frequency from 8.80 to 12.89 GHz with positive gain all over the band and optimized using the ground plane length. Here By varying the ground plane sizes, we finally optimized at 3mm ground length and achieved the highest gain at 12 GHz at 3.74 dB.

## 2. MATERIALS AND METHODS

Metamaterials with unique electromagnetic properties enable many advantages in antenna design in satellite applications. While using the antenna for wideband applications, there is an effect of mutual coupling, removed by negative index materials. The capacitive and inductance nature of the Metamaterial gives the bandgap frequency,

$$Fr = \frac{1}{2\Pi\sqrt{Lc}} \quad (1)$$

### 2.1. DESIGN METHODOLOGY

These steps must be followed while designing the antenna for desired applications. The proposed antenna design methodology with metamaterial structure is explained clearly in Fig.1.

- First, we must select the operating frequency for a particular application for antenna design.
- Design an antenna using a high-frequency structure simulator and evaluate the antenna.
- Design the miniaturized metamaterial structure by satisfying the artificial material properties as -ε and -μ.
- Now, optimize the antenna parameters by varying the position of the meta structure on the antenna until the desired antenna requirements are met.
- Simulate and analyze the results. Fabricate the antenna with the final consideration of the design structure.

Finally, the miniaturized antenna is designed for a wide band with positive gain and low return loss.

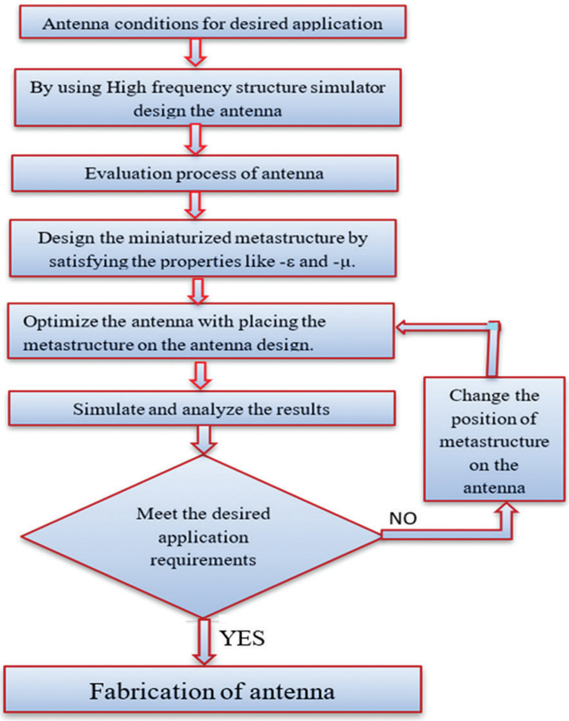

Antenna conditions for desired application
By using High frequency structure simulator design the antenna
Evaluation process of antenna
Design the miniaturized metastructure by satisfying the properties like -ε and -μ.
Optimize the antenna with placing the metastructure on the antenna design.
Simulate and analyze the results
Change the position of metastructure on the antenna
Meet the desired application requirements
NO
YES
Fabrication of antenna

**Fig. 1.** Design methodology structure of an antenna

### 2.2. ANTENNA DESIGN & CONFIGURATION

In this session, we proposed a miniaturized antenna with a length of 15 mm and a width of 10mm circular antenna with FR4 Epoxy as substrate $\varepsilon_r$=4.4 with 1.6 mm thickness. The circular patch is designed with a radius of 1.8 mm, a feed line of 6mm, and a width of 1 mm is used.

Consider the circular antenna with the radius '$r$' and effective radius '$a_r$' taking the effective radius into account, antenna radiation characteristics are improved in relationships of the return losses, gain, and bandwidth [27]. The design formulas are below.

$$a = \frac{F}{\left\{1+\frac{2h}{\pi\varepsilon_r F}\left[lnln\left[\frac{\pi F}{2h}\right]+1.7726\right]\right\}^{\frac{1}{2}}} \tag{2}$$

$$F = \frac{8.791\times10^9}{f_r\sqrt{\varepsilon_r}} \tag{3}$$

$$\varepsilon_{eff} = \frac{1}{2}(\varepsilon_r + 1) + \frac{1}{4}\frac{(\varepsilon_r-1)}{\sqrt{1+\frac{12h}{a}}} \tag{4}$$

$$a_r = \frac{1.8412\,c}{2\pi f_r\sqrt{\varepsilon_r}} \tag{5}$$

### 2.3. DESIGNING OF THE PROPOSED ANTENNA

The antenna was designed with the dimensions shown in Table 1. This antenna is made from a circular patch with a radius of R1=1.8 mm. Regarding that patch, We added three split ring resonators of various radii, R2, R3 & R4, with 2.18 mm,3 mm & 4 mm, respectively. The 6mm length and 1mm width feed line excited the patch. We take a rectangular ground plane of dimensions 10 mm x 3 mm.

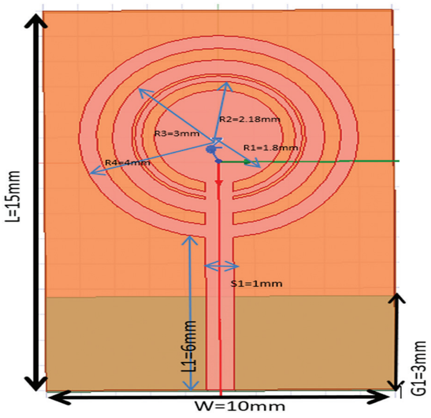


**Fig. 2.** Proposed antenna front view

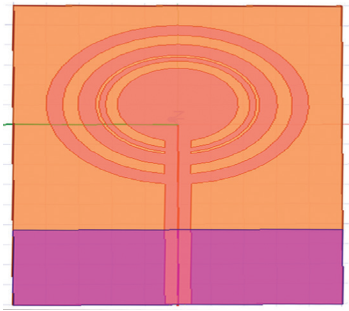


**Fig. 3.** Proposed antenna bottom view

Fig. 2, 3 signifies the front and bottom view of the proposed design antenna. This Structure, wide bandwidth operated in the region of the X band from 8.80 to 12.89 GHz frequency with a positive gain of 3 dB all over the band. Return loss is observed as very low at 9.71 GHz and 12 GHz ranges. Ground plane length G1 also varied and got optimized at 3 mm.

**Table 1.** Proposed antenna dimensions

| S.NO | Parameter (mm) | Proposed Antenna |
|---|---|---|
| 1 | L | 15 |
| 2 | W | 10 |
| 3 | L1 | 6 |
| 4 | S1 | 1 |
| 5 | G1 | 3 |
| 6 | R1 | 1.8 |
| 7 | R2 | 2.18 |
| 8 | R3 | 3 |
| 9 | R4 | 4 |

### 2.4. DESIGN PRINCIPLE

The design of a circular patch antenna is considered an equivalent circuit as follows.

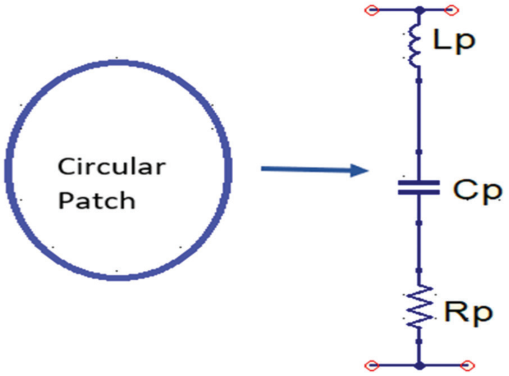


**Fig. 4.** Patch equivalent circuit

Fig. 4 shows the series combination of the circular patch's inductance, capacitance, and resistance, denoted by Lp, Cp &Rp, respectively

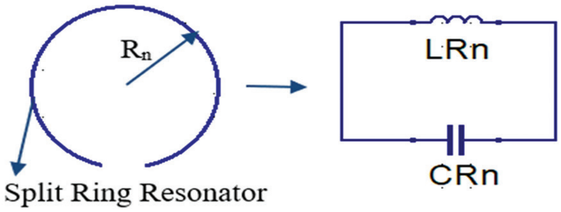
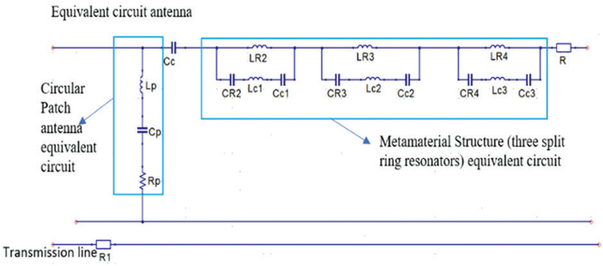


**Fig. 5.** Three split ring resonators equivalent circuits

**Fig. 5.** represents the S.R.R. (Split Ring Resonator) equivalent circuit with radius Rn, where 'n' characterizes the nth ring resonator. Each ring resonator has a parallel combination of inductance $LR_n$ and capacitance $CR_n$. Here are three-ring resonators with three different radii, $R_2$, $R_3$ & $R_4$, and having a similar variety of inductance & capacitance $LR_2$ & $CR_2$, $LR_3$ & $CR_3$. $LR_4$ & $CR_4$, respectively. Fig. 6 shows the equivalent circuit of the antenna.



Equivalent circuit antenna
Circular Patch antenna equivalent circuit
Lp
Cp
Rp
Cc
LR2
CR2
Lc1
Cc1
LR3
CR3
Lc2
Cc2
LR4
CR4
Lc3
Cc3
R
Metamaterial Structure (three split ring resonators) equivalent circuit
Transmission line
R1

**Fig. 6.** Equivalent circuit of the proposed antenna

The coupling capacitor is denoted by the term Cc in this context. Each ring resonator is connected in series.

The resonance frequencies are [19],

$$F_Z^2 = \frac{1}{2\Pi\sqrt{LR_2CR_2}} \tag{6}$$

$$F_Z^3 = \frac{1}{2\Pi\sqrt{LR_3CR_3}} \tag{7}$$

$$F_Z^4 = \frac{1}{2\Pi\sqrt{LR_4CR_4}} \tag{8}$$

$$Fz_1 = \frac{1}{2\pi}\sqrt{\frac{CR_2+Cc_1}{Cc_1CR_2(Lc_1+LR_2)}} \tag{9}$$

$$Fz_2 = \frac{1}{2\pi}\sqrt{\frac{CR_3+Cc_2}{Cc_2CR_3(Lc_2+LR_3)}} \tag{10}$$

$$Fz_3 = \frac{1}{2\pi}\sqrt{\frac{CR_4+Cc_3}{Cc_3CR_4(Lc_3+LR_4)}} \tag{11}$$

$$F_M = \frac{1}{2\pi\sqrt{\left(L_{eq}+\left[\left(\frac{Lt_1}{Lt_2}\right)\left(\frac{Ct_1}{Ct_2}\right)\right]+\left[\left(\frac{Lt_3}{Lt_4}\right)\left(\frac{Ct_3}{Ct_4}\right)\right]\right)}} \tag{12}$$

$$L_{t1} = LR_2\ LR_3 + LR_3\,Lc_2 + LR_3\,Lc_1 + Lc_1Lc_2 \tag{13}$$

$$L_{t3} = LR_3\ LR_4 + LR_3\,Lc_3 + LR_4\,Lc_2 + Lc_2Lc_3 \tag{14}$$

$$L_{t2} = LR_2 + LR_3 + Lc_2 + Lc_1 \tag{15}$$

$$L_{t4} = LR_3 + LR_4 + Lc_3 + Lc_2 \tag{16}$$

$$\begin{gathered} C_{t1} = CR_2\ CR_3\,CR_1 + CR_2\,Cc_2Cc_1 \\ +CR_2\,Cc_2\,CR_3 + Cc_1CR_3\,Cc_2 \end{gathered} \tag{17}$$

$$\begin{gathered} C_{t2} = CR_2\ CR_3 + CR_2\,Cc_2 \\ +Cc_1\,CR_3 + Cc_1 \end{gathered} \tag{18}$$

$$\begin{gathered} C_{t3} = CR_2\ Cc_2CR_4 + CR_2\,Cc_2Cc_3 \\ +CR_4\,Cc_3\,CR_3 + Cc_2CR_4\,Cc_3 \end{gathered} \tag{19}$$

The following circuit component parameters are as LR3=2.5 nH, LR4=7.0 nH, LR4=12.0 nH, Lc1=1.75 nH, Lc2=0.5 nH, Lc3=0.02 nH, CR2=0.108 pF, CR3=0.029 pF, CR4=0.05 pF, Cc1=1.6 pF, Cc2=0.0362 pf. The resonance frequency of metamaterial structure F.M. is 8.524 GHz. The patch circuit frequency can be expressed as follows,

$$F_p = \frac{1}{2\pi\sqrt{\left(L_p+\left[\left(\frac{C_PC_C}{C_P+C_C}\right)\right]\right)}} \tag{20}$$

The antenna's resonance frequency was 10.39 GHz, with the values of $Lp$ , $C_P$ & $C_C$ as 2.5nH,1.5 pF & 0.1 pF respectively.

Half power frequencies in L.C.R. series circuits are $F_{.M.}$ & $F_{.P.}$ taken as,

$$F_0 = \sqrt{F_M . F_P} \tag{21}$$

## 3. RESULTS

This session covered S11 parameter extraction based on ground length, antenna gain, and antenna radiation pattern, as it varies with operating frequency. The session concludes by discussing the comparison of simulated and measured results.

### 3.1. ANTENNA'S GROUND PLANE OPTIMIZATION

The antenna must be effective enough to transmit the signal. Impedance matching is critical in many of the techniques used for this. We used simple optimetrics by varying the antenna length factor from 3 mm to 5 mm.

G1 is 3mm in length and has a wide bandwidth range from 8.80 GHz to 12.89 GHz with a return loss of -25 dB.

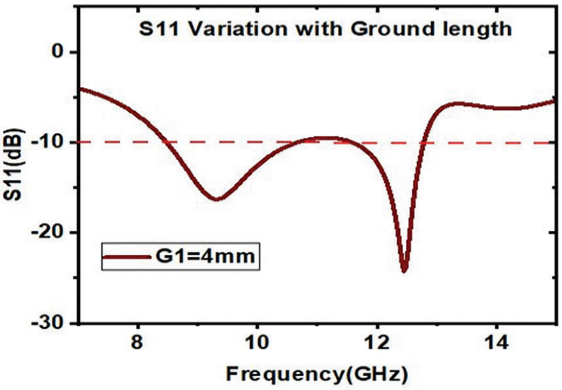


**Fig. 7.** Ground plane effect on S11 at G1 is 4 mm

By changing the length G1 of the ground plane to 4mm and observing the antenna resonating at frequencies 8.43 GHz to 10.64 GHz, and 11.58 to 12.92 GHz with -17 dB & -29 dB, respectively shown in Fig. 7.

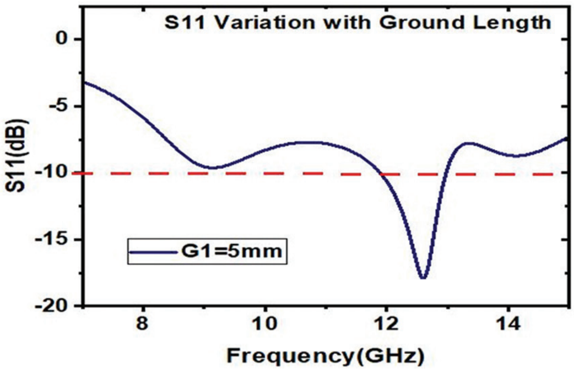


**Fig. 8.** Ground plane effect on S11 at G1 is 5 mm

By Changing the length G1 of the ground plane to 5mm, then observed antenna resonating at dual-band frequencies from 11.8 to 13 GHz with return loss -18 dB, respectively, shown in Fig. 8.

### 3.2. REFLECTION COEFFICIENT S11 EXTRACTION

The proposed antenna is operated at a frequency from 8.80 GHz to 12.95 GHz.

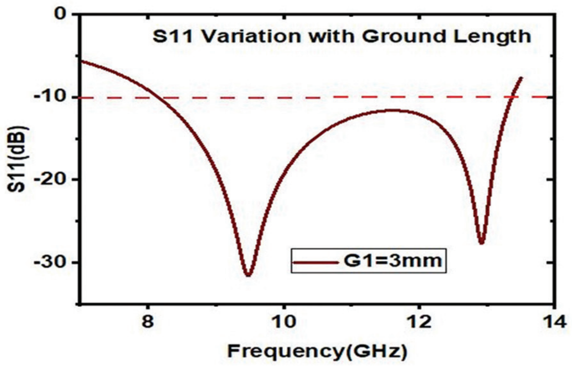


**Fig. 9.** S11 plot of the antenna at G1=3mm

The circular patch antenna with TSRR got optimized at 3mm. Fig. 9 shows the S11 plot of an antenna at an optimized ground length of 3mm, where achieved wide bandwidth with good Return Loss of -31 dB at 9.8 GHz and -24 dB at 12.5 GHz.

### 3.3. PROPOSED ANTENNA VOLTAGE STANDING WAVE RATIO (VSWR)

The crucial parameter,i.e., impedance matching, is measured with the VSWR

The VSWR of proposed antennas also varied at the acceptable value, i.e., 1.5 at the 9.71 GHz frequency range shown in Fig.10.

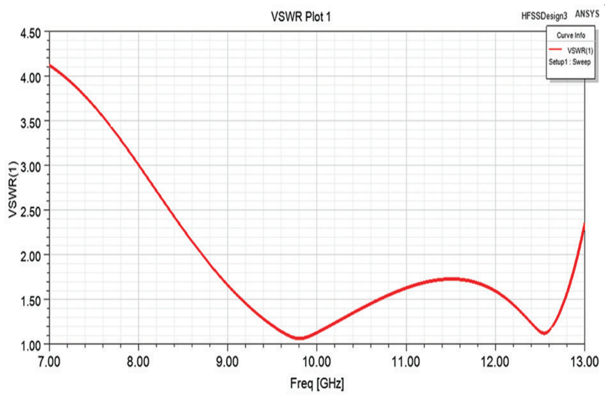


**Fig. 10.** VSWR plot of antenna

### 3.4. ANTENNA'S GAIN

The gain of the proposed antenna is positive throughout the operating band. Below, Fig. 11 & 12 shows the gain plots of the antennas at 9.71 GHz & 12 GHz, respectively.

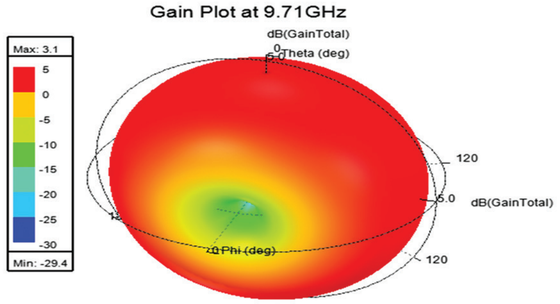


**Fig. 11.** Proposed antenna's gain plot at 9.71 GHz

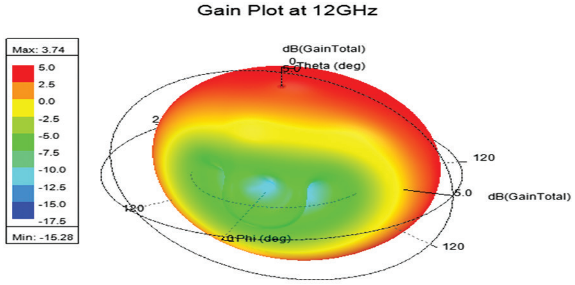


**Fig. 12.** Proposed antenna's gain plot at 12 GHz

The circularly polarized patch antenna has good immunity to signals in all directions, and its gain makes orientation in a particular focus possible. Fig. 11 shows a 3.1 dB gain of 9.71 GHz, and Fig. 12 offers the highest gain at 12 GHz at 3.74 dB for the proposed antenna.

### 3.5. GAIN VS. FREQUENCY PLOT

The below fig 13. represents the plot for frequency vs. gain.

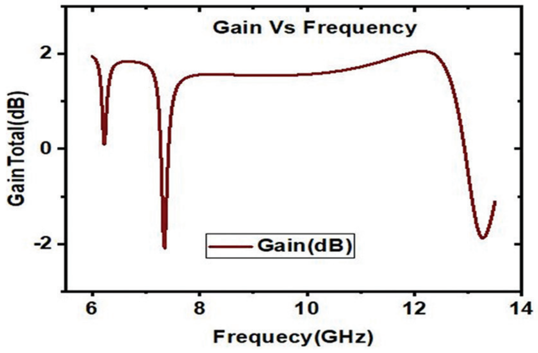


**Fig. 13.** Gain vs. Frequency plot

Here proposed antenna achieved a positive gain all over the X- band from 7.76 to 12.8 GHz. At phi 0 degrees and theta 0-degree direction, we got a maximum increase of 2 dB at 12 GHz.This is used for military and commercial applications.

### 3.6. RADIATION PATTERN

A circular patch has the advantage of having the signal distribution with equal signal strength in all directions possible.

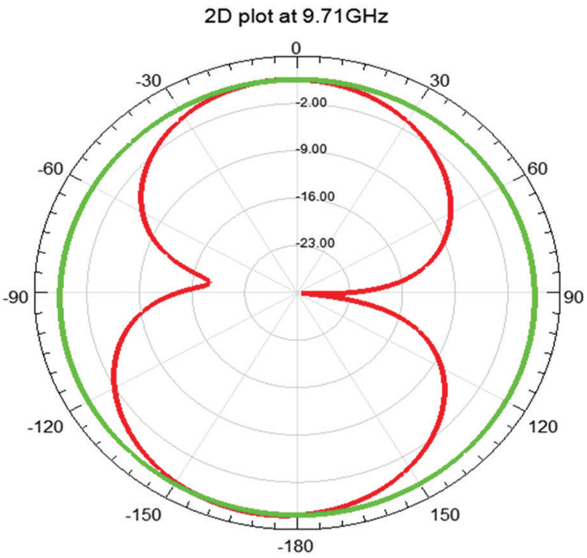
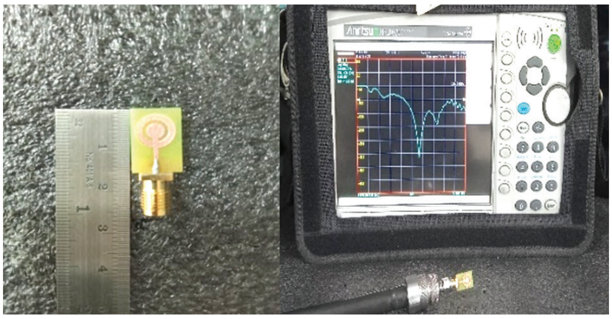


**Fig. 14.** Proposed antenna radiation pattern at 9.71 GHz

This determines the signal strength towards the particular direction, which detect the object's location. Isotropic pattern radiates equally in all directions, taking as a reference for other sources to compare. The proposed antenna got an isotropic radiation pattern at 9.71 GHz, which is the unique advantage of the proposed antenna used for Radar applications.

## 4. FABRICATION RESULTS

The designed and simulated antenna with dimensions 15 x 10 mm$^2$ is fabricated using substrate FR4 Epoxy of thickness 1. 6 mm. The front view of the fabricated antenna is shown below in Fig. 15.



a) b)

**Fig. 15.** Fabricated Antenn a) prototype b) vector analyzer setup

Vector Network Analyzer used for testing the Fabricated antenna. The below fig shows the comparison of the simulated and fabricated results.

Fig. 16 shows the simulated and measured parameters S11 & VSWR are compared and achieve the appropriate, acceptable range.

The comparison of the measured frequency band 9.41 GHz-12.95 GHz to the simulated frequency band 8.80 GHz - 12.89 GHz, covering extra X-band, is achieved.

## 5. DISCUSSION

Table 2. gives the comparison of results thus obtained by the proposed antenna and previous works. Compared to the [7,13,21], The bandwidth has been increased. And antenna has a higher gain than previous works [20,21]. Compared to [7,20,21], miniaturization of the antenna also achieved an average of 81.94%.

### 5.1. LIMITATIONS

The Proposed antenna operated at X-band applications with an average gain of 3 dB. The proposed antenna is an isotropic radiation pattern for satellite and radar applications and was non-isotropic at the remaining bands.

**Table 2.** Comparison of proposed work with previous works

| Ref | Dimensions (mm) | Frequency Bands | Gain |
|---|---|---|---|
| [7] | 30×40 | 9.7 GHz | 0.7 dBi |
| [10] | 13x13 | 9.5 GHz | 4 dBi |
| [12] | 50 × 45 | 4.5 to 5.33 GHz<br>6.98 to 13.65 GHz | 15.1 dBi |
| [13] | 15x15 | 3.33 GHz,<br>5.01 GHz,<br>5.28 GHz,<br>7.46 GHz &<br>9.48 GHz | 0.4, 0.28, 3.49, 4.19 & 2.05 dBi |
| [17] | 24 x50 | 2.01 to 2.15 GHz,<br>7.83 to 8.52 GHz,<br>9.91 to 10.01 GHz,<br>11.21 to 12.84 GHz | Five dBi |
| [18] | 29.5 x 22 | 3.5 GHz, 4.41 GHz,<br>5.8 GHz, 8.26 GHz,<br>10.48 GHz<br>13.35 GHz and<br>14.42 GHz | 2.68 dBi |
| [19] | 45 × 31 | 2.2 GHz to 9.8 GHz | 5 dBi |
| [20] | 30x30 | 8 to 12 GHz | 2 dBi |
| [21] | 20x30 | 8 to 12 GHz | 2 dBi |
| Proposed | 15x10 | 8.80 to 12.89 GHz | 3.74 dBi |

## 6. CONCLUSION AND FUTURE SCOPE

In this paper, a Miniaturized Metamaterial-based X-band range antenna is fabricated and simulated. We got wideband nature by adding Three split-ring resonators to the circular patch. The proposed Structure has been miniaturized by 81.94%, adding Split rings to the circular patch. Because of the circular patch signal strength equal, i.e., isotropic in all directions, achieved at 9.71 GHz frequency and addiction of negative index materials, mutual coupling at the wideband is reduced and achieved a positive gain of 3dB all over the X band ranging from 8.80 GHz to 12.89 GHz, And also discussed the optimetrics of the Ground plane. The efficiency of the antenna is 95.6%. The Fabricated and simulated results are nearly identical and valid for X-band applications. Adding metasurfaces and fractal techniques will improve the antenna's gain. It will be helpful in Radar array applications.

## 7. REFERENCES:

[1] M. Li, N Behead, "Ultra-wideband, true-time-delay, metamaterial-based microwave lenses", Proceedings of the 2012 IEEE International Symposium on Antennas and Propagation, Chicago, IL, USA, 8-14 July 2012, pp. 1-2.

[2] R. J. Hadi, C. Sandhagen, A. Bangert, "Wideband high-gain multilayer patch antenna-coupler with metamaterial superstrate for x-band applications", Proceedings of the 44th European Microwave Conference, Rome, Italy, 6-9 October 2014, pp. 636-639.

[3] B. P. Mishra, S. Sahu, S. K. S. Parashar, S. K. Pathak, "A compact wideband and high gain GRIN metamaterial lens antenna system suitable for C, X, Ku band application." Optik, Vol. 165, 2018, pp. 266–274.

[4] G. V. Veselago, "The Electrodynamics of Substances with a Simultaneously Negative value of epsilon and Mu", I.O.P. Science, Vol. 10, 1968, pp. 509-514.

[5] D. J. Robbins, Pendry J.B, J Stewart W, A. J. Holden, "Magnetism from Conductors and Enhanced Non-Linear Phenomena", IEEE Transactions on Microwave Theory and Techniques, Vol. 47, 1969, pp. 2075-2084.

[6] R. D. Smith, C. M. Soukoulis, D. C. Vier, T. Koschny, "Electromagnetic parameter retrieval from inhomogeneous metamaterials", Physical Review, Vol. 71, 2005, pp. 1-17.

[7] R. A. Sadeghzadeh, O. Borazjani, M. Naser-Moghadasi1, J. Rashed-Mohassel "Bandwidth improvement of planar antennas using a single-layer metamaterial substrate for X-band application", International Journal of Microwave and Wireless Technologies, Vol. 1, 2020, pp. 1-9.

[8] L. Peng, Z.-Q. Li, C.-L. Ruan "A Novel Compact and Polarization-Dependent Mushroom-Type E.B.G. Using CSRR for Dual/Triple-Band Applications", IEEE Microwave and Wireless Components Letters, Vol. 20, No. 9, 2010, pp. 489-491.

[9] Ahmad A. Gheethan, IEEE, Paul A. Herzig, Gokhan Mumcu, "Compact 2 2 Coupled Double Loop G.P.S. antenna Array Loaded with Broadside Coupled Split Ring Resonators", IEEE Transactions on Antennas and Propagation, Vol. 61, No. 6,2013, pp. 3000-3008.

[10] D. Pattar, P. Dongaokar, S. L. Nisha, S. Amith, "Design and Implementation of Metamaterial Based Patch antenna", Proceedings of the IEEE International Conference for Innovation in Technology, Bengaluru, India, 6-8 November 2020, pp. 6-8.

[11] A. N. Estep, A. Alù, A. N. Askarpour, "Experimental Demonstration of Negative-Index Propagation in a Rectangular Waveguide Loaded with Complementary Split-Ring Resonators", IEEE Antennas and Wireless Propagation Letters, Vol. 14, 2015, pp. 119-122.

[12] P. Jain et al., "I-shaped Metamaterial Antenna for X-band Applications", Proceedings of the Progress In Electromagnetics Research Symposium - Spring, St Petersburg, Russia, May 2017, pp. 2800-2803.

[13] V. Y B. Reddy, A. M. Prasad, K. V. Swamy, "Metamaterial Inspired Compact Penta-band antenna For WiMAX, WLAN, Satellite band and X-band Applications", Proceedings of the IEEE International Conference on Electronics, Computing and Communication Technologies, September 2020, pp. 1-6.

[14] M. Lashab, N. A. Jan, F. Benbdelaziz, C. E. Zebiri "Electrically Small Planar antennas Based on Metamaterial", Antenna Fundamentals for Legacy Mobile Applications and Beyond,, Springer International Publishing, Vol. 4, 2018, pp. 71-98.

[15] A. Sowjanya, D. Vakula, "Microstrip Band Pass Filter Using Symmetrical Split Ring Resonator for X band Applications", Proceedings of the IEEE Indian Conference on Antennas and Propogation, July 2019, pp. 1-4.

[16] S. Khan, T. F. Eibert, "A Multifunctional Metamaterial-Based Dual-Band Isotropic Frequency-Selective Surface", IEEE Transactions on Antennas and Propagation, Vol. 66, No. 8, 2018, pp. 4042-4051.

[17] N. Gunavathi, S. P. J. Christydass, "CSRR Inspired one x2 Metamaterial MIMO antenna for X- Band Application", International Journal of Advanced Science and Technology, 2020, pp. 5880-5888.

[18] M. Kumar, R. K. Saraswat, "A metamaterial loaded hybrid fractal multiband Antenna for wireless applications with frequency band reconfigurability characteristics", Frequenz, Vol. 74, No. 11-12, 2020, pp. 401-416.

[19] E. A. Serria, M. I. Hussein, "Implications of Metamaterial on Ultra-Wide Band Microstrip antenna Performance", Crystals, Vol. 10, No. 8, 2020.

[20] M. Vinoth, R. Vallikannu, "Design and Analysis of Metamaterial Patch antenna 5G and X Band Applications", Proceedings of the International Conference on Computer Communication and Informatics, 2021, pp. 1-6.

[21] R. Kiruthika, T. Shanmuganantham, "Comparison of Different Shapes in Microstrip Patch antenna for X-band Applications", Proceedings of the International Conference on Emerging Technological Trends, Kollam, India, 21-22 October 2016, pp. 1-6.

# Radar Signal Recognition Based on Multilayer Perceptron Neural Network

Original Scientific Paper

**Raja Kumari Chilukuri**
Department of Electronics and Communication Engineering, Koneru Lakshmaiah Education Foundation, Guntur, Andhra Pradesh 522502, India
Department of ECE, VNRVJIET, Hyderabad 500090, India
chrajakumari@gmail.com

**Hari Kishore Kakarla**
Department of Electronics and Communication Engineering, Koneru Lakshmaiah Education Foundation, Guntur, Andhra Pradesh 522502, India
kakarla.harikishore@kluniversity.in

**K Subba Rao**
Department of Electronics and Communication Engineering (Retd.), Osmania University, Hyderabad 500007, India
kakarlasubbarao51@gmail.com

***Abstract*** *– Low Probability of Intercept (LPI) radars are developed on an advanced architecture by making use of coded waveforms. Detection and classification of radar waveforms are important in many critical applications like electronic warfare, threat to radar and surveillance. Precise estimation of parameter and classification of the type of waveform will provide information about the threat to the radar and also helps to develop sophisticated intercept receiver. The present work is on classification of modulation waveforms of LPI radar using multilayer perceptron neural (MLPN) network. The classification approach is based on the following two steps. In the first step, the waveforms are analysed using cyclstationary technique which models the signal in bi-frequency (BF) plane. Using this algorithm, the BF images of the signals are obtained. In the second step, the BF images are fed to a feature extraction unit to get the salient features of the waveform and then to the multilayer perceptron neural (MLPN) network for classification. Nine types of noise free modulation waveforms (Frank, four polyphase codes and four poly time codes) are classified using the images obtained in the first step. The success rate achieved is 100 % for noise free signals. The experiment is repeated for various noise levels up to -12dB SNR. The noisy signals, before feeding to the MLPN network, are denoised using two types of denoising filters connected in cascade and the classification success rate achieved is 93.3% for signals up to -12dB SNR.*

***Keywords****: LPI radar, signal recognition, cyclostationary (CS), cyclic autocorrelation function (CACF), spectral correlation density (SCD), Bi-frequency (BF), contour plot, denoising, multilayer perceptron neural (MLPN) network, confusion matrix, Artificial Neural Networks.*

## 1. INTRODUCTION

Low Probability of Intercept (LPI) radars are developed on an advanced architecture by making use of specially designed coded waveforms which results in low power. The low power levels of LPI radars yield low probability as a synergetic by product. The detection of LPI radars by hostile intercept receivers is highly challenging owing to wide frequency bands and very low peak power. The interception and measurement of LPI signals in hostile radiometric receivers is a difficult task. Recent work has been focused on the early detection of LPI radar signals to defend against an eventual attack [1, 2]. Unfortunately, interception alone cannot resolve the issue. It is crucial to classify the type of radar and link this radar class to such a platform and/or a weapon system in order to completely detect the radar. LPI radars use special type of waveforms and inhibit the non-cooperative receiver from signal interception and detection. The waveforms of LPI radar signals are challenging for standard electronic reconnaissance methods to differentiate precisely due to the characteristics of low power, wide bandwidth, high resolution, frequency change, etc. Identification of LPI radar signals and improving the recognition ability of reconnaissance equipment is the difficult task in electronic warfare [3].

In order for the electronic attack (EA) or electronic support (ES) system to take immediate action against the attacker, precise estimation of parameters is very important. Also understanding the type of waveform will provide information about the threats to the radar. The ability to re-guide and re-transmit without affecting the electronic system is made much easier by the identification of parameters [4]. Development of sophisticated receivers for interception, detection, and analysis of waveforms is possible only by the knowledge of the parameters and the type of the waveform. Technologies like multi-input multi output (MIMO) radar, ES, and EA systems could also be developed with the help of the parameters [5, 6].

In [7], the authors have estimated the modulation parameters of LPI radar using cyclostationary (CS) technique. CS method is very good for the analysis of LPI radar waveforms as these waveforms are periodic. Polytime coded signals (T1-T4) are analysed and estimated the parameters of all the codes with an accuracy of 94%. It is presumed that the radar signals are free from noise. Initially, the time domain signals are transformed into bi-frequency (BF) domain using CS techniques and the spectral correlation density (SCD) function is computed. From the contour plot of the SCD function, the parameters of the radar signal are manually extracted and the results are found to be good. But generally, the received signal is corrupted with lot of noise, thereby decreasing the detection or measurement efficiency [8,9]. But by preprocessing the noisy signal using denoising filters, the measurement accuracy could be improved.

In [10], the authors have analysed the radar signals and assessed the parameters of noisy signals using CS techniques. Two different kinds of denoising filters are stacked in cascade to pre-process the noisy waveforms and to improve the signal quality. The denoised waveforms are analysed using CS algorithm and the coefficients of the SCD function are evaluated. Modulation parameters of 9 types of waveforms (Frank, $P_1$-$P_4$ and $T_1$-$T_4$ are extracted with an accuracy of 95% up to -12 dB signal to noise ratio (SNR). The process of identifying the radar type and related missiles can be made after classification and parameter extraction. Most of the existing classification techniques are based on time -frequency (TF) images.

In [6], Choi-William's distribution (CWD) is employed to analyse the signals, and the extracted features from TF images are fed to the Elman neural network (ENN) for classification. The overall success rate (SR) achieved is 94.7% at -2 dB SNR. The authors have analysed 8 types of modulation signals (P1-P4, Frank, LFM, BPSK, and Costas). In [11], the authors used Alex net and classified 10 types of radar signals up to -6 dB SNR and achieved 92.5% success rate.

In [12], the authors have used improved MLPN network on original radar signal and achieved the classification success rate of more than 90% when the SNR is 0 dB. The success rate decreases to about 80% when the SNR is - 5 dB. In [13], the authors employed multiple features images joint decision (MFIJD) model to extract the pixel feature and to get the feature image of the LPI radar signal. The model is created by fusing the original signal, double short-time autocorrelation feature image, and short-time autocorrelation feature image. The TF images are simultaneously fed to the hybrid model classifier and achieved an overall SR of 87.7% at -6dB SNR for 11 types of radar signals.

In [14], the authors have proposed Choi-William's distribution (CWD) to convert radar signals in to time- frequency images. The TF images are simultaneously fed to the automatic modulation classification algorithm based on dense convolutional neural networks (AAMC-DCNN) and achieved an overall success rate of 93.4% at -8dB SNR for 8 types of modulation signals.

In [15], the authors employed Cohen class time-frequency distribution (CTFD) model to extract TF images. 2-D Wiener filtering, bilinear interpolation, and Otsu methods are applied to remove the background noise. The pre-processed TF images are fed to the convolutional neural network (CNN) and achieved an overall success rate of 96.1% at -6 dB SNR for 12 types of modulation signals.

In [9], the authors have developed a model for automatic recognition of modulations of LPI radar signals. Smooth pseudo-Wigner-Ville distribution is used to transform the time-domain signals to TF images. In order to create high dimensional matrices, the TF images are then fed into a triplet convolutional neural network (TCNN) which improves the NW's training process and hence the classifier's ability. Simulation studies showed that the recognition success rate is 94% at -10 dB SNR for 10 signals.

In this paper, a model is developed for automatic recognition of 9 types of radar signal modulations under high noisy conditions. The noisy radar signals are denoised first using two types of denoising filters and the denoised signals are converted into bi-frequency (BF) images using cyclostationary techniques and then the BF images are fed to an MLPN network for classification.

## 2. CYCLOSTATIONARY (CS) ALGORITHM:

Non-stationary signals are effectively analysed using time-frequency algorithms, which can evaluate the signals simultaneously in both time and frequency domains. Many time-frequency (TF) algorithms are discussed in the literature [16]. But cyclostationary (CS) algorithm transforms the signal in to the cycle frequency-frequency domain or bi-frequency (BF) domain. CS method is used here as it is efficient for periodic signals like radar waveforms.

It offers additional properties which are not available in time–frequency domain. The main property of CS is that they have spectral correlation with frequency shifted versions of itself at certain frequency shifts [7]. The two most important metrics in CS analysis are the cyclic autocorrelation function (CACF) and the spectral correlation density (SCD) functions. The SCD function accurately captures the statistical behavior of the signal in the bi-frequency domain. Many useful characteristics of LPI radar signal can be determined from cyclic auto correlation and the SCD function. It finds applications in many areas like parameter estimation, array processing, signal identification, direction estimation and time of arrival, signal detection [13]. CS is used in detection and identification of weak spread spectrum communication signals. It also offers additional capability in the detection and classification of LPI radar signals [1].

Let the signal to be analyzed be *x(t)*. Eq. (1) is used to determine its CACF.

$$R_x^{\alpha}(\tau) \triangleq \lim_{T \to \infty} \frac{1}{T} \int_{-T/2}^{T/2} x\left(t+\frac{\tau}{2}\right) x^{*}\left(t-\frac{\tau}{2}\right) e^{-j2\pi\alpha t} dt \quad (1)$$

where '$\alpha$' represents cycle frequency. The SCD coefficients are computed using eq. (2)

$$\begin{aligned} S_x^{\alpha}(f) &\triangleq \int_{-\infty}^{\infty} R_x^{\alpha}(\tau)\, e^{-j2\pi f\tau} d\tau \\ &= \lim_{T \to \infty} \frac{1}{T} X\left(f+\frac{\alpha}{2}\right) X_T^{*}\left(f-\frac{\alpha}{2}\right) \end{aligned} \quad (2)$$

The discrete SCD coefficients of finite signals are evaluated using eq. (3)

$$S_{X_N}^{\gamma}(n,k) = \frac{1}{N} \sum_{n=0}^{N-1} X_N\left(n, k+\frac{\gamma}{2}\right) X_N^{*}\left(n, k-\frac{\gamma}{2}\right) \quad (3)$$

where,

$$X_N(n,k) = \sum_{n=0}^{N-1} W(n) x(n) e^{-\frac{j2\pi kn}{N}} \quad (4)$$

and '*N*' is the length of the signal, *W(n)* is the window and '$\gamma$' is the discrete cycle frequency.

SCD is a function of two parameters-cycle frequency and frequency. Fig. 1 shows the block diagram to measure the parameters of noisy radar signals. The input signal is denoised first using two types of denoising filters. The SCD coefficients of the denoised signal, x(n) are estimated using eq. (3) and the bi-frequency (BF) image of the SCD function is plotted. Fig. 2 shows the BF image of noise free Frank code with a carrier frequency of 1 GHz and the parameters are measured as shown in the Fig. 2. (a) [10].

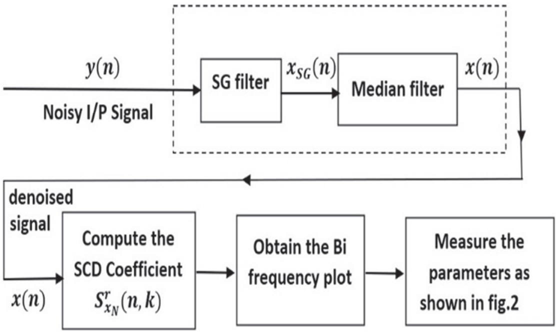
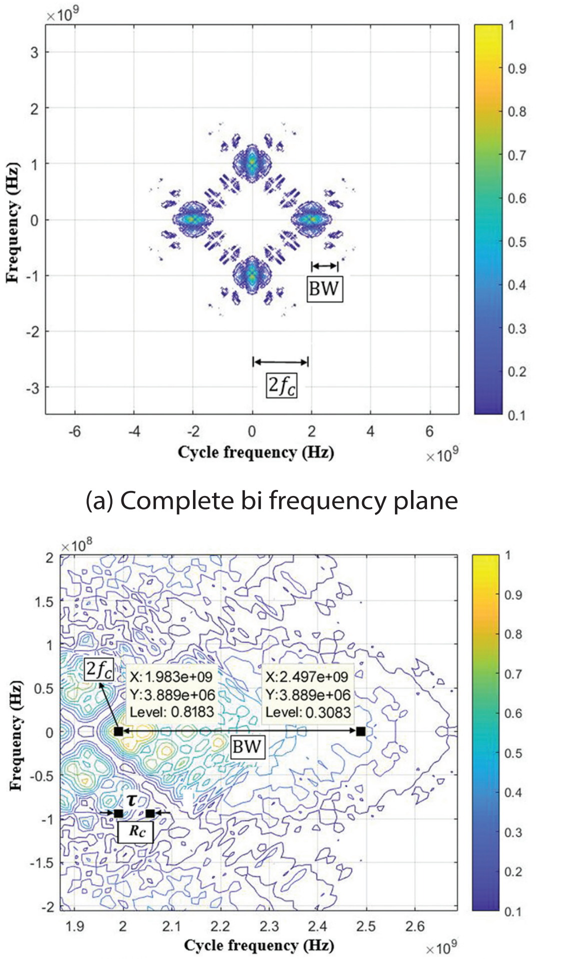

$y(n)$
Noisy I/P Signal
SG filter
$x_{SG}(n)$
Median filter
$x(n)$
denoised signal
$x(n)$
Compute the SCD Coefficient $S_{x_N}^{r}(n,k)$
Obtain the Bi frequency plot
Measure the parameters as shown in fig.2

**Fig. 1.** Block diagram of measurement of modulation parameters.

The parameters measured are carrier frequency ($f_C$), bandwidth (BW) and code rate ($R_C$). Contour plots of noise free, $P_1$ code and $T_1$ code is shown in Fig. 3 and 4 respectively. From Fig. 2, 3 and 4, it may be observed that the shapes of BF images are different for different types of modulations and these images are the basis for classification of signals.



(a) Complete bi frequency plane

$f_C = \left(\frac{1.983}{2}\right)$ GHz, BW= (2.497-1.98) GHz , $R_C = \frac{1}{\tau}$

(b) Enlarged version of right most butterfly of Fig. 2 (a)

**Fig. 2.** Contour plot of noise free Frank code for carrier frequency, $f_C$=1 GHz

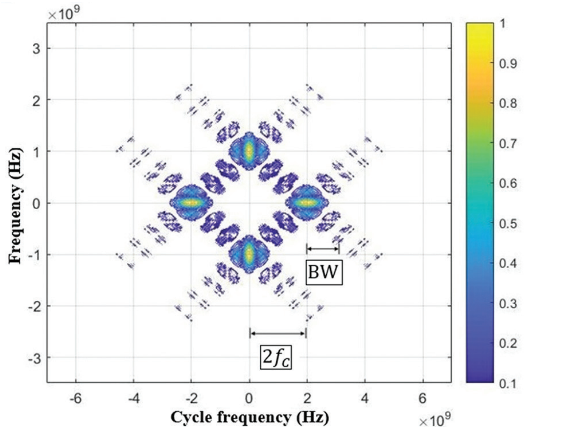


**Fig. 3.** Contour plot of noise free $P_1$Code for carrier frequency, $f_C$=1 GHz

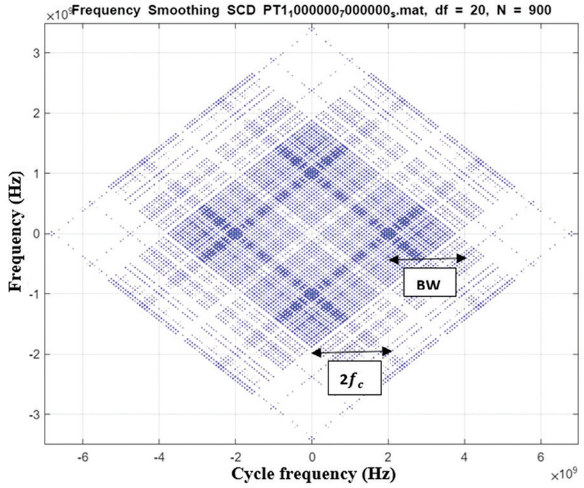


**Fig. 4.** Contour plot of noise free $T_1$ Codefor carrier frequency, $f_C$=1 GHz

## 3. FEATURE EXTRACTION USING PRINCIPAL COMPONENT ANALYSIS (PCA):

PCA is one of the dimensionality reduction algorithms used to reduce the class features. It reduces the size of the input data matrix using projection matrix to represent the data in mean square sense. Linear combination of the eigenvectors obtained from the data covariance matrix is used to represent the data in PCA.

The PCA maps an ensemble of P, N-dimensional vectors $X=[x_1, x_2, x_3, \ldots\ldots x_P]$ onto an ensemble of P, D-dimensional vectors $Y=[y_1, y_2, y_3, \ldots\ldots y_P]$, where $D<N$. Using linear transformation one can show that

$$Y=A^H X \tag{5}$$

where $A$ is a square matrix with i orthogonal column vectors, $i$=1, 2, ..., $P$ and $H$ is the Hermitian operation.

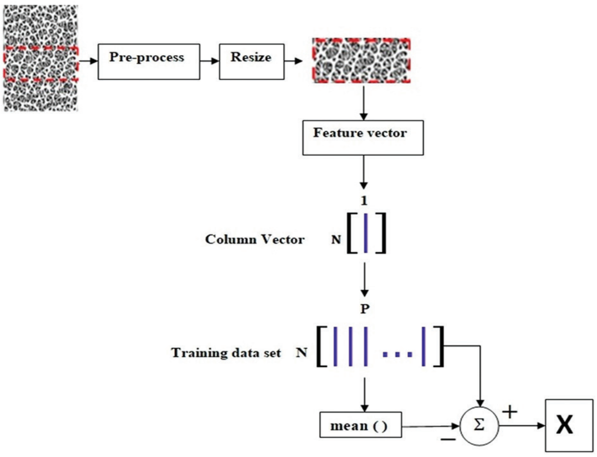


**Fig. 5.** Feature vector generation

The BF images of LPI signals obtained from the CS method are resized to 60 x 60 for all the input signals for pre-processing. In this work a total of $P$=135 input signals are taken for pre-processing followed by feature extraction process as shown in Fig. 5. Each input signal is represented in column vector and stacked together to get a matrix of size 3600 x 135. Later the mean of the training matrix is calculated column wise and the mean is subtracted from the training data set matrix giving the matrix '$X$'. $P$ is the number of training signals and $N$=3600 is the length of the input vector. $X$ is of dimension 3600×135. The number of features is reduced from 3600 to $D$=25. Hence $Y$ is of dimension 25×135. Non-zero eigenvectors of $X$ are obtained using singular value decomposition (SVD) method. SVD states that any $N$×$P$ matrix $X$ can be decomposed as

$$X = U \sum V^H \tag{6}$$

where $U$ is the $N$×$N$ unitary matrix, $V$ is the $P$×$P$ unitary matrix and $\Sigma$ is the matrix of non-negative real singular values. Note that

$$X^H X = V\Sigma^H (U)^H U\Sigma V^H = V(\Sigma^H \Sigma)V^H \tag{7}$$

Equation (7) indicates that the eigenvectors of $X^H X$ are contained in the '$V$' matrix and the eigenvectors of $XX^H$ are contained in the '$U$' matrix where '$U$' is given by

$$U = XV\Sigma^{-1} \tag{8}$$

It can be shown that the non-zero eigen values of the higher dimensional covariance matrix $XX^H$ are computed by computing SVD of smaller dimensional covariance matrix $X^H X$. After getting the eigenvector matrix '$U$' and the eigen values from the input data matrix '$X$' using SVD, the transformation matrix $A$ is obtained from'$U$' using the largest eigen values as shown in Fig. 6. In order to find the largest eigen values a threshold $Th_\lambda$ is selected and is named as eigenvalue threshold constant. The optimum value of $Th_\lambda$ is found to be 0.02. Training and testing signals are projected on to matrix $A$ to get a lower dimensional feature vector with size DX1 for one signal. All such signals form a DXP matrix and given as input to MLP neural network for classification [17].

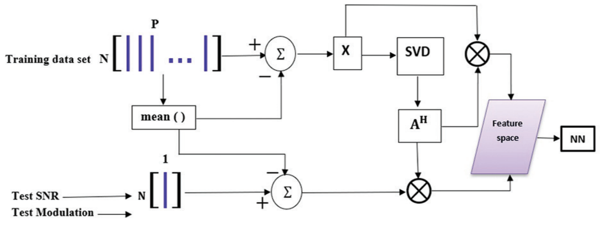


**Fig. 6.** Block diagram of PCA

## 4. CLASSIFICATION NETWORKS

Multilayer perceptron neural (MLPN) network is a feed forward network with an interconnection of non-linear parallel individual computing units. The inputs are propagated layer upon layer in a forward direction resulting in a non-linear mapping of the inputs at the output layer [18].

MLPN network is efficient for small number of hidden layers and also require short training time than deep neural network. The main advantage of MLPN is that it can be used to solve complex non-linear problems and also it makes quick predictions after training. The same accuracy ratio can be achieved even with smaller samples. Hence MLPN is used for classification of radar signals.

An MLP has three distinctive characteristics:

1. The model of each neuron in the network includes a nonlinear activation function.
2. The network contains one or more layers of hidden neurons that are not part of the input or output of the network.
3. The network exhibits a high degree of connectivity, determined by the synapses of the network.

The MLPN network is represented as

$$y_k(l) = \emptyset(\sum_{h=1}^{H} w_{kh}\, \emptyset(\sum_{i=1}^{l} w_{hi} x_i(l))) \quad (9)$$

where $x_i$ is the input,

$y_k$ *is* the output,

$i$ is the number of input nodes,

$l$ is the sample number,

$k$ is the output index

and $h$ is the number of layers.

The weight values between neurons $i$ and $k$ and $i$ and $h$ are represented as $w_{kh}$ and $w_{hi}$ respectively, and the activation function is represented as Ø. A single global training technique using supervised learning determines all weight values $w$ in the MLP simultaneously [19]. For distinct layers of neurons, the activation functions can change and is monotonic. As the number of network layers is relatively low, the activation function used is sigmoid function which is defined as:

$$\emptyset(x) = \frac{1}{(1+e^{-x})} \quad (10)$$

A two-layer feed-forward neural network with one hidden layer and with nine neurons in the output layer, one for each type of modulation, as shown in Fig. 7 is developed for classification of images.

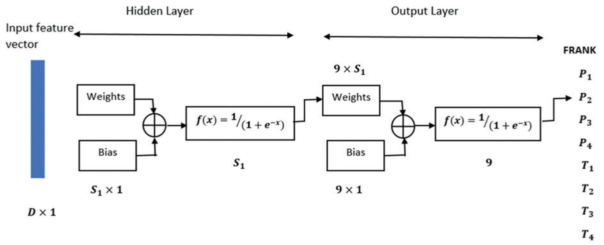


**Fig. 7.** Block diagram of Two-Layer Perceptron Neural Network

For each detection method, the feature vector dimension Dx1 is obtained using the principal component analysis. The optimum number of neurons is chosen after considering several different numbers for the hidden layer [20]. Supervised training of the MLP network uses the gradient of the performance function to determine the weights. The gradients are determined using back propagation algorithm.

To increase the network's convergence speed of the training algorithms, variable learning rate technique is used. The pace of learning is maintained constant during training using typical steepest decent method. The correct learning rate setting is extremely important for best performance. Network regularization $R$ is used to improve the network generalization. The network regularization $R$ is calculated using eq. (11).

$$R{=}g*M_{SE}{+}(1{-}g) \quad (11)$$

where $M_{SE}$ is the mean sum of square of the network errors and $g$ is the performance ratio ($g$=0.0197). The regularization performance target was established at $R$=0.9816. The best value is selected for each training set using a variety of training iterations (epochs). The size of the weight vector is 50×25 since the number of hidden layers are 50 and the number of features is 25.

## 5. SIMULATION RESULTS

The overall block diagram for estimation of parameters and classification of signals is shown in Fig. 8. The BF images obtained from the detector unit are fed to the feature extraction unit and then to the MLPN network for classification.

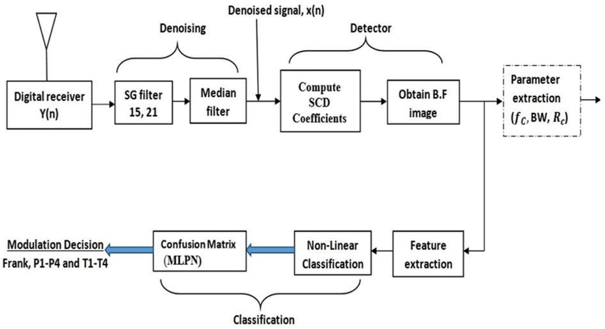
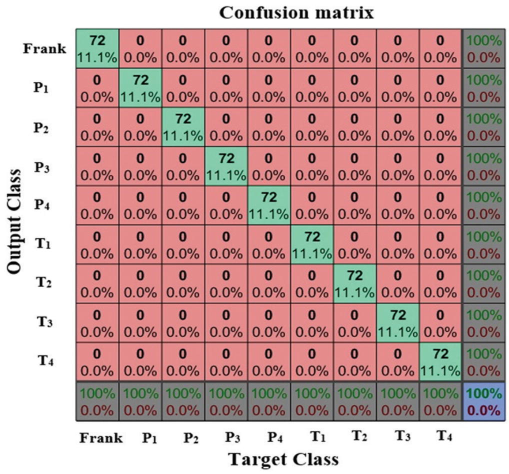


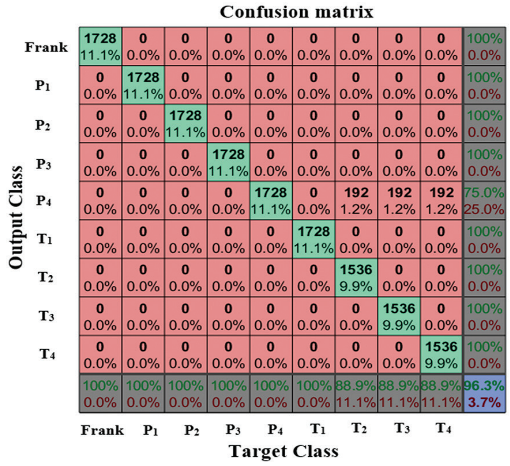
**Fig. 8.** Parameter extraction and classification

Test signals with -6 dB SNR are used for optimization. The optimum selection is based on the highest average probability of correct classification. Nine types of modulation signals (Frank, $P_1$-$P_4$ and $T_1$-$T_4$ are used for classification. The experiment is carried out with three different carrier frequencies (0.8 GHz, 1 GHz and 1.2 GHz). The SNR of each signal is varied from noise free signal to -12 dB insteps of 3 dB. Thus, a total of 135 (9x3x5) signals are considered. For better training and testing of the two-layer MLPN network, the signals are repeated 216 times. Thus, making the total number of signals to be 29,160. Out of 29,160 signals, 70% of them are used for training, 15% for testing and the remaining 15% for validation. Ten test runs are used to build the classification statistics. To randomize the weight matrices of each test, the networks are reset using the ideal network parameters. The maximum number of epochs (iterations) is kept at 1000. The number of neurons in the hidden layer is varied from 10 to 150 and the optimum number is found to be 50 for high noisy signals. The optimum eigen value threshold constant is found to be 0.02 for noisy signals. Confusion matrices are generated for the classification test at each SNR level. The classification success rate (SR) is obtained from the confusion matrix.



| Output \ Target | Frank | $P_1$ | $P_2$ | $P_3$ | $P_4$ | $T_1$ | $T_2$ | $T_3$ | $T_4$ | |
|---|---|---|---|---|---|---|---|---|---|---|
| Frank | 72<br>11.1% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 100%<br>0.0% |
| $P_1$ | 0<br>0.0% | 72<br>11.1% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 100%<br>0.0% |
| $P_2$ | 0<br>0.0% | 0<br>0.0% | 72<br>11.1% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 100%<br>0.0% |
| $P_3$ | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 72<br>11.1% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 100%<br>0.0% |
| $P_4$ | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 72<br>11.1% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 100%<br>0.0% |
| $T_1$ | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 72<br>11.1% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 100%<br>0.0% |
| $T_2$ | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 72<br>11.1% | 0<br>0.0% | 0<br>0.0% | 100%<br>0.0% |
| $T_3$ | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 72<br>11.1% | 0<br>0.0% | 100%<br>0.0% |
| $T_4$ | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 72<br>11.1% | 100%<br>0.0% |
| | 100%<br>0.0% | 100%<br>0.0% | 100%<br>0.0% | 100%<br>0.0% | 100%<br>0.0% | 100%<br>0.0% | 100%<br>0.0% | 100%<br>0.0% | 100%<br>0.0% | 100%<br>0.0% |

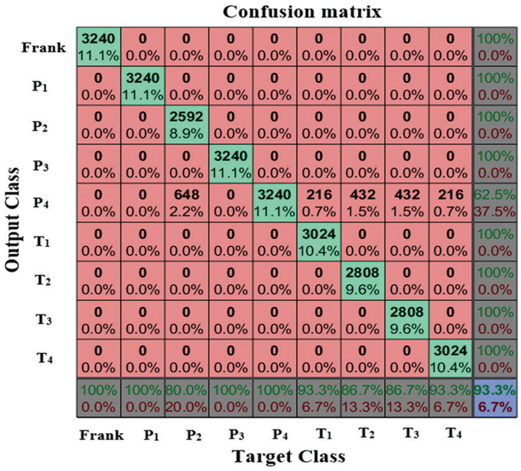
**Fig. 9.** Confusion matrix of noise free signals



| Output \ Target | Frank | $P_1$ | $P_2$ | $P_3$ | $P_4$ | $T_1$ | $T_2$ | $T_3$ | $T_4$ | |
|---|---|---|---|---|---|---|---|---|---|---|
| Frank | 1728<br>11.1% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 100%<br>0.0% |
| $P_1$ | 0<br>0.0% | 1728<br>11.1% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 100%<br>0.0% |
| $P_2$ | 0<br>0.0% | 0<br>0.0% | 1728<br>11.1% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 100%<br>0.0% |
| $P_3$ | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 1728<br>11.1% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 100%<br>0.0% |
| $P_4$ | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 1728<br>11.1% | 0<br>0.0% | 192<br>1.2% | 192<br>1.2% | 192<br>1.2% | 75.0%<br>25.0% |
| $T_1$ | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 1728<br>11.1% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 100%<br>0.0% |
| $T_2$ | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 1536<br>9.9% | 0<br>0.0% | 0<br>0.0% | 100%<br>0.0% |
| $T_3$ | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 1536<br>9.9% | 0<br>0.0% | 100%<br>0.0% |
| $T_4$ | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 1536<br>9.9% | 100%<br>0.0% |
| | 100%<br>0.0% | 100%<br>0.0% | 100%<br>0.0% | 100%<br>0.0% | 100%<br>0.0% | 100%<br>0.0% | 88.9%<br>11.1% | 88.9%<br>11.1% | 88.9%<br>11.1% | 96.3%<br>3.7% |

**Fig. 10.** Confusion matrix for signals up to -6dB

It is found that the classification success rate is 100% for noise free signals when the numbers of hidden layers are 10. Fig. 9 shows the confusion matrix of noise free signals. The experiment is repeated for various noise levels up to -12 dB in steps of 3 dB. Figs. 10 and 11 show the confusion matrices of noisy signals up to -6 dB and -12 dB respectively.



| Output \ Target | Frank | $P_1$ | $P_2$ | $P_3$ | $P_4$ | $T_1$ | $T_2$ | $T_3$ | $T_4$ | |
|---|---|---|---|---|---|---|---|---|---|---|
| Frank | 3240<br>11.1% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 100%<br>0.0% |
| $P_1$ | 0<br>0.0% | 3240<br>11.1% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 100%<br>0.0% |
| $P_2$ | 0<br>0.0% | 0<br>0.0% | 2592<br>8.9% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 100%<br>0.0% |
| $P_3$ | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 3240<br>11.1% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 100%<br>0.0% |
| $P_4$ | 0<br>0.0% | 0<br>0.0% | 648<br>2.2% | 0<br>0.0% | 3240<br>11.1% | 216<br>0.7% | 432<br>1.5% | 432<br>1.5% | 216<br>0.7% | 62.5%<br>37.5% |
| $T_1$ | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 3024<br>10.4% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 100%<br>0.0% |
| $T_2$ | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 2808<br>9.6% | 0<br>0.0% | 0<br>0.0% | 100%<br>0.0% |
| $T_3$ | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 2808<br>9.6% | 0<br>0.0% | 100%<br>0.0% |
| $T_4$ | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 3024<br>10.4% | 100%<br>0.0% |
| | 100%<br>0.0% | 100%<br>0.0% | 80.0%<br>20.0% | 100%<br>0.0% | 100%<br>0.0% | 93.3%<br>6.7% | 86.7%<br>13.3% | 86.7%<br>13.3% | 93.3%<br>6.7% | 93.3%<br>6.7% |

**Fig.11.** Confusion matrix for signals upto -12dB

**Table 1.** Classification results of various noisy signals

| Noise level of the signal (1) | No. of hidden layers (2) | Success rate (SR) (3) |
|---|---|---|
| Noise free signals only | 10 | 100% |
| Upto-3 dB SNR | 10 | 96.3% |
| Upto-6 dB SNR | 10 | 96.3% |
| Upto-9 dB SNR | 10 | 94.4% |
| Upto-12 dB SNR | 50 | 93.3% |

**Table 2.** Comparison of the results with the literature values.

| S. No (1) | Reference number (2) | Type of TF/BF algorithm used (3) | Type of Network used (4) | No. of modulation signals used (5) | Max. noise level SNR (dB) (6) | Success rate (SR) (7) |
|---|---|---|---|---|---|---|
| 1 | [10], Oct. 2016 | Choi–Williams distribution (CWD) | Elman neural network (ENN) | 8 | -6 | 92.5% |
| 2 | [19], Aug. 2018 | Cohen class time-frequency distribution (CTFD) | Convolutional neural networks (CNN) | 12 | -6 | 96.1% |
| 3 | [12], 2019 | Original Radar signal | Improved MLPN | 7 | 0 | 90% |
| 4 | [6], Jan.2020 | Multiple feature images joint decision (MFIJD) | Hybrid model classifier | 11 | -6 | 87.7% |
| 5 | [18], Jan. 2021 | Choi–Williams distribution (CWD) | Dense convolutional neural networks (DCNN) | 8 | -8 | 93.4% |
| 6 | [9], Nov. 2021 | Smooth pseudo-Wigner–Ville distribution (WVD) | Triplet convolutional neural network (TCNN) | 10 | -10 | 94.7% |
| 7 | Proposed method | Cyclostati-onary technique | Multilayer perceptron neural network (MLPN) | 9 | -12 | 93.3% |

Table. 1 shows the classification results for various noisy signals. Column 1 shows the maximum noise level. Column 2 shows the maximum number of hidden layers considered and the last column shows the classification success rate. The success rate achieved for signals up to -6 dB SNR is 96.3%. The maximum classification success rate achieved for signals up to -12 dB SNR is 93.3% and the number of hidden layers is 50.

Table. 2 shows the comparison of the results with the literature values. Column 1 shows the serial number. Column 2 shows the reference number and month and year of publication. Column 3 shows the type of algorithm used to get the TF/BF image and column 4 gives the type of neural network used. Columns 5 and 6 show respectively the maximum number of modulation signals and the maximum SNR considered. The last column shows the classification success rate achieved. For S. No. 2, though the success rate is the highest (96.1%), the noise level considered is only up to -6 dB SNR. The next highest success rate is 94.7% for S. No. 5 and the noise level considered is also less (-10 dB SNR). For the proposed method, the success rate achieved is 93.3% with the noise level considered up to -12 dB SNR. It means even for high noisy signals (compared to S. No. 5), the success rate achieved is nearly same. Hence the proposed method is superior.

## 6. CONCLUSIONS

Detection and classification of radar waveforms are important in many critical applications like electronic warfare, threat to radar and surveillance. LPI radar waveforms are classified using cyclostationary techniques and multilayer perceptron neural (MLPN) network. The main advantage of MLPN is that it can be used to solve complex non-linear problems and also it makes quick predictions after training. The classification process is carried out in three steps. Firstly, the noisy signals are denoised using denoising filters. Later the signals are analysed using cyclostationary algorithm and the BF images are obtained. In the third step, the images are fed to the MLPN network for classification. Nine types of modulation waveforms are considered under various noise conditions up to -12 dB SNR. Before feeding to the MLPN network for classification, the BF images are fed to the feature extraction unit to reduce the number of features to 25. With the proposed method, the classification success rate achieved is 100% for noise free signals. But as the noise level increases, the success rate decreases. The maximum success rate achieved is 93.3% for signals up to -12 dB SNR. When compared with literature values, the proposed method is better as the classification success rate is good even in low SNR conditions.

## 7. ACKNOWLEDGEMENT:

The authors express their sincere thanks to the management and principal, Vallurupalli Nageswara Rao Vignana Jyothi Institute of Engineering and Technology, Hyderabad for providing necessary facilities to carry out this work.

## 8. REFERENCES

[1] P. E. Pace, “Detecting and classifying low probability of intercept radars”, 2nd Edition, Norwood: Artech House, 2009.

[2] M. I Skolnik, “Introduction to radar systems”, 3rd Edition, McGraw-Hill Education, New York, 2003.

[3] W. Si, J. Luo, Z. Deng, “Radar Signal Recognition and Localization Based on Multiscale Lightweight Attention Model”, Journal of Sensors, Vol. 2022, 2022, pp. 1-13.

[4] M. S. Sunder, K. Subbarao, “Cyclostationary Analysis of Polytime Coded Signals for LPI Radars”, International Journal of Research in Engineering and Technology, Vol. 4, No. 6, 2015, pp. 544-560.

[5]. J. E Fielding, “Polytime coding as a means of pulse compression”, IEEE Transactions on Aerospace and Electronic Systems, Vol. 35, No. 2, 1999, pp. 716–721.

[6]. M. Zhang, L. Liu, M. Diao, “LPI Radar Waveform Recognition Based on Time-Frequency Distribution”, Sensors, Vol. 16, No. 10, 2016, pp. 1-20.

[7] R. K. Chilukuri, H. Ki. Kakarla, K. Subbarao, “Estimation of Modulation Parameters of LPI Radar Using Cyclostationary Method”, Journal of Sensing and Imaging, Vol. 21, No. 51, 2020, pp.1-20.

[8] T. R. Kishore, K. D. Rao, “Automatic Intrapulse Modulation Classification of Advanced LPI Radar Waveforms”, IEEE Transactions on Aerospace and Electronic Systems, Vol. 53, No. 2, 2017, pp. 901–914.

[9] L. Liu, X. Li, “Radar signal recognition based on triplet convolutional neural network”, EURASIP Journal on Advances in Signal Processing, 2021, pp. 1–16.

[10] R. K. Chilukuri, H. K. Kakarla, K. Subbarao, “Estimation of intra pulse modulation parameters of LPI radar under noisy conditions”, Journal of International Journal of Microwave and Wireless Technologies, Vol. 14, No. 9, 2022, pp. 1177-1194.

[11] G. Limin, C. Xin, “Low Probability of Intercept Radar Signal Recognition Based on the Improved Alex Net Model”, Proceedings of the 2nd International Conference on Digital Signal Processing, Tokyo, Japan, 25-27 February 2018, pp. 119-124.

[12] F. Li, Y. Wang, L. Zhao, Z. Yang, "Radar modulation recognition based on MLP neural network", Proceedings of the International Conference on Microwave and Millimeter Wave Technology, Guangzhou, China, 13 February 2020, pp. 1-3.

[13] Z. Ma, Z. Huang, A. G. Huang, "LPI Radar Waveform Recognition Based on Features from Multiple Images", Sensors, Vol. 20, No. 2, 2020, pp. 1-23.

[14] W. Si, C. Wan, C. Zhang, "Towards an accurate radar waveform recognition algorithm based on dense CNN", Multimedia Tools and Applications, Vol. 80, No. 2, 2021, pp. 1779–1792.

[15] Z. Qu, X. Mao, Z. Deng, "Radar Signal Intra-Pulse Modulation Recognition Based on Convolutional Neural Network", IEEE Access, Vol. 6, 2018, pp. 43874-43884.

[16] X. Zhang et al. "Radar Signal Intra pulse Modulation Recognition Based on a Denoising-Guided Disentangled Network", Remote Sensing, Vol. 14, No. 5, 2022, pp. 1-15.

[17] T. Farrell, G. Prescott, "A Method for Finding Orthogonal Wavelet Filters with Good Energy Tiling Characteristics", IEEE Transactions on Signal Processing, Vol. 47, No. 1, 1999, pp. 220-223.

[18] M. Shyamsunder, "Classification and estimation of modulation parameters of LPI Radar signals", Osmania University, Hyderabad, India, Ph.D. thesis, 2020.

[19] Z. Ma, W. Yu, P. Zhang, Z. Huang, A. Lin, Y. Xia, "LPI Radar Waveform Recognition Based on Neural Architecture Search", Computational Intelligence and Neuroscience, Vol. 2022, 2022, pp. 1-15.

[20] Q. Guo, X. Yu, G. Ruan, "LPI Radar Waveform Recognition Based on Deep Convolutional Neural Network Transfer Learning", Symmetry, Vol. 11, No. 4, 2019, pp. 1–14.

# Hybrid H-DOC: A bait for analyzing cyber attacker behavior

Original Scientific Paper

**Amal M. R.**
Noorul Islam Centre for Higher Education,
Research Scholar, Department of Computer Science and Engineering
Kumarakoil, Tamil Nadu, 629175, India
amalmr589@gmail.com

**Venkadesh P**
Noorul Islam Centre for Higher Education,
Assistant Professor, Department of Computer Science and Engineering
Kumarakoil, Tamil Nadu, 629175, India

***Abstract*** *– Cyber security is a vital concern for companies with internet-based cloud networks. These networks are constantly vulnerable to attack, whether from inside or outside organization. Due to the ever-changing nature of the cyber world, security solutions must be updated regularly in order to keep infrastructure secure. With the use of attack detection approaches, security systems such as antivirus, firewalls, or intrusion detection systems have become more effective. However, conventional systems are unable to detect zero-day attacks or behavioral changes. These drawbacks can be overcome by setting up a honeypot. In this paper, a hybrid Honeynet model deployed in Docker (H-DOC) bait has been proposed that comprises both low interaction and high interaction honeypot to attract the malicious attacker and to analyze the behavioral patterns. This is a form of bait, designed to detect or block attacks, or to divert an attacker's attention away from the legitimate services. It focuses only on the SSH protocol, as it is widely used for remote system access and is a popular target of attacks. The proposed Hybrid H-DOC method identify ransomware activity, attack trends, and timely decision-making through the use of an effective rule and tunes the firewall. The attack detection accuracy of the proposed Hybrid H-DOC method when compared with IDH, Decepti-SCADA, AS-IDS and HDCM is 13.97%, 11.82%, 8.60% and 5.07% respectively.*

***Keywords****: cybersecurity, docker, containers, high interactive honeypots, low interactive honeypots*

## 1. INTRODUCTION

Cyber security entails the protection of resources that are connected to the Internet. Every day, malicious activities on the Internet is becoming more common [1]. Cyberattacks are on the rise at an exponential rate, as millions of attacks are identified annually, requiring more complex and automated analysis techniques because existing technology cannot handle the volume of data or the diversity of attacks [2]. Cyber risks are complicated and time-consuming to understand and address. The analysis of honeypot data can identify cyber threats [3].

Honeypot technology is a dynamic and ever-growing technology [4]. Honeypot is a network computer and server configured such that to appear vulnerable and to interact highly to attackers, to attract the attackers with open flaws and known vulnerabilities [5]. In computing security, honeypots are frequently used by scientists and security specialists, depending on their level of involvement. There is a unique feature of honeypots that makes any communication with them illegitimate because they are not providing any genuine services [6]. Real world scenario of Honeypot is shown in Fig. 1.

Generally, honeypots are used and are widely distributed. However, there are a number of difficulties that need to be addressed. Security, flexibility, and a limited number of IP addresses are all factors to consider [7, 8]. Honeypots are also prone to potential attackers avoiding them because of their nature. Therefore, it was decided to use operating system level virtualization, otherwise known as containerization, to execute the selected honeypots [9,10]. Containers are virtual environments in which a program and its dependencies are packaged together [11]. The operating system and kernel can be shared by containers, making them less resource-intensive than virtual machines [12-14].

In addition, by running in user space, they minimize system burden [15,16].

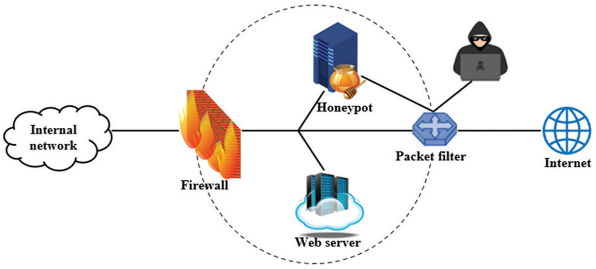


**Fig. 1.** Real world scenario of Honeypot

A Docker container is used in this experiment. In addition to x86-64 and ARM, Docker supports numerous architectures natively, meeting the compatibility criterion. By using Docker Compose, the load balance on the target system were also managed and create, deploy, and orchestrate the instances using the API easily and quickly [17,18]. This study suggests a hybrid honeynet model implemented in Docker. Additionally, the security of Honeypot implementations within Docker containers is investigated.

The rest of the paper is organized in the following manner. Section II represents the literature review in detail. Section III describes the Hybrid H-Doc bait in detail. Section IV describes the security analysis. Section V describes the conclusion and future work.

## 2. LITERATURE REVIEW

The history of cyber security experimentation (CSE) platform was traced back to the early 21st century. Due to the cumulative amount of cyberattacks, countries are investigating the development of a CSE platform. (IDS) Intrusion Detection System, (VDS) Vector Deep Surveillance and honeypot system. Among the above said CSE platforms honeypot is the highly efficient scenario. The "state of the art" of present honeypot solutions is presented in this section.

In 2018, Almohannadi, H., et al. [19] proposed a new threat intelligence technique that evaluates honeypot log data to identify attacker behaviour and find attack trends. They've set up a honeypot on an AWS cloud to collect cyber incident log data in order to achieve this goal. Elasticsearch technology, specifically an ELK (Elasticsearch, Logstash, and Kibana) stack, is used to analyze the log data.

In 2019, Yin, et al.,[20] present a new architecture for a cyber security experimentation platform on the basis of Docker. This software has the scalability and flexibility necessary for large-scale cyber simulations. This feature enables users to customize cybernode's topologies, software environment, and also support the customization of important experiment indicators. It is possible to transmit important experiment indications in real-time, thereby reducing the total cost and facilitating the analysis process.

In 2021, Buzzio-Garcia, J. [21] suggests the utilization of Docker as a high-interaction honeypot, so that threats can be detected at both the network and host levels. It was developed using open-source tools to ensure scalability, safety, and dynamic functionality. A real-world test has demonstrated that it is capable of capturing harmful data for examination at the network and host level employing tools like VirusTotal.

In 2022, Sivamohan, S., et al [22] used Docker container technology with a honeynet-based IDS to create an efficient active protection architecture. The creation of honeynet technology is crucial to cloud security and threat detection. Based on the results of the experiment, it appears that this defense system can identify and log the attacker's activities, revealing new attack strategies and even zero-day vulnerabilities.

From the existing methods, it is identified that, there is no solution focused solely on the SSH protocol. An SSH connection encrypts connections between two end points and provides password or public-key authentication. A secure alternative to unsecure file transmission methods and legacy login protocols (such as telnet and rlogin) (such as FTP). The position of authorized key files and port forwarding in SSH, however, are not ideal. Port forwarding allows an attacker to get around firewalls that have been set up to restrict access to the server's network. Due to their encrypted SSH connection, the attackers are undetectable. So, it is important to focus on the SSH protocol in order to reduce the above-mentioned issues. Therefore, in this paper a hybrid H-Doc bait has been proposed which concentrates on the SSH protocol.

## 3. PROPOSED HYBRID H-DOC BAIT

In this research, an attacker's behavior as well as metadata are utilized to address the problem. The procedure involved in this research process for implementation are as follows:

- Setting Up EC2 instance
- Implementing Docker on Cloud
- Setting Up the Hybrid Honeynet in Docker
- Tuning the firewall

### 3.1. SETTING UP EC2 INSTANCE

Among the most popular instances are General Purpose ones, which are an excellent way to get started with AWS or cloud computing. Their most common uses include web servers, development environments for mobile apps, and enterprise applications such as CRMs and ERPs. Within this class, the most important distinction is between instances that have a Fixed performance and those that have a Burstable performance. Using burstable performance EC2s, one can easily grow their computational power.

### 3.2. IMPLEMENTING DOCKER ON CLOUD

A Docker container has several advantages, including its ability to be deployed in development, test, staging, and production environments, and its ability to be integrated into distributed systems. Applications are constructed with Docker technology, which is delivered to end users via AWS EC2 services. Scalability and management of Docker containers are best handled by Amazon EC2 Container Service. Docker containers with the EC2 Container Service are used to run processes on Amazon EC2 instances using optimistic, shared state scheduling. Amazon ECS allows you to run containers across many hosts, isolate applications and users, and scale quickly to meet your applications' and users' changing needs.

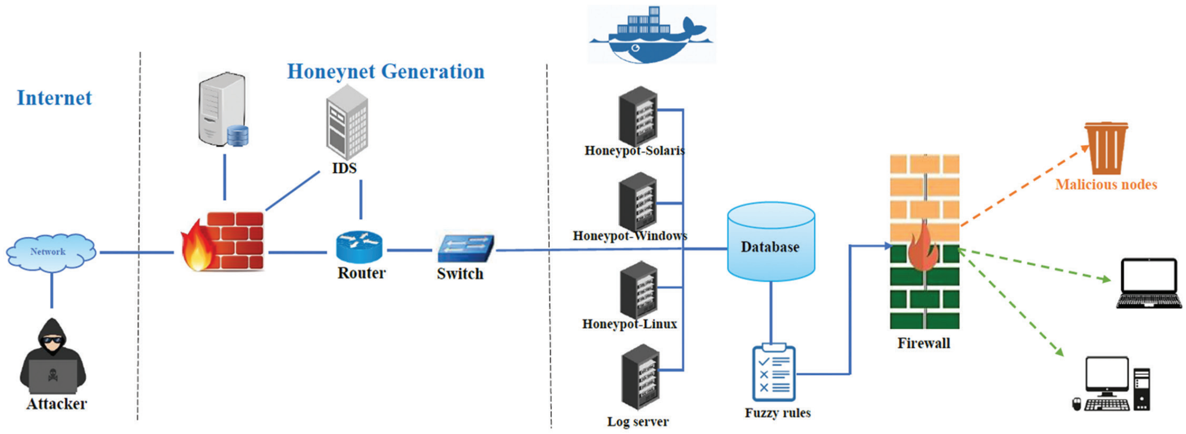


**Fig. 2.** Proposed framework

### 3.3. SETTING UP THE HONEYPOT IN DOCKER

Package the honeypot environment in a docker container to deploy it in various nodes which contain docker for faster deployment mainly in UNIX based distros.

### 3.4. TUNING THE FIREWALL

Real-time data flows will be handled by a uniform, high-throughput, low-latency platform developed by the project. There are tools for processing continuous and timely events as well as extracting high-level knowledge events from lower-level events, is known as complex events. An inference engine uses working memory and fuzzy rules to make decisions. Apache Kafka has been used, which is a big data processing tool. This can process big streams of data. There is a fuzzy rule base that comprises all the rules, and working memory keeps track of the most recent state of the system. As soon as a feature extraction packet is received, it is sent to an inference engine for identification; if they are considered attacks, the firewall blocks them.

Fig. 2 represents the overall framework of the proposed method. When the attacker tries to enter the network through SSH, it will be assigned to hybrid honeypots which contains both low interaction and high interaction honeypots. Unused and unwanted containers will be removed through container removal. Loggings contains the visiting informations of the attacker.

### 3.5. ATTACK ENTRY VIA SSH

The purpose of this study is to investigate into SSH connections to honeypots from any IP address, usually over port 22. By using the honeypots, the attacker is given access to a Linux shell console. Devices with IP addresses that connect to a honeypot are considered attackers. This paper defines a session as any SSH connection between an attacker and a honeypot that is approved by the honeypot. A honeypot is attacked by connecting to an SSH port, usually port 22, and establishing an SSH protocol session with the attacker. During the session, the attacker can enter commands, download and execute files, and so on, to communicate with the honeypot.

### 3.6. SSH SCENARIO

An SSH session was established with a sophisticated simulated attacker. The following was found:

- The traffic was first forwarded to the low interaction honeypot after installation.
- Expert system assesses whereas, to send the traffic to high interaction honeypot or to stay in the low interaction honeypot system. After connection, the attacker can navigate to the honeypot terminal with the fake file system.
- By using the command 'vi,' a fingerprinting attack quickly identified a popular fingerprint indication for honeypots.
- The honeypot container logs recorded all interactive contacts with the attacker session, which were stored and sent to syslog.

### 3.7. HYBRID HONEYNET MODEL

- Honeypots are grouped into clusters called Honeynets to prevent them from becoming independent units which is shown in Fig. 3. The benefits of such setups include Real-time correlation of sensor data, One point of sensor control, Central storage of event data. However, such a distributed model has a few drawbacks Complexity of infrastructure, greater security, risk Inefficient management that should be considered as well. In order to overcome the disadvantages, hybrid honeynet has been proposed.

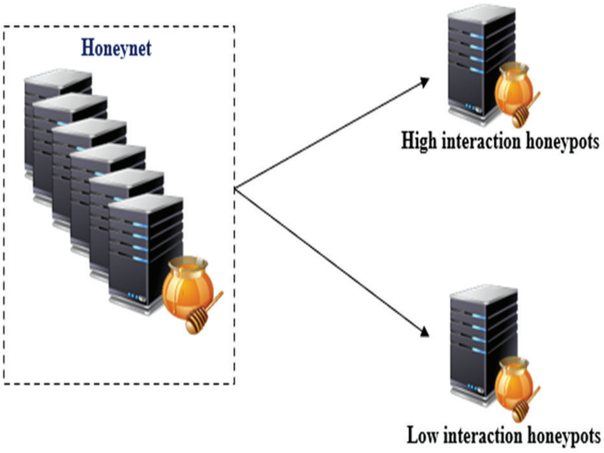


**Fig. 3.** Schema of hybrid honeynet

### 3.8. LOW INTERACTION HONEYPOT (LIH)

During aggressive expansion, low-interaction honeypots are simple, yet can save time due to intruder detection, and the honeypot imitate can be reduced with specific commands. Honeyd, meanwhile, is a honeypot with a low-interaction level. In order to imitate services with low interaction, attackers can take advantage of the low interaction honeypot. Because of the minimal amount of contact, this type of honeypot gathers data from the first step of an attack. Information about the threat's reason for attacking is seldom obtained.

### 3.9. HIGH INTERACTION HONEYPOTS (HIH)

Honeypots with high interaction are the opposite of honeypots with low interaction in deception technology. In contrast to merely simulating particular protocols or services, the attacker actually attacks real systems. This makes it less likely that they will understand they are being monitored or diverted. Since these systems are only available as decoys, all communications discovered are hostile by its very nature, simplifying it to finding threats and track an attacker's activity. "Lyrebird" is a honeypot framework that is highly interactive. All communications between the attacker and the computer are recorded in clear text, so the attacker has access to real vulnerable programs. The Schema of the honeynet with a single Low and high interaction honeypot is shown in Fig. 4.

### 3.10. REQUIREMENTS OF THE SERVER WITH EXPERT SYSTEM

Sessions should satisfy one of two hypotheses:

- The sessions could be diverted to a LIH
- The sessions could be diverted to a HIH
- To make decisions, you must use the data provided by the simulated environment, low interaction honeypot, including country of origin, IP address reputation, downloaded malware, and so on. Numerical quantities are displayed, such as the number of times the malware was identified by the antivirus software or the number of times the shell command was entered.
- Decision needs to be made quickly, within a few seconds. In addition to being simple to install, the solution must be low performing and able to run on a variety of Linux platforms.

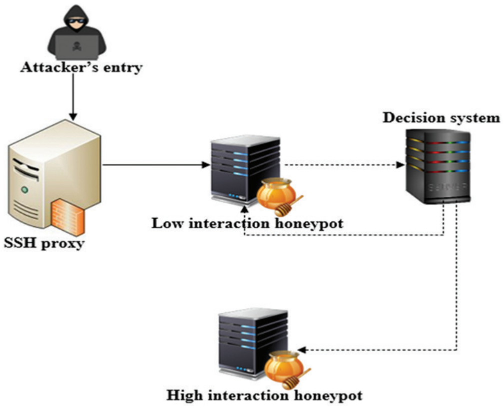


**Fig. 4.** Schema of the honeynet with a single Low and high interaction honeypot

### 3.11. BEHAVIOR OF THE ATTACKER

The behavior of an attacker is defined as the sum of all attacks they have carried out in the domain of interest. The attack behaviour $h_1$ of an attacker $a_x$ in any domain $n_y$ is represented as follows for each attack $k_x$.

$$h_1 = n_y \, a_x \, \Delta k_x \mathrm{a_x} \tag{1}$$

For example, denial-of-service attacks can be carried out on operating systems, databases, hardware, or apps which is shown in Fig. 5.

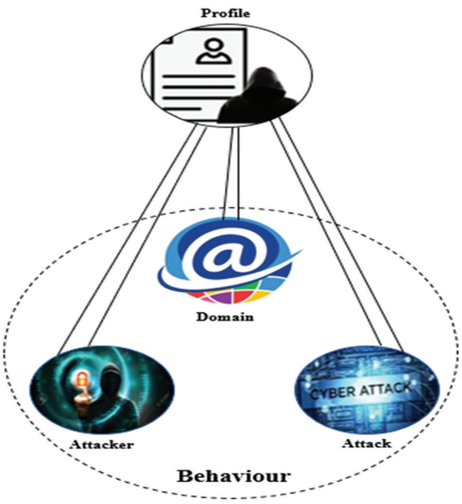


**Fig. 5.** Relation between profile and behavior of attacker

## 3.12. PROFILE OF THE ATTACKER

Every $a_x$ will have a behavior, i.e., there will be a behaviour $h_1$for every $a_x$. A profile $F$ of an attacker indicates the sum of all the attacker's behavior, $a_x.\sum h_1$shows the pattern of $a_x$' s actions.

$$F = \sum h_1 \tag{2}$$

### 3.12.1. Properties Of Profile F

***Property 1:***

Profile $F_n$ is a collection of $h_1$

***Proof:***

Set of all possible attacks $a_x$ will define $P_m$

Set of all possible $a_x$ is a member of any or all $h_1$

A subset of $a_x$ can be any or all $h_1$ members.

A specific $k_x$ behavior is specified by a subset of $a_x$ assigned to a particular $F_1$. Thus, $k_x$ may have only one or multiple $h_1$. For example, if an attacker sends spam email first, he will be assigned to the behavior $h_1$. Because the same attacker is responsible for Virus, he has been given the $h_2$ behaviour. The profile for the attacker is summation of behavior $h_1$, $h_2$ which is given in equation 3.

$$Profile\ F = \mathrm{h}_1 + \mathrm{h}_2 \tag{3}$$

So, profile is a group of behaviors.

***Property 2:***

A cyber attacker cm always performs an action that leads to a purpose, leaving evidence (behaviour) behind. This property was attained by extending Locards exchange principle,

$$\mathrm{h}_1 = \mathrm{c_m k_x} + \Sigma\ a_x \mathrm{k_x} \tag{4}$$

In contrast, no attack is possible in any domain without a motive involving a set of attack vectors.

## 3.13. CONTAINER IMPLEMENTATION EFFICIENCY

A container mechanism is not a new concept; the well-known chroot system was first introduced in the 1970s with Unix operating systems, intended to limit the scope of programs. Container implementations such as OpenVZ, Linux Containers, FreeBSD's Jail, and subsequently Rocket and Docker were among the first. In comparison to virtual and physical computers, container instances are extremely light because they lack an operating system and execute just the functions, i.e., services, required for a container's operation. The containers run on the same kernel as the container management system because they are both based on the same operating system. This solution has the following advantages over physical and virtual servers:

- Implementation of infrastructure at a rapid pace
- Minimal footprint
- A high degree of flexibility
- Easy orchestration
- high density

In order to ensure container security, three main mechanisms are used:

### 3.13.1. Namespaces

The primary and most important security protection for containers is the namespace, as it prevents the containers from learning about host resources, particularly about other container processes or resources that also implement the containers. Docker employs a variety of namespaces in this regard, including User, Net, mnt, and IPC.

### 3.13.2. CGroups

The Cgroup method ensures that all containers have access to the same resources (CPU, memory, IO). Denial of service attacks that result from bad application behavior or malicious actions occurring in any compromised container are prevented by securing the host or other containers. According to the study's examples, the basic launch of publicly available Docker images generally lacks Cgroup settings.

### 3.13.3. Capabilities

With the capabilities system, it is easy to control access to containers. Some actions can be performed inside containers, but control actually extends beyond them. In order to avoid the attack footprint as much as possible, the container capabilities are to be minimized to the bare minimum

## 3.14 FUZZY RULE BASE

The three fuzzy input variables are used to decrease the fuzzy rules. The fuzzy rule base can be applied to reasoning. The security team will be able to predict the possibility of an attack on the low-interaction honeypot by creating this fuzzy rule foundation and fuzzy reasoning engine.

## 4. RESULTS AND DISCUSSION

The proposed Hybrid H-Doc method was deployed on an AmazonWebService-EC2 instance. These systems do not function as firewalls or IDS/IPS systems, but they provide information on attacks, as well as how to avoid or identify them. Consequently, firewalls and IDS/IPS systems can utilize this knowledge to enhance their capacity to counter such analyzed attacks. Honeypot output can be used to find new threat signatures, blacklist IP addresses, map protocol abnormalities, and so on. The honeypot is a useful analytical and scientific tool as a result.

An innovative hybrid honeynet concept is presented in this study. From the existing methods it is concluded that none of them provide a realistic means to identify the level of complexity of an attack in real time on the basis of its behavior and metadata. A test case scenario is used to evaluate the model's functionality:

The following sections list the outcomes of specific experiments assessed using these criteria.

**Table 1.** Attacker's information stored in logs

| IP address | location | Count of sessions | Severe source | Last seen |
|---|---|---|---|---|
| 59.162.172.25 | India, Telangana | 532 | Yes | 2021–10-12 09:46:32 |
| 23.25.132.45 | India, Maharashtra | 846 | Yes | 2021–10-22 14:54:28 |
| 46.206.183.15 | Austria, Wien | 481 | Yes | 2021–10-2823:50:14 |
| 37.118.125.26 | Italy, Marche | 654 | Yes | 2021–10-22 15:44:23 |

The table below shows the format of the assaults on the honeypot that were logged and kept in the database. A similar table, as represented in Table 2, was recorded for every row in Table 1 to log all the sessions per IP address.

| Timestamp | IP address | Sessions | User | Password | Success |
|---|---|---|---|---|---|
| 2021–10-12 09:46:32 | 59.162.172.25 | 45d21 w5423... | Root | #14@es | Yes |
| 2021–10-22 14:54:28 | 23.25.132.45 | 124j14 s21a25... | Root | @89dr | Yes |
| 2021–10-2823:50:14 | 46.206.183.15 | 541k36 4e12sh.. | Root | Gog@13 | No |
| 2021–10-22 15:44:23 | 37.118.125.26 | 952s21 r236e4... | Root | Kih!26 | Yes |

Fig. 6 represents the probability of attackers attacking the honeypot system. The honeypot looks like a genuine computer system, complete with apps and data, leading attackers to believe it is a legitimate target. A honeypot gives IT security teams more insight and helps them respond to attacks that the firewall cannot stop.

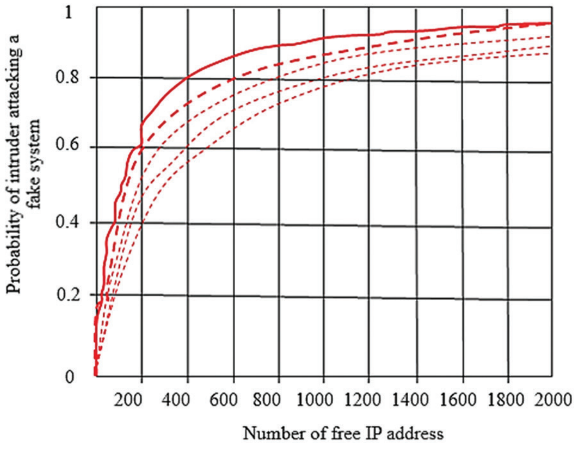


**Fig. 6.** Probability of intruder attacking Hybrid H-DOC

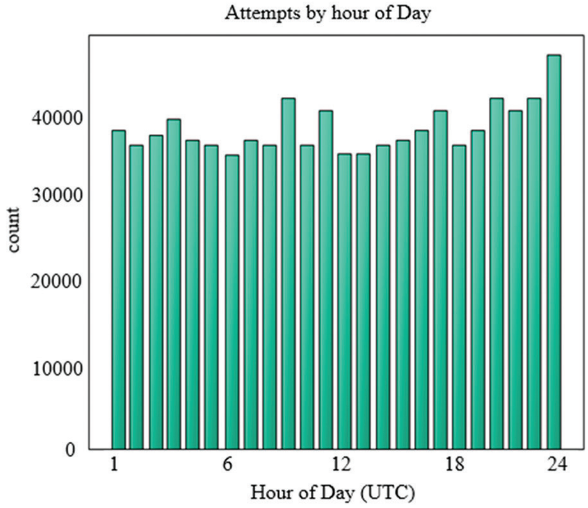


**Fig. 7.** Count of attacks by attacker in an hour of a day

Fig. 7 represents the count of attacks by attacker in an hour of a day which was recorded in the honeypot.71.14 percent of all cases in the prior year were caused by malware, while 28.86 percent were caused by PUAs. Nearly 86 percent of all spambot occurrences were caused by the Gamut spambot.

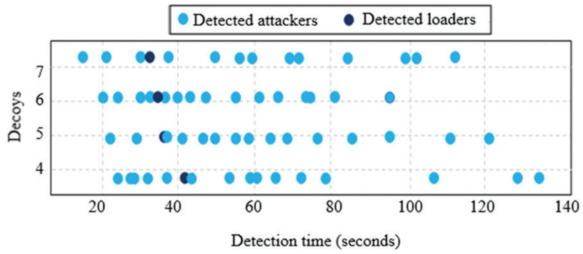


**Fig. 8.** Effectiveness of decoys

Fig. 8 illustrates how decoys can be useful. A graph showing the recognition of bot attacks and the loader can be seen. There may be four, five, six, or seven decoys in each subnet depending on the circumstances. Using seven decoys per subnet, the loader and attacker's detection time is greatly reduced.

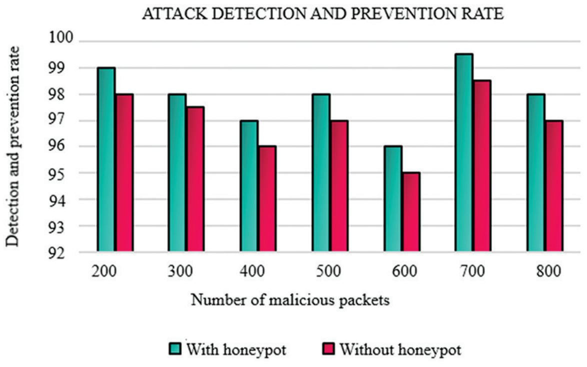


**Fig. 9.** Rate of attack detection and prevention

The results of the attack detection rate in the two situations are shown in Fig. 9. A performance analysis of attack detection is performed on a total of 800 malicious packets. Results indicate that the suggested framework is more effective when honeypot is implemented.

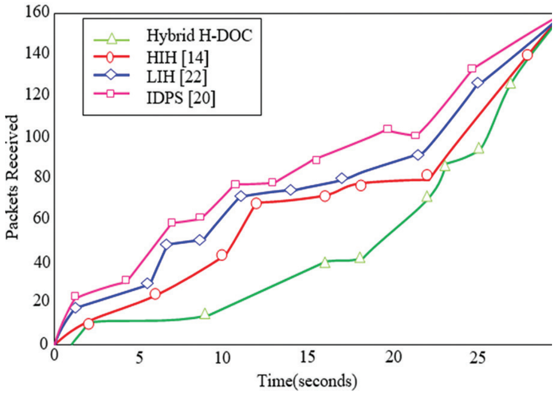


**Fig. 10.** Comparison of Packet arrival rate

A comparison of the packet arrival times for IDS, LIH, HIH and hybrid H-DOC is shown in Fig. 10. The workload of a hybrid honeypot is lower than that of a standard honeypot, according to the study. As a result, performance of false alarms and workload is improved under all network loads.

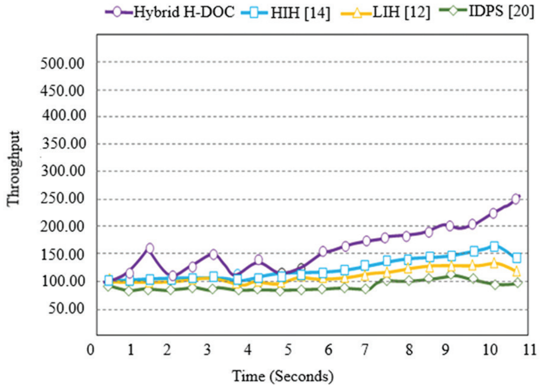


**Fig. 11.** Throughput of honeypots

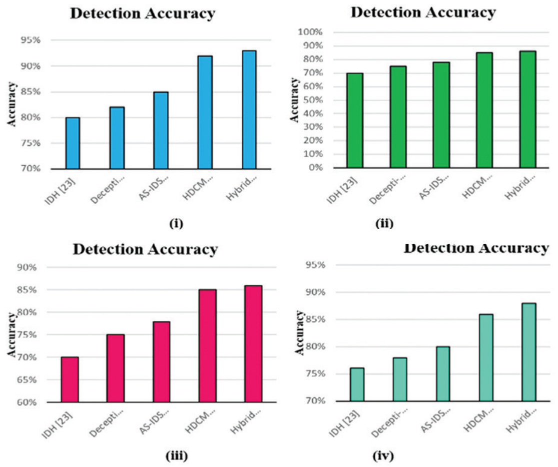


**Fig. 12.** Comparison of Detection Accuracy (i) When number of attackers=10 (ii) number of attackers=20 (iii) When number of attackers=50 (iv) When number of attackers= 100

The Fig. 12 demonstrates the comparison of detection accuracy of the proposed Hybrid H-DOC bait with the existing methods such as Intrusion Detection Honeypot (IDH), Decepti-SCADA (supervisory control and data acquisition), Anamoly and signature Based IDS (AS-IDS), and Honeypot deployment contract-theoretic model (HDCM). The detection accuracy for the proposed Hybrid H-DOC bait is higher when compared to other existing methods.

## 5. CONCLUSION

In this paper a hybrid honeynet model has been proposed. Honeypots are set up instantly using Docker technology for practical testing. This system is simple to use, very effective, and capable of recording data from the attacker and capturing malicious attacks. By updating policies, security administrators can enhance their ability to protect the entire application system. The proposed method uses a server with expert system will decide whether the traffic to be given to low interaction systems or too high, so the efficiency is high. The performance evaluation accomplished has demonstrated the feasibility of the proposed solution. We further plan to deploy the honeypot to collect real-world attack data. The collected data will be used for threat intelligence analysis as well as the automated translation of such intelligence into functional cybersecurity configurations, such as rules for firewalls and/or intrusion detection systems.

## 6. REFERENCES

[1] M. Mirza, M. Usman, R. P. Biuk-Aghai, S. Fong, "A modular approach for implementation of honeypots in cyber security", International Journal of Applied Engineering Research, Vol. 11, No. 8, 2016, pp. 5446-5451.

[2] N. El Kamel, M. Eddabbah, Y. Lmoumen, R. Touahni, "A smart agent design for cyber security based on honeypot and machine learning", Security and Communication Networks, Vol. 2020, 2020.

[3] C. Gupta, "HoneyKube: designing a honeypot using microservices-based architecture", University of Twente, Enschede, Netherlands, Master's thesis, 2021.

[4] E. Chovancová, N. Ádám, "The Security of Heterogeneous Systems based on Cluster High-interaction Hybrid Honeypot", Proceedings of the IEEE 23rd International Conference on Intelligent Engineering Systems, 25-27 April 2019, pp. 81-86.

[5] M. S. Durairajan, R. Saravanan, S. S. Chakkaravarthy, "Low Interaction Honeypot: A Defense

Against Cyber Attacks", Journal of Computational and Theoretical Nanoscience, Vol. 13, No. 8, 2016, pp. 5446-5453.

[6] D. Sever, T. Kišasondi, "Efficiency and security of docker based honeypot systems", Proceedings of the 41st International Convention on Information and Communication Technology, Electronics and Microelectronics, Opatija, Croatia, 21-25 May 2018, pp. 1167-1173.

[7] R. K. Shrivastava, B. Bashir, C. Hota, "Attack detection and forensics using honeypot in IoT environment", Proceedings of the International Conference on Distributed Computing and Internet Technology, Springer, Cham, 2019, pp. 402-409.

[8] R. Surendiran, "A Secure Command Based Approach to find Stolen Mobiles", Research Review International Journal of Multidisciplinary, Vol. 3, No. 10, 2018, pp. 454-456.

[9] I. M. M. Matin, B. Rahardjo, "Malware detection using honeypot and machine learning", Proceedings of the 7th International Conference on Cyber and IT Service Management, Jakarta, Indonesia, 6-8 November 2019, pp. 1-4.

[10] G. K. Sadasivam, C. Hota, B. Anand, "Detection of severe SSH attacks using honeypot servers and machine learning techniques", Software Networking, Vol. 2018, No. 1, 2018, pp. 79-100.

[11] N. Bhagat, B. Arora, "Intrusion detection using honeypots", Proceedings of the Fifth International Conference on Parallel, Distributed and Grid Computing, Solan, India, 20-22 December 2018, pp. 412-417.

[12] Y. Otoum, A. Nayak, "As-ids: Anomaly and signature based ids for the internet of things", Journal of Network and Systems Management, Vol. 29, No. 3, 2021, pp. 1-26.

[13] G. S. Shiny B. M. Kumar, "E2IA-HWSN: Energy Efficient Dual Intelligent Agents based Data Gathering and Emergency Event Delivery in Heterogeneous WSN Enabled IoT", Wireless Personal Communications, Vol. 122, No. 1, pp. 379-408.

[14] A. Appathurai, R. Sundarasekar, C. Raja, E. J. Alex, C. A. Palagan, A. Nithya, "An efficient optimal neural network-based moving vehicle detection in traffic video surveillance system", Circuits, Systems, and Signal Processing, Vol. 39, No. 2, 2020, pp. 734-756.

[15] N. Innab, E. Alomairy, L. Alsheddi, "Hybrid system between anomaly based detection system and honeypot to detect zero day attack", Proceedings of the 21st Saudi Computer Society National Computer Conference, Riyadh, Saudi Arabia, 25-26 April 2018, pp. 1-5.

[16] S. Suratkar, K. Shah, A. Sood, A. Loya, D. Bisure, U. Patil, F. Kazi, "An adaptive honeypot using q-learning with severity analyzer", Journal of Ambient Intelligence and Humanized Computing, Vol. 13, 2021, pp. 1-12.

[17] S. S. Chakkaravarthy, D. Sangeetha, M. V. Cruz, V. Vaidehi, B. Raman, "Design of intrusion detection honeypot using social leopard algorithm to detect IoT ransomware attacks", IEEE Access, Vol. 8, 2020, pp. 169944-169956.

[18] N. Cifranic, R. A. Hallman, J. Romero-Mariona, B. Souza, T. Calton, G. Coca, "Decepti-SCADA: A cyber deception framework for active defense of networked critical infrastructures", Internet of Things, Vol. 12, 2020, pp. 100320.

[19] H. Almohannadi, I. Awan, J. Al Hamar, A. Cullen, J. P. Disso, L. Armitage, "Cyber threat intelligence from honeypot data using elastic search", Proceedings of the IEEE 32nd International Conference on Advanced Information Networking and Applications, Krakow, Poland, 16-18 May 2018, pp. 900-906.

[20] Y. Yin, Y. Shao, X. Wang, Q. Su, "A Flexible Cyber Security Experimentation Platform Architecture Based on Docker", Proceedings of the IEEE 19th International Conference on Software Quality, Reliability and Security Companion, Sofia, Bulgaria, 22-26 July 2019, pp. 413-420.

[21] J. Buzzio-Garcia, "Creation of a High-Interaction Honeypot System based-on Docker containers", Proceedings of the Fifth World Conference on Smart Trends in Systems Security and Sustainability, London, United Kingdom, 29-30 July 2021, pp. 146-151.

[22] S. Sivamohan, S. S. Sridhar, S. Krishnaveni, "Efficient Multi-platform Honeypot for Capturing Real-time Cyber Attacks", Intelligent Data Communication Technologies and Internet of Things, Springer, Singapore, 2022, pp. 291-308.

# Design and Implementation of a flexible Multi-purpose Cryptographic System on low cost FPGA

Original Scientific Paper

**Ahmed Maache**
Laboratory of Signals and Systems
Institute of Electrical Engineering and Electronics
University M'Hamed Bougara of Boumerdes, Algeria
a.maache@univ-boumerdes.dz

**Abdesattar Kalache**
Laboratory of Signals and Systems
Institute of Electrical Engineering and Electronics
University M'Hamed Bougara of Boumerdes, Algeria
kalacheabdessattar@gmail.com

***Abstract*** *– The design of cryptographic hardware that supports multiple cryptographic primitives is common in literature. In this work, a new design is presented consisting of a multi-purpose cryptographic system featuring both 128-bit pipelined AES-CORE (Advanced Encryption Standard) for high-speed symmetric encryption and a Keccak hash core on a low-cost FPGA. The KECCAK-CORE's security and performance parameters are tunable in the sense that capacity, bitrate, and the number of rounds can be user-defined. Such flexibility enables the core to suit a large range of security requirements. The structure of Keccak's sponge construction is exploited to enable different modes of operation. An example application outlined in this work is Pseudo Random Number Generation (PRNG). With few adjustments, the KECCAK-CORE was also operated as a post-processing unit for True Random Number Generation (TRNG) that uses the analog Lorenz chaotic circuit as a physical entropy source. The multi-purpose design was implemented in VHDL targeting an IntelFPGA Cyclone-V FPGA. For AES symmetric encryption, a maximum throughput of 31.1Gbps was achieved and a logic usage of 25146LEs (23% of the FPGA) in the case of the pipelined variant of AES-CORE. For the KECCAK-CORE, maximum throughput figures of 5.81, 8.4, and 11Gbps were achieved for the three SHA-3 variants 512, 384, and 256-bit respectively, with an area usage of 8947LEs (8%). The system as a whole occupies an area of 26909LEs (26%). The random sequences generated by the system operating in PRNG and TRNG post-processing modes successfully passed the National Institute of Standards and Technology (NIST) statistical test suite (NIST SP 800-22). The information entropy analysis performed on the post-processed TRNG sequences indicates an average of 0.935.*

***Keywords****: Advanced encryption standard (AES), Pipelining, FPGA, Keccak, RNG, secure hash algorithm-3 (SHA -3)*

## 1. INTRODUCTION

Security is crucial in today's embedded systems. To ensure data integrity and confidentiality, various cryptographic algorithms have been developed. The structure of these algorithms is highly parallelizable, which makes them favorable for hardware implementations. Also, the emphasis on high communication bandwidths requires the use of hardware-accelerated cryptographic algorithms most commonly ASIC (Application Specific Integrated Circuit) accelerators which offer the highest performance and power efficiency. However, these are not cost-efficient for individual use. FPGAs (Field Programmable Gate Arrays) are flexible alternatives that provide reconfiguration and lower costs for small production. In addition, they can facilitate the design/prototype of cryptographic hardware that combines multiple cryptographic primitives [1], [2], [3]. This flexibility enables the system to support various cryptographic operations and protocols required by applications such as IEEE 802.11, Bluetooth, Transport Layer Security (TLS), Global System for Mobile (GSM), ISO/IEC 29192, etc.

The advantages of using flexible multi-purpose cryptosystems that use ASIC and FPGA lie in their ability to provide a common implementation that supports various requirements for different new technologies such as Wireless Sensor Networks (WSN) and the Internet of Things (IoT). These technologies often need to overcome the challenge of deploying efficient cryptographic algorithms on their resource-constrained nodes [4]. The use of resource-shared multi-purpose cryptographic designs can help in overcoming such challenges and offer area/power reductions rather than using multiple distinct cores. This flexibility can also assist in protecting against different attacks by switching to a more suitable security level of the same algorithm if needed. In terms of performance, throughput requirements may also dictate that the system should switch to a more appropriate configuration [1].

The multi-purpose design proposed in this paper includes two main cryptographic classes: block cipher for symmetric-key encryption/decryption and hash functions. The symmetric encryption is performed using AES128. A new Keccak design and implementation (SHA-3) is presented, which supports multiple modes of operations with tunable security/performance parameters and extendable output length. The security/performance parameters include the capacity, bitrate, and number of rounds. A practical mode of operation supported by the proposed construction is examined, that is a reseedable cryptographically secure PRNG, which is beneficial when high throughput bursts of random bits are needed. Furthermore, this work studies the use of the KECCAK-CORE as a post-processing unit for physical entropy bits (sampled from the analog Lorenz chaotic system) in TRNG to eliminate statistical defects inherited in the *digitized analog signals* (*das*). The post-processed bits can be used to seed the PRNG, the two generators can be used to generate in-system signals required for encryption such as keys and nonces. For consistency and simplicity, the general nature NIST's statistical test suite is used to evaluate both RNGs [5].

Contributions of this work are summarized as follows:

- The implementation of a pipelined AES operated in CTR (Counter) mode with no RTL (Register Transfer Level) register balancing. The architecture provides high throughput while maintaining a good throughput-to-area ratio.
- The hardware implementation/evaluation of the most recent provably secure PRNG in cryptography, the sponge-based PRNG.
- Investigation of the cryptographic hash function based post-processing by taking advantage of the implemented Keccak algorithm to debias the chaotic Lorenz system. Even though hashing algorithms' implementations are relatively demanding in hardware, this is not an issue in the case of this work since the hash function was already implemented for other purposes.

This paper is organized as follows. In Section 2, related literature is outlined. Section 3 covers some needed cryptographic background concepts related to AES, KECCAK, and RNG. Section 4 explains the design and implementation of the proposed system and its supported modes of operation. Section 5 presents the obtained results in terms of performance, area usage, and randomness/entropy-related tests. Section 6 compares these results to other related works. Conclusions are laid out in Section 7.

## 2. RELATED WORK

Several implementations of resource-shared AES with AES-like hash functions have been presented in the literature. Authors in [1] proposed a resource-shared crypto-coprocessor of AES with SHA-3, in which they fit four non-pipelined AES-128 units with SHA3-256 by integrating lookup tables and sharing the unified XOR sections. However, this design is limited in terms of AES throughput because it is non-pipelined. Also, the Six Input Equation optimization used cannot take advantage of other devices with different LUT input sizes. In [6], a resource-sharing design of AES and Fugue was outlined. However, Fugue was a second-round candidate that got eliminated due to security flaws and its average throughput-to-area ratio [7]. Other studies [8-11] have implemented resource-shared AES with Grøstl-256 on different FPGA platforms. Compared to Keccak, Grøstl-256 has relatively small security margins, lower throughput, throughput/area, and energy consumption-per-bit on FPGAs and ASICs [12]. Also, many of these designs are non-pipelined, which results in low area usage with relatively low throughput. Furthermore, the hashing units are of fixed output lengths and security parameters, which in turn limits their suitability for different applications. HLS (High-Level Synthesis) was used in [13] to implement a dynamically configurable SHA-3 accelerator in terms of digest length and capacity. However, the use of HLS is relatively less efficient than pure HDL (Hardware Description Language) design flow. High-speed pipelined SHA-3 designs were presented in [14-18]. However, they support no flexibility, and hence a narrower application scope.

Authors in [19] proposed an ultra-high throughput fully pipelined AES operated in CTR mode. A 60Gbps inner and outer pipelined AES architecture was proposed in [20]. More sophisticated timing optimization techniques were described in [21] where an 82Gbps pipelined AES was designed using 2-slow balancing techniques. While these works provide ultra-high throughputs, the area usage is not proportional to the increase in throughput, hence, a low throughput-to-area ratio. In [3], a highly customized VLSI (Very Large Scale Integration) design of an advanced AES Cryptoprocessor is presented. The design supports multiple modes of operations targeting the European Processor Initiative (EPI) that features multiple hardware cryptographic accelerators, including SHA, which are controlled by a secure RISC-V processor.

PRNGs are well studied in the literature. FPGA stream cipher/LFSR-based PRNGs were proposed in [22], [23]. Chaos-based PRNGs were presented in [24], [25] where the three-dimensional chaotic systems were implemented digitally, which implies a deterministic fully predictable system with no sensitivity to initial conditions. Bits from the three variables were post-processed to generate pseudo-random numbers. The above-stated works provide very compact hardware implementation and good performance. On the other hand, hardware and software designs of cryptographically secure PRNGs were proposed. Authors in [26] demonstrated high throughput and low power FPGA implementations of two PRNGs, one of which is the computationally secure Blum Blum Shub generator. In contrast to the algorithmic nature of PRNGs, TRNGs are physical. However, an algorithmic debiasing step performed on the digitized bits is required; also referred to as TRNG post-processing. Authors in [27] relied on SHA-256 to debias bits generated by ring oscillators which is fundamentally different from chaotic systems.

## 3. BACKGROUND

### 3.1. AES ALGORITHM

AES can provide up to the TOP SECRET level of security with high performance on both software and hardware. It features multiple key lengths of 128, 192, and 256 bits each with the number of rounds 10, 12, and 14 respectively. Initially, the encryption starts by XORing the key and the input data before feeding the result through the rounds. Each round consists of four steps: SubBytes, ShiftRows, MixColumns, and AddRoundKey, except for the last round which requires no column mixing. SubBytes step is a nonlinear byte-to-byte correspondence described by the S-BOX which is obtained by calculating the multiplicative inverse of the input byte followed by an affine transform. ShiftRows is a transformation of the state via circular shifts with different values at each row. The previous two steps provide the required confusion to avoid any differentiability between the input and output. MixColumns is a transformation that operates on the State column-by-column, treating each column as a four-term polynomial [28] to ensure diffusion. The AddRoundKey step consists of XORing the resultant state from Mixcolumns with the RoundKey provided by the key expansion.

CTR is a feedback-free block cipher mode where the ciphertext is a result of XORing the plaintext with the output of encrypted successive counter values concatenated with a nonce. The use of a single XOR operation on the plaintext results in a symmetry between the encryption and decryption. Unlike feedback modes, CTR mode is highly parallelizable, random read accessible, and suffers no error propagation problems making its implementation straightforward.

### 3.2. SHA-3 AND KECCAK FAMILY

AES SHA-3 is a subset of the broader cryptographic primitive family Keccak of hash functions. It is based on the novel Hermetic Sponge Construction approach. Although it features various state widths b = {25, 50, 100, 200, 400, 800, 1600}-bit, only permutation of b = 1600 bits was submitted [29]. The Sponge Construction is divided into two phases: *absorption* and *squeezing* phase as seen in Fig. 1. It uses $b = r + c$ bits of state, where $r$ is the rate at which states are updated with message bits between each application of the permutation function, $c$ is the capacity and defines the security level. Increasing the capacity results in a higher security level with a performance penalty and vice-versa [29]. Its value is set by the user depending on the application. The input state of Keccak-f[b] is arranged in a three-dimensional matrix of *5×5×w*, where *w* defines the length of the lane and equals *b/25*.

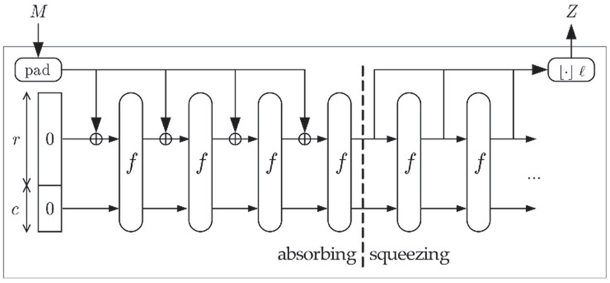


**Fig. 1.** The Sponge Construction [30]

In the absorption phase, the input block of length $r$ is XORed with the state lane-wise. If the length of the current input block is less than $r$, padding is required. After the input data is completely absorbed, the output is obtained by truncating the state. The output length is a user choice in certain applications. For hash functions, the lengths are 224, 256, 384, and 512-bit. If the required output length is greater than the bitrate, it is obtained by truncating the outputs after feeding through Keccak-f until the required length is satisfied, which is known as the squeezing phase [31]. Each round $R$ of Keccak-*f*[*b*] consists of five-step mappings $R = \iota\ o\ \chi\ o\ \pi\ o\ \rho\ o\ \theta$ where:

$$\begin{aligned}
[\theta] : C[x][z] &\leftarrow a[x][0][z] \oplus a[x][1][z] \oplus ... \oplus a[x][4][z] \\
D[x][z] &\leftarrow C[x-1][z] \oplus C[x+1][z-1] \\
a[x][y][z] &\leftarrow a[x][y][z] \oplus D[x][z] \\
[\rho][\pi] : B[y][(2.x + 3.y)][z] &\leftarrow A[x][y][rp(x,y)] \\
[\chi] : a[x][y][z] &\leftarrow B[x][y][z] \oplus (B[x+1][y][z] \oplus 1) \\
&\quad .B[x+2][y][z] \\
[\iota] : a[0][0][z] &\leftarrow a[0][0][z] \oplus RC_i[z]
\end{aligned} \tag{1}$$

where *a[x,y,z]* represents a particular lane of a state; *B*, *C,* and *D* are the intermediate results, and $RC_i$ is the round constant. Note that $\pi$ and $\rho$ steps are combined because they are merely two consecutive permutations of the state. The values of this permutation are hardwired in the *rp* array. The number of rounds to be executed $n_r$ is calculated as $12 + 2(\log_2(b/25))$.

### 3.3. RANDOM NUMBER GENERATION

**3.3.1. Sponge-based PRNG:** PRNG is a deterministic algorithm that, given a truly random binary sequence of length $n$ referred to as a seed, outputs a binary sequence of length $N > n$ that is completely determined by the seed and 'looks' random. Thus, if the seed is compromised, the output sequences of the PRNG are known. Therefore, a requirement for a seed is randomness, hence a common technique is to seed a PRNG with a TRNG and then use the PRNG afterward. This is because PRNG has superior performance relative to TRNG. Another requirement for the seed is a sufficient length for it to be insusceptible to brute-force recovery. Moreover, the seed should contain enough entropy for the application since the seed entropy defines an upper limit for the entropy that the PRNG can deliver. Ideally, a PRNG can be constructed with a random oracle that responds deterministically to every unique query with a truly random response chosen uniformly from its output domain. Non-empty seeds are fed to the PRNG to update the state and random output bits are fetched afterward. It should be known that previous seeds should be stored, hence the name history-keeping mode [32]. The history is encoded by seed-complete encoding function *e(h)*. The encoded history is provided to the random oracle during the fetch call producing a random sequence that is truncated at every feed-fetch cycle and so on (Fig. 2). The memory needed for history-keeping mode grows linearly with the number of past queries; rendering it impractical.

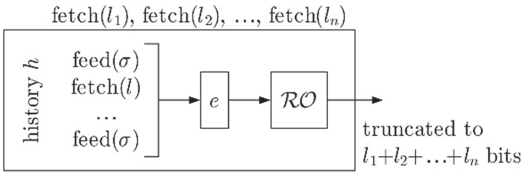


**Fig. 2.** History-keeping mode PRNG [32]

Instead of a random oracle, sponge construction can be used as in [32], which yields a history-keeping mode similar to reseedable PRNG. The sponge construction relies on a fixed length state which implies no memory growth. Furthermore, sponge construction features similarities with the mode; where every input absorption of length $r$ into the state is a feed request and the same goes for fetch requests (Fig. 1). In addition, the last $c$ bits of the state are never directly affected by the input blocks. Capacity $c$ determines the attainable security level of the construction.

**3.3.2. TRNG:** TRNG extracts bits from a non-deterministic physical process. Naturally, the physical process produces a continuous-time analog signal, which gets digitized uniformly to yield *das*. Due to their physical nature, TRNGs are typically a separate piece of hardware connected to the application via an interface (USB or PCI bus...). Examples of physical processes used are Josephson's Junction [33], Johnson's noise [34], and Zener diode's shot noise [35]. Unlike PRNGs, TRNGs suffer from uneven probabilities of zeros and ones. The difference of probabilities of 0s and 1s is termed bias $b =$ *(p(1)-p(0))/2*. For that, a post-processing step is needed where the digitized data is transformed into uniformly distributed random numbers i.e $b = 0$. Post-processing also helps to eliminate other statistical defects introduced by the physical source. While the output's entropy of the PRNG is bounded up by the seed's entropy, TRNGs' output entropy increases after each random number is generated.

### 3.4. CHAOTIC SYSTEMS AS ENTROPY SOURCES

Chaotic systems are by no means random. They exhibit a deterministic behavior since this class of systems can be described neatly as a system of nonlinear ordinary differential equations. Since the system cannot be solved analytically, mathematicians resort to phase space to have a qualitative rather than quantitative understanding of the system. Solutions to the system's differential equation can be represented as a trajectory in phase space. The system being chaotic implies that two trajectories will be $||E(t)|| \sim ||E(t_0)||.e^{\lambda t}$ apart from each other, where $t > t_0$ and $\lambda$ is Lyapunov exponent. This encapsulates the sensitive dependence on initial conditions resulting from the exponential divergence of nearby trajectories [36]. This limits the horizon of prediction up to a time $t_h$, known as the Lyapunov time; which corresponds also to the loss of one bit of information [37]. In addition to the sensitivity to initial conditions, chaotic systems exhibit nonperiodic behavior. In this work, the Lorenz system will be used as an entropy source for the RNG, which is a simplified mathematical model for atmospheric convection (details are in Section 4.3.2.).

A successive sampling of the Lorenz system results in a seeding material for the PRNG or a das for the TRNG. The entropy of the material depends on the sampling frequency, sampling resolution, and quantization loss of the quantization function. The source entropy should be at least equal to the security strength of the instantiation [38]. The instantiation shall provide reseeding in case the internal state of the PRNG is compromised. Periodic reseeding shall reduce the likelihood of a security threat.

## 4. SYSTEM DESIGN AND IMPLEMENTATION

The multi-purpose cryptographic system consists of two cryptographic cores. The first is *AES-CORE* dedicated mainly to 128-bit symmetric-key encryption operated in CTR mode. The second is a tunable *KECCAK-CORE*, based on KECCAK[b=1600], that supports different modes of operation. The normal mode of operation of the KECCAK-CORE is a certified hash function (SHA-3) that outputs digests of a fixed predefined length. However, its utility can be extended to include an RNG, which can be used to generate high-quality keys/seeds required by the AES-CORE CTR mode. For that reason, the output register of the KECCAK-CORE is not only connected to the device output ports but also to the input port of the AES-CORE. An overall system diagram is seen in Fig. 3.

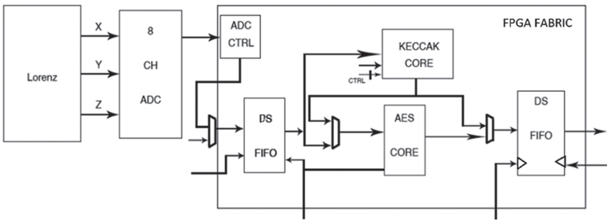


**Fig. 3.** Multipurpose Cryptographic System Overview

The system is synchronized to an external interfacing circuitry by two Double Sided FIFOs (First-In-First-Out), one for the input ports and one for the output ports. The input DSFIFO is a 16-bit write-side and 128-bit read-side (which is equal to the system's data bus width). The DSFIFO-based synchronization is dedicated only to data. Control signals synchronization, on the other hand, is performed with register synchronization chains. The output DSFIFO's write-side is 128-bit muxed with the output registers of both cores, meanwhile, the read-side is 16-bit.

The on-board ADC is used to provide the system with accessibility to physical entropy sources (Lorenz chaotic circuit). It is an 8-channel 12-bit SAR clocked at 1Mhz. The system can be adjusted to support any number of ADC channels with FIFO synchronization, however, the implemented system supports only one channel for the sake of simplicity and thus only one of the three system's variables can be sampled at once. Interfacing the ADC is done using IntelFPGA ADC Controller IP (Intellectual Property), which is very straightforward. The IP provides parallel access to all channels simultaneously. The channel is muxed into the input FIFO of the system. The detailed RTL view of the system is shown in Fig. 4. The following subsections will explain the implementation details of each part of the system.

### 4.1. AES-CORE IMPLEMENTATION

**4.1.1. Non-pipelined CTR AES-128:** AES-128 requires successive ten rounds that only differ in the round key derived from the KeyExpansion. Each round key is calculated based on the previous key and a unique round constant. The KeyExpansion procedure in the ten rounds is also identical. The four round steps alongside the key scheduler were implemented as combinational logic. SubBytes step was implemented as 16 parallel LUTs representing the S-box table. ShiftRows is a rewiring of the state, hence, its implementation is straightforward and requires no hardware. MixColumns step requires multiplication by "two" and "three" in *GF($2^8$)* which is achieved easily since multiplication by "two" is a shift to the left while multiplication by "three" is a shift and XOR. The AddRoundKey step is an XOR operation between the state and the round key. Non-pipelined AES128 core is implemented with a single block of round encryption and KeyExpansion. Round constants are provided by an FSM along with a control signal to exclude the MixColumns step from the 10th round. The ciphertext is a result of the plaintext XORed with the latched output of round 10; the 32-bit counter of CTR mode is then incremented, concatenated with the 96-bit nonce, and fed into the round block. The 128-bit encryption requires 11 clock cycles and the next plaintext should be provided during that window. A similar core was used in [39] to enc/dec real-time video data as an example application.

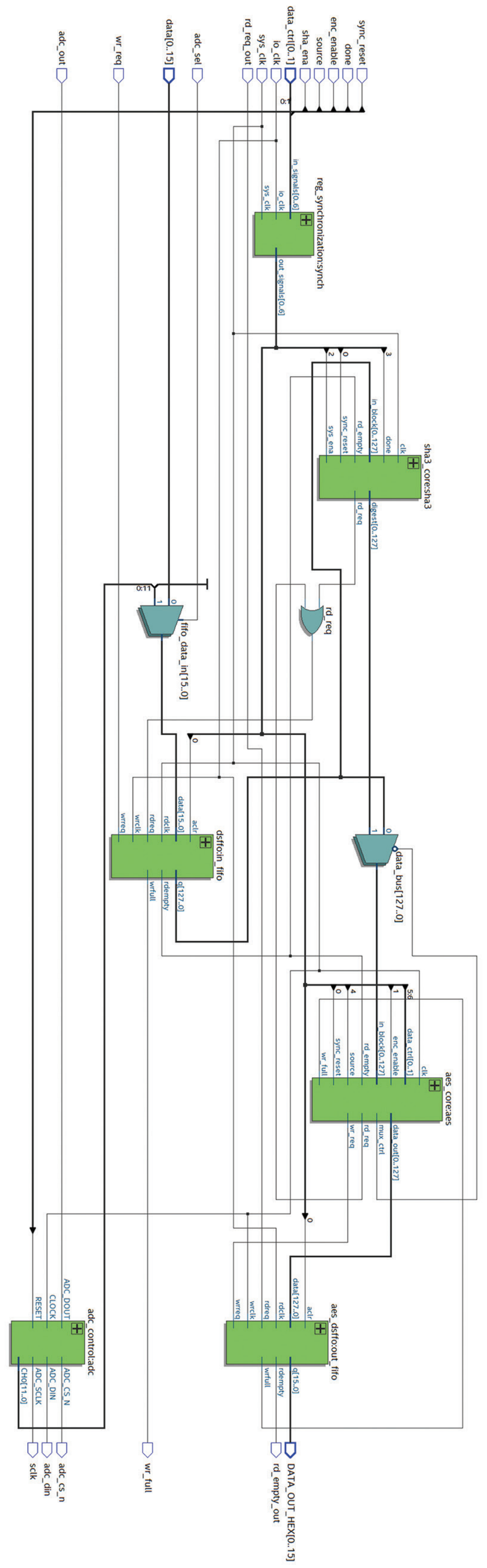


**Fig. 4.** RTL view of the on-FPGA system

**4.1.2. Pipelined CTR AES-128:** AES's implementation has a wide range of requirements and constraints on throughput and area. For the implementation to be compatible with high-speed applications, high throughput architectures are required. One method to improve the throughput is pipelining. This design is obtained by placing registers between the unrolled encryption and Key Scheduler rounds. Encryption and KeyExpansion rounds are identical to those of the non-pipelined variant. However round-key and control signals are hard-wired in each round and no FSM control unit is needed, hence, minimizing the critical path delay. Encryption rounds run parallel to the KeyExpansion. Therefore, timing has to be exact for each round key from the RoundExpansion to arrive at its corresponding encryption round. The key, seed, and plaintext registers are multiplexed into the input port of the core along with the corresponding control signals for compactness. The input port itself is demultiplexed to the output of the Sponge PRNG and the device input pins. Therefore, both seed and key can be obtained from the RNG or provided by an external interfacing circuitry. Double Sided FIFOs are used to separate the clock domains. This architecture encrypts a 128-bit block every clock cycle with a latency of 11 clock cycles.

## 4.2. KECCAK-CORE IMPLEMENTATION

For the implementation of Keccak-f, the straightforward unfolded structure was chosen as seen in Fig. 5.

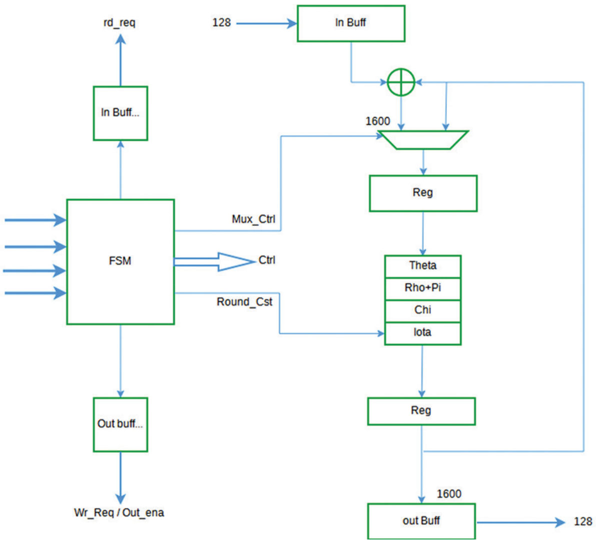


**Fig. 5.** KECCAK-CORE Implementation

A multiplexer controlled by a round counter determines whether the state is updated with the result of the round function or with the new input block during the absorption phase. The input block is stored in an input buffer in a form of a SIPO (Serial-In-Parallel-Out) shift register. At every clock, 128-bit input enters the register in parallel with Keccak-f computation to avoid overhead. The state is stored in a 1600-bit register that is updated after each round. The five steps forming the round function are implemented using combinational logic with no inner round pipelining.

The round constants are provided by an FSM. After the last round, the state is truncated to 256-bit to output the hash digest.

The number of rounds $n_r$ is user-defined with two constraints: it must be at least 24, and the number of rounds can only be multiples of 12. The rationale for that is not a security concern. It is related to flexibility and usability of the implementation in applications where slowing down the hash operation is beneficial. The user-defined $n_r$ mode is obtained by reusing the states in the FSM. The control unit will loop indefinitely between the 24 rounds' states as long as a control signal is pulled high. The control signal is checked at the 12th and 24th states, which explains the multiples of 12. The output digest buffer is connected to the output DSFIFO of the device and the input port of the AES-CORE.

Four modes of operation for the KECCAK-CORE are defined. First of which is a '*randomized hash function*' where data is fed as 16-bit chunks to the input DSFIFO of the device and should be padded beforehand. The input bits are first registered in the input buffer of KECCAK-CORE before being forwarded to the absorption phase of Keccakf [$b = 1600, n_r = 24$]. A digest is then obtained using this configuration. Note that the values of *c* and *r* have not been stated, even though they define the level of security/performance of the hash function and the SHA-3 hash variant (output length), because they are user-defined. The second mode of operation is a '*slow one-way function/key derivation function*', which is obtained by simply increasing $n_r$, hence, slowing down the function. The remaining two modes are a '*reseedable PRNG*' and a '*post-processing unit for TRNG*', which are explained in the next subsection.

## 4.3. RNG IMPLEMENTATION

**4.3.1. Sponge-based PRNG:** The PRNG design should be flexible enough in terms of performance and security to suit most cryptographic applications. For instance, the requirements of generating random bits for nonces are different from those to be used as cryptographic keys and so on. Furthermore, since RNGs require sources of entropy as mentioned earlier, the implemented RNG's parameters have to be tunable according to the available entropy source. Lastly, the RNG has to have access to multiple physical entropy sources simultaneously while keeping the implementation as compact as possible. All of this can be achieved by tweaking the Keccak hash function.

First, the security/performance tradeoffs of the PRNG should be adjustable according to the application. As mentioned earlier, the capacity value determines a ceiling to the security level that the sponge function provides [29] and defines the resistance of the construction to state recovery attacks; an increase in capacity *c* implies a decrease in the bitrate *r* and a drop in performance occurs as a result. One practical way to adjust the capacity/bitrate is to alter the input buffer shift register in KECCAK-CORE by masking the last c bits. This method is quite simple and requires minor resources. However, a huge entropy loss is conceived due to the masked bits being no longer stored in the input DSFIFO. This entropy loss is not tolerated in the case of large capacities or when a continuous flow of entropy from the sources is required.

A better approach is to control the read signal from the KECCAK-CORE to the input DSFIFO. By tweaking the control unit at a few rounds' states, the read signal is pulled high at a given interval of rounds defined by an external 3-bit signal. The featured capacity values are {256,1024} with a stride of 128. Secondly, since the number of rounds dictates the security margin, it is adjustable as mentioned earlier. This may also be helpful in the case of entropy sources with high correlations even though Keccak's security margin is already thick and should fulfill its security claims even in the case of a decrease in the number of rounds [40].

Lastly, the implemented RNG is reseedable, meaning that an additional entropy source can be added after random bits have been generated. Instead of throwing away the current state of the PRNG, reseeding combines the current state of the generator with the new seed material [41]. The implementation can be operated as a reseedable sponge PRNG, using the sponge function of KECCAK-CORE in a cascaded way without the state being reinitialized. This implementation features two requests: feed and fetch. At first, the seed material is fed to the input DSFIFO as chunks of 16 bits before them being absorbed by the sponge's input buffer shift register as chunks of 128 bits. When the input seed length *l* is equal to *r*, where *r* is user-defined as discussed earlier, the input buffer is XORed with the current state and forwarded to Keccak-f. This sequence is repeated until the DSFIFO issues a read-empty signal indicating that the seed material is completely absorbed. Next, the sponge is switched to the squeezing phase where the desired output length is obtained [32]. After applying the iterated Keccak-f, the state is stored in a shift register that is controlled externally to iterate through the desired length of the output *l* ≤ *r*. The output shift register is connected to a 128-bit digest buffer that feeds to an output DSFIFO with a 128-bit write-side and 16-bit read-side controlled by the user to fetch the random bits.

One can fetch one random bit directly after feeding each seed with length *l* < *r* through Keccak-f[*b* = *1600*]. In that case, the implementation is operating in the duplex construction mode. Unlike a sponge function that is stateless in between calls, the duplex construction accepts calls that take an input string and return an output string depending on all inputs received so far. The instance of the duplex construction (Fig. 6) is known as a duplex object [41]. This can go further by letting duplex objects non-identical to one another. This asymmetry is supported by the implementation and can be obtained by altering the capacity and number of rounds control signals.

The output of the PRNG is connected to the multiplexed input port of AES-CORE to deliver the generated 128-bit keys and 96-bit nonces required for the current encryption session. The Sponge performance/security parameters must be chosen accordingly.

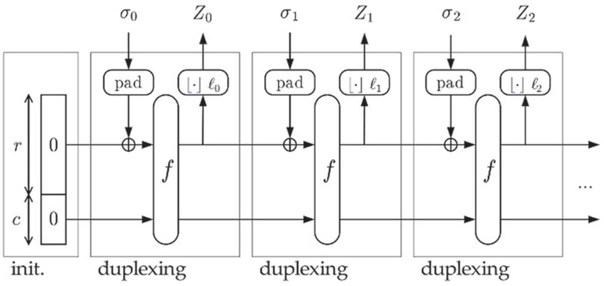


**Fig. 6.** The Duplex Construction [30]

**4.3.2. Post-processing the Lorenz System:** Chaotic dynamical systems are chosen as an entropy source for their unpredictability over sufficiently long periods. The Lorenz system was selected owing to its popularity and simple analog implementation. Fig. 7 shows an analog Lorenz circuit with non-linear feedback loops from the outputs of integrator circuits to two analog multipliers AD633. The output of the integrators (implemented using LM358P op-amps) represent the values *x, y,* and *z* of the Lorenz system given by Equation 2. It can be noted that this system does not exhibit chaotic behavior until a specific range of values for the system parameters *ρ, σ,* and *β*. These physical parameters are set by the values of capacitors and resistors, hence, they can be tuned. The range of voltages can be adjusted to suit the ADC range by changing the resistance connected to the op-amp inputs [42].

$$\begin{aligned}\frac{dx}{dt} &= \frac{1}{R_1C}.(y-x)\\ \frac{dy}{dt} &= \frac{1}{R_3C}.x - \frac{1}{R_5C}.y - \frac{1}{100R_4C}.xz\\ \frac{dz}{dt} &= \frac{1}{100R_6C}.xy - \frac{1}{R_7C}.z\end{aligned} \tag{2}$$

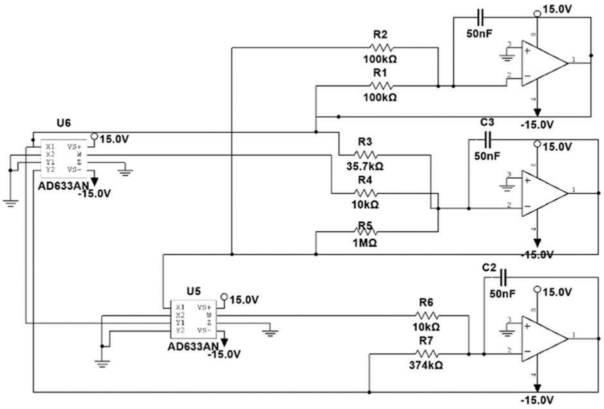


**Fig. 7.** Analog Lorenz circuit (*ρ=28, σ=10, β=8/3)*

The outputs *x,y,* and *z* can be connected to three ADC channels of the FPGA board. In the conducted experiments, only the *x* output is connected and sampled with a frequency of *1Mhz* even though the power spectrum of the circuit is concentrated on the lower frequencies. This will eventually lead to correlations in the values of *x* due to continuity, which will be more apparent due to SAR's quantization loss. Also, relying on two signals simultaneously does not increase the source's entropy due to the high mutual entropy between any pair of variables of Lorenz equations. If multiple physical processes are desired, one way is to take advantage of causal chains with a high degree of independence [43].

The *das* suffers from statistical defects, which is a manifest of the system's continuities and autocorrelations due to the system's memory. Another plausible problem is the upper bound limit on entropy which was shown in the works of [44], [45] on chaotic univariate maps. Fortunately, the two works are not concerned with continuous chaotic multivariate systems, and no work disproved the conjecture for this category. The entropy source may eventually produce zero information asymptotically [46]. In addition, a good evaluation of the TRNG's entropy takes into account the sampling frequency, the ADC resolution, the aging of components that may cause the noise source to completely break down, and the non-ideal behavior of the different components (for instance the low-pass behavior of the op-amps used) [43]. Investigating this point is out of the scope of the present work. This work, however, provides an assessment of the performance of the KECCAK-CORE post-processing step on eliminating statistical defects and examines the results of the RNG in non-ideal conditions.

A sponge-based TRNG design is straightforward and similar to that of a sponge-based PRNG. The major difference is that in the process of producing true random numbers, the KECCAK-CORE's sponge function resembles a post-processing step. The number of rounds is fixed to a minimum of 24 rounds. In contrast to PRNG mode, no output is squeezed before the end of absorbing the entire input block of the desired entropy, and the state is reinitialized afterward. This implies that there is no need for the capacity parameter *c* and the bitrate *r* to be set to meet the performance requirements of the application. However, if the user does not discard the state, or if the mode is switched to a PRNG seeded with the physically generated das, the user should readjust the capacity *c* to provide the resistance required against attacks.

## 5. RESULTS AND DISCUSSION

All cores were manually implemented in VHDL (VHSIC Hardware Description Language) and synthesized using Intel Quartus Prime 20.1 on Cyclone-V FPGA (5CSXFC6D6F31C6) using the DE-10 Standard board.

### 5.1. AES-CORE

The non-pipelined AES-CORE hit an $f_{max}$ of 215.3Mhz which implies a critical path delay of 4.64ns from the control unit state register to the datapath register before the encryption unit. Each round of encryption is completed in one clock cycle, which means that the entire encryption operation takes 11 clock cycles per block. That yields a maximum throughput of 2.33Gbps. This core is lightweight with 3% area consumption. The synthesis optimization mode was set to 'balanced' and the advanced physical optimization was turned on. In general, this core is low-area with good throughput.

The pipelined AES-CORE hit an $f_{max}$ of 260.92MHz, a 21% improvement over the non-pipelined one. This is because the signals in the pipelined architecture are hardwired in each round and no control unit is needed; thus minimizing the maximum delay on the critical path. The maximum throughput is roughly 31Gbps, which is 13 times faster than its non-pipelined counterpart. As seen in Table 1, the core utilizes around 25kLEs -Logic Elements- (equivalent to 9973ALMs -Adaptive Logic Modules-) which accounts for 23% of the device's total logic elements.

### 5.2. KECCAK-CORE

The KECCAK-CORE used 8947LEs (3415ALMs) which account for 8% of the FPGA, making it lightweight and suitable for embedded applications. The usage of dedicated logic registers accounts for over 70% of the total logic usage. This is due to the inherited reliance on permutation in the Keccak family. In addition to low-area usage and high flexibility, the core clocks at 271.69MHz.

The maximum throughput of the core is of the form:

$$Throughput = \frac{TP_{max}}{n_r + 1}.(1 - \frac{c}{b}) \quad (3)$$

where *b = 1600*, $TP_{max}$ is a constant that depends on the maximum frequency $f_{max}$. It can be viewed as the throughput of the core with *c = 0* and $n_r = 1$, resulting in $TP_{max}$ =404.73Gbps. Of course, operating the core in these conditions is dangerous and never recommended given that the security margin against distinguishers is coupled with the number of rounds [12], [47], [48]. While lowering the capacity decreases the level of attainable security in the sponge construction. Moreover, eliminating the capacity introduces a vulnerability to length extension attacks in Hash-based Message Authentication Codes (HMACs) as well as violating the assumptions made by the original authors in [30]. The values of $n_r$ and *b* are substituted as defined by NIST's standard [31], and the capacity for each SHA-3 variant is equal to twice the digest length *l* yielding:

$$Throughput = \frac{TP_{max}}{25}.(1 - \frac{2.l}{1600}) \quad (4)$$

Results obtained for the three SHA-3 variants are shown in Table 1.

**Table 1.** Results of the AES/SHA-3 implementations

| Core | Area (LEs) | fmax (MHz) | Thr. (Mbps) | Mbps/ LE |
|---|---|---|---|---|
| AES Non-pipelined | 3548 | 215.30 | 2385.9 | 0.67 |
| AES Pipelined | 25146 | 260.92 | 31846.4 | 1.26 |
| SHA-3 256 | 8947 | 271.69 | 11.00 | 2.60 |
| SHA-3 384 | 8947 | 271.69 | 8.40 | 1.98 |
| SHA-3 512 | 8947 | 271.69 | 5.81 | 1.38 |

### 5.3. SPONGE-BASED PRNG

The performance of the PRNG depends on the capacity. However, operating the KECCAK-CORE in a reseedable PRNG mode is fundamentally different from that of a hashing standard mode. Capacity and bitrate can be user-defined, but one should set the parameters along with a proper limit on the number of output blocks squeezed before each seed refresh. For instance, The *KECCAK[r,c]* PRNG provides a resistance of $2^c$ against state recovery if the number of output bits between times where the state has gained at least *c* bits of fresh seeding material has an upper bound of $r.2^{r/2}$ [32].

Pseudo-random numbers are generated using the previously implemented KECCAK-f[b=1600], more specifically KECCAK[*r=1088, c=512*]. The random sequence is obtained by squeezing the state after providing the empty string as input. The system is clocked at a 100MHz clock frequency. Each squeezed block is 256-bit, taken every 24 clock cycles. If the state is reseeded with *n* high entropy bits, it yields a resistance of $2^n$ to any state recovery attack. Random bits are extracted using Quartus Prime's Signal Tap Logic Analyzer as shown in Fig. 8. The Figure shows the random number generated by the sponge-based PRNG which is the output state of sha-3 256-bit core (*sha3|in_state[0..255]* at the top) every time sample (24 clock cycles) and its individual output bits (*sha3|in_state[i]*). These generated bits were collected for randomness and entropy analysis.

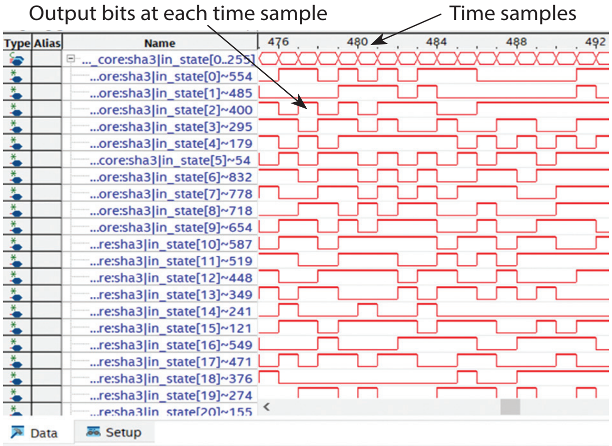

Output bits at each time sample
Time samples
Type
Alias
Name
476
480
484
488
492
..._core:sha3|in_state[0..255]
...ore:sha3|in_state[0]~554
...ore:sha3|in_state[1]~485
...ore:sha3|in_state[2]~400
...ore:sha3|in_state[3]~295
...ore:sha3|in_state[4]~179
...core:sha3|in_state[5]~54
...ore:sha3|in_state[6]~832
...ore:sha3|in_state[7]~778
...ore:sha3|in_state[8]~718
...ore:sha3|in_state[9]~654
...re:sha3|in_state[10]~587
...re:sha3|in_state[11]~519
...re:sha3|in_state[12]~448
...re:sha3|in_state[13]~349
...re:sha3|in_state[14]~241
...re:sha3|in_state[15]~121
...re:sha3|in_state[16]~549
...re:sha3|in_state[17]~471
...re:sha3|in_state[18]~376
...re:sha3|in_state[19]~274
...re:sha3|in_state[20]~155
Data
Setup

**Fig. 8.** Random Bits Extraction via Signal Tap

To assess the quality of the pseudo-random numbers generated, the NIST SP 800-22 verification is used [5]. Results in Table 2 indicate that the sequence passed all tests with all P-values > 0.01, which implies that the sequence is random.

On the analog side, the Lorenz chaotic system is implemented using discrete components. Its attractor is observed after plotting variables *x* and *y* in XY mode on a digital oscilloscope as seen in Fig. 9. The *x* variable is connected to an ADC channel. Due to the high sampling frequency which is much higher than the highest frequency component in the signal (< 10Khz which is the cutoff frequency of the LM358P at unity gain), das taken at short periods and fed into the post-processing unit can suffer from serious statistical defects. The same verification tests were also used to assess the quality of the generated random numbers. The verification was performed on 450000 bits without an internal state reset. Results in Table 2 indicate that the random sequence passed all tests.

**Table 2.** NIST's Statistical tests results

| Test | PRNG P-value | TRNG P-value |
|---|---|---|
| Frequency Test (Monobit) | 0.01397 | 0.30962 |
| Frequency Test within a Block | 0.12058 | 0.49488 |
| Run Test | 0.56053 | 0.35807 |
| Longest Run of Ones | 0.13052 | 0.12262 |
| Binary Matrix Rank | 0.29228 | 0.12260 |
| Discrete Fourier Transform | 0.69992 | 0.38835 |
| Non-Overlapping Template Matching | 0.36968 | 0.54103 |
| Overlapping Template Matching | 0.84336 | 0.13222 |
| Maurer's Universal | 0.31458 | 0.16966 |
| Linear Complexity | 0.22099 | 0.41366 |
| Serial test | 0.70625 | 0.51154 |
| Approximate Entropy | 0.17980 | 0.04060 |
| Cumulative Sums (Forward) | 0.01225 | 0.43620 |
| Cumulative Sums (Reverse) | 0.01225 | 0.54158 |
| Random Excursions | 0.14555 | 0.28605 |
| Random Excursions Variant | 0.26355 | 0.83167 |

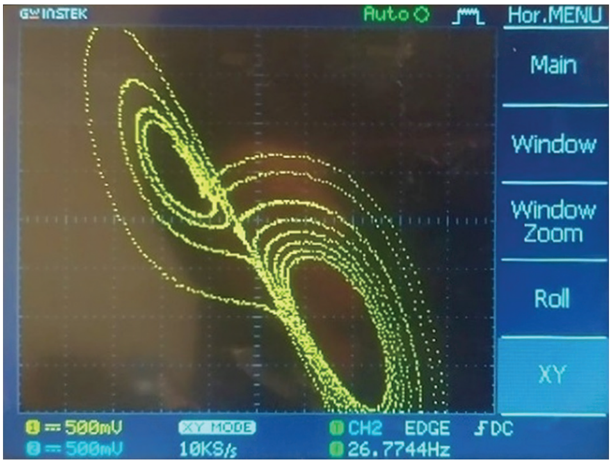


**Fig. 9.** The Lorenz Attractor XY on the oscilloscope

The throughput of the post-processing unit depends mainly on the physical process itself and the resolution of the ADC. If the process's frequency content is assumed to be at frequencies higher than the sampling frequency with high entropy, then the TRNG is bottlenecked by the board's SAR ADC maximum sampling frequency 20Mhz, in this case, the throughput will be around 12.71Mbps. If all 8 channels are used (by taking advantage of causal chains), the throughput is around 101.72Mbps. Moreover, if better precision is required, a higher-resolution ADC can be utilized. However, the sampling frequency is limited by the system's maximum clock frequency which is in the order of 100Mhz. If the interfacing circuit's frequency goes higher than that, no timing violation will occur because of the high maximum frequency of the FIFO's BRAM, however, the interfacing circuit will be severely bottlenecked.

### 5.5. INFORMATION ENTROPY ANALYSIS

The entropy of the TRNG sequence is estimated using Maurer's universal statistical test [49]. The test is performed on a sequence of length n (in this case n = 450000 bits). Practically, the compression estimator of NIST 800-90B [50] was used, which is an extension of Maurer's test to approximate the lower bound on the min-entropy. The algorithm is described in detail in [50], [51]. The min-entropy ensures a conservative measurement of the per-bit entropy of the source since it corresponds to the difficulty of predicting the most likely outcome. Finally, the min-entropy value is found to be H=5.61 per 6-bit, which corresponds to a per-bit min-entropy of 0.935.

The min-entropy calculated is lower than the ideal value of one. However, considering the high-frequency sampling of the low-frequency implementation of the analog Lorenz system, in addition to the low-resolution ADCs coupled with the low pass behavior of the op-amps used, the achieved entropy result can be considered respectable. Also, the sponge-based post-processing demonstrated its effectiveness in removing statistical defects in non-ideal conditions. However, better results can be achieved by feeding the system with high-frequency physical sources.

### 5.6. FPGA RESOURCE USAGE

The proposed design is composed of a tunable Keccak core and a pipelined AES-128 core. Individually, the cores utilize 3415 + 9973 = 13388ALMs (6024 Slices). However, the synthesized complete design consumed only 10310ALMs (4639 Slices), a 23% area reduction. This indicates an efficient logic packing of the two cores in the target device. The post-fitting netlist overall logic usage is 26 % for the pipelined AES variant and 11% for the non-pipelined variant, less than 1% of Block RAM, and 10% of the device's total pins. Table 3 summarises the FPGA post-place-and-route resource usage of the whole system.

**Table 3.** FPGA resource usage summary

| Resource | Usage | % |
|---|---|---|
| ALMs used in final placement | 10,310 / 41,910 | 25 |
| Dedicated logic registers | 7,167 / 83,820 | 9 |
| Combinational ALUT usage for logic | 12,606 | |
| -- 7 input functions | 0 | |
| -- 6 input functions | 6,907 | |
| -- 5 input functions | 1,540 | |
| -- 4 input functions | 1,966 | |
| -- <=3 input functions | 2,193 | |
| M10k blocks | 8 / 553 | 1 |
| Total block memory bits | 12,288/ 5,662,720 | <1 |
| I/O pins | 50 / 499 | 10 |

## 6. COMPARISON TO RELATED WORK

To date, the work presented in [1] is the only study that combined AES with SHA3-256 with the design being resource-shared. Most reported works on AES/Hash function designs tend to resource-share AES with Grøstl due to their common structure similar to the case of Fuge and Whirlpool hash functions. The resource-shared implementations are compact and generally more hardware efficient. For a fair comparison, the Throughput-Per-Slice (TPS) metric is calculated for the aforementioned resource-shared cryptosystem designs, and then compared with the proposed design herein in both hashing and symmetric encryption.

Most of the resource-shared works were synthesized on Xilinx Virtex-5/6 families. Meanwhile, the present implementation was performed on the IntelFPGA Cyclone-V device. Thus, an equivalent conversion between Slice and Logic Element is necessary. A single ALM is equivalent to 0.45 Virtex-5/6 slice and Cyclone-III/IV's LE is equivalent to 0.12 Virtex-5/6 slice [52], while 1 BRAM is equivalent to 128 slices [53]. Another note regarding comparing device performance, the Cyclone-V FPGA is a speed grade 6 device, meaning that it supports a maximum global clock frequency of 550Mhz which exactly matches Virtex-5's max frequency, whereas, 601Mhz is the maximum global frequency of the Virtex-6 devices. This would hint that the obtained throughput figures might be higher should Virtex-6 FPGA be used.

The comparison of the proposed design with other AES/Hash-function works is demonstrated in Table 4. The proposed design recorded the highest area utilization, a few times higher than the compact AES/Grøstl designs and 1.89× higher area utilization than the highest throughput AES/Grøstl reported in [11], and 1.35× higher than that of AES/SHA3-256 design presented in [1]. However, the recorded results are anticipated due to the relatively complex nature of both cores (pipelining of AES and flexibility of Keccak). Moreover, the works presented in the table are resource-shared. Consequently, they are supposed to yield more compact implementations. However, despite the flexibility of the presented KECCAK-CORE, it yields several times higher throughput than all Grøstl, Whirlpool, and Fuge hash functions when operating as SHA3-256. The throughput of the tunable KECCAK-CORE is close to that of the untunable core presented in [1], which is, to date, the highest hashing throughput recorded in AES/hash-function designs on FPGAs.

The proposed system achieved the second-highest TPS of 2.43, which is higher than all reported Grøstl implementations. Whereas for AES enc/dec, the highest throughput and TPS figures were recorded with 3.79× and 2.8× higher encryption throughput and TPS respectively than those reported in [1].

**Table 4.** Results comparison with other unified designs

| Study | Device | Cores | Area (LE/Slice+BRAM) | Equiv. Slices | TP (Mbps) | TPS ( Mbps/Slice ) |
|---|---|---|---|---|---|---|
| Järvinen [6] | Cyclone-III | Fuge-256 | 4520LEs | 552 | 972 | 1.76 |
| | | AES-128 enc | | | 778 | 1.41 |
| Kochar [54] | Virtex-5 | Whirlpool | 6742LUTs | 2247 | 410 | 0.18 |
| | | AES-128 enc | | | 205 | 0.09 |
| At [8] | Virtex-7 | Grøstl-256 | 185+1 | 313 | 98 | 0.31 |
| | | AES-128 enc/dec | | | 229 | 0.73 |
| Pelnar [9] | Virtex-6 | Grøstl-256 | 302+0 | 302 | 13.24 | 0.04 |
| | | AES-128 enc | | | 13.8 | 0.05 |
| | | AES-128 dec | | | 9.99 | 0.03 |
| Guo [10] | Cyclone-IV | Grøstl-256 + 4×AES-128 enc | 15135LEs | 1847 | 3877 | 2.10 |
| Järvinen [6] | Cyclone-III | Grøstl-256 | 13723LEs | 1675 | 1434 | 0.86 |
| | | 4×AES-128 enc | | | 2869 | 1.71 |
| Rogawski [55] | Cyclone-III | Grøstl-256 + 4×AES-128 enc/dec | 23758LEs | 2851 | 2378 | 0.83 |
| Rogawski [11] | Virtex-6 | Grøstl-256 + 4×AES-128 enc/dec | 2447+0 | 2447 | 4212 | 1.72 |
| Kundi [1] | Virtex-6 | SHA3-256 | 1380+16 | 3428 | 14876.13 | 4.34 |
| | | 4×AES-128 enc/dec | | | 8400.64 | 2.45 |
| This work | Cyclone-V | Keccak (tunable) | 10310ALMs | 4639 | 11264 | 2.43 |
| | | Pipelined AES-128 | | | 31846 | 6.86 |

## 7. CONCLUSION

A multi-purpose cryptographic design has been presented consisting of a 128-bit AES-CORE symmetric encryption and a tunable KECCAK-CORE that can be user-configured by modifying the capacity, bitrate, and the number of rounds. The design achieved the second-highest TPS of 2.43 compared to other works, with an area usage of roughly 26% of the low-cost Cyclone-V FPGA. In addition to the basic functions of symmetric encryption and hash function, the system was also operated as a reseedable PRNG and a post-processing unit for a chaotic-based TRNG. Both PRNG and TRNG random sequences successfully passed NIST's statistical tests. The information entropy analysis performed on the post-processed TRNG sequences indicates a respectable average of H=5.61 per 6-bit (0.935), suggesting that the sponge-based post-processing showed its effectiveness in removing statistical defects in non-ideal conditions. In future work, a more efficient design could be investigated by utilizing the sponge as a symmetric key algorithm.

**Acknowledgement**

The authors acknowledge the support of Wameedth Scientific Club for providing some of the components and IntelFPGA University program for donating the DE-10 board.

## 8. REFERENCES

[1] D.-S. Kundi, A. Khalid, A. Aziz, C. Wang, M. O'Neill, W. Liu, "Resource-Shared Crypto-Coprocessor of AES Enc/Dec with SHA-3", IEEE Transactions on Circuits and Systems I: Regular Papers, Vol. 67, No. 12, 2020, pp. 4869-4882.

[2] K. Shahzad, A. Khalid, Z. E. Rákossy, G. Paul, A. Chattopadhyay, "CoARX: A Coprocessor for ARX-based Cryptographic Algorithms", Proceedings of the 50th IEEE Design Automation Conference, Austin, TX, USA, 29 May 2013, pp. 1-10.

[3] P. Nannipieri, S. D. Matteo, L. Baldanzi, L. Crocetti, L. Zulberti, S. Saponara, L. Fanucci, "VLSI Design of Advanced-Features AES Cryptoprocessor in the Framework of the European Processor Initiative", IEEE Transactions on VLSI Systems, Vol. 30, No. 2, 2022, pp. 177-186.

[4] B. Ilyas, S. M. Raouf, S. Abdelkader, T. Camel, S. Said, H. Lei, "An Efficient and Reliable Chaos-Based IoT Security Core for UDP/IP Wireless Communication", IEEE Access, Vol. 10, 2022, pp. 49625-49656.

[5] L. Bassham et al. "A statistical test suite for random and pseudorandom number generators for cryptographic applications", https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=906762 (accessed: 2022)

[6] K. Järvinen, "Sharing Resources Between AES and the SHA-3 Second Round Candidates Fugue and Grøstl", Proceedings of the 2nd SHA-3 Candidate Conference, 23 August 2010, p. 2.

[7] M. Sonmez, R. Perlner, L. Bassham, W. Burr, D. Chang, S. jen Chang, M. Dworkin, J. Kelsey, S. Paul, R. Peralta, "Status report on the second round of

the SHA-3 cryptographic hash algorithm competition", NIST, Technical Report 7764, 2011.

[8] N. At, J. L. Beuchat, E. Okamoto, I. San, T. Yamazaki, "A low area unified hardware architecture for the AES and the cryptographic hash function Grøstl", Parallel and Distributed Computing, Vol. 106, 2017, pp. 106-120.

[9] M. Pelnar, M. Muehlberghuber, M. Hutter, "Putting together What Fits together - GrÆStl", Smart Card Research and Advanced Applications, Springer Berlin Heidelberg, 2013, pp. 173-187.

[10] K. Guo, H. M. Heys, "A Pipelined Implementation of the Grøstl Hash Algorithm and the Advanced Encryption Standard", Proceedings of the 26th IEEE Canadian Conference on Electrical and Computer Engineering, Regina, Canada, 5-8 May 2013, pp. 1-4.

[11] M. Rogawski, K. Gaj, E. Homsirikamol, "A High-Speed Unified Hardware Architecture for 128 and 256-bit Security Levels of AES and the SHA-3 Candidate Grøstl", Microprocessors and Microsystems, Vol. 37, No. 6, 2013, pp. 572-582.

[12] S. jen Chang, R. Perlner, W. Burr, M. Sonmez, J. Kelsey, S. Paul, L. Bassham, "Third-round report of the SHA-3 cryptographic hash algorithm competition", NIST, Technical Report 7896, 2012.

[13] K. E. Ahmed, M. M. Farag, "Hardware/software co-design of a dynamically configurable SHA-3 System-on-Chip (SoC)", Proceedings of the IEEE International Conference on Electronics, Circuits, and Systems, Cairo, Egypt, 6-9 Dec 2015, pp. 617-620.

[14] H. Mestiri, I. Barraj, M. Machhout, "A High-Speed KECCAK Architecture Resistant to Fault Attacks", Proceedings of the 32nd International Conference on Microelectronics, Aqaba, Jordan, 14-17 December 2020, pp. 1-4.

[15] T. Newe, M. Rao, D. Toal, G. Dooly, E. Omerdic, A. Mathur, "Efficient and High Speed FPGA Bump in the Wire Implementation for Data Integrity and Confidentiality Services in the IoT", Sensors for Everyday Life: Healthcare Settings, Springer International Publishing, 2017, pp. 259-285.

[16] R. Shahid, M. Sharif, M. Rogawski, K. Gaj, "Use of Embedded FPGA Resources in Implementations of 14 Round 2 SHA-3 Candidates", Proceedings of the International Conference on Field-Programmable Technology, New Delhi, India, 12-14 December 2011, pp. 1-9.

[17] K. Gaj, E. Homsirikamol, M. Rogawski, R. Shahid, M. U. Sharif, "Comprehensive evaluation of high-speed and medium-speed implementations of five SHA-3 finalists using Xilinx and Altera FPGAs", Cryptology ePrint Archive, Paper 2012/368, 2012.

[18] B. Jungk, M. Stottinger, M. Harter, "Shrinking KECCAK Hardware Implementations", Proceedings of the SHA-3 2014 Workshop, Univeristy of California, Sanata Barbara, CA, USA, 22 Aug 2014.

[19] A. Soltani, S. Sharifian, "An Ultra-high Throughput and Fully Pipelined Implementation of AES Algorithm on FPGA", Microprocessors and Microsystems, Vol. 39, No. 7, 2015, pp. 480-493.

[20] X. Zhang, M. Li, J. Hu, "Optimization and Implementation of AES Algorithm based on FPGA", Proceedings of the IEEE 4th International Conference on Computer and Communications, Chengdu, China, 7-10 December 2018, pp. 2704-2709.

[21] R. Farashahi, B. Rashidi, S. Sayedi, "FPGA based Fast and High Throughput 2-Slow Retiming 128-bit AES encryption algorithm", Microelectronics Journal, Vol. 45, No.8, 2014, pp. 1014-1025.

[22] T. Tuncer, E. Avaroğlu, "Random Number Generation with LFSR based Stream Cipher Algorithms", Proceedings of the 40th International Convention on Information and Communication Technology, Electronics and Microelectronics, Opatija, Croatia, 22-26 May 2017, pp. 171-175.

[23] H. Tang, T. Qin, Z. Hui, P. Cheng, W. Bai, "Design and implementation of a configurable and aperiodic pseudo random number generator in FPGA", Proceedings of the IEEE 2nd International Conference on Circuits, System and Simulation, Guangzhou, China, 14-16 July 2018, pp. 47-51.

[24] R. Hobincu, O. Datcu, "FPGA Implementation of a Chaos based PRNG Targeting Secret Communication", Proceedings of the International Symposium on Electronics and Telecoms, Timisoara, Romania, 8-9 November 2018, pp. 1-4.

[25] S. Gomar, M. Ahmadi, "A digital pseudo random number generator based on a chaotic dynamic

system", Proceedings of the 26th International Conference on Electronics, Circuits and Systems, Genoa, Italy, 27-29 November 2019, pp. 610-613.

[26] B. Paul, G. Trivedi, P. Jan, Z. Němec, "Efficient PRNG design and implementation for various high throughput cryptographic and low power security applications", Proceedings of the 29th International Conference Radioelektronika, Pardubice, Czech Republic, 16-18 April 2019, pp. 1-6.

[27] S. Łoza, Ł. Matuszewski, "A true random number generator using ring oscillators and SHA-256 as post-processing", Proceedings of the International Conference on Signals and Electronic Systems, Poznan, Poland, 11-13 September 2014, pp. 1-4.

[28] NIST, Security requirements for cryptographic modules, https://doi.org/10.6028/NIST.FIPS.140-2 (accessed: 2022)

[29] G. Bertoni, J. Daemen, M. Peeters, G. V. Assche, "The KECCAK SHA-3 Submission", https://keccak.team/files/Keccak-submission-3.pdf (accessed: 2022)

[30] G. Bertoni, J. Daemen, M. Peeters, G. V. Assche, "Cryptographic sponge functions", https://keccak.team/files/CSF-0.1.pdf (accessed: 2022)

[31] M. Dworkin, "SHA-3 standard: Permutation-based hash and extendable-output functions", https://doi.org/10.6028/NIST.FIPS.202 (accessed: 2022)

[32] G. Bertoni, J. Daemen, M. Peeters, G. V. Assche, "Sponge-based Pseudo-Random Number Generators", Cryptographic Hardware and Embedded Systems, Springer Berlin Heidelberg, 2010. , pp. 33-47

[33] H. Shimakage, Y. Tamura, "Chaotic oscillations in Josephson Junctions for Random Number Generation", IEEE Transactions on Applied Superconductivity, Vol. 25, No. 3, 2015, pp. 1-4.

[34] J. B. Johnson, "Thermal Agitation of Electricity in Conductors", Physical Review, Vol. 32, No. 1, 1928, pp. 97-109.

[35] M. Stipčević, "Fast Nondeterministic Random Bit Generator based on Weakly Correlated Physical Events", Review of Scientific Instruments, Vol. 75, No. 11, 2004, pp. 4442-4449.

[36] S. H. Strogatz, "Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry and Engineering", 2nd Edition, CRC Press, 2015.

[37] P. Gaspard, "Chaos, Scattering and Statistical Mechanics", Cambridge University Press, 2005.

[38] E. Barker, J. Kelsey, Recommendation for random number generation using deterministic random bit generators, https://doi.org/10.6028/NIST.SP.800-90Ar1 (accessed : 2022)

[39] A. Maache, A. Touati, A. Ouali, "Implementation of an AES-based Real-time Video Encryption/Decryption using FPGA/HPS", Proceedings of the 19th International Multi-Conference on Systems, Signals & Devices, Setif, Algeria, 6-10 May 2022, pp. 702-706.

[40] G. Bertoni, J. Daemen, M. Peeters, G. Van Assche, R. Van Keer,B. Viguier, "Kangarootwelve: Fast hashing based on keccak-p", Applied Cryptography and Network Security, Springer International Publishing, 2018, pp. 400-418.

[41] G. Bertoni, J. Daemen, M. Peeters, G. Van Assche, "Duplexing the sponge: Single-pass authenticated encryption and other applications", Selected Areas in Cryptography, Springer International Publishing, 2012, pp. 320-337.

[42] Horowitz Group, Build a Lorenz attractor, http://seti.harvard.edu/unusual_stuff/misc/lorenz.htm (accessed: 2022)

[43] W. Schindler, W. Killmann, "Evaluation Criteria for True (Physical) Random Number Generators used in cryptographic applications", Cryptographic Hardware and Embedded Systems - CHES 2002, Springer Berlin Heidelberg, 2003, pp. 431-449.

[44] G. J. Wallinger, Information theory and chaotic systems, https://www.spsc.tugraz.at/sites/default/files/InformationTheory_ChaoticSystems_final.pdf (accessed: 2022)

[45] C. Liu, L. Ding, Q. Ding, "Research About the Characteristics of Chaotic Systems based on Multiscale Entropy", Entropy, Vol. 21, No. 7, 2019, p. 663.

[46] M. Stipčević, Ç. K. Koç, "True Random Number Generators", Open Problems in Mathematics and Computational Science, pp. 275-315, Springer, 2014.

[47] C. Boura, A. Canteaut, C. D. Canniere, "Higher-order differential properties of Keccak and Luffa", Fast Software Encryption, Springer Berlin Heidelberg, 2011, pp. 252-269.

[48] G. Bertoni, J. Daemen, M. Peeters, G. V. Assche, “Note on zero-sum distinguishers of Keccak-f”, https://keccak.team/files/NoteOnKeccakParametersAndUsage.pdf (accessed: 2022)

[49] U. M. Maurer, “A Universal Statistical Test for Random Bit Generators”, Journal of Cryptology, Vol. 5, No. 2, 1992, pp. 89-105.

[50] M. S. Turan, E. Barker, J. Kelsey, K. A. McKay, M. L. Baish, M. Boyle, “Recommendation for the entropy sources used for random bit generation”, NIST Special Publication, Vol. 800, No. 90B, 2018, p. 102.

[51] Y. Kim, C. Guyot, Y.-S. Kim, “On the Efficient Estimation of Min-Entropy”, IEEE Transactions on Information Forensics and Security, Vol. 16, 2021, pp. 3013-3025.

[52] Intel Corporation, “Stratix III FPGAs vs. Xilinx Virtex-5 devices: Architecture and performance comparison”, White paper WP01007, 2007.

[53] P. Bulens, F. X. Standaert, J. J. Quisquater, P. Pellegrin, G. Rouvroy, “Implementation of the AES-128 on Virtex-5 FPGAs”, Progress in Cryptology, Springer, 2008, pp. 16-26.

[54] T. Kochar, S. Nandi, S. Biswas, “A single chip implementation of AES cipher and Whirlpool hash function”, Proceedings of the Annual IEEE India Conference, India, 18-20 December 2009, pp. 1-4.

[55] M. Rogawski, K. Gaj, “A high-speed unified hardware architecture for AES and the SHA-3 candidate Grøstl”, Proceedings of the 15th Euromicro Conference on Digital System Design, Cesme, Turkey, 5-8 September 2012, pp. 568-575.

# Tuna Swarm Optimization with 3D-chaotic map and DNA encoding for image encryption with lossless image compression based on FPGA

Original Scientific Paper

**Sunil B. Hebbale**
Faculty, KLECET Chikodi, Dt. Belagavi,
Karnataka 591201, India.
sunilhebbale123@gmail.com

**V. S. Giridhar Akula**
KORM College of Engineering,
Thadigotla, Kadapa, Andhra Pradesh, India
giridharakula456@gmail.com

**Parashuram Baraki**
Department of CS&E Smt. Kamala and
Sri.Venkappa M. Agadi College of Engineering and Technology,
Lakshmeshwar, Dist. Gadag, Karnataka, India
parashurambaraki456@gmail.com

***Abstract*** *– Images and video-based multimedia data are growing rapidly due to communication network technology. During image compression and transmission, images are inevitably corrupted by noise due to the influence of the environment, transmission channels, and other factors, resulting in the damage and degradation of digital images. Numerous real-time applications, such as digital photography, traffic monitoring, obstacle detection, surveillance applications, automated character recognition, etc are affected by this information loss. Therefore, the efficient and safe transmission of data has become a vital study area. In this research, an image compression–encryption system is proposed to achieve security with low bandwidth and image de-noising issues during image transmission. The Chevrolet transformation is proposed to improve image compression quality, reduce storage space, and enhance de-noising. A 3D chaotic logistic map with DNA encoding and Tuna Swarm Optimization is employed for innovative image encryption. This optimization approach may significantly increase the image's encryption speed and transmission security. The proposed system is built using the Xilinx system generator tool on a field-programmable gate array (FPGA). Experimental analysis and experimental findings show the reliability and scalability of the image compression and encryption technique designed. For different images, the security analysis is performed using several metrics and attains 32.33 dB PSNR, 0.98 SSIM, and 7.99721 information entropy. According to the simulation results, the implemented work is more secure and reduces image redundancy more than existing methods.*

***Keywords****: Image encryption, chaotic maps, decryption algorithm, optimization algorithm, DNA encoding, Logical operations, image compression*

## 1. INTRODUCTION

The Internet has evolved into an essential sort of technology in the modern day. It applies to various fields. Multimedia is a prevalent form of communication in modern society. Numerous real-world applications, such as medical imaging systems, radar transmission, military systems, etc., may use this technology. Multimedia storage and transmission are still crucial. The transmission of data over the Internet must also be secure, which is a key factor [1-3]. Currently, security technologies such as data encryption, digital signature, and trusted routing are utilized to transmit data securely. Images are a popular kind of media in all sectors. Consequently, encrypted communication is vital as well as of great interest. The three main areas of study are compression of images, encryption, and image processing. De-noising, encryption, and image compression are all subject areas of this article, which will deal with image compression and de-noising [4-6].

The primary goal of compressing images is to reduce both storage space and network traffic. The source encoder transforms the input image into a minimal representation. They are built using several transforms, including DFT, DCT, and DWT. The DWT is a computationally efficient method, and the JPEG-2000 standard has embraced it [7, 8]. Even though the DWT can reach efficient compression ratios, it has significant disadvantages, such as a lack of directionality, aliasing, shift variance, and oscillations. The complicated WT may overcome these restrictions but at a high computational cost. In addition, the FWPT may be applied to signals with large fractional frequency components, although it has a greater computing cost than the DWT [9, 10].

In addition to compression, images should be encrypted before transferring through insecure lines of communication. Since the encryption process destroys the link between pixels, it is often performed after compression [11, 12]. There are several accessible encryption protocols, and there is always a trade-off between computational cost and encryption strength. The encryption of images based on a chaotic system was extensively explored, along with its intrinsic properties revealing the intimate link between cryptography and chaotic systems and cryptography.

There are two types of cryptosystems based on chaotic maps: Two-dimensional (2D) and one-dimensional (1D) chaotic maps. 1D chaotic map-based encryption approaches are insufficiently secure due to various constraints. Their basic orbits and starting and control parameters were predictable [13-15]. Therefore, traditional 1D chaotic systems are inappropriate for encrypting images. Recently, a complex chaotic system based on Chebyshev and 1D Sine maps and a Sine Cosine composite function system is described. 2D chaotic maps feature more complicated architecture and better chaos than 1D maps, making them suited for data encryption. 2D chaotic maps include the 2D Logistic-Modulated-Sine Coupling-Logistic Chaotic Map, two-dimensional Logistic-Adjusted-Sine map, and the 2D sine chaotification system. The implementation of these maps was feasible [16, 17]. These suggested 2D maps perform better and have broader chaotic ranges than the traditional existing mapping. Additionally, greater chaotic behaviors are seen in hyper-chaotic systems. Furthermore, a random and dynamic combination of the hybrid hyper-chaotic map is suggested to attain excellent encryption performance [18-20].

The existing methods that were presented simply encrypt the plain image. The most efficient and secure approach for transmitting data is compression followed by encryption with noise reduction. To provide noiseless, secure, and rapid image transmission, the proposed model conducts image de-noising, compression, and security. To enhance image quality, minimize storage space, and achieve superior de-noising, tetrolet transformation-based image compression is recommended in this research. Then, encryption and decryption of grayscale images using 3D chaotic maps and the DNA encoding technique are proposed. In addition, Tuna Swarm Optimization is presented, which may significantly increase the image's encryption speed and transmission security.

In digital signal processing systems, FPGA has become the primary paradigm for high-performance system implementation, particularly in image processing applications. In addition, FPGA can build high-performance signal processing system designs at fast speeds. In other words, the Xilinx system generator (XSG) increases the capability of FPGA and offers efficient tools for designing an image encryption model in a manner that is compatible with MATLAB/Simlinkin.

The practical design of the proposed image encryption and compression algorithm is the main goal of this work. As a result, a high level of security must be guaranteed. Every system or piece of work already in existence in this subject has to do with software or DSP. Therefore, the particular system's portability is not very simple. Here, the proposed approach is applied to the huge VLSI domain. We are concentrating on the suggested work's front-end design and putting it into FPGA. This will increase speed and effectiveness while decreasing area, energy, latency, and expense. Therefore, an encryption-based chaotic system is built in this research using FPGA. The following is a summary of the important contributions to this paper:

- De-noising and lossless compression of image is implemented by Chevrolet transformation.
- Proposed 3D chaotic logistic map for the diffusion of pixels during encryption.
- The initial condition for the 3D employs an ASCII key with 64 bits. Bits undergo "bit-XOR" to obtain a decimal value.
- A cryptographic hash function SHA-512 is utilized to produce a hash value.
- The proposed Tuna Swarm Optimization can greatly improve encryption speed and secure image transmission.

The remaining sections are structured as follows: The relevant work reported by various researchers is summarized in section 2. The suggested image compression and encryption technique are detailed in Section 3. Section 4 describes the experimental implementation of both the chaotic system and the proposed encryption architecture using Xilinx. In section 5, the conclusion and future work are explored.

## 2. LITERATURE SURVEY

There are several image compression and encryption technique exists. However, achieving both strong security and compression at the same time is still a difficult challenge since both are conflicting. In this part, the idea of image encryption using chaotic maps, as well as image denoising and compression, is discussed.

Bhargava and Gangadharan [21] developed an HRRBF-based FABF on an FPGA utilizing RLG technology with integrated geometric and photometric weight computation and kernel result calculation techniques. RPGs were used to build the reversibly modified carry select adder, the multiplier, and the reversible adder subtractor which enhances speed and lowers delay, power, area, and execution time. TCAM was used as the main memory to speed up processing.

Maazouz, Toubal, Bengherbia, Houhou, and Batel [22] constructed a new discrete chaotic system using the hybrid classification approach; the state variables of this system were then utilized to build a new S-box substitution matrix. The significance of this matrix in establishing the algorithm's degree of security cannot be overstated. The advantages led further to the improvement of the cryptographic properties of the developed algorithm, which included the Feistel model and some AES components.

Kumar, Reddy, Rinaldi, Parameshachari, and Arunachalam [23] provided low-power, high-speed hardware solutions for the FPGA implementation of AES to protect data. This work does not use LUTs to implement AES SubBytes and InvSubBytes transformations. This design uses combinational logic circuits to convert SubBytes and InvSubBytes. This architecture removes unwanted delays and adds sub-pipelining to improve AES algorithm speed by not using LUTs. The modified positive polarity reed muller (MPPRM) design reduces total hardware needs. With MPPRM architecture in SubBytes phases, a mixed column architecture is implemented.

Zodpe and Sapkal [24] suggested a method for improving the encryption quality of the AES algorithm, as well as its implementation on FPGA. Initially, the modified AES algorithm's S-box values are created using the PN Sequence Generator. Second, the first encryption/decryption key is derived from the output of the PN Sequence Generator.

Hafsa, Gafsi, Malek, and Machhou [25] suggested combining a complicated chaotic PRNG with MAES. For a high-quality encryption key, a chaotic PRNG was recommended. Unpredictability, entropy, and complexity characterize the created key. Using four S-boxes enhanced the complexity of the MAES sub-bytes operation. Mix-columns and shift-rows transformations were removed to enhance complexity. Only four encryption rounds are done in a loop, saving time. The encryption data route executes a 32-byte block in parallel and one clock cycle.

Kumar, Balmuri, Marchewka, Bidare Divakarachari, and Konda [26] improved AES's key expansion to speed up subkey generation. The fork-join model of key expansion (FJMKE) structure was created to increase the speed of subkey creation while minimizing the hardware requirements of AES by eliminating the frequent calculation of secret keys. JFK-AES produces subkeys in half the time of the usual design.

Bhargava and Gangadharan [27] developed FPGA-based MRBF-based FBF architecture to reduce computing complexity, space, and power. MRBF was implemented using a quantum-dot cellular automaton-based carry select adder, which reduces space, power consumption, and latency. In addition, image de-noising utilizing MRBF-FBF was tested.

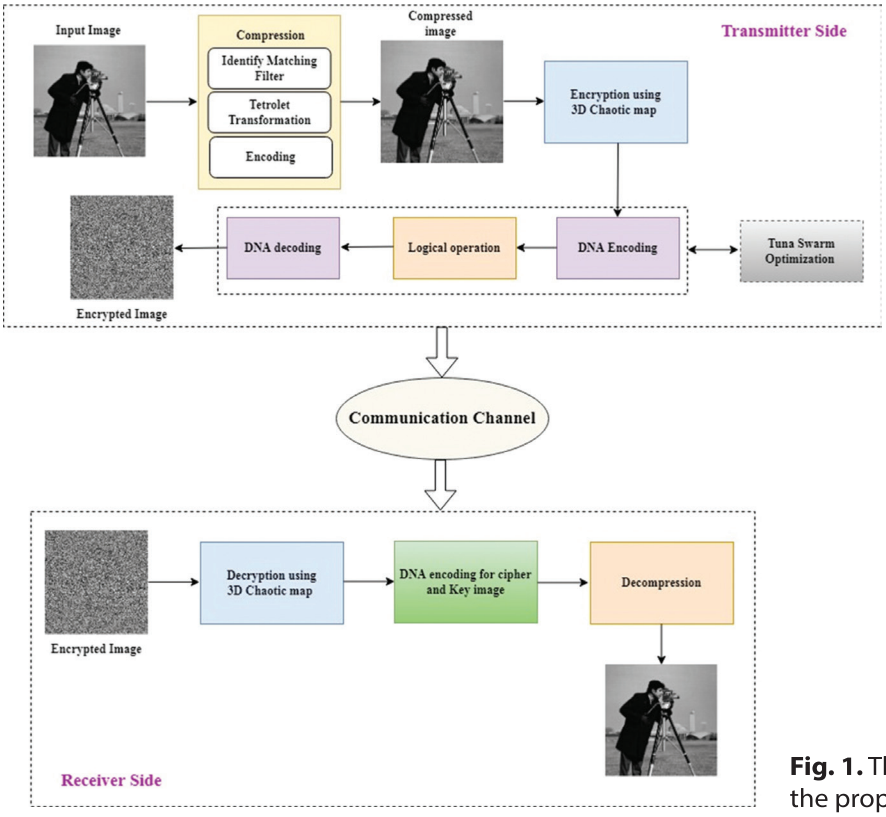


**Fig. 1.** The architecture of the proposed methodology

## 3. METHODOLOGY

The quality of digital images is diminished by noise interference. As image quality degrades, image diagnosis becomes more challenging. The transmission of massive data sets is the greatest hurdle in digital image transmission. The compression approach decreases the image's file size by decreasing the number of pixels, making it more suitable to store and deliver information across a variety of communication channels. Additionally, reducing network traffic will increase efficiency. The Chevrolet transformation is employed to remove noise and redundancy from digital images. This compressed image has been encrypted using chaotic maps. Fig. 1 depicts the block diagram for the proposed framework.

The primary components of the proposed work are the input image, image compression, compressed image, encryption module, encrypted image, inverse process, and reconstruction of the image. Haar model becomes Chevrolet by matching and rearranging tiles. The image is partitioned into 4x4 blocks. Each block's tetromino division reflects its image geometry. By combining four similar squares that are linked at one edge.

### 3.1. MATCHING TILE IDENTIFICATION

The Chevrolet transform is based on the Chevrolet decomposition method. The 4x4 input matrix of the image undergoes a standard Haar transformation.

The sum of the twelve detail coefficients is denoted as

$$CD_{sum} = \left|s(diagonal)\right| + \left|s(horizontal)\right| + \left|s(vertical)\right| \quad (1)$$

Where $CD_{sum}$ represents the present information. The subsequent tile is used to determine the new tiles. Using the Haar transform, new information is acquired and stored in NDsum. After reviewing every tile, the procedure concludes. After that, tiles that match are chosen.

### 3.2. TETROLET TRANSFORM

For Chevrolet coefficients, the input image is divided into 4x4 pixels. Level-1 Haar approximation contains matched tetromino tiles of sub-images. Diagonal, horizontal, and vertical image transformation coefficients are approximations. Repeat the process to generate the images using the Tetrolet transform [28] is expressed as

$$I =\subset Z_2 \quad (2)$$

$$a = (a[x,y])(x,y) \in I \quad (3)$$

Where $N$=2$J$, $J \in N$

$$N_4(x,y) := 1,y),(i+1,y),(x,y-1),(x,y+1) \quad (4)$$

An edge index has three neighbors, but a vertex index has just two.

$$J : I \rightarrow with\ J((x,y)) :=\ jN + x \qquad A\ set\ E = r \in N \quad (5)$$

$$\forall(x,y) \in I_v \exists(x,y) \in I_v : (x,y) \in N_4(x,y) \quad (6)$$

Here, $I_v$= let $L$: $I_v$ →be the condition $I_v$ = 4 for every set.

#### 3.2.1 Tilings by tetrominoes

The tetrominoes are formed using the four-unit squares. This is joined around its edges, not simply at its corners. There are five distinct forms independent of rotations and reflections; these are known as free tetrominoes. The conventional 2D-Haar wavelet decomposition yields a unique tetromino partition.

Haar Wavelet model handles sample or native sample average changes in the input image matrix. Except for $k$=$l$=$0$, Haar transform coefficients reflect rows and columns of native pixel averages. This square measurement resulted in multiple "edge extractions" The second Haar transformation processes images as rows and columns. The transition involves adding and subtracting level one's parts. For stage one, the transformation can just be a straightforward component addition. However, the fundamental elements of the transformation matrix are present at levels greater than one. These parts result in decimal numbers (pixel values are integers), and even small decimal numbers can have a significant impact on higher-level value changes. To circumvent this limitation, the higher-level Approximation matrix is used.

$$k=2a+b-1 \quad (7)$$

Based on the aforementioned formula, $a$ and $b$ are defined clearly for $k$. Hence, the approximation matrix for level 3 ($N$=8) comprises 15 coefficients.

$$Haar\ transform = \begin{bmatrix} \begin{bmatrix} A & V \\ H & D \\ A & V \\ H & D \end{bmatrix} & \begin{bmatrix} A & V \\ H & D \\ A & V \\ H & D \end{bmatrix} \end{bmatrix} \quad (8)$$

Therefore, the grayscale image is compressed and noise free. These images are then encrypted further using 3D chaotic maps encoded with DNA.

### 3.3. IMAGE ENCRYPTION

The proposed encryption algorithms are detailed in this section. This method consists of the following encryption process:

- 3D chaotic map with a generation of symmetric keys
- Permutation
- Diffusion
- Key image generation using 3D and pixel confusion

#### 3.3.1. 3D chaotic map with symmetric keys generation

The proposed method initializes the p, q, and r dimensions of a 3D-chaotic map using three symmetric keys. Symmetric keys are identical for both encryption and decryption. This method requires the sender and the recipient to have the same secret key. Consequently, in the proposed work, the receiver must hold identical keys to decode the encrypted image.

Three separate ways for creating symmetric map initialization keys, each with its approach, are discussed. Each method initializes the $p$, $q$, and $r$ dimensions of a 3D chaotic logistic map with a random decimal integer between 0 and 1.

#### 3.3.1.1 SHA-512

In the proposed methodology, SHA-512 [29] is utilized and the SHA-512 hash function of the compressed image generates the initial values and intermittent parameters of the chaotic system. The hash sequence of the plain image with 512 bits is obtained when the compressed image is given: $K= [k_1, k_2, \ldots, k_{64}]$.

$$Checksum = 2 + \frac{mod((h1+h2+h3+h4)*10^{14},256)}{255} \tag{9}$$

Where h1, h2, h3, and h4 are 8-bit sequences generating SHA-512 64 bits overall. Modulo functions as 'mod'. Then, the chaotic system is generated by the initial values.

#### 3.3.1.2 Chebyshev key

This key initializes the 3D chaotic map's x dimension. Chebyshev's polynomial chaos map [30] is a chaos map. Orthogonal Chebyshev polynomials are used. This map's chaos is utilized for encryption. n-degree polynomials are defined by the equation (10):

$$k_{n+1} = \cos(p^{*} \arccos(k_n)) \tag{10}$$

Applying the Chebyshev polynomial formula provided in Equation 10 using the aforementioned parameters $k_1$ and $p$ for $n$ times. This map then produces a 1 × 65,356 -dimensional vector with chaotic negative and positive decimal values. The vectors are then transformed from decimal to binary values and shifted by one unit circularly. The matrix is transformed back to its original decimal form. The logical operation bit XOR is performed between the matrices. All the acquired values are put together, and the sum is transformed into a 64-bit binary. All the acquired values are put together, and the sum is transformed into a 64-bit binary. The acquired 64-bit binary is subdivided into eight sectors, and every eight bits is transformed into a decimal number. Applying the SHA-512 hash algorithm and utilizing Equation [10] generates a checksum result. The checksum value generated by the Chebyshev key generation process is utilized to initialize the '$p$' parameter of the 3D chaotic map.

#### 3.3.1.3 Prime key

The primary key [31] initializes the second dimension '$y$' of the 3D chaotic mapping. Enter ASCII 64-bit characters. The characters entered are transformed into a 128-bit binary value. Only 64 bits are extracted and split into eight halves. Every bit is transformed into a decimal number. Applying the SHA-512 hash algorithm and utilizing Equation [9] generates a checksum result. The ASCII key generation method's checksum value is used to initialize the '$z$' parameter of a 3D chaotic mapping. After initializing the values, the map is iterated $T_2$ times to produce chaotic values. Equation 11 normalizes $p$, $q$, and $r$ vectors.

$$\begin{cases} x = \left\lfloor \mathrm{mod}\left((x^{*}T_2), image\,height\right) \right\rfloor \\ y = \left\lfloor \mathrm{mod}\left((y^{*}T_2), image\,height\right) \right\rfloor \\ z = \left| \mathrm{mod}\left((z^{*}T_2), image\,height\right) \right| \end{cases} \tag{11}$$

Where Image height = 256 for image size 512 x 512 and $T_2$ is a large number

#### 3.3.2 Permutation of pixels

Pixel position permutation is a technique for unusually repositioning an image's pixels. This procedure may be carried out either arbitrarily or by employing a permutation key. The proposed approach employs the one and two dimension sequences of Level 1's 3D chaotic map as permutation keys for the permutation of columns and rows of the compressed image respectively.

#### 3.3.2.1 Row permutation

To explain the permutation process, an example with eight-by-eight matrices is shown. The permutation of a row has four stages:

Step 1: Load the original image A and obtain its width and height, denoted by W and H, respectively.

Step 2: Iterate H times the x sequence present in equation (11) to get the permutation key of rows.

Step 3: Sort x in ascending order depending on its index to get the row permutation vector PROW = ($Pr_1$, $Pr_2$,..., $Pr_H$).

Step-4: Generate the permuted image for the row $I$ utilizing the permutation vector $P_{ROW}$ as specified in equation (7):

$$P(i, j + k(i)) = I(i, j) \tag{12}$$

Here, the permutated matrix row-wise is $P$, the input matrix is $I$, and the rows and columns index of the input matrix is $(i, j)$.

#### 3.3.2.2 Permutation of column

In column permutation, the same steps are taken as in row permutation:

Step 1: To construct the column permutation key, repeat the y sequence defined in equation (12) W times.

Step 2: Obtain the column permutation vector $P_{Column}$ = ($Pc_1$, $Pc_2$... $Pc_W$) by sorting the series depending on its index in ascending order.

Step 3: Using the permutation vector $P_{Column}$ and the equation below, obtain the permuted image for column $C$:

$$C(i + l(j), j) = P(i, j) \tag{13}$$

### 3.3.3 DNA encoding and decoding

DNA is a high molecular weight, double-stranded polymer with a double helix shape. Including adenine (A), thymine (T), guanine (G), and cytosine (C) as nitrogen-containing bases (C). A and T, as well as G and C, have complementary relationships.

8-bit binary represents each grayscale pixel. 00 and 11 are binary complements, as are 01 and 10. Each pixel in a grayscale image may be turned into a four-bit DNA sequence. Table 1 shows how a DNA nucleotide may be encoded using the complementary pairing concept.

**Table 1.** Encoding and decoding rules of DNA

| Rule | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| **00** | A | A | T | T | C | C | G | G |
| **01** | G | C | C | G | A | T | A | T |
| **10** | C | G | G | C | T | A | T | A |
| **11** | T | T | A | A | G | G | C | C |

### 3.3.4 XOR logical operation

DNA-encoded images and key images are XORed to encrypt and randomize the pixels. The key image is generated from a 3D chaotic map's 'z' dimension. The dimension 'z's-generated chaotic sequence is an [M × N] matrix. Where the input image is represented as M and N. Third step of the proposed encryption procedure encodes the key image with DNA using eight complementary principles. The DNA-encoded key image and the Level 3 image are logically XORed.

### 3.3.5 DNA diffusion and Tuna Swarm Optimization for optimized encrypted image

TSO algorithm [32] is proposed to obtain the channel's optimum DNA-encoded key image. Tuna is a marine carnivorous fish. Sizes of tuna species vary significantly. Tuna is an ocean-top predator that eats midwater and surface fish. They have an efficient and unique swimming technique (called fishtail form) that the body stays rigid while the tail swings swiftly. Despite swimming fast, the single tuna is slower than the agile fish. Tuna employ "group travel" to hunt. Intelligence helps them find and attack the target. These species have evolved smart foraging approaches.

The primary strategy is spiral foraging. When feeding, tuna swim in a spiral pattern to drive their prey into shallower water, where they may be successfully attacked. Next, is referred to as parabolic foraging. Each tuna swims in succession, creating a parabolic curve that encircles its target.

#### 3.3.5.1 Initialization

TSO begins the optimization process by randomly creating starting populations in the search space,

$$X_i^{\text{int}} = rand.(ub - 1b) + 1b, \quad i = 1,2,\ldots,NP, \tag{14}$$

Where, lower and upper boundaries of search space as 1b and ub, with the initial individual as $X_i^{\text{int}}$, rand is a random vector from 0 to 1 with uniform distribution and tuna populations represented as NP.

#### 3.3.5.2 Spiral Foraging

When predators attack herring, sardines, and other small schooling fish, the entire group creates a dense, continually shifting structure, making it difficult to target a single individual. The tuna swarm creates a spiral to hunt. Most fish in the group lack directional sense, but when a few fish swims together in a given direction, the nearby fish will adjust their route sequentially until they form a large group with the same goal and begin hunting. Along with circling after food, tuna groups communicate. Each tuna follows the previous one, so they may exchange information. Based on these assumptions, the spiral foraging formula is:

$$X_i^{t+1} = \begin{cases} \alpha_1.\left(X_{best}^t + \beta.\left|X_{best}^t - X_i^t\right|\right) + \alpha_2.X_i^t, & i = 1, \\ \alpha_1.\left(X_{best}^t + \beta.\left|X_{best}^t - X_i^t\right|\right) + \alpha_2.X_{i-1}^t, & i = 2,3,\ldots,NP, \end{cases} \tag{15}$$

$$\alpha_1 = a + (1 - a).\frac{t}{t_{\max}}, \tag{16}$$

$$\alpha_2 = (1 - a) - (1 - a).\frac{t}{t_{\max}}, \tag{17}$$

$$\beta = e^{bl}.\cos(2\pi b), \tag{18}$$

$$l = e^{3\cos(((t_{\max} + 1/t) - 1)\pi)}, \tag{19}$$

where b is a 0-1 random number, Maximum iterations $t_{max}$, iteration number is t, a determines how closely tuna follow the ideal individual in the first phase, ), $\alpha_1$ and $\alpha_2$ regulate the likelihood of individuals to migrate to the optimum and preceding individuals, current best (food) as $X_{best}^t$, $X_i^{t+1}$ is the i^th individual in the iteration $t + 1$.

Tuna may use the search area around food as they forage in a spiral. Following the best forager is not as effective as group foraging when the best fails to find food. To start a spiral search, we generate a random search coordinate. Each member may explore a wider region, and TSO can explore globally. The particular mathematical model is presented here:

$$X_i^{t+1} = \begin{cases} \alpha_1.\left(X_{rand}^t + \beta.\left|X_{rand}^t - X_i^t\right|\right) + \alpha_2.X_i^t, & i = 1, \\ \alpha_1.\left(X_{rand}^t + \beta.\left|X_{rand}^t - X_i^t\right|\right) + \alpha_2.X_{i-1}^t, & i = 2,3,\ldots,NP, \end{cases} \tag{20}$$

where $X_{rand}^t$ is a random search space reference.

Metaheuristic algorithms initially explore globally before focusing on local exploitation. As the number of iterations increases, TSO adjusts spiral foraging reference locations from random to optimal. It is given in equation (21),

$$X_i^{t+1} = \begin{cases} \alpha_1 . \left( X_{rand}^t + \beta . \left| X_{rand}^t - X_i^t \right| \right) + \alpha_2 . X_i^t, & \mathrm{i} = 1 \\ \alpha_1 . \left( X_{rand}^t + \beta . \left| X_{rand}^t - X_i^t \right| \right) + \alpha_2 . X_{i-1}^t, & i = 2,3,\ldots, NP, \end{cases} \quad \text{if } rand < \frac{t}{t_{\max}} \\ \begin{cases} \alpha_1 . \left( X_{rand}^t + \beta . \left| X_{rand}^t - X_i^t \right| \right) + \alpha_2 . X_i^t, & i = 1 \\ \alpha_1 . \left( X_{rand}^t + \beta . \left| X_{rand}^t - X_i^t \right| \right) + \alpha_2 . X_{i-1}^t, & i = 2,3,\ldots, NP, \end{cases} \quad \text{if } rand \geq \frac{t}{t_{\max}} \tag{21}$$

### 3.3.5.3 Parabolic Foraging

Additionally, eating in a spiral pattern, tunas can feed cooperatively in a parabolic shape. Concerning food, tuna creates a parabolic pattern. Moreover, tuna look for food by scanning their surroundings. Assuming that the chance of selection is 50% for each of these methods, they are executed concurrently. It is mathematically expressed as:

$$X_i^{t+1} = \begin{cases} X_{best}^t + rand . \left( X_{best}^t - X_i^t \right) + TF . p^2 . \left( X_{best}^t - X_i^t \right), & \text{if } rand < 0.5, \\ TF . p^2 . X_i^t, & \text{if } rand \geq 0.5, \end{cases} \tag{22}$$

$$P = \left( 1 - \frac{t}{t_{\max}} \right)^{(t / t_{\max})}, \tag{23}$$

Where TF is a random number with a value of 1 or – 1.

### 3.3.5.4 Optimization through TSO

TSO is used in this stage to acquire the optimum mask sequence. At the beginning of the TSO optimization procedure, the search space population is generated at random. In each cycle, each individual chooses randomly between two foraging methods or regenerates the search space with probability *r*. In parameter setup simulation experiments, *r* will be examined. All TSO individuals are updated and computed till the completion of the optimization procedure, then the optimal tuna and its fitness value are obtained. Ultimately, the vector with the highest fitness value creates the optimized masks.

### 3.3.6 Final Encryption

The application of logical operations yields a new DNA-encoded matrix that is subsequently DNA decoded to decimal values. Using the same DNA complementary principles, the pixels are turned into binary bits, which may then be translated into decimal integers. After applying four stages of encryption, the resulting image is an optimal cipher image that is broadcast across the channel to the destination.

### 3.3.7 Decryption

To recover the image at the receiving end, reverse procedures are done. As symmetric keys are used in the proposed approach, identical keys are used at both the receiver and transmitting ends.

### 3.4 FPGA Implementation

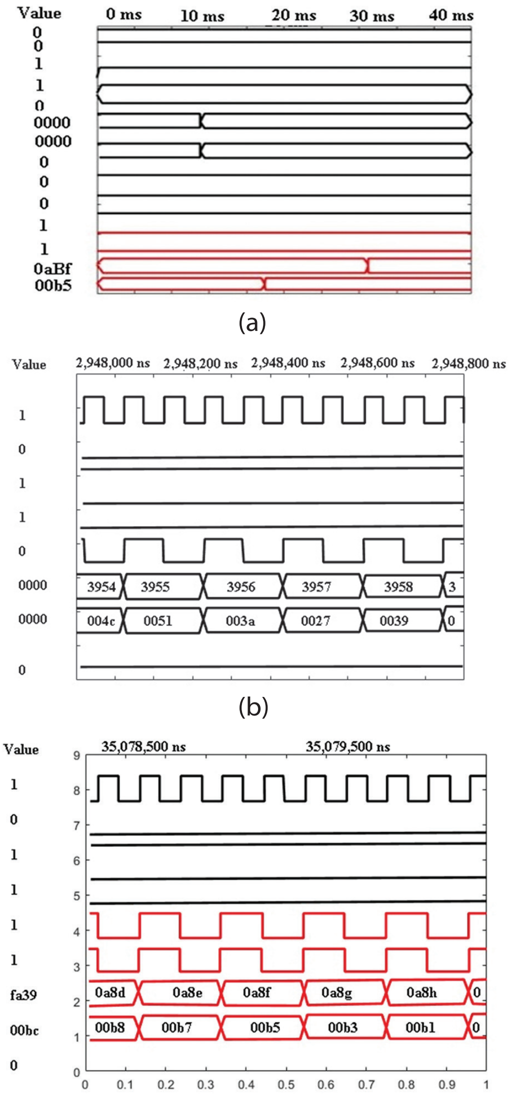
The complete design was encoded using the Verilog programming language. This design used 30820 slices and 61088 LUTs. Over 50 to 80 clock cycles of delay are required for the initial round. After that, we get the output at each clock cycle. The proposed method achieves a minimum clock period of 4.246 nanoseconds and a high clock frequency of 256.022 MHz, resulting in an efficiency of 14.15 Mbps/slice.

## 4. SIMULATION RESULTS AND DISCUSSIONS

In software, the proposed compression and encryption technique was implemented, as well as security and compression performance evaluations. MATLAB is used for software implementation, with the Xilinx tool for FPGA implementation. The simulation results are shown in Figure 2 (1)-(c).

### 4.1 IMPLEMENTATION DETAILS

The proposed encryption is implemented in MATLAB version 10 using the Windows 7 operating system and an Intel Core i7 processor. The following analysis has been conducted. This section presents the evaluation results, highlighting image quality analysis, statistical analysis, and performance of the algorithm.



(a)

(b)

(c)

**Fig.2**. (a-c): Xilinx simulation results

The input gray image is encrypted using 3D chaos, permutation of pixels, optimum DNA encoding, and XOR logic. Chebyshev, prime, and ASCII of 64-bit are employed to create the 3D chaotic mapping during permutation. Permutation of a column and row pixels is performed to the chaotic map's initial two dimensions, $p$, and $q$. To diffuse an image, additional DNA encoding rules are employed. This optimization approach may significantly enhance the image's encryption speed and transmission security. The chaotic map's third dimension '$z$' is employed for XOR operation. The key images are DNA-encoded. The confusion matrix is generated by XOR the encoded input and the key image.

## 4.2. STATISTICAL ANALYSIS

The implementation is carried out on eight 512 × 512 grayscale images with 256 gray levels. The images include Lena, a pepper, a baboon, a yacht, a boat, an airplane, Barbara, and a clock. Our methodology provides very secure denoised image compression, as shown by the outcomes of our experiments. The encryption and decryption of three images are shown in Table 2, and it reveals that the cipher image does not disclose any data from the compressed image. The following are subsections detailing the experimental results.

### 4.2.1. Compression ratio

The PSNR value is the objective vertex that assesses the image quality after compression and decryption, and its mathematical description is as follows:

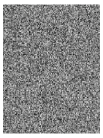
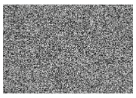
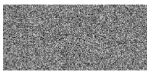
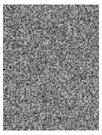
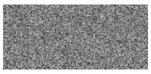
$$MSE = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} (F(x,y) - f(x,y))^2$$

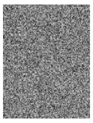
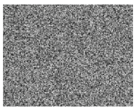
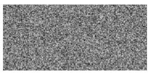
$$PSNR = 10.\log_{10}\left(\frac{255^2}{MSE}\right) \tag{24}$$

Where width and length are denoted as *W* and *H*, the original image as *f (x, y)*, and decrypted image as *F(x, y)*.

Table 2 shows the PSNR values for the various compression ratio values of various images. The image becomes increasingly distorted when PSNR decreases. When the CR is 4:3 or 2:1, the recovered image is essentially identical to the original image. When the CR is 4:1, the decrypted image is still recognizable, hence the image transmission is satisfied by the proposed compression encryption approach. Moreover, the performance of the proposed approach is increased by the use of the FPGA than using MATLAB. As the compression ratio increases, image quality degrades. Low compression improves image quality. The proposed tetrolet transform compresses normal visual data effectively.

**Table 2.** PSNR values for various CRs

| Original image | Compression ratio | Encryption image | Decryption Image | PSNR with FPGA | PSNR with MATLAB |
|---|---|---|---|---|---|
| | 4:3 |  | | 42.5943 | 40.8791 |
| | 2:1 |  | | 39.0267 | 37.6532 |
| | 4:1 |  | | 37.6125 | 35.9912 |
| | 4:3 |  | | 43.8712 | 40.9977 |
| | 2:1 | | | 41.9264 | 38.6541 |
| | 4:1 |  | | 39.5187 | 35.7889 |

| | | | | | |
|---|---|---|---|---|---|
| | 4:3 |  | | 40.816 | 37.5637 |
| | 2:1 |  | | 38.1024 | 35.2908 |
| | 4:1 |  | | 35.0013 | 33.3356 |

#### 4.2.2 Histogram Analysis

Histogram analysis of a cipher image shows the strength of its security. The histogram of encrypted images has a uniform distribution of pixel values, whereas the histogram of cipher images is nonlinear. As a result, it can be said that the encryption is secure.

The varied bar heights in the histogram image represent the various occurrence frequencies of the data. Figure 3 displays the results of a histogram analysis of the suggested encryption and decryption procedure. It can be seen from the image that the plain image's non-uniform histogram has more common pixels in the 200–50 pixel range, meaning that certain pixels are less frequent than the rest. The distribution is uniform and distinct from the histograms of a plain image while the histogram of the encrypted image has an equal count of each pixel's occurrence. When an image's histogram is flat, it offers no important data to attackers. With the use of the histograms in Figure 3, we can state with confidence that the suggested encryption method is remarkably resistant to assaults.

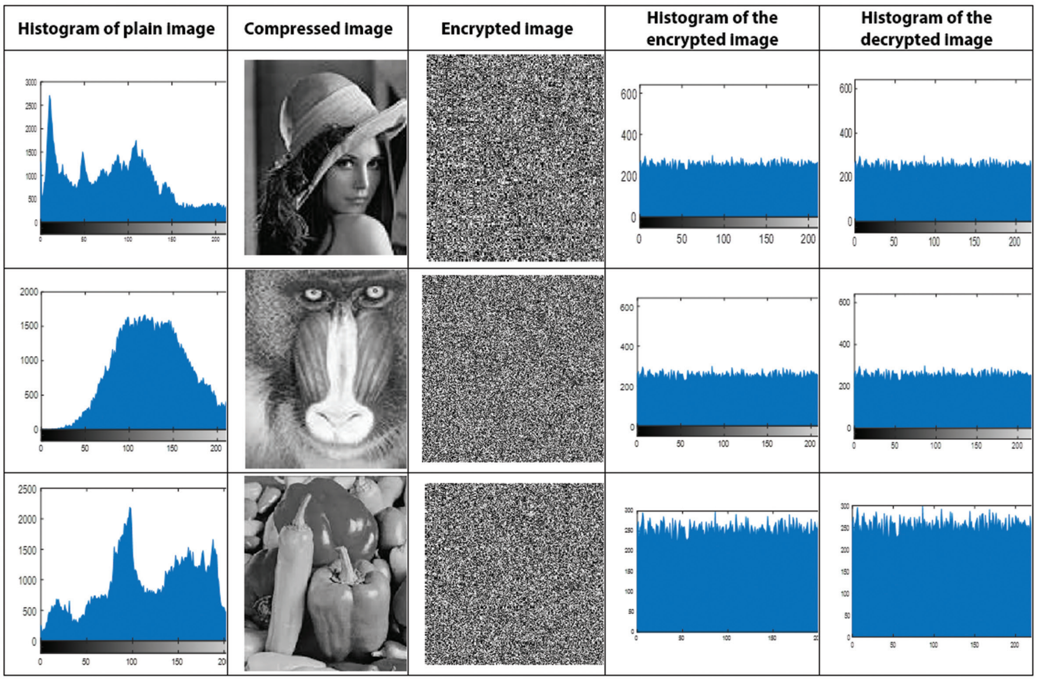


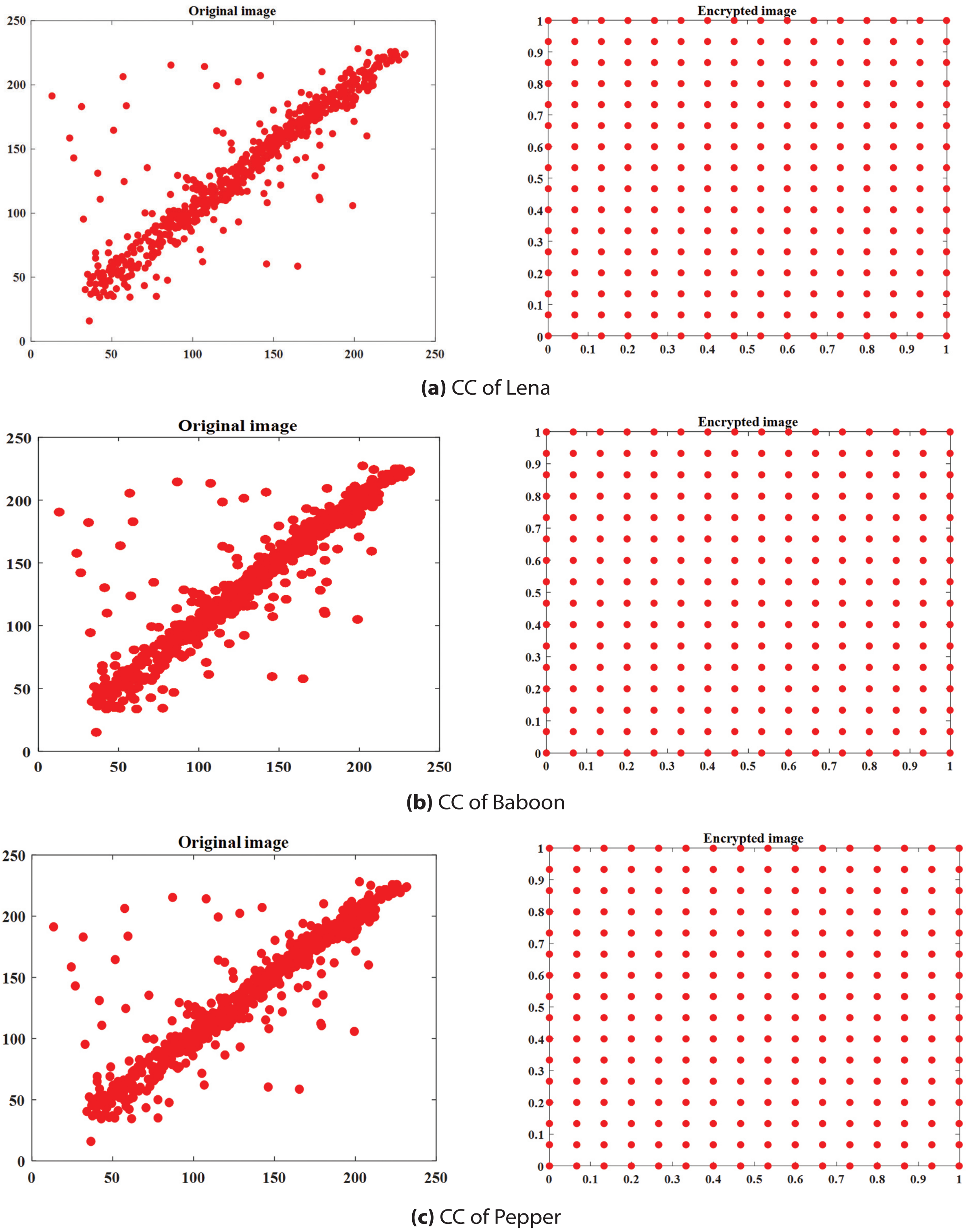
**Fig. 3.** Histogram analysis for the Encryption and Decryption process

#### 4.2.3. Correlation coefficient test

Another important test in statistical analysis is the correlation coefficient. The correlation coefficient indicates the degree of similarity between neighboring pixels in a particular image. In cryptography, the dependency between neighboring pixels must be destroyed; hence the CC near 0 is always desired in a cryptosystem. The outcome of the proposed technique for CC is shown in Figure 4. Using the following formula, the correlation coefficient can be determined:

$$CC = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N}(x_i - \bar{x})^2 \times \sum_{i=1}^{N}(y_i - \bar{y})^2}} \quad (25)$$

Where two adjacent pixels as $x_i$ *and* $y_i$, *and the* number of pixels is *N*.



**(a)** CC of Lena

**(b)** CC of Baboon

**(c)** CC of Pepper

**Fig. 4.** (a) Correlation of Lena's image. (b) Correlation of Baboon image. (c) Correlation of Pepper image

#### 4.2.4. Keyspace analysis

This value impacts the resilience of a cryptographic algorithm against brute-force attacks. It estimates the key sample space used to choose the encryption key. Therefore, to reduce brute force's feasibility, the sample area for the key should be made extremely spacious. The proposed method employs the SHA-512 function, which provides a key space of size $2^{512}$, which seems to be strong enough to resist brute-force attacks.

#### 4.2.5. Entropy test

Entropy (H) is the average quantity of information in each pixel, as per information theory as is expressed as:

$$H = \sum_{i=1}^{N} p(x_i) \times \log_2\left(\frac{1}{p(x_i)}\right) \tag{26}$$

where the total number of pixels is $N$.

A perfect cryptosystem generates ciphers with equal probabilities, resulting in an entropy of eight pixels encoded on 8 bits (pixel values between 0 and 255). Table 3 lists the various entropy values for the evaluated test images. This table indicates that the entropy of the encrypted images is quite near to the ideal value, demonstrating the algorithm's resilience to entropy threats.

**Table 3.** Results of information entropy compared with state-of-art Methods

| Test Image | Proposed using MATLAB | Proposed using FPGA | Ref. [33] | Ref. [34] | Ref. [35] | Ref. [36] |
|---|---|---|---|---|---|---|
| Lena | 7.9991 | 7.9993 | 7.9891 | 7.9991 | - | - |
| Pepper | 7.9995 | 7.9998 | 7.9929 | 7.9987 | - | - |
| Baboon | 7.9990 | 7.9992 | - | 7.9990 | 7.9969 | - |
| Yacht | 7.9984 | 7.9986 | - | - | 7.9977 | - |
| Boat | 7.9987 | 7.9989 | - | - | 7.9979 | - |
| Airplane | 7.9962 | 7.9964 | - | - | - | 7.9980 |
| Barbara | 7.9950 | 7.9952 | - | - | - | 7.9969 |
| Clock | 7.9900 | 7.9902 | - | - | - | 7.9980 |

#### 4.2.6. Strength against noise attack

In communication environments, cipher text images are frequently easily disrupted and destroyed by noise. Therefore, it must be able to ensure the resilience of encryption text while sending it. Thus, even if the cipher text is altered by interference, it is still possible to extract some relevant information.

To measure the effectiveness of our de-noising approach, the tetrolet transformation, we corrupt the grayscale image with varying quantities of salt and pepper noise. Using DCSR, GSR, wavelet, and NESTA, Table 4 displays the PSNR and SSIM quantitative evaluation results related to the proposed approach for various noise level values.

**Table 4.** Comparative results for image denoising with existing algorithms using FPGA

| Methods | Noise Level | PSNR | SSIM |
|---|---|---|---|
| Proposed | 20% | 35.21 | 0.99 |
| | 50% | 33.69 | 0.99 |
| | 60% | 30.56 | 0.99 |
| | 70% | 29.89 | 0.95 |
| NESTA Algorithm | 20% | 18.88 | 0.39 |
| | 50% | 16.71 | 0.38 |
| | 60% | 16.33 | 0.37 |
| | 70% | 16.03 | 0.36 |
| Wavelet denoising | 20% | 31.68 | 0.89 |
| | 50% | 28.61 | 0.86 |
| | 60% | 26.69 | 0.79 |
| | 70% | 24.84 | 0.74 |
| GSR Algorithm | 20% | 26.70 | 0.80 |
| | 50% | 24.66 | 0.78 |
| | 60% | 23.61 | 0.74 |
| | 70% | 22.40 | 0.69 |
| DCSR Algorithm | 20% | 33.97 | 0.99 |
| | 50% | 30.08 | 0.95 |
| | 60% | 27.03 | 0.93 |
| | 70% | 25.24 | 0.92 |

Due to the tetrolet transform's superior sparse representation for image geometrical structural properties and excellent power focusing capacity, it has a significant denoising capability. With these advantages, the performance of the proposed approach is superior to other existing techniques. Among all the techniques, the DCSR algorithm achieves similar results to the proposed approach when the noise level is lower (20%). However, when the noise level is increased, the performance of this approach is lower than the proposed approach. Figures 5 and 6 exhibit fluctuations in PSNR and SSIM output in response to varying noise levels, demonstrating the efficacy of our approach.

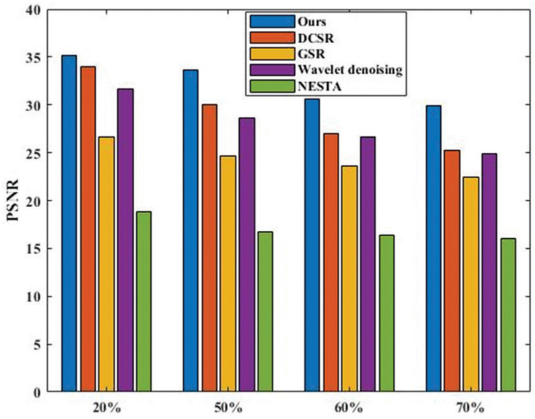


**Fig.5.** Analysis of PSNR values for various noise reduction techniques using FPGA

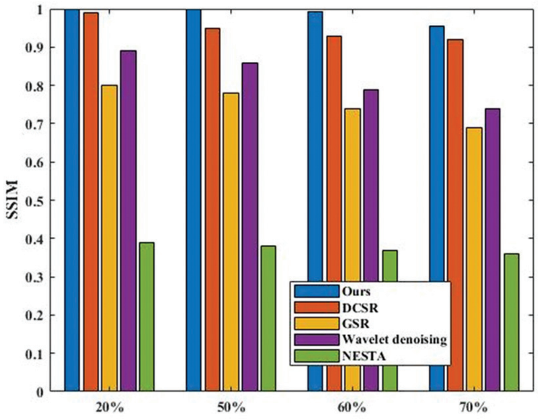


**Fig. 6.** Analysis of SSIM values for various noise reduction techniques using FPGA

These PSNR and SSIM results reveal that the demonstrated technique consistently receives the best scores, proving that the restoration result obtained by it is the best both practically and graphically.

**Table 5.** Comparative results for image denoising with existing algorithms using MATLAB

| Methods | Noise Level | PSNR | SSIM |
|---|---|---|---|
| Proposed | 20% | 33.88 | 0.97 |
| | 50% | 31.23 | 0.97 |
| | 60% | 28.48 | 0.97 |
| | 70% | 26.12 | 0.93 |
| NESTA Algorithm | 20% | 16.54 | 0.36 |
| | 50% | 14.98 | 0.35 |
| | 60% | 12.51 | 0.35 |
| | 70% | 14.93 | 0.34 |
| Wavelet denoising | 20% | 28.43 | 0.86 |
| | 50% | 26.12 | 0.85 |
| | 60% | 24.51 | 0.76 |
| | 70% | 22.90 | 0.71 |
| GSR Algorithm | 20% | 24.89 | 0.78 |
| | 50% | 23.71 | 0.75 |
| | 60% | 20.12 | 0.72 |
| | 70% | 18.39 | 0.65 |
| DCSR Algorithm | 20% | 30.78 | 0.97 |
| | 50% | 28.76 | 0.92 |
| | 60% | 25.43 | 0.91 |
| | 70% | 22.98 | 0.89 |

From Tables 4 and 5, it is observed that the performance of the proposed approach is enhanced using FPGA. The PSNR and SSIM values are quite decreased in MATLAB implementation. Finally, we concluded that the results acquired by FPGA are more clear and more accurate than the outcomes obtained by MATLAB.

#### 4.2.7. Time complexity Analysis

Another crucial metric to examine when evaluating the efficiency of the encryption system is time complexity. The plain image is compressed and encrypted as part of the encryption process, and the encrypted image is decrypted and rebuilt as part of the decryption phase. Table 5 displays the time complexity taken for reconstruction, decryption, encryption, and compression.

**Table 6.** Encryption and decryption time (sec)

| Encryption process | | |
|---|---|---|
| Compression | Encryption | Total |
| 0.84 | 0.18 | 1.02 |
| **Decryption process** | | |
| Decryption | Reconstruction | Total |
| 0.19 | 29.08 | 29.27 |

## 5. CONCLUSION AND FUTURE WORKS

This research proposes a novel approach for encrypting compressed grayscale images utilizing optimized DNA encoding and 3D-chaotic logistics mapping. Proposed the Chevrolet algorithm to compress the grayscale image with excellent quality and low noise. Here, several steps in the encryption process are carried out using a 3D chaotic map. DNA encoding is done using complementary rules on the input and key images. The encoded input image and key image are now combined logically using the XOR method. Proposed a Tuna Swarm Optimization that might significantly improve the image's encryption speed and transmission security. This study tested the proposed method on eight images and analyzed the outcomes. Our approach is tested on salt-and-pepper-noise-corrupted images. Comparing Chevrolet transformation with cutting-edge PSNR and SSIM algorithms. Histogram analysis, key space analysis, entropy, correlation coefficient, and time complexity supported the proposed algorithm's performance. The proposed method outperforms state-of-the-art methods with 32.33dB PSNR, 0.98% SSIM, and 7.99721 information entropy, demonstrating it is more secure using FPGA implementation. In the future, we may design more complex algorithms using various basic quantum chaotic systems to improve the encryption process. We can do this by utilizing chaotic quantum systems, hyper-chaos, etc.

**Conflict of Interests:** The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

**Acknowledgments**

We declare that this manuscript is original, has not been published before, and is not currently being considered for publication elsewhere.

**Funding:** The authors received no specific funding for this study

**Availability of Data and Material:** Not applicable

**Code Availability:** Not applicable

**Authors' Contributions:** The authors confirm sole responsibility for the following: study conception and design, data collection, analysis and interpretation of results, and manuscript preparation.

**Ethics Approval:** This material is the authors' original work, which has not been previously published elsewhere. The paper reflects the authors' own research and analysis truthfully and completely.

## 6. REFERENCES

[1] W. Sirichotedumrong, H. Kiya, “Grayscale-based block scrambling image encryption using ycbcr color space for encryption-then-compression systems”, APSIPA Transactions on Signal and Information Processing, Vol. 8, No. 1, 2019, pp. 12-23.

[2] Y. Naseer, T. Shah, A. Javeed, “Advanced image encryption technique utilizing compression, dynamical system, and S-boxes”, Mathematics and

Computers in Simulation, Vol. 178, No. 2, 2020, pp. 207-217.

[3] J. S. Khan, S. K. Kayhan, "Chaos and compressive sensing-based novel image encryption scheme", Journal of Information Security and Applications, Vol. 58, No. 1, 2021, p. 102711.

[4] E. Setyaningsih, R. Wardoyo, A. K. Sari, "Securing color image transmission using a compression-encryption model with a dynamic key generator and efficient symmetric key distribution", Digital Communications and Networks, Vol. 6, No.4, 2020, pp. 486-503.

[5] M. Shi, S. Guo, X. Song, Y. Zhou, E. Wang, "Visual secure image encryption scheme based on compressed sensing and regional energy", Entropy, Vol. 23, No. 5, 2021, p. 570.

[7] X. Chai, J. Bi, Z. Gan, X. Liu, Y. Zhang, Y. Chen, "Color image compression and encryption scheme based on compressive sensing and double random encryption strategy", Signal Processing, Vol. 176, 2020, pp. 107-684.

[8] V. Upadhyaya, M. Salim, "Joint approach based quality assessment scheme for compressed and distorted images", Chaos, Solitons & Fractals, Vol. 160, No.1, 2022, pp. 112-278.

[9] V. Upadhyaya, M. Salim, "Compressive Sensing: An Efficient Approach for Image Compression and Recovery", Recent Trends in Communication and Intelligent Systems, Vol.1, No.1, 2020, pp. 25-34.

[10] M. Lahdir, H. Hamiche, S. Kassim, M. Tahanout, K. Kemih, S. A. Addouche, "A novel robust compression-encryption of images based on SPIHT coding and fractional-order discrete-time chaotic system", Optics & Laser Technology, Vol. 109, No.1, 2019, pp.534-546.

[11] Y. Xie, J. Yu, S. Guo, Q. Ding, E. Wang, "Image encryption scheme with compressed sensing based on the new three-dimensional chaotic system", Entropy, Vol. 21, No. 9, 2019, p. 819.

[12] Y. G. Yang, B. W. Guan, Y. H. Zhou, W. M. Shi, "Double image compression-encryption algorithm based on fractional order hyperchaotic system and DNA approach", Multimedia Tools and Applications, Vol. 80, No. 1, 2021, pp. 691-710.

[13] H. Hu, Y. Cao, J. Xu, C. Ma, H. Yan, "An image compression and encryption algorithm based on the fractional-order simplest chaotic circuit", IEEE Access, Vol. 9, No.1, 2021, pp. 22141-22155.

[14] K. Wang, X. Wu, T. Gao, "Double color image compression–encryption via compressive sensing", Neural Computing and Applications, Vol. 33, No. 19, 2021, pp. 12755-12776.

[15] W. Huang, D. Jiang, Y. An, L. Liu, X. Wang, "A novel double-image encryption algorithm based on Rossler hyperchaotic system and compressive sensing", IEEE Access, Vol. 9, No.1, 2021, pp. 41704-41716.

[16] A. Sahasrabuddhe, D. S. Laiphrakpam, "Multiple image encryption based on 3D scrambling and hyper-chaotic system", Information Sciences, Vol. 550, No. 1, 2021, pp. 252-267.

[17] J. Mou, F. Yang, R. Chu, Y. Cao, "Image compression and encryption algorithm based on the hyper-chaotic map", Mobile Networks and Applications, Vol. 1, No. 1, 2019, pp. 1-13.

[18] B. K. Shukur, S. Hadi, W. Al-Hameed, "An Efficient Approach Implementation to Image Compression and Encryption", Eurasian Journal of Engineering and Technology, Vol. 8, No. 1, 2022, pp. 1-13.

[19] L. Gong, K. Qiu, C. Deng, N. Zhou, "Image compression and encryption algorithm based on a chaotic system and compressive sensing", Optics & Laser Technology, Vol. 115, No. 1, 2019, pp. 257-267.

[20] J. Karmakar, D. Nandi, M. K. Mandal, "A novel hyper-chaotic image encryption with sparse-representation-based compression", Multimedia Tools and Applications, Vol. 79, No. 37, 2020, pp. 28277-28300.

[21] H. Dong, E. Bai, X. Q. Jiang, Y. Wu, "Color image compression-encryption using the fractional-order hyperchaotic system and DNA coding", IEEE Access, Vol. 8, No.1, 2020, pp. 163524-163540.

[22] G. U. Bhargava, S. V. Gangadharan, "FPGA implementation of modified recursive box filter-based fast bilateral filter for image denoising", Circuits, Systems, and Signal Processing, Vol. 40, No. 3, 2021, pp. 1438-1457.

[23] M. Maazouz, A. Toubal, B. Bengherbia, O. Houhou, N. Batel, "FPGA implementation of a chaos-based

image encryption algorithm", Journal of King Saud University-Computer and Information Sciences, Vol. 1, No. 1, 2022, pp. 23-45.

[24] T. M. Kumar, K. S. Reddy, S. Rinaldi, B. D. Parameshachari, K. Arunachalam, "A low area high-speed FPGA implementation of AES architecture for cryptography application. Electronics, Vol. 10, No. 16, 2021, pp. 20-23.

[25] H. Zodpe, A. Sapkal, "An efficient AES implementation using FPGA with enhanced security features", Journal of King Saud University-Engineering Sciences, Vol. 32, No. 2, 2020, pp. 115-122.

[26] A. Hafsa, M. Gafsi, J. Malek, M. Machhout, "FPGA implementation of improved security approach for medical image encryption and decryption", Scientific Programming, Vol. 3, No. 1, 2021, pp. 34-56.

[27] T. M. Kumar, K. R. Balmuri, A. Marchewka, P. B. Divakarachari, S. Konda, "Implementation of Speed-Efficient Key-Scheduling Process of AES for Secure Storage and Transmission of Data", Sensors, Vol. 21, No. 24, 2021, p. 8347.

[28] G. U. Bhargava, S. V. Gangadharan, "FPGA implementation of modified recursive box filter-based fast bilateral filter for image denoising", Circuits, Systems, and Signal Processing, Vol. 40, No. 3, 2021, pp. 1438-1457.

[29] J. Krommweh, "Tetrolet transform: A new adaptive Haar wavelet algorithm for sparse image representation", Journal of Visual Communication and Image Representation, Vol. 21, No. 4, 2010, pp. 364-374.

[30] M. McLoone, J. V. McCanny, "Efficient single-chip implementation of SHA-384 and SHA-512", Proceedings of the IEEE International Conference on Field-Programmable Technology, Hong Kong, China, 16-18 December 2002, pp. 311-314.

[31] L. Kocarev, Z. Tasev, "Public-key encryption based on Chebyshev maps", Proceedings of the International Symposium on Circuits and Systems, Bangkok, Thailand, 25-28 May 2003.

[32] A. ShokouhSaljoughi, H. Mirvaziri, "A new method for image encryption by 3D chaotic map", Pattern Analysis and Applications, Vol. 22, No. 1, 2019, pp. 243-257.

[33] M. Tan, Y. Li, D. Ding, R. Zhou, C. Huang, "An Improved JADE Hybridizing with Tuna Swarm Optimization for Numerical Optimization Problems", Mathematical Problems in Engineering, Vol. 1, No. 1, 2022.

[34] Y. Naseer, T. Shah, A. Javeed, "Advanced image encryption technique utilizing compression, dynamical system, and S-boxes", Mathematics and Computers in Simulation, Vol. 178, No.1, 2020, pp. 207-217.

[35] W. Alexan, M. ElBeltagy A. Aboshousha, "Rgb image encryption through cellular automata, s-box, and the Lorenz system", Symmetry, Vol. 14, No. 3, 2022, p. 443.

[36] J. Hao, H. Li, H. Yan, J. Mou, "A New Fractional Chaotic System and its application in image encryption with DNA mutation", IEEE Access, Vol. 9, No.1, 2021, pp. 52364-52377.

[37] R. Guesmi, M. A. Farah, "A new efficient medical image cipher based on a hybrid chaotic map and DNA code", Multimedia tools and applications, Vol. 80, No. 2, 2021, pp. 1925-1944.

**Appendix-1**

| **Acronyms** | Abbreviations |
|---|---|
| **FPGA** | Field-Programmable Gate Array |
| **DFT** | Discrete Fourier Transform |
| **DCT** | Discrete Cosine Transform |
| **DWT** | Discrete Wavelet Transform |
| **FWPT** | Full Wavelet Packet Transform |
| **XSG** | Xilinx System Generator |
| **HRRBF** | Hybrid Recursive Reversible Box Filter |
| **FABF** | Fast Adaptive Bilateral Filter |
| **RLG** | Reversible Logic Gates |
| **AES** | Advanced Encryption Standard |
| **LUT** | Look Up Table |
| **MPPRM** | Modified Positive Polarity Reed-Muller |
| **PRNG** | Pseudorandom Number Generator |
| **MAES** | Modified Advanced Encryption Standard |
| **FJMKE** | Fork-Join Model Of Key Expansion |
| **MRBF** | Modified Recursive Box Filter |
| **FBF** | Fast Box Filter |
| **TSO** | Tuna Swarm optimization |

# Improved Security of a Deep Learning-Based Steganography System with Imperceptibility Preservation

Original Scientific Paper

**Ammar Mohammedali Fadhil**
Middle Technical University,
Institute of Technology , Department of Information and Communication Technology ,
Alzaafaraniya, Baghdad, Iraq
ammaral-khafaji@mtu.edu.iq

**Hayder Nabeel Jalo**
Middle Technical University,
Institute of Technology , Department of Information and Communication Technology ,
Alzaafaraniya, Baghdad, Iraq,
hayderjalo@mtu.edu.iq

**Omar Farook Mohammad**
Al-Hadba University College,
Computer Technology Engineering Department
Mosul, Iraq
ofmalobaidy@hcu.edu.iq

***Abstract*** – *Since its inception, the steganography system (SS) has continuously evolved and is routinely used for concealing various sensitive data in an imperceptible manner. To attain high performance and a better hiding capacity of the traditional SS, it has become essential to integrate them with diverse modern algorithms, especially those related to artificial intelligence (AI) and deep learning (DL). Based on this fact, we proposed a DL-based SS (DLSS) to extract some significant features (like pixel locations, importance, and proximity to the imperceptibility) from the cover image using a neural network (NN) in a hierarchical form, thus selecting the candidate pixels for embedding afterwards. The pixel weight was expressed in terms of the position, imperceptibility, and its relationship with adjacent pixels to be a stego image. Performance evaluation revealed that the proposed DLSS achieved imperceptibility of 84 dB for images in training mode of a standard dataset.*

***Keywords***: *deep learning, steganography, neural network, embedding, imperceptibility*

## 1. INTRODUCTION

With the advent of the information communication technology (ICT), the transfer of various sensitive data in form of images, videos, and audio occurs primarily over the Internet. Meanwhile, many problems have been encountered related to the security and reliability of such accessible communication of information [1]. Thus, researchers in the field of information security became concerned about the legality of such information transfer and the freedom to have information, ensuring privacy-preserved data transfer [2]. In this rationale, the importance of diverse applications-based information transmission at the level of local and global networks appeared as one of the main focuses of the study. The main objective of this study is to protect the information (so-called privacy-preserved data communication) from penetration and plagiarism. In recent years, research on information penetration and data security revealed ever-increasing threats from hackers and adversaries, enforcing the rapid development of various protection techniques including the steganography system (SS) [3]. Previously, information security systems used data encryption algorithms to send data from one party to another, where such algorithms included encryption keys that contained all the information required to decrypt the information [4].

Digital data provide a comfortable environment for editing and modifying the data that can be copied without losing data quality and content. The computer-processed digital data can be delivered from one device to another without any errors or external interference [5]. However, digital data distribution poses serious concern due to attacks or manipulation by unauthorized users, which leads to the loss of relevance, thus weakening its security aspect. Lately, the Internet access related insecurity of digital content has posed great challenges to all software developers, researchers, users, and distributors. A large part of the digital world is engrossed in the Internet, where several applications are implemented, considering the Internet service as a proven method of data communication between users. As the contemporary communication technologies empowered by the Internet and cloud computing have become an integral part of daily life, there is an urgent need to establish smart algorithms for highly secured and privacy-preserved information transfer over the Internet [6]. It has been realized that weak information security is mainly due to data transfer through insecure public channels. Thus, there must be some secure means to protect that information from unauthorized uses or illegal access by adversaries or hackers. To overcome this problem, dedicated research efforts have been made to achieve secure and confidential information communication, with information hiding technology constantly growing and becoming more complex. Information concealment technologies include digital media like images, video, and audio, providing an excellent carrier of hidden confidential information [7].

Using the data hiding technique, secret information and messages can be transmitted in a secure way through cover media, undetectable to viewers, hackers, and trackers. Over the decades, data hiding methods have been widely used to transfer confidential medical, military, agricultural, and other data. The SS has been most commonly used for concealing textual data securely that hide a specific text in one of the media, making them imperceptible [8] to others, except those who have the key to solving the algorithm. Until the secret data transmitted by the sender are received at the authorized recipient end, it remains hidden inside the medium [9]. Several daily life applications on the Internet use digital images, thus offering a suitable environment for data hiding. One can define the SS either perceptibly or imperceptibly through a high degree of security [10]. Any SS is very complex because it manipulates imperceptibly and efficiently the data of the transmission media. Consequently, the data hiding process suffers from various limitations related to the image size and the accuracy of transmitted information.

Despite the invisibility of hidden data, they are somehow visible to observers; however, useful information remains undisclosed without a key [11]. The major component of data concealment in the SS is called cryptography. It ensures that the information hidden in the digital medium cannot be perceived by the human eye, which is why the observer cannot detect the message included in the medium [12]. The main goal of any SS is to improve the security of data transmitted from the sender to the receiver. The purpose of using the SS is to hide confidential data from the public and make the transmitted image free of any information. Watermarking is the second part of the data hiding process [13], where by simple changes, the hidden data are often presented in the form of simple images. Cryptography [14] is a process of encrypting data (images or written text), whereby the data pose a challenge to the observer and cannot be decrypted. (Fig. 1) shows a classification scheme of information hiding.

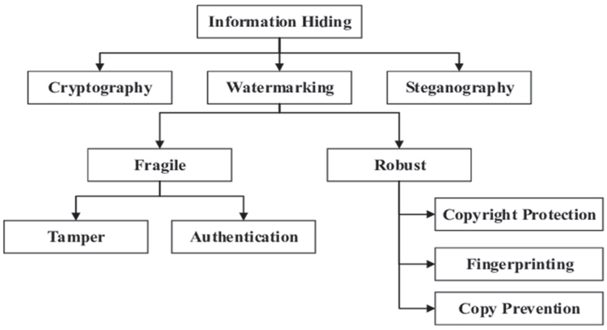


**Fig. 1.** Classification scheme of information hiding

Although each type of information hiding systems has its advantages and disadvantages, most researchers attempt to overcome the shortcomings of the SS. The main idea behind the improvement of the SS is to transmit information with high capacity, high security, and low imperceptibility. Table 1 gives a comparison of three types of information hiding schemes in terms of their main characteristics.

**Table 1.** Comparison of three types of information hiding schemes.

| Attributes | Watermark | Cryptography | Steganography |
|---|---|---|---|
| Media | Image is popular, maybe text and video | Mostly text, sometimes image | Digital form of images, video, and audio |
| Imperceptibility | No | Yes | No |
| Visibility | Medium | High | Less |
| Key | Yes | No | Yes |
| Criteria | Capacity and imperceptibility | Security | Security, capacity, and imperceptibility |
| Results | Watermark | Cipher | Stego |
| Application | Authentication | Commerce | Many applications |
| Readability | Simi | Full | Full with extraction |

Any information hiding system is composed of two components, a sender and a recipient [15]. The secret message (text) is embedded into the medium (a stego image) and then transmitted by the sender. Then, the message from the medium (a stego image) is extracted by the recipient. The stego image embedded in the message is identical with the original image. The process of embedding and extracting requires specific algorithms that contain a key called a stego key. In short, by embedding, the sender generates a stego image and a key, while the recipient extracts the desired information (a secret message, and produces the original image) from the stego image by using the stego key (Fig. 2).

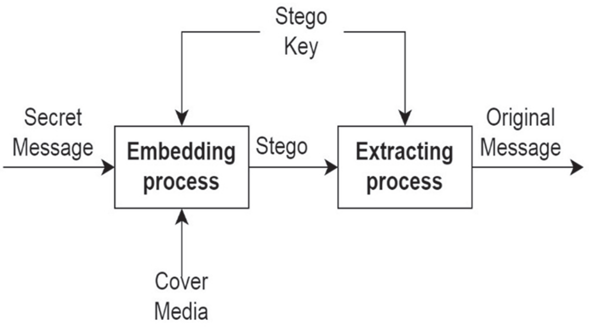


**Fig. 2.** Basic SS architecture

Most of the previous contributions related to the process of distributing secret data in the image. In this paper, we propose a method based on one of the artificial intelligence algorithms, i.e., deep learning. The cover image is initially divided into small images based on the color contrast of the image. After that the location of the secret bit is obtained by applying the neural network algorithm, and then we can use the weight factor to select the best pixel to embed the secret bit among more than one pixel.

One of the most important stages in steganography is the selection of the pixel to embed the secret bit, which must be done in such a way as to preserve imperceptibility. Therefore, the deep learning method was used to find the best pixel for embedding and thus to increase imperceptibility.

### 1.1. DEEP LEARNING (DL)

By using artificial intelligence (AI) it is possible to combine various modern technologies. The main goal of such unification is to mimic through artificial systems the cognitive abilities of humans and their intelligent behavior for further exploitation, especially in terms of solving complex problems such as detection, object recognition and self-driving [16]. Machine learning systems (MLS) consist of two main procedures, including feature extraction, and are designed for training [17]. Developers design a feature extraction protocol to extract different features from the data input into the system followed by their training to learn the system using the classifiers. This is performed to achieve a suitable function ensuring the absolute security of the private data transmitted by the sender and retrieved by the recipient. Despite the effectiveness of MLS in solving complex problems, they suffer from various limitations [18]. Many methods in the literature use deep learning techniques to solve different problems [19, 20]. In order to overcome these shortcomings, it is necessary to develop a suitable feature extractor. Creating a proper feature extractor is challenging because it requires experience and in-depth knowledge of the problem developers face. In addition, a particular feature cannot be generalized to another problem. In order to overcome this problem, DL has been recently widely used as part of ML. Fig. 3 illustrates a typical AI architecture.

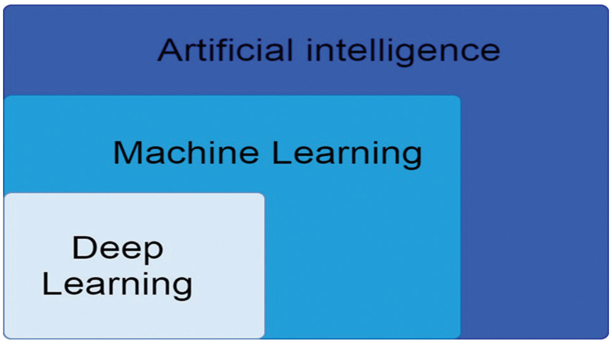


**Fig. 3.** General framework of AI

DL based on an improved artificial neural network (ANN) model is used in this paper. The ANN model is used to input the data into the neural system, and then the received output is fed back as an input, creating an experience for the system. Weight vectors such as $w=(w_1,w_2,..,w_m)$, with $w_i \in R$, can be manipulated at the neural system to produce the input vector such as $x=(xw_1, xw_2,.., xw_m)$, and applied as the following nonlinear function to obtain the output:

$$y = \sum_{i=1}^{n} w_i \times x_i + b, \tag{1}$$

where $y$ is the output, which is the sum of weight ($w_i$) times the input vectors ($x_i$) plus bias ($b$).

## 2. RELATED WORK

Intensive studies have been conducted on the principle of hiding data as text in images or other media. Researchers have developed various modern technologies and linked them to data hiding or the SS [21], which used an inverted bit stream to increase imperceptibility, but the capacity was very low. Here, DL played a considerable role in data steganography advancement, especially for data in JPEG image format and other types of text [22] considering the number of attacks in terms of security to avoid secret data manipulation. Attempts were made to create a new paradigm of data steganography analysis using the concept of feature learning, a novel CNN-based method for feature extraction and classification, as well as techniques to improve imperceptibility [23]. One of the most important advantages of this method is the reliability of more than one type of image, but this method suffers from its inability to account for hacker attacks. The DL-based SS [24, 25] was implemented to improve the NN classifier, which achieves improved security and data hiding capacity in the network. The information hiding method is good and effective, and it cannot embed a large amount of data, which most traditional methods suffer from. A new model [26] based on training the embedded images and an AI-assisted classifier could attain an embedding ratio of 70% in the training stage and of 30% in the examination stage, indicating an excellent outcome. The management of the training and the testing mode with the compatibility process between them were successful and the reason for increasing the security of embedding secret data with limited embedding data, which is the basis of the steganography system. A convolutional NN model was proposed [27] that consisted of three main stages, including calculation and data analysis, extraction of significant features, and classification of the extracted features in the digital image in order to embed hidden data into them, Although extracting important features from a cover image improves data security, it does not help to increase the imperceptibility of the stego image. A comprehensive review of the most recent existing reports in the literature related to data hiding (especially steganography) showed the use of diverse methods that mostly depend on the DL algorithms based on the celebrated ANN algorithms [28, 29]. DL algorithms [30] have been used to cover a digital image that include object border pixels within digital images, thus accurately classifying these pixels according to feature weights for further embedding. In all cases, embedding in a section or part of the cover image reduces the image capacity for secret data, which is important to sign for the method used to be feasible. Moreover, DL was directly applied [9] to the SS for the purpose of constructing an encoder and a decoder, enabling the learning of reversible steganography by distributing data according to the NN algorithm. However, it still needs to be improved in terms of security and robustness, which are the disadvantages of this method.

From the above, we can propose a method that takes advantage of existing methods and at the same time avoids the problems associated with the steganography system. The method depends on the method of selecting the hiding data position in the image (pixels) through deep learning (impact of a smart variable) to avoid the classical distribution and increase the imperceptibility. Furthermore, dividing the image into subimages helps to avoid statistical attacks faced by the steganography system, thereby maintaining the security of the transmitted data.

## 3. PROPOSED STEGANOGRAPHY METHOD

An image steganography system processes a specific image enclosed by pixels, where each pixel has a decimal value consisting of one byte or 8 bits. Human eyes can easily recognize grey in four bits called the most significant bits (MSB) and cannot recognize the other four bits called the least significant bits (LSB). (Fig. 4) displays the process of hiding a secret message.

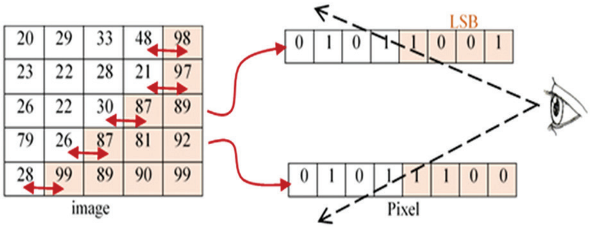


**Fig. 4.** Image recognition by human eyes

Hiding text in a given image involves two steps: in the first step, the text is converted into a series of binary bits from 0 and 1, while in the second step, these bits are embedded in the digital data of image pixels. Each pixel consists of 8 bits, which can contain one to two bits of a text message. Most of the existing methods take into account the embedding place, but with the same technique.

Feature selection is the most significant step in ML that works together with the NN algorithm. The candidate pixels act as NN inputs for embedding, where only those pixels that satisfy the condition $P_{con.}=P_1-P_2=16$ (decimal value) can be added to the LSB. Therefore, the number of pixels can be stored in the form of a vector (called the input layer) for input into the NN code (Fig. 5).

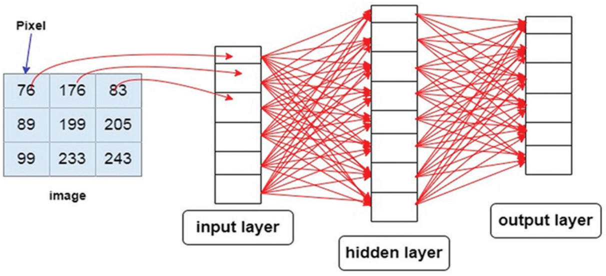


**Fig. 5.** The structure of the NN with the cover image

The image (Fig. 5) is comprised of pixels represented by their decimal values (act as an input layer of the NN). The NN has three major layers including the input, hidden, and output layers. Pixel values in the proposed NN are inserted into the input layer, producing a filter from the candidate pixels for embedding, where information is saved in the stego key. The output vector is delivered again to the image by maintaining the coordinate of the pixel $(x, y)$ as the address. The cover image at the start is divided into several sub-images each having a definite size depending on the generated random function. The image is divided into sub-images so that the data are in multiple vectors. In this way several neural networks are achieved according to the number of vectors or images. Inputs $x$ and corresponding weighs $(w)$ are related by the following:

$$V = \sum = (x_1 \times w_1) + (x_2 \times w_2) \ldots + (x_n \times w_n) \quad (2)$$

$$z = \sum = x.w = w^T x \quad (3)$$

$$z = w^T x + b\,, \quad (4)$$

where $\hat{y}=g(z)$ is the output layer. $V$ is considered as a vector of weight w and pixel data $x$.

Many sub-images imply various NNs, and each sub-image can be handled by a single NN. However, as shown in (Fig. 6), DL can deal with multiple NNs.

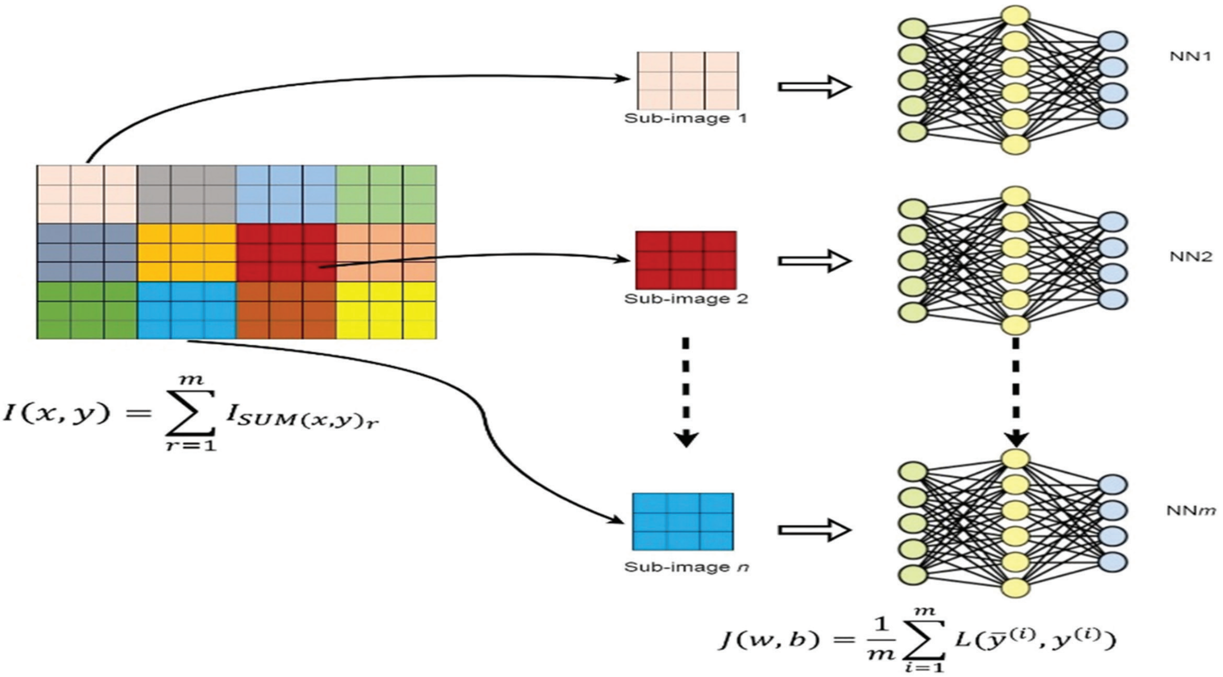

Sub-image 1
NN1
Sub-image 2
NN2
Sub-image $n$
NN$m$
$I(x,y) = \sum_{r=1}^{m} I_{SUM(x,y)_r}$
$J(w,b) = \frac{1}{m}\sum_{i=1}^{m} L(\bar{y}^{(i)}, y^{(i)})$

**Fig. 6.** DLNN system for the entire cover image

A convolutional layer with NNs considers the main issue, representing the main elements in the system, as well as the features, i.e., a set of features given by each NN is produced during the processing of the hidden layer. Convolution selects the pixel value of *I(x,y)* with the assumed weight derived from the adjacent pixels called kernel K∈R$^{(2kf+1)\times(2ks+1)}$ such that kf and ks are sub-images of dimension (3x3). The stego pixel takes the form:

$$S(i,j) = S.k = \sum_{u=1}^{kf}\sum_{v=1}^{ks} I_{(i+u,j+v)}K_{(u,v)}, \quad (5)$$

where $K$ is the kernel of the corresponding coordinate of cover pixel (*u,v*) aimed at producing stego pixel *S*(*i,j*).

The number of hidden layers inside the system can be controlled, providing several feedback inputs until the appropriate stego pixels are due to the manipulation of cover pixels achieved under the corresponding weights. Fig. 7 displays a typical procedure for getting the secret text.

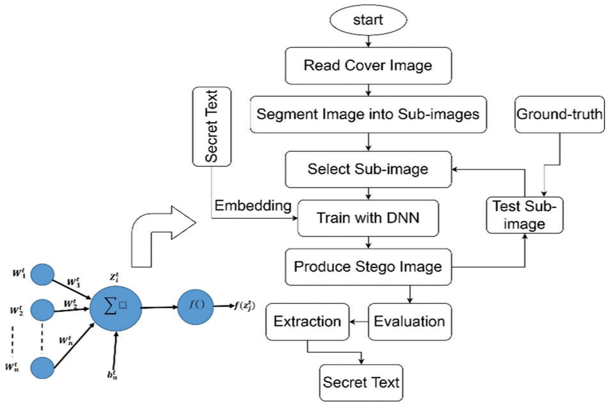


**Fig. 7.** General overview of the system

The proposed system first reads the cover or original image and then divides it into sub-images following a specific condition before being fed to each individual *NN*. First, the *NN* is selected under the features derived from sub-image pixels. Next, each *NN* contributes to a deep neural system with the submission of new features improved during the processing.

## 4. RESULTS AND DISCUSSION

Fig. 8 shows cover images used by the proposed DLSS together with the corresponding stego images. Numerous images of size (256 × 256) and (512 × 512) pixels from the standard dataset were used for performance evaluation.

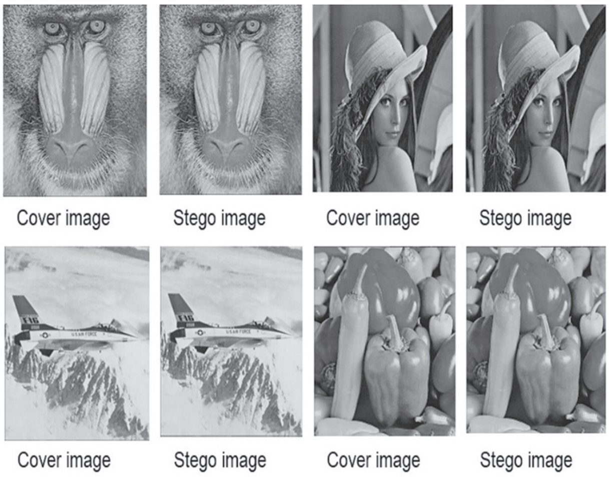


**Fig. 8.** Standard cover and stego images used in the proposed DLSS

Images like Lena, Baboon, Peppers, and Jet with different payload capacities were tested. The stego image is defined as the secret message inside the original image that remains indistinguishable from anything, meeting the purpose of steganography. The efficiency of the system was evaluated in terms of imperceptibility, payload capacity, secret bits embedded in the imperceptible part of the pixel (LSB) of the cover image, the peak signal-to-noise ratio (PSNR), and the mean square error (MSE).

As already mentioned (Fig. 7), human eyes cannot differentiate the cover from the stego image, which makes it difficult to observe the inner secret. For this reason, hackers or intruders use a statistical technique to figure out the secret message included in the stego image. Over the years, numerous image steganography techniques have been developed to obtain an optimal algorithm that can achieve the best results. However, specific benchmark criteria are needed to compare the performance of the proposed DLSS with the existing state-of-the-art methods. Although most existing SS can successfully hide private information, making it indistinguishable to human eyes, these techniques suffer from various statistical issues that need to be overcome. So, to properly validate the results obtained from the proposed DLSS, it is important to determine payload capacity of the secret message, indicating the robustness of the stego image (carrying data without distortion) against various attacks. In this study, PSNR and MSE parameters were used for validation.

### 4.1 PSNR

In terms of MSE of the proposed DLSS, PSNR values (in dB) were evaluated as follows:

$$PSNR = 10\log_{10}\left(\frac{max^2}{MSE}\right), \tag{6}$$

where max indicates the maximum pixel intensity value of 255, and PSNR is a measure of image resolution and distortion derived from the mean square error (MSE). For both greyscale and color images, PSNR as high as 70 dB and above is considered to be very good, in the range of 30 to 50 dB is acceptable, and below 30 dB is unacceptable. Table 2 summarizes the obtained imperceptibility and payload capacity results for different images in both greyscale and color images. Table 3 shows PSNR values for various standard images of different pixel sizes achieved by the proposed DLSS.

**Table 2.** Performance evaluation of the proposed DLSS.

| Image | Image resolution | Payload capacity (bytes) | Embedding ratio | Pixel representation | | | | | | | | PSNR (dB) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Lena | 256 × 256 (pixel) | 32765 | 6.25% | 1 | 0 | 1 | 1 | 0 | 1 | 0 | ½ | 76 |
| | 256 × 256 (pixel) | 53743 | 12.5% | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 61 |
| | 256 × 256 (pixel) | 64752 | 18.75% | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 42 |
| Lena | 512 × 512 (pixel) | 32765 | 6.25% | 1 | 0 | 1 | 1 | 0 | 1 | 0 | ½ | 84 |
| | 512 × 512 (pixel) | 53743 | 12.5% | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 70 |
| | 512 × 512 (pixel) | 64752 | 18.75% | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 54 |

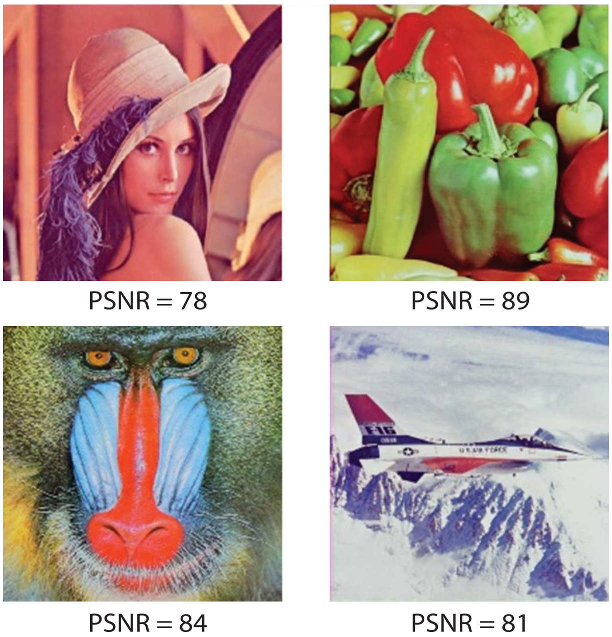
In the first NN iteration, when a 3K secret message was embedded, the secret bit appeared in the first bit of the LSB with every two pixels getting a one-pixel candidate. The neural system could present pixels with the ability to embed secret bits within the cover image. For an embedding ratio of 12.5%, all candidate pixels could replace the LSB (first bit) with the secret bit that appeared from the text message. In contrast, for an embedding ratio of 18.75%, the system made it possible to occupy two pixels from the LSB for embedding secret bits (Table 3).

**Table 3.** Results of images used in the proposed method with their sizes.

| Image | Image size (pixels) | PSNR (dB) |
|---|---|---|
| Baboon | 256 × 256 | 79 |
| Peppers | 256 × 256 | 78 |
| Jet | 256 × 256 | 75 |
| Baboon | 512× 512 | 88 |
| Peppers | 512× 512 | 86 |
| Jet | 512× 512 | 83 |

One of the most important features of steganography results is imperceptibility, which is measured by PSNR. This factor depends on the strength of the distribution of secret data by image pixels, which results in the deep learning method in a smart distribution of data within the image.

PSNR values are found to depend on image information, e.g., a baboon image includes several pixel variations thus nominating the neural system to embed multiple pixels. One of the significant features used by a deep neural network (DNN) is the difference between certain pixels and adjacent pixels (4 or 8 neighbors). Thus, the Lena image achieved a lower PSNR value because of the uniform pixel values and the embedding ratio of 18% of the image. In contrast, the Baboon image had too many pixels variations, allowing a large number of pixels to be selected to store secret bits. In this study, the DNN was used to increase payload capacity while keeping the imperceptibility (image quality) of greyscale images (one channel represented by a one-pixel value) intact. The same strategy was used for color images where the process included three channels according to red, green, and blue (RGB), indicating that each pixel consists of 24 bits (8 bits for each channel). In addition, as illustrated in (Fig. 9), PSNR values in color images were higher than in grey images. It was asserted that the training system containing different images over 205 is worth improving the variable hidden layer with the neural system, thereby improving image imperceptibility.


PSNR = 78

PSNR = 89

PSNR = 84

PSNR = 81

**Fig. 9.** PSNR of (256 × 256) color images with a 6.25% embedding ratio

### 4.2. MSE

The MSE values of the proposed DLSS were evaluated based on the difference between the original image (prior to embedding) and the stego image (carrying a secret message) by the expression:

$$MSE = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} (G1_{ij} - G2_{ij})^2, \qquad (7)$$

where M and N are the row and the column of the image, and G1 and G2 are the cover and the stego image pixels, respectively, representing the ith row and the jth column. The obtained MSE value for the greyscale image was 100.0 (worst), and reduced to 0 (better). A 10-bit image with a corresponding pixel value of [0,1023] became undetectable, which indicates the excellent performance of the DLSS because the main purpose of any steganography system is to reduce the MSE value as much as possible. The achieved MSE value of 0 clearly indicates that there is no difference between the cover image and the stego image.

Table 4 compares the present results with results obtained in other studies described earlier in the literature. In order to demonstrate the superior nature of the proposed DLSS compared to the existing state-of-the-art SS, the achieved results were additionally compared with the most significant findings published in the literature. Different steganography techniques have shown different performances in terms of payload capacity and PSNR values, indicating that the best one is the current DLSS. The obtained improvement in payload capacity and PSNR values was attributed to the excellent embedding ratio by the DLNN. The PSNR value with an embedding ratio of 6.25% (green bar) was higher due to fewer secret messages or less information being embedded into the image, causing a smaller image degradation or distortion effect. In addition, the blue bar represented a higher embedding ratio to obtain a low PSNR. It is important to note that some studies did not evaluate all capacities, hence the missing bars. The high imperceptibility achieved by the proposed DLSS was mainly due to the use of a large number of iterations and new features such as pixel variations and differences between pixel values. In conventional methods, pixel values are limited such that the difference is fixed in advance (for example, difference = pix1(value) - pix2(value) = 40). In the proposed DLSS with a DNN, all variables were changed through system training by increasing and decreasing the number of hidden layers and nodes.

**Table 4.** Comparison of the present results with results obtained in other studies described earlier in the literature, using the USC-SIPI database with a 6.25% embedding ratio.

| Authors | Year | Image used | Method | PSNR |
|---|---|---|---|---|
| [31] Diar D. et al. | 2021 | Lena | LSB+CRT+PVD | 73.0 |
| [32] U. Pilania & P. Gupta | 2020 | Lena &Baboon | IWT-SVD Scheme | 54.1 |
| [33] M. Oudah et al. | 2020 | Lena | DWT Transform | 70.5 |
| [34] Q. Li et al. | 2020 | Baboon | Chaos Encryption | 61.2 |
| [35] M. Kumar, & H. Nagar | 2021 | Lena | Hybrid LSB+ AES cryptography | 75.2 |
| [36] A. Hindi, et al. | 2019 | Baboon | Index XOR LSB | 74.4 |
| [37] S. Almutairi, et al. | 2019 | Baboon | 2 bits LSB | 79.3 |
| **Proposed** | | Lena +Baboon | DNN + region segmentation | 84.3 |

## 5. CONCLUSION

In this paper, a robust DLSS is proposed for extracting different significant features from the cover image using DL combined with a NN in a hierarchical form. In this way, candidate pixels are selected for embedding. The use of DL in steganography has made the hiding process more secure, allowing more payload capacity to be embedded for digital images. By this method, DL as an AI algorithm could extract features that are later processed according to weight and importance. Steganography consisted of the cover and stego images, where the cover image was used to extract significant features of the images. Here, pixel position provided the basis for work in addition to image weight and imperceptibility. The NN was used to determine the suitability of embedding sites, resulting in high imperceptibility (84.3 dB for a 512 ×512 image) of the stego image. The results revealed that it is possible to embed huge amounts of information instead of the previously approved random embedding. Performance evaluation showed that the proposed DLSS outperformed the existing steganography in terms of PSNR, MSE, and imperceptibility values, indicating high data security against attacks with capacities of a 12.5% and 18.75% embedding ratio. It is worth taking a valuable part of the image or dividing the image into several parts to further improve the DLSS. Furthermore, it may be interesting to adopt the NN algorithm by selecting a section and hierarchical division.

## ACKNOWLEDGMENT

The authors are grateful to the Middle Technical University (MTU), and the Al-Hadba University College, Iraq, for technical assistance.

## 6. REFERENCES

[1] Y. Li, Y. Tu, J. Lu, Y. J. S. Wang, "A Security Transmission and Storage Solution about Sensing Image for Blockchain in the Internet of Things", Sensors, Vol. 20, No. 3, 2020, p. 916.

[2] K. Del Villar, E. Close, R. Hews, L. Willmott, B. White, "Voluntary assisted dying and the legality of using a telephone or internet service: The impact of Commonwealth 'Carriage Service' offences", Monash University, Vol. 47, 2021, p. 125.

[3] Y. Zhang, X. Le, Y. Jian, W. Lu, J. Zhang, T. Chen, "3D Fluorescent Hydrogel Origami for Multistage Data Security Protection", Advanced Functional, Vol. 29, No. 46, 2019, p. 1905514.

[4] A. A. Yazdeen, S. R. Zeebaree, M. M. Sadeeq, S. F. Kak, O. M. Ahmed, R. R. Zebari", FPGA Implementations for Data Encryption and Decryption via Concurrent and Parallel Computation: A Review", Qubahan Academic, Vol. 1, No. 2, 2021, pp. 8-16.

[5] D. M. Abdullah et al. " Secure Data Transfer over Internet Using Image Steganography: Review", Asian Journal of Research in Computer Science, 2021, pp. 33-52.

[6] J. Molina-Ríos, N. Pedreira-Souto, "Comparison of development methodologies in web applications", Information and Software Technology, Vol. 119, 2020, p. 106238.

[7] R. Tabares-Soto et al., "Digital media steganalysis", Digital Media Steganography, Elsevier, 2020, pp. 259-293.

[8] M. Hussain, A. W. A. Wahab, Y. I. B. Idris, A. T. Ho, K.-H. Jung, "Image steganography in spatial domain: A survey", Signal Processing: Image Communication, Vol. 65, 2018, pp. 46-66.

[9] C.-C. Chang, X. Wang, S. Chen, I. Echizen, V. Sanchez, C.-T. Li, "Deep Learning for Predictive Analytics in Reversible Steganography", arXiv:2106.06924, 2021.

[10] Y. Ke, J. Liu, M.-Q. Zhang, T.-T. Su, X.-Y. Yang, "Steganography Security: Principle and Practice", IEEE Access, Vol. 6, 2018, pp. 73009-73022.

[11] S. A. Parah, J. A. Sheikh, J. A. Akhoon, N. A. Loan, G. M. Bhat, "Information hiding in edges: A high capacity information hiding technique using hybrid edge detection", Multimedia Tools and Applications, Vol. 77, No. 1, 2018, pp. 185-207.

[12] Y. Li, S. Yao, K. Yang, Y.-A. Tan, Q. Zhang, "A High-Imperceptibility and Histogram-Shifting Data Hiding Scheme for JPEG Images", IEEE Access, Vol. 7, 2019, pp. 73573-73582.

[13] C. Qin, Z. He, H. Yao, F. Cao, L. Gao, "Visible watermark removal scheme based on reversible data hiding and image inpainting", Signal Processing: Image Communication, Vol. 60, 2018, pp. 160-172.

[14] S. Singh and Y. Sharma, "A Review on DNA based Cryptography for Data hiding", Proceedings of the International Conference on Intelligent Sustainable Systems, Palladam, India, 21-22 February 2019, pp. 282-285.

[15] A. Chatterjee, S. K. Pati, "Data Hiding with Digital Authentication in Spatial Domain Image Steganography", Computational Intelligence in Pattern Recognition, Springer, 2020, pp. 897-907.

[16] Y. Ma, Z. Wang, H. Yang, L. Yang, "Artificial intelligence applications in the development of autonomous vehicles: a survey", IEEE/CAA Journal of Automatica Sinica, Vol. 7, No. 2, 2020, pp. 315-329.

[17] B. T. Atiyha, S. Aljabbar, A. Ali, A. Jaber, "An improved cost estimation for unit commitment using back propagation algorithm", Malaysian Journal of Fundamental and Applied Sciences, Vol. 15, No. 2, 2019, pp. 243-248.

[18] M. I. Fazal, M. E. Patel, J. Tye, Y. Gupta, "The past, present and future role of artificial intelligence in imaging", European Journal of Radiology, Vol. 105, 2018, pp. 246-250.

[19] D. Božić-Štulić, M. Braović, D. Stipaničev, "Deep learning based approach for optic disc and optic cup semantic segmentation for glaucoma analysis in retinal fundus images", International Journal of Electrical and Computer Engineering Systems, Vol. 11, No. 2, 2020, pp. 111-120.

[20] D. Velasco-Montero, J. Fernández-Berni, R. Carmona-Galán, Á. Rodríguez-Vázquez, "Performance assessment of deep learning frameworks through metrics of CPU hardware exploitation on an embedded platform", International journal of electrical and computer engineering systems, Vol. 11, No. 1, 2020, pp. 1-11.

[21] A. M. Fadhil, "Bit inverting map method for improved steganography scheme", Universiti Teknologi Malaysia, 2016, PhD thesis.

[22] M. Chaumont, "Deep learning in steganography and steganalysis", Digital Media Steganography, Elsevier, 2020, pp. 321-349.

[23] G. Sulong, A. Mohammedali, "Recognition of human activities from still image using novel Classifier", Journal of Theoretical and Applied Information Technology, Vol. 71, No. 1, 2015.

[24] Y. Zou, G. Zhang, L. Liu, "Research on image steganography analysis based on deep learning", Journal of Visual Communication and Image Representation, Vol. 60, 2019, pp. 266-275.

[25] H. Ruiz, M. Chaumont, M. Yedroudj, A. O. Amara, F. Comby, G. Subsol, "Analysis of the Scalability of a Deep-Learning Network for Steganography "Into the Wild"", Proceedings of the International Conference on Pattern Recognition, 2021, pp. 439-452.

[26] A. Mohammedali, "Human Activities Recognition in Still Image", LAP LAMBERT, Tech Science Press, 2016.

[27] J. Yang, K. Liu, X. Kang, E. K. Wong, Y.-Q. Shi, "Spatial Image Steganography Based on Generative Adversarial Network", arXiv:1804.07939, 2018.

[28] J. Ye, J. Ni, Y. Yi, "Deep Learning Hierarchical Representations for Image Steganalysis", IEEE Transactions on Information Forensics and Security, Vol. 12, No. 11, 2017, pp. 2545-2557.

[29] C. Zhang, C. Lin, P. Benz, K. Chen, W. Zhang, I. S. Kweon, "A Brief Survey on Deep Learning Based Data Hiding, Steganography and Watermarking", arXiv:2103.01607, 2021.

[30] B. Ray, S. Mukhopadhyay, S. Hossain, S. K. Ghosal, R. Sarkar, "Image steganography using deep learning based edge detection", Multimedia Tools and Applications, Vol. 80, No. 24, 2021, pp. 33475-33503.

[31] D. R. I. M. Setiadi, "PSNR vs SSIM: imperceptibility quality assessment for image steganography", Multimedia Tools and Applications, Vol. 80, No. 6, 2021, pp. 8423-8444.

[32] U. Pilania, P. Gupta, "Analysis and implementation of IWT-SVD scheme for video steganography", Micro-Electronics and Telecommunication Engineering, Springer, 2020, pp. 153-162.

[33] M. K. Oudah, A. N. Abed, R. S Khudhair, S. M. Kaleefah, "Improvement of Image Steganography Using Discrete Wavelet Transform", Engineering and Technology Journal, Vol. 38, No. 1, 2020, pp. 83-87.

[34] Q. Li et al., "A Novel Grayscale Image Steganography Scheme Based on Chaos Encryption and Generative Adversarial Networks", IEEE Access, Vol. 8, 2020, pp. 168166-168176.

[35] M. Kumar, S. Kumar, H. Nagar, "Enhanced Text and Image Security Using Combination of DCT Steganography, XOR Embedding and Arnold Transform", Design Engineering, 2021, pp. 732-739.

[36] A. Y. Hindi, M. O. Dwairi, Z. A. AlQadi, Technology, "A Novel Technique for Data Steganography", Engineering, Technology & Applied Science Research, Vol. 9, No. 6, 2019, pp. 4942-4945.

[37] S. Almutairi, A. Gutub, M. Al-Ghamdi, "Image Steganography to Facilitate Online Students Account System", Review of Business and Technology Research, Vol. 16, No. 2, 2019, pp. 43-49.

# An Optimized Quadratic Support Vector Machine for EEG Based Brain Computer Interface

Original Scientific Paper

**Omar N. Maher**
Mansoura University,
Computers and Control Systems Engineering Dept.,
Faculty of Engineering,
Mansora, Dakahlia, Egypt
o.elbklawi@gmail.com

**Amira Y. Haikal**
Mansoura University,
Computers and Control Systems Engineering Dept.,
Faculty of Engineering,
Mansora, Dakahlia, Egypt
amirayh@mans.edu.eg

**Mostafa A. Elhosseini**
Mansoura University,
Computers and Control Systems Engineering Dept.,
Faculty of Engineering, Mansora, Dakahlia, Egypt
Taibah University,
College of Computer Science and Engineering in
Yanbu, Yanbu, Madinah, Saudi Arabia
melhosseini@mans.edu.eg

**Mahmoud Saafan**
Mansoura University,
Computers and Control Systems Engineering Dept.,
Faculty of Engineering, Mansora, Dakahlia, Egypt
Saafan2007@mans.edu.eg

***Abstract*** *– The Brain Computer Interface (BCI) has a great impact on mankind. Many researchers have been trying to employ different classifiers to figure out the human brain's thoughts accurately. In order to overcome the poor performance of a single classifier, some researchers used a combined classifier. Others delete redundant information in some channels before applying the classifier as they thought it might reduce the accuracy of the classifier. BCI helps clinicians to learn more about brain problems and disabilities such as stroke to use in recovery. The main objective of this paper is to propose an optimized High-Performance Support Vector Machines (SVM) based classifier (HPSVM-BCI) using the SelectKBest (SKB). In the proposed HPSVM-BCI, the SKB algorithm is used to select the features of the BCI competition III Dataset IVa subjects. Then, to classify the prepared data from the previous phase, SVM with Quadratic kernel (QSVM) were used in the second phase. As well as enhancing the mean accuracy of the dataset, HPSVM-BCI reduces the computational cost and computational time. A major objective of this research is to improve the classification of the BCI dataset. Furthermore, decreased feature count translates to fewer electrodes, a factor that reduces the risk to the human brain. Comparative studies have been conducted with recent models using the same dataset. The results obtained from the study show that HPSVM-BCI has the highest average accuracy, with 99.24% for each subject with 40 channels only.*

***Keywords****: brain computer interface, classification, quadratic support vector machine, feature selection, SelectKBest.*

## 1. INTRODUCTION

Brain Computer Interface (BCI) is a computerized system that can cooperate between the signals created by the human brain's thoughts and the computer [1]. The incorporating signals developed into actions. BCI collects and transmits electrical signals used in controlling electrical wheel cheers for disabled people; it also helps clinicians learn more about brain problems and illnesses such as stroke to use in recovery [2]. BCI comes in three types; Invasive, which injects electrodes into the grey matter; partially Invasive, in which electrodes are implanted in the brain surface. Non-Invasive one comes in a wearable device full of external sensors and electrodes and eases to communicate with computers. Many Competitions have appeared in this field, and all aim to find out the human brain's thoughts with high accuracy [3]. BCI competition III dataset IVa is one of the most common datasets subjected to extensive study by researchers recently.

Many recent studies have been applied in the BCI field. Amin Hekmatmanesh et al.[4] present a literature review that discusses brain-controlled vehicles. The study shows that electroencephalogram (EEG) signals are used to detect brain signals from the motor cortex area. Different Artificial Intelligence (AI) optimization algorithms are applied then to classify EEG signals. The biomedical signals are then used to control vehicles. R. Agarwal et al.[5] present a literature review that discusses human emotions and how to classify them using EEG signals and different datasets. The study discusses different classification accuracies according to different brain regions. S. Sodagudi et al.[6] proposed a new hybrid method to classify EEG signal data. The hybrid method consists of two stages. First, the Kernel Extreme Learning-based Multi-Layer Perceptron (KEL MLP) was used to extract brain activity features. Then Bayesian Quadratic Discriminant Transfer Neural Network (BQDTNN) was used as a classification technique. W. Al-Salman et al.[7] Constructing a new method to classify the six sleep stages using EEG signals. The method consists of using Discrete Wavelet Transform (DWT) to analyze EEG signals and extract brain wave features. Then, the extracted features were applied to Least Square Support Vector Machine LS-SVM to classify sleep stages. C. Wang et al.[8] Applied a new method to enhance classification accuracy by using Shannon Complex Wavelets (SCW) with Convolutional Neural Networks (CNN). The method consists of three stages. First, EEG signals have been recorded. Then SCW is used to calculate the time-frequency matrix. Finally, CNN was used to classify the BCI data.

The feature selection algorithm filters out unnecessary data or redundant features and chooses a subset of specific features or variables that lead to better performance in classification accuracy and training time. Feature selection methods are divided into three main categories: filter method, wrapper method, and embedded method [9], [10]. The filter methods are known to be the quickest in execution but imprecise. The wrapper method uses a computational model that rates subsets based on the miss classification rate. Embedded methods figure out which features contribute best to the model during the construction process. SelectKBest (SKB) feature selection algorithm is univariate. It uses different univariate statistical tests such as (Analysis Of Variance (ANOVA) F-value, Chi-square, and mutual information methods) to select the best features from the dataset [11].

Classification algorithms are accustomed to categorizing data into a class or category. Classification comes into three types: binary classification, multiclass classification, and multilabel classification. Binary classification algorithms are used to classify datasets that have only two classes, the normal state usually called "class 0" and the abnormal state usually called "class 1". Multiclass classification algorithms are used to classify datasets that have more than two classes. Many algorithms used for binary classification can be used for multiclass classification. Multilabel classification algorithms are used to classify datasets that have two or more classes, where each input might have one or more class labels predicted. SVM is one of the most common binary classifiers and it was proposed first by Vapnik [12]. It aims to find the best separable line that can divide the data into two groups. SVM work very well with linear data [13]. However, if the data is non-linear; a kernel function must be added to SVM such as a Quadratic kernel, Cubic kernel, and Gaussian kernel.

The main contribution of this paper is summarized in applying a modern feature selection method (SKB) on the dataset to reduce the unnecessary channels, Then, training selected features with a Quadratic SVM (QSVM) classifier. The proposed approach decreases the computational cost and time needed to train BCI datasets and predict the class. This paper has been organized as follows: Section 2 covers the literature studies related to this work. Section 3 declares the used dataset, and illustrates the presented feature selection, classification algorithms, the structure of the proposed approach, and the utilized performance metrics. Section 4 contains simulation results for the experiment as well as conducting a comparative analysis. Finally, Section 5 introduces the conclusions.

## 2. RELATED WORK

Before presenting the proposed approach, a group of previous studies that use different algorithms to classify BCI competition III dataset IVa has been summarized.

Sahar Selim et al. [14] proposed a method consisting of Common Spatial Pattern as feature extraction, Attractor Metagene (AM) with Bat optimization Algorithm (BA) as feature selection, and SVM has been used as a classifier (CSP\AM-BA-SVM). Finally, This hybrid algorithm obtains average classification accuracy of 85% with few EEG channels, but that requires large computational time.

Amardeep Singh et al. [15] proposed a Symmetric Positive Definite (SPD) as matrices based on the motor imagery classification method. SPD performed very well with a small sample set. Their method was applied to BCI Competition III dataset IVa and obtained an average accuracy of 87.21% of the subjects. However, this method obtained better ac-curacy just in a small sample set.

Yongkoo Park et al. [16] proposed a method that extracts features using Filter Bank CSP (FBCSP) and then selects the optimal channels which include the best features. Finally, the selected features were classified by LS-SVM. Their method was applied to BCI competition III dataset IVa and the average accuracy of the 5-subjects was 86.73%. However, one of the limitations of this method was badly performing with multiclass data.

Kais Belwafi et al. [17] proposed an algorithm Dynamic Self-Adaptive Algorithm (DSAA), which depends on the LS method. The study applied to BCI competition III dataset IVa with an average of 81.95%. The filtering method performed well only with online systems.

Amin Hekmatmanesh et al. [18] proposed an improved CSP algorithm to recognize and classify BCI Competition III dataset IVa data by aggregating four algorithms. The algorithms are Discriminative FBCSP with the Discriminative Sensitive Learning Vector Quantization (DFBCSP-DSLVQ), the Soft margin SVM (SSVM) classifier, and the Generalized Radial Bases Functions (GRBF) to create a method called DFBCSP DSLVQ SSVM GRBF with an average accuracy of 92.70%. However, for multi-classes, the error ratio rises when using this method.

Considering the feasibility of classifying datasets through efficient evaluation, we can see several serious limitations of the results. These problems can be summarized as; high computational time for the algorithm to be executed, and choosing bad channels when it contains a large amount of common noise. Moreover, feature selection algorithms may not perform well with multiclass motor imagery tasks. However, the hybrid algorithm successfully overcomes two of these challenges by classifying the dataset with high performance in an adequate training time. (Table 1) summarizes previously discussed algorithms focusing on various pros and limitations.

**Table 1.** Related work pros and Limitations summary

| Author | Mean accuracy | Pros | Limitations |
|---|---|---|---|
| Sahar Selim et al. [14] | 85% | Use only 0.1 of EEG channels with high accuracy | Requires considerable computational time |
| Amardeep Singh et al. [15] | 87.21% | Performs well with small a sample set | Performs badly with large sets |
| Yongkoo Park et al. [16] | 86.73% | Performs well with binary classes | Badly performing with multiclass data |
| Kais Belwafi et al. [17] | 81.95% | The filtering method performs well with online systems. | The filtering method performs badly only with offline systems. |
| Amin Hekmatmanesh et al. [18] | 92.70% | Performs well with binary classes | For multi-classes, the error ratio rises when using this method. |

## 3. MATERIALS AND METHODS

Facing the previous difficulties of classifying datasets led to considering alternatives to achieve more accuracy with acceptable computational time. Therefore, by combining two algorithms, we found out that, after all, creating a union can boost both processes' strengths and overcome both weaknesses. The two main components of this union are QSVM and SKB algorithms.

### 3.1. BCI COMPETITION III DATASET IVA

BCI Competition III Dataset IVa has been collected from five healthy subjects. Those subjects were sat in a comfortable chair with arms placed on comfortable armrests [19]. The Data set include data from the four initial sessions with no feedback. The subject sat with open eyes opposite a screen that presents a letter for 3.5 seconds, as declared in (Fig. 1).

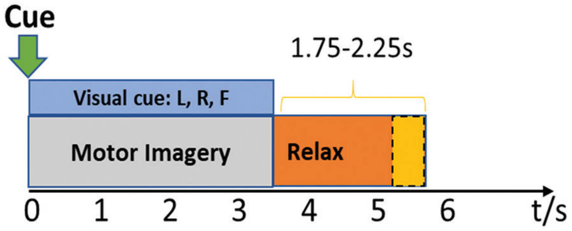


**Fig. 1.** Dataset timeline

Three letters equal three motor imageries the subject must perform [20]. For example, where, (L, R) left or Right hand and (F) foot. The subject relaxed for (1.75-2.25) seconds randomly between performed tasks. The dataset consists of continuous signals of 118 EEG channels according to the 10/20 system as shown in (Fig. 2) and markers that indicate the time points of 280 cues for each of the 5 subjects (aa, al, av, aw, ay). The data was recorded using Ag/AgCl electrode cap.

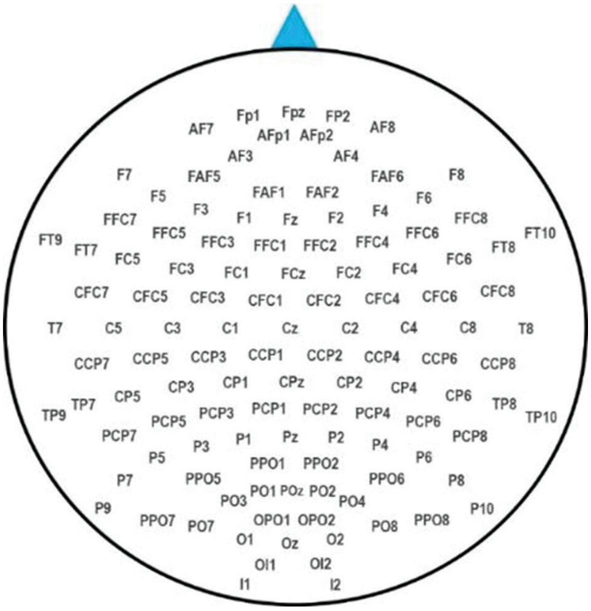


**Fig. 2.** 118 EEG channels

### 3.2. ALGORITHMS INVOLVED

This section discussed the paper's algorithms from a theoretical view. The algorithms that have been used in the hybrid approach are SKB as a feature selection algorithm and QSVM as a classifier.

#### 3.2.1. Feature selection algorithm

Ag/AgCl electrode cap covers 118 channels in the human brain as declared in (Figure 2). While using BCI Competition III Dataset IVa, we noticed that some channels contain redundant information and others have only noisy information. SKB algorithm has been used to remove redundant and noisy channels according to the chi-square value. SKB keeps only 40-channel for each subject.

SKB Algorithm is a modern algorithm used in the 20th century. SKB chooses the powerful features by ranking the whole features according to statistical tests such as (ANOVA) F-value, Chi-square,…etc.) [21], [22]. Then select the best features that represent the data. This study used the chi-square test-based method to select the best features. Chi-square is given by:

$$X_c^2 = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i} \tag{1}$$

Where n is the number of features, c is the freedom degree, $O_i$ is the observed values and $E_i$ is the expected values if there is no association between the two events [23]. The Chi-square test is used to test how much two events depend on each other. From (Equation 1), we can conclude that if there are two independent features, the observed count and expected count is very close values, leading to a small Chi-square value. The greater the correlation of features, the higher the value of Chi-square in promoting the selection of these features. (Algorithm 1) declare SKB steps in brief.

**Algorithm 1:** SKB

```
for each Subject
  Select the **Score Function SF // SF = Chi-square**
    Apply **Chi-square** statistical equation
    $X_c^2 = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i}$
    Rank All Features due to Chi-square value
    Select the number of K
  Select only the Best features according to K-value
end
```

#### 3.2.2. Classification algorithm

Classification algorithms are accustomed to categorizing data into a class or category. SVM is one of the most common classifiers. SVM Separates the two classes based on the distance between the objects and the hyperplane (Distance Margin). SVM works with a technique called the kernel functions that convert low dimensional input space to a higher dimensional space [12]. Linear SVM can classify linear data only, but if we have non-linear data, we should add a kernel with SVM. The results show that the quadratic kernel is the best one with BCI competition III dataset IVa.

QSVM classifies the data into two groups with hyper-plane equation as declared in (Equation 2).

$$f(X) = \frac{1}{2} X^T W X + b^T X + c \tag{2}$$

Where $W$ is a weight vector, $X$ is the input vector, $b$ is bias and $T$ is the transpose. As shown in (Figure 3), QSVM has three decision boundaries [24]; the group of nodes lies on the hyper-plane described in (Equation 3), the group of nodes lies in the positive class described in (Equation 4) and the group of nodes lies in the negative class described with (Equation 5).

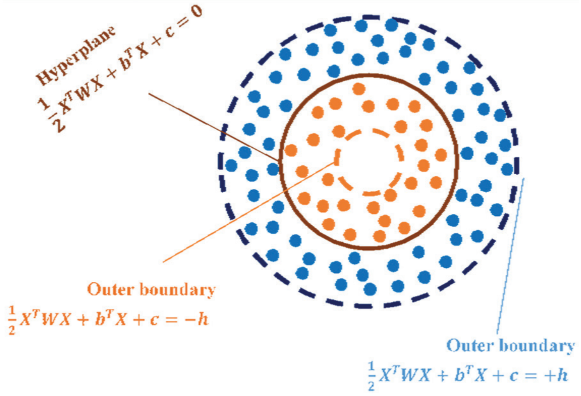

Hyperplane
$\frac{1}{2} X^T W X + b^T X + c = 0$
Outer boundary
$\frac{1}{2} X^T W X + b^T X + c = -h$
Outer boundary
$\frac{1}{2} X^T W X + b^T X + c = +h$

**Fig. 3.** Quadratic surface taxonomy

$$\frac{1}{2} X^T W X + b^T X + c = 0 \tag{4}$$

$$0 < \frac{1}{2} X^T W X + b^T X + c \leq +h \tag{5}$$

$$0 > \frac{1}{2} X^T W X + b^T X + c \geq -h \tag{6}$$

Where $W$ is a weight vector, $X$ is the input vector, b is the bias, $T$ is a transpose, and (-$h$, +$h$) represents the hyper-plane of the inner and outer quadratic surface.

Cross-validation is commonly used to improve model prediction in machine learning. With this technique, we start dividing each subject in the BCI dataset randomly into k parts (k-fold cross-validation) [25]. In this study, we use 5-fold cross-validation. Four parts were used as training sets and the left one was used as a testing set. This process repeats five times with different sets each time.

#### 3.2.3. HPSVM-BCI Approach

This section discusses High-Performance SVM-BCI (HPSVM-BCI) framework and the following method to implement the HPSVM-BCI approach. HPSVM-BCI framework contains the dataset subjects as we described before and its dimensions. The framework contains the classification algorithms that have been applied to the dataset subjects using 5-fold cross-validation such as Linear Discriminant (LD), Quadratic Discriminant (QD), Logistic Regression (LR), Naïve Byes (NB), Linear SVM (LSVM), QSVM, Cubic SVM (CSVM), and Deep Neural Network (DNN) as shown in (Fig. 4). Finally, the framework illustrates the performance metrics that have been used to evaluate classification algorithms and the feature selection algorithm that has been applied to the winner.

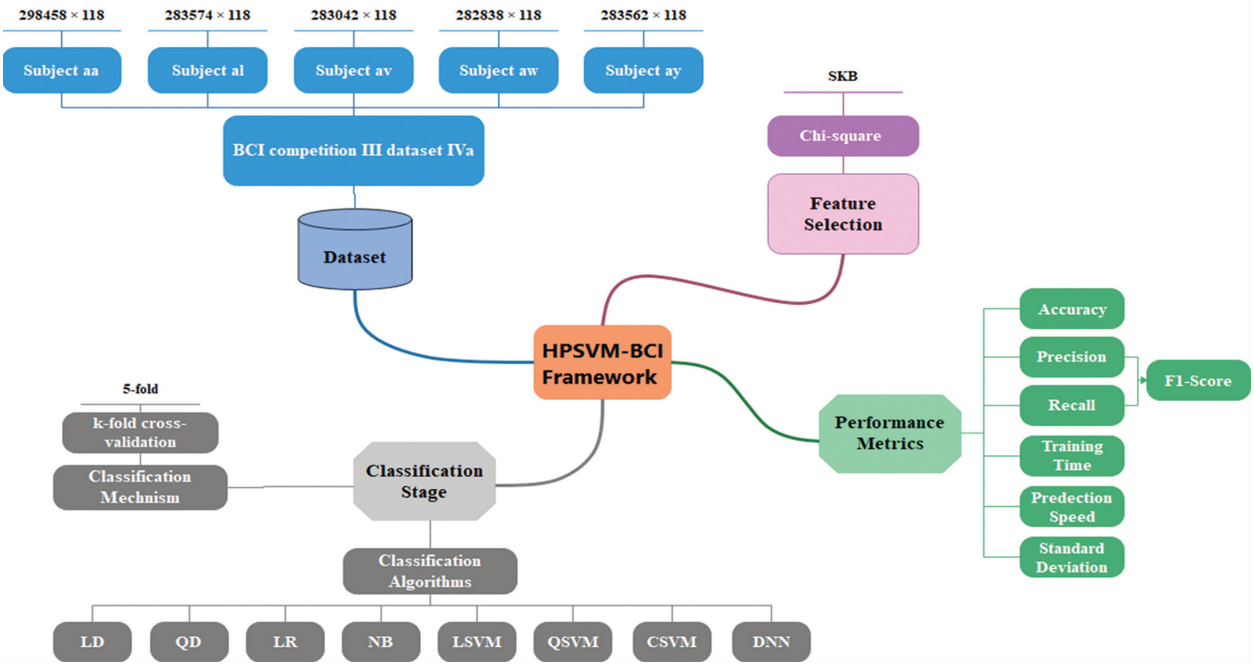


**Fig. 4.** HPSVM-BCI mechanism.

The proposed approach is a combination of both the feature selection algorithm (SKB) and the winner classification algorithm (QSVM). The HPSVM-BCI Process flow diagram is declared in (Fig. 5). SKB has been applied to the original dataset to evaluate the importance of each feature according to the Chi-square equation. The best features are selected then and a prepared dataset has been created. The prepared dataset has been subject to the classification stage. QSVM separates the prepared dataset into 5-fold cross-validation. Four parts randomly have been used as input to QSVM as training data. The last part has been used as testing data to evaluate the classifier. The classification stage has been repeated five times each with random training and testing data and prepares the data for the classification stage. The whole operation repeated for each subject on BCI competition III dataset IVa.

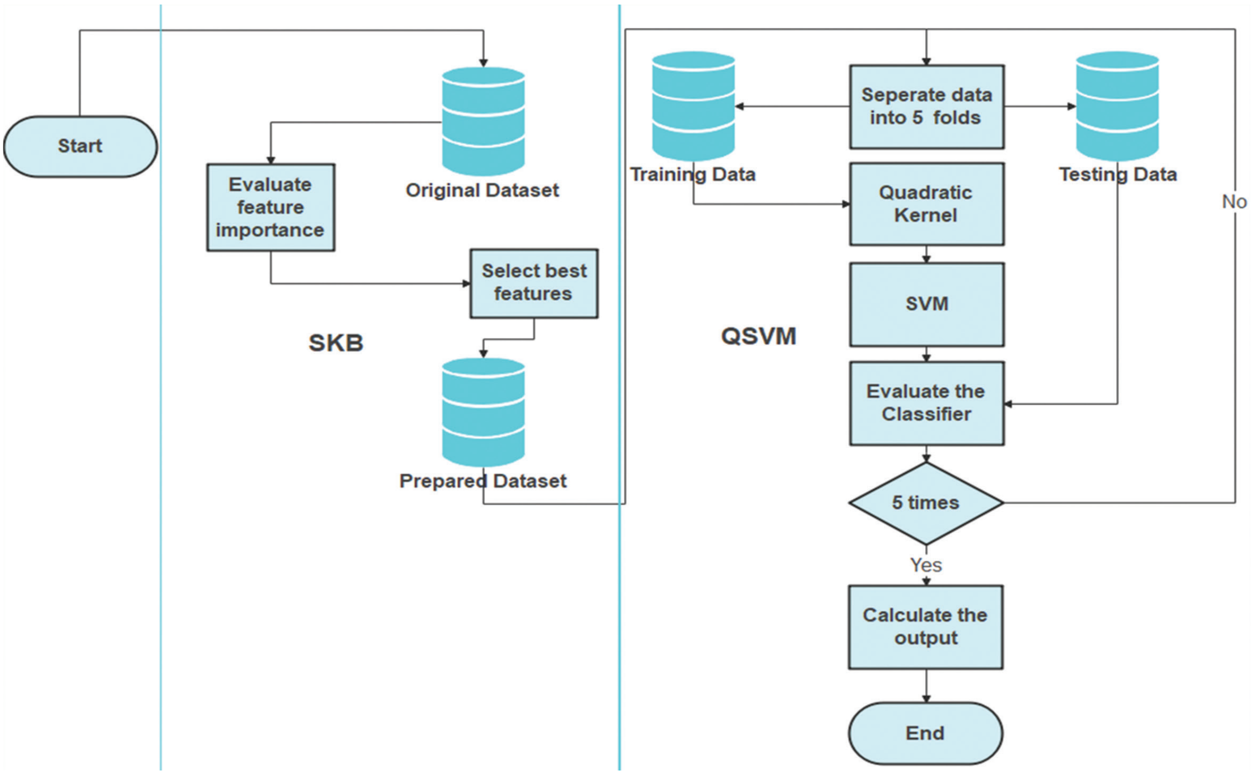


**Fig. 5.** HPSVM-BCI mechanism

## 3.3. PERFORMANCE METRICS

The computer results for this research have been evaluated according to different metrics; confusion matrix, F1 score, training time, and prediction speed.

### 3.3.1. Confusion matrix

The confusion matrix describes the effects of a forecast over a classification problem. Confusion metrics are very important metrics in evaluating classifier performance [26]. The accuracy equation is described in (Equation 6):

$$Accuracy = \frac{T_P+T_N}{T_P+T_N+F_P+F_N} \qquad (6)$$

Where $T_P$ is a True Positive, $T_N$ is a True Negative, $F_P$ is a False Positive and $F_N$ is a False Negative.

### 3.3.2. Precision and Recall

Precision is defined as the ratio between the TP and all the Positives. It also helps to measure the relevant data points. The recall is defined as the fraction of retrieved instances among all relevant instances.

$$\boldsymbol{precision = \frac{T_P}{T_P+F_P}} \qquad (7)$$

$$\boldsymbol{recall = \frac{T_P}{T_P+F_N}} \qquad (8)$$

### 3.3.3. F1 Score

F1 Score aims for a balance between Precision and Recall [27], and there are many negative classified cases.

The Precision and Recall equations were described in (Equations 7, and 8). F1 Score equation:

$$F1 = \frac{2}{(1/Precision + 1/Recall)} \qquad (9)$$

### 3.3.4. Training Time

It is the whole time that the model needs to be trained.

### 3.3.5 Prediction Speed

It is the number of observations that the AI model can deliver every second.

## 4. COMPUTER SIMULATIONS AND RESULTS

This section consists of three parts; first, displays the results of applying several algorithms on the dataset and obtains the winner. Second, the winner algorithm has been compared with the suggested approach HPSVM-BCI. Finally, a comparison between the proposed approach against the related work.

This experiment discusses several algorithms that have been executed with 5k-fold cross-validation on BCI Competition III Dataset Iva as illustrated in (Table 2). Performance metrics have been calculated 50 times and the average value is calculated for each metric. Standard deviation (SD) was also calculated for classification accuracy to show the algorithm's stability.

QSVM proved its ability in dealing with high-complexity data, such as Electroencephalography datasets, but it takes a huge training time. Accordingly, we suggest adding SKB to select the most relevant features of datasets. (Table 3) shows that HPSVM-BCI achieved higher average accuracy and average F1-Score than QSVM except

with subject "al" and reduces the average training time from (127,341 to 49,642) sec as shown in (Figure 6. a) this means that training time decreases by 250%. The mean prediction speed increases as well from (8,536 to 16,024.6) obs/sec as shown in (Fig. 6. b).

**Table 2.** The average values of performance metrics for several algorithms

| Method | Performance Metrics | Subjects | | | | | Mean |
|---|---|---|---|---|---|---|---|
| | | aa | al | av | aw | ay | |
| LD | Accuracy (%) | 78.82 | 76.21 | 83.87 | 87.20 | 89 | 83 |
| | F1 Score (%) | 74.63 | 76 | 76.52 | 87.56 | 90.50 | 81.99 |
| | Training Time (sec) | 54 | 65 | 12 | 17 | 4 | 30.40 |
| | Prediction Speed (obs/sec) | 75000 | 84000 | 130000 | 62000 | 110000 | 92200 |
| | SD (±%) | 0.16 | 0.18 | 0.15 | 0.19 | 0.15 | 0.17 |
| QD | Accuracy (%) | 87.77 | 88.92 | 87.91 | 91 | 90.88 | 89.29 |
| | F1 Score (%) | 87.14 | 88.66 | 85.52 | 91.13 | 91.25 | 88.74 |
| | Training Time (sec) | **53** | **64** | **9** | **16** | **4** | **29.20** |
| | Prediction Speed (obs/sec) | 75000 | 81000 | 140000 | 62000 | 110000 | 93600 |
| | SD (±%) | 0.12 | 0.12 | 0.11 | 0.11 | 0.11 | 0.12 |
| LR | Accuracy (%) | 79.12 | 76.11 | 87.83 | 92.65 | 97.09 | 86.56 |
| | F1 Score (%) | 74.86 | 76.12 | 82.41 | 92.91 | 97.90 | 84.84 |
| | Training Time (sec) | 278 | 324 | 177 | 75 | 16 | 174 |
| | Prediction Speed (obs/sec) | 110000 | 100000 | 99000 | 100000 | 170000 | 115800 |
| | SD (±%) | 0.23 | 0.22 | 0.21 | 0.23 | 0.20 | 0.22 |

|  |  |  |  |  |  |  |  |
|---|---|---|---|---|---|---|---|
| **NB** | Accuracy (%) | 50.92 | 51.83 | 45.72 | 55.55 | 48.08 | 50.42 |
|  | F1 Score (%) | 60.31 | 36.62 | 55.80 | 63.80 | 47.23 | 52.75 |
|  | Training Time (sec) | 78 | 92 | 33 | 20 | 3 | 45.20 |
|  | Prediction Speed (obs/sec) | 110000 | **130000** | **150000** | 110000 | **180000** | **136000** |
|  | SD (±%) | 0.14 | 0.15 | 0.14 | 0.15 | 0.14 | 0.14 |
| **LSVM** | Accuracy (%) | 80 | 77 | 87.42 | 92.51 | 96.37 | 86.66 |
|  | F1 Score (%) | 76.1 | 76.5 | 81.2 | 92.8 | 97.4 | 84.8 |
|  | Training Time (sec) | 35510 | 67507 | 8322 | 925 | 83 | 22469 |
|  | Prediction Speed (obs/sec) | 210 | 96 | 450 | 1600 | 9500 | 2371 |
|  | SD (±%) | 0.078 | 0.09 | 0.087 | 0.087 | 0.127 | 0.094 |
| **QSVM** | Accuracy (%) | **99.12** | 98.91 | **99.30** | 99. 41 | 99.41 | **99.20** |
|  | F1 Score (%) | **98.88** | 99 | **98.91** | 99.44 | 99.68 | **99.18** |
|  | Training Time (sec) | 35291 | 86898 | 4086 | 991 | 75 | 25468 |
|  | Prediction Speed (obs/sec) | 1200 | 580 | 1900 | 11000 | 28000 | 8536 |
|  | SD (±%) | 0.06 | **0.06** | 0.09 | 0.11 | 0.11 | 0.09 |
| **CSVM** | Accuracy (%) | 51.32 | **99.61** | 59.27 | **99.52** | **99.72** | 81.88 |
|  | F1 Score (%) | 41.21 | **99.61** | 25.83 | **99.46** | **99.82** | 73.18 |
|  | Training Time (sec) | 16722 | 85728 | 7936 | 1362 | 93 | 22368 |
|  | Prediction Speed (obs/sec) | **180000** | 1700 | 120000 | 18000 | 34000 | 70740 |
|  | SD (±%) | 0.12 | 0.08 | 0.12 | 0.24 | 0.19 | 0.15 |
| **DNN** | Accuracy (%) | 76.24 | 92.36 | 64.57 | 87.68 | 90.83 | 82.34 |
|  | F1 Score (%) | 78.24 | 92.42 | 67.57 | 89.62 | 91.83 | 83.94 |
|  | Training Time (sec) | 1199 | 990 | 115 | 244 | 212 | 552 |
|  | Prediction Speed (obs/sec) | **115400** | 125200 | 139300 | **118500** | 145100 | 128700 |
|  | SD (±%) | **0.06** | 0.06 | **0.08** | **0.07** | **0.07** | **0.07** |

**Table 3.** The average values of performance metrics for QSVM vs HPSVM-BCI

| **Subject** |  |  | **QSVM** |  |  | **Prediction Speed (obs/sec)** |
|---|---|---|---|---|---|---|
|  | **Precision (%)** | **Recall (%)** | **F1-Score (%)** | **Accuracy (%)** | **Training Time (sec)** |  |
| **aa** | **98.90** | 98.90 | 98.9 | 99.12 | 35291 | 1200 |
| **al** | **99.58** | 98.33 | **99.01** | **98.91** | 86898 | 580 |
| **av** | 99.11 | 98.70 | 98.90 | 99.30 | 4086 | 1900 |
| **aw** | **99.52** | 99.31 | 99.34 | 99.41 | 991 | 11000 |
| **ay** | 99.52 | **99.71** | 99.60 | 99.41 | 75 | 28000 |
| **Subject** |  |  | **HPSVM-BCI** |  |  | **Prediction Speed (obs/sec)** |
|  | **Precision (%)** | **Recall (%)** | **F1-Score (%)** | **Accuracy (%)** | **Training Time (sec)** |  |
| **aa** | 98.83 | **99.12** | **90.00** | **99.20** | **17104** | **2190** |
| **al** | 98.31 | **99.19** | 98.71 | 98.70 | **30017** | **1302** |
| **av** | **99.30** | **98.81** | **99** | **99.41** | **2042** | **3781** |
| **aw** | 99.40 | **99.60** | **99.50** | **99.39** | **439** | **20350** |
| **ay** | **99.77** | 99.52 | **99.71** | **99.50** | **40** | **52500** |

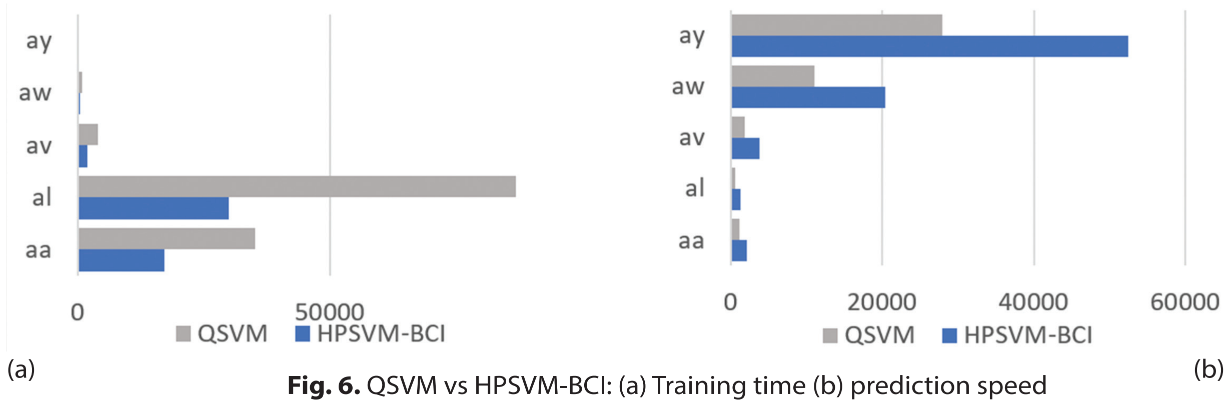


**Fig. 6.** QSVM vs HPSVM-BCI: (a) Training time (b) prediction speed

The suggested method HPSVM-BCI has been compared with literature studies that execute different types of classifiers at BCI Competition III dataset IVa. (Table 4) displays that HPSVM-BCI overwhelms the entire literature studies for aa, av, aw, and ay subjects by 5.60%, 17.43%, 5.73%, and 3.43, respectively. However, in the 'al' subject, SPD and CSP\AM-BA-SVM overcome the proposed method by only 1.30%. Accordingly, the mean accuracy for HPSVM-BCI is the best accuracy with 99.18%.

**Table 4.** The average values of performance metrics for QSVM vs HPSVM-BCI

| Author | Method | Subjects' average accuracy | | | | | Mean |
|---|---|---|---|---|---|---|---|
| | | aa | al | av | aw | ay | |
| Amin Hekmatmanesh et al. [18] | DFBCSP DSLVQ<br>SSVM GRBF | 93.51% | 98.59% | 81.82% | 93.63% | 96.14% | 92.72% |
| Amardeep Singh et al. [15] | SPD | 81.31% | **100%** | 76.46% | 87.13% | 91.29% | 87.22% |
| Yongkoo Park et al. [16] | FBCSP + LS-SVM | 92.92% | 89.27% | 71.39% | 83% | 94.14% | 86.71% |
| Kais Belwafi et al. [17] | DSAA | 69 .55% | 96.38% | 60.52% | 70.53% | 78.60% | 82% |
| Sahar Selim et al. [14] | CSP\AM-BA-SVM | 86.63% | **100%** | 66.78% | 90.60% | 81% | 85% |
| | **Proposed Method** | **99.20%** | 98.70% | **99.41%** | **99. 39%** | 99.50% | **99.24%** |

## 5. CONCLUSIONS

The goal of BCI is to integrate machine intelligence with the brain via electrodes. The field is now flooded with competitions that aim to uncover the human brain's thinking with high accuracy. One of the most widely used datasets for BCI competition III Dataset IVa has been extensively investigated by researchers. We aim to improve the classifications of the BCI dataset in this study. This can be achieved by developing a new approach HPSVM-BCI, which features two steps; selecting the best features and classifying the data. In SKB, in the first step, the features are sorted by Chi-square value, and then the best features are selected for classification by QSVM. After that, the quadratic function is used to determine the best surface for splitting into two classes. This improves the mean accuracy of data and reduces computational time, training time, and prediction time for HPSVM-BCI. As a result, the number of electrodes that reduce the risk of human brain injury is also decreasing.

## 6. REFERENCES:

[1] K. Venkatachalam, A. Devipriya, J. Maniraj, M. Sivaram, A. Ambikapathy, S. A. Iraj, "A novel method of motor imagery classification using eeg signal", Artificial Intelligence in Medicine, Vol. 103, 2020, p. 101787.

[2] M. Miao, W. Zhang, W. Hu, R. Wang, "An adaptive multi-domain feature joint optimization framework based on composite kernels and ant colony optimization for motor imagery EEG classification", Biomedical Signal Processing and Control, Vol. 61, 2020, p. 101994.

[3] H. Göksu, "BCI oriented EEG analysis using log energy entropy of wavelet packets", Biomedical Signal Processing and Control, Vol. 44, 2018, pp. 101-109.

[4] A. Hekmatmanesh, P. H. J. Nardelli, H. Handroos, "Review of the state-of-the-art of brain-controlled vehicles", IEEE Access, Vol. 9, 2021, pp. 110173-110193.

[5] R. Agarwal, M. Andujar, S. Canavan, "Classification of emotions using EEG activity associated with different areas of the brain", Pattern Recognition Letters, Vol. 162, 2022, pp. 71-80.

[6] S. Sodagudi, S. Manda, B. Smitha, N. Chaitanya, M. A. Ahmed, N. Deb, "EEG signal processing by feature extraction and classification based on biomedical deep learning architecture with wireless communication", Optik, Vol. 270, 2022, p. 170037.

[7] W. Al-Salman, Y. Li, A. Y. Oudah, S. Almaged, "Sleep stage classification in EEG signals using the clustering approach based probability distribution features coupled with classification algorithms", Neuroscience Research, 2022. (in press)

[8] C. Wang, Y. Wu, C. Wang, Y. Zhu, C. Wang, Y. Niu, Z. Shao, X. Gao, Z. Zhao, Y. Yu., "MI-EEG classification using Shannon complex wavelet and convolutional neural networks", Applied Soft Computing, Vol. 130, 2022, p. 109685.

[9] J. S. Kirar, R. K. Agrawal, "Composite kernel support vector machine based performance enhancement of brain computer interface in conjunction with spatial filter", Biomedical Signal Processing and Control, Vol. 33, 2017, pp. 151-160.

[10] J. Chen, P. K. Kudjo, S. Mensah, S. A. Brown, G. Akorfu, “An automatic software vulnerability classification framework using term frequency-inverse gravity moment and feature selection”, Journal of Systems and Software, Vol. 167, 2020, p. 110616.

[11] M. S. Zulfiker, N. Kabir, A. A. Biswas, T. Nazneen, M. S. Uddin, “An in-depth analysis of machine learning approaches to predict depression”, Current Research in Behavioral Sciences, Vol. 2, 2021, p. 100044.

[12] V. Vapnik, "The nature of statistical learning theory", Springer Science & Business Media, 2013.

[13] D. Madroñal, R. Lazcano, R. Salvador, H. Fabelo, S. Ortega, G. M. Callicó, E. Juarez, C. Sanz., “SVM-based real-time hyperspectral image classifier on a manycore architecture”, Journal of Systems Architecture, Vol. 80, 2017, pp. 30-40.

[14] S. Selim, M. M. Tantawi, H. A. Shedeed, A. Badr, “A CSP\AM-BA-SVM Approach for Motor Imagery BCI System”, IEEE Access, Vol. 6, 2018, pp. 49192-49208.

[15] A. Singh, S. Lal, H. W. Guesgen, “Small sample motor imagery classification using regularized Riemannian features”, IEEE Access, Vol. 7, 2019, pp. 46858-46869.

[16] Y. Park, W. Chung, “Optimal channel selection using covariance matrix and cross-combining region in EEG-based BCI”, Proceedings of the 7th International Winter Conference on Brain-Computer Interface, Gangwon, Korea, 18-20 February 2019, pp. 1-4.

[17] K. Belwafi, S. Gannouni, H. Aboalsamh, H. Mathkour, A. Belghith, “A dynamic and self-adaptive classification algorithm for motor imagery EEG signals”, Journal of Neuroscience Methods, Vol. 327, 2019, p. 108346.

[18] A. Hekmatmanesh, H. Wu, F. Jamaloo, M. Li, H. Handroos, “A combination of CSP-based method with soft margin SVM classifier and generalized RBF kernel for imagery-based brain computer interface applications”, Multimedia Tools and Applications, Vol. 79, No. 25, 2020, pp. 17521-17549.

[19] B. Blankertz et al. “Results of the BCI Competition III”, BCI Meeting, 2005, p. 30.

[20] Z. Qiu, J. Jin, H.-K. Lam, Y. Zhang, X. Wang, A. Cichocki, “Improved SFFS method for channel selection in motor imagery based BCI”, Neurocomputing, Vol. 207, 2016, pp. 519-527.

[21] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, “Scikit-learn: Machine learning in Python”, Journal of Machine Learning Research, Vol. 12, 2011, pp. 2825-2830.

[22] C. D. Whitmire, J. M. Vance, H. K. Rasheed, A. Missaoui, K. M. Rasheed, F. W. Maier, “Using machine learning and feature selection for alfalfa yield prediction”, AI, Vol. 2, No. 1, 2021, pp. 71-88.

[23] F. Thabtah, F. Kamalov, S. Hammoud, S. R. Shahamiri, “Least Loss: A simplified filter method for feature selection”, Information Sciences, Vol. 534, 2020, pp. 1-15.

[24] I. Dagher, “Quadratic kernel-free non-linear support vector machine”, Journal of Global Optimization, Vol. 41, No. 1, 2008, pp. 15-30.

[25] H. Ling, C. Qian, W. Kang, C. Liang, H. Chen, “Combination of Support Vector Machine and K-Fold cross validation to predict compressive strength of concrete in marine environment”, Construction and Building Materials, Vol. 206, 2019, pp. 355-363.

[26] M. Hasnain, M. F. Pasha, I. Ghani, M. Imran, M. Y. Alzahrani, R. Budiarto, “Evaluating trust prediction and confusion matrix measures for web services ranking”, IEEE Access, Vol. 8, 2020, pp. 90847-90861.

[27] W. Castro, J. Oblitas, M. De-La-Torre, C. Cotrina, K. Bazán, H. Avila-George, “Classification of cape gooseberry fruit according to its level of ripeness using machine learning techniques and different color spaces”, IEEE Access, Vol. 7, 2019, pp. 27389-27400.

# A Deep Learning Approach for Automated COVID-19 Detection

Original Scientific Paper

**Amanpreet Singh**
Department of Electronics and Communication Engineering
Punjabi University, Patiala, Punjab, India.
asodhi.pta@gmail.com

**Charanjit Singh**
Department of Electronics and Communication Engineering
Punjabi University, Patiala, Punjab, India.
channisingh@yahoo.com

***Abstract*** *– Nowadays, COVID-19 is a life-threatening virus for human beings, and the reason behind it is its attack on the respiratory system. A large number of cases of infection were reported with minor to no symptoms. So, detection of the disease at an earlier stage can decrease the death rate in the patients. Chest X-Rays scans can be used primarily for analyzing the infection. X-ray technology is chosen over CT scans because its equipment is readily available, results can be obtained quickly, and the process is quite affordable in terms of cost. This paper proposed a solution using a deep learning approach to detect COVID-19 infection in human lungs using Chest X-Ray scans. Here, we have used CLAHE (Contrast Limited Adaptive Histogram Equalization) to enhance the contrast of X-ray images and then Convolutional Neural Network on CLAHE processed images to improve the accuracy of the overall model.*

*Further, these scans are classified using machine learning classifiers among COVID-19 infected and normal. The proposed model is trained and validated on a publicly available COVID-19 X-ray dataset containing 15917 X-ray Images. Confusion matrices and ROC curves have been generated to analyze the model's efficiency. Training and validation graphs are developed to calculate the other parameters like validation accuracy and training Accuracy. The model's accuracy is 99.8%, which is better than its existing state-of-the-art approaches. These results show that this model is promising for physicians to classify the chest X-Rays scans of infected patients with COVID-19.*

***Keywords****: Deep Learning, Medical Images, COVID-19 Detection, X-Ray Images*

## 1. INTRODUCTION

COVID-19 is a lethal disease caused by a newly confirmed Coronavirus infection. In December 2019, it was first transmitted to humans, and it would be transmitted from person to person through droplets that form when a person speaks, coughs, or sneezes [1-6]. This virus primarily attacks the lungs and can harm the muscles of an infected human being.

Because of global climate change, there are so many diseases from which people are already suffering, and at the same time, the impact of coronavirus is immense. Healthcare professionals and researchers in different regions across the globe are working to find a stable solution and improve testing capacity by employing multifunctional tests to control the spread of contamination and protect themselves from the same.

Recently, RT-PCR (reverse transcriptase-polymerase chain reaction) diagnostics have proved effective in detecting infection. Though, this method has the disadvantages of a longer detection time and a slower virus detection speed. [7, 8]. Many scholars are working globally to overcome the limits of RT-PCR tests and expand the diagnosis process of COVID-19. Deep learning algorithms with CNN are also used to diagnose viruses through image classification techniques. According to the WHO recommendations, chest X-rays effectively diagnose the clinical symptoms of infected people who have been improved [9].

Recent studies show that CNNs are very useful and have a perfect effect in identifying COVID-19 through image processing. CNN can be a multi-layer neural network that recognizes image patterns with the help of different image pre-processing tools. Some CNN Models like Resnet50 [10], AlexNet [11], VGG16 [12], and VGG19 are also available and perform well in classifying COVID-19 chest X-Rays scans.

Here we have proposed a fusion of CLAHE (Contrast Limited Adaptive Histogram Equalization) along with the CNN (Convolutional Neural Network) to increase the model's accuracy. CLAHE is the developed version of the adaptive histogram equation used to improve the contrast of the images by performing a stretching-out mechanism on frequent intensity values of the image. In the proposed method, we have used CLAHE to pre-process the medical images, and then processed images are provided to the CNN network for classification. A detailed explanation of CLAHE and the CNN network is provided in the upcoming sections. The main contributions of the work are:

- The work presented here provides an improved Deep learning model trained to spot COVID-19 contamination using chest X-Ray images and classify them into infected and normal subjects. This method proposes a new fusion of the CLAHE (Contrast Limited Adaptive Histogram Equalization) CNN classification. In addition, the coloring of indexed images is used in pre-processing to enhance the accuracy of this model.
- We have used data augmentation strategies for COVID-19 discovery to avoid overfitting issues.
- A publicly available large X-ray image dataset is used in this work, showing better accuracy and other metrics than existing methods.

## 2. RELATED LITERATURE

Over the past few months, many researchers have examined and analyzed chest X-rays using machine learning algorithms to detect the infection. Several AI learning-based approaches are available for COVID-19 detection using X-ray scans. In the healthcare area, deep learning [DL], a sub-branch of artificial intelligence, is a current and increasingly evolving CAD (Computer-Aided Design) tool to help clinicians/radiologists better predict disease. DL methods can guide practitioners in advancing the quality of COVID-19 detection.

Chowdhury et al. [13] worked on breast X-rays and created a framework, PDCOVIDNet, based on dilated parallel conventional neural network. The proposed method used small convolution in a similar stack to capture and stretch the required properties to obtain an accuracy of 96.58%.

Khan et al. [14] presented a new X-ray analytical architecture such as COVID-19 with pre-charged machine learning models such as VGG16, ResNet50, DensNet121, and VGG19, in which VGG16 and 19 have shown the best accuracy. This proposed model consists of 2 phases, such as pre-processing and data dissemination and learning transfer, and lastly indicates an accuracy of around 99.3%.

Minae et al. [15] reported wide-ranging research showing COVID-19 infection in chest X-ray imaging using four integrated models: SqueezeNet, ResNet18, ResNet50, and DensNet-121. This plan used the data expansion to create an altered variety of the COVID-19 image to raise the number of testers and ultimately achieve 90% specificity and 98% sensitivity.

Sekeroglu et al. [16] designed a prototype using different machine-learning techniques that performed 38 experiments to identify infection using high-precision X-ray imaging. Of these, he served ten experiments, five various deep learning algorithms, and 14 trials with hi-tech pre-trained systems for educational exchange. These procedures found an accuracy of 98.50%, an accuracy of 99.18%, and a sensitivity of 93.84%. They conclude that the process developed by CNN can complement the detection of COVID-19 in low-resolution images with minimal processing and no pre-processing.

Khalifa et al. [17] introduced a method of classifying coronavirus cure goals in the human brain grounded on handling type and therapeutic level by using deep learning (DL) and machine learning (ML). The processing distribution accuracy obtained by the model reaches 98.05% in comparison with other ML models, such as SVM and DT. The DCNN model lacked an accuracy rate (98.2%) compared with the DT (98.5%) for estimating the clinical trials. Broadcast models (e.g., Alexnet) were used in the study.

School et al. [18] reported a possible development of hybrid delivery methods using CNN and marine hunters for COVID-19 imaging obtained by international chest radiologists. They used the CNN design model to extract features and the competitive marine predator algorithm to select the most important images. However, scientific research has yet to determine the fusion pathway to improve the distribution and presentation of the COVID-19 image.

Mohammad Marufur Rahman et al. [19] proposed a HOG (histogram of oriented gradient) and CNN-based model for the classification of COVID-19 and Pneumonia from X-ray images and achieved an accuracy of 96.74% in image classification.

Most reports have used X-ray scanning to spot the contamination, highlighting the significance of chest X-ray scans as an accurate means for physicians and electrotherapists. Though, in some cases, the distributions do not provide the desired results due to inconsistencies in the control data and the inability to qualify from the image. To eliminate these limitations, in this research, we have proposed a combination of CLAHE and CNN to improve the system's overall accuracy.

## 3. PROPOSED METHODOLOGY

Over the past few years, multiple classifier systems have gained everybody's consideration in the domain of artificial intelligence. These systems are proved very effective in resolving many existing complications, like health care and computer vision difficulties. These systems combine different features from different models to boost the system's overall efficiency. In this proposed system, we combine CLAHE with CNN to increase the accuracy of the entire system.

### 3.1. SYSTEM ARCHITECTURE

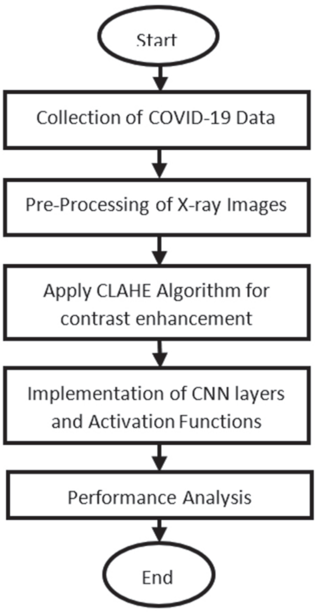

Start
Collection of COVID-19 Data
Pre-Processing of X-ray Images
Apply CLAHE Algorithm for contrast enhancement
Implementation of CNN layers and Activation Functions
Performance Analysis
End

**Fig.1.** Flow chart of the Proposed Model

Fig. 1 above shows the flow chart of the proposed model. This proposed system takes X-ray scans as input. The first step is to resize the images and convert indexed images to RGB, as CLAHE works much better on Colored images than any other format. Then this colored converted image is sent to the CLAHE for contrast enhancement; then, we applied the CNN model to the same image. These two features were merged and used as strategies to form a distribution model. A detailed explanation of the whole procedure is given in the following sections. Fig 2. below shows the architecture of the proposed system.

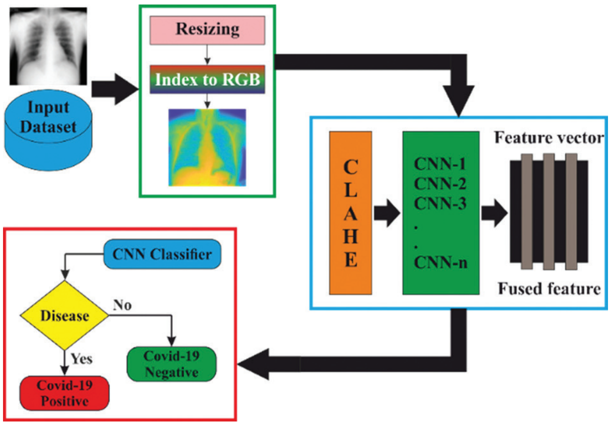


**Fig. 2.** Proposed System Architecture

As we can see in Fig. 2 above, firstly input image goes to the resizing block and then to the RGB block for conversion; after conversion, the same image goes to the CLAHE block for contrast enhancement so the powerful CNN layers can perform their necessary actions on it. Then data is fed to the CNN network, where images are classified as positive or negative for COVID-19.

### 3.2. DATASET USED

The dataset used in the process is publicly available on Kaggle, named COVID-19 X-Ray dataset. This dataset contains 15917 X-Ray scans of infected and non-infected patients, 2186 and 13731, respectively. The files contain images of various sizes that range from 512×512 to 657×657 pixels.

### 3.3. DATA PRE-PROCESSING

Firstly the dataset is normalized, resizing the images to 70 × 70 is done, and then the images are mixed and divided into training and test data. The training data contains 11142 images, which are divided into two folders named COVID and NONCOVID. The COVID folder under training data contains 1530 images, and the NONCOVID folder contains 9612 images. Similarly, test data contains 4775 images and two subfolders, COVID and NONCOVID, which include 656 and 4119 images, respectively. It can be precisely seen in Table 1 below.

**Table 1.** Number of images in different categories used in the training process

| Dataset | COVID | NONCOVID | Total |
|---|---|---|---|
| Train | 1530 | 9612 | 11142 |
| Test | 656 | 4119 | 1800 |

Training and test dataset images were selected by excruciating the complete dataset of images (i.e., training and test dataset combined). If multiple images are present for the same patient, we have ensured that images are marked as either training or test data so results can be manageable because of patient overlap. Then we converted the indexed images into color images using the MATLAB function ind2rgb(X, map). After all, this dataset is ready for the training of the model.

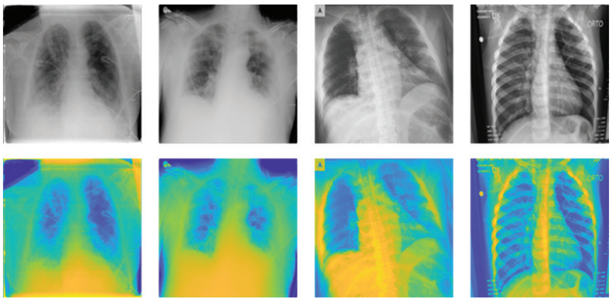


**Fig. 3.** Some Sample Indexed and Converted Color Images

### 3.4. CLAHE (CONTRAST LIMITED ADAPTIVE HISTOGRAM EQUALIZATION)

CLAHE is a variant of Adaptive Histogram Equalization (AHE) that deals with contrast over-enhancement. CLAHE works with small zones of the image, termed mosaics. Adjacent tiles are then joined using bilinear interpolation to eliminate artificial boundaries. This algorithm can be used to enhance the contrast of images.

We can also use CLAHE on color images, which generally works on the luminance channel. The results are much better after fitting to just the luminance channel of an HSV image. The standard architecture of CLAHE is shown in Fig. 4 below.

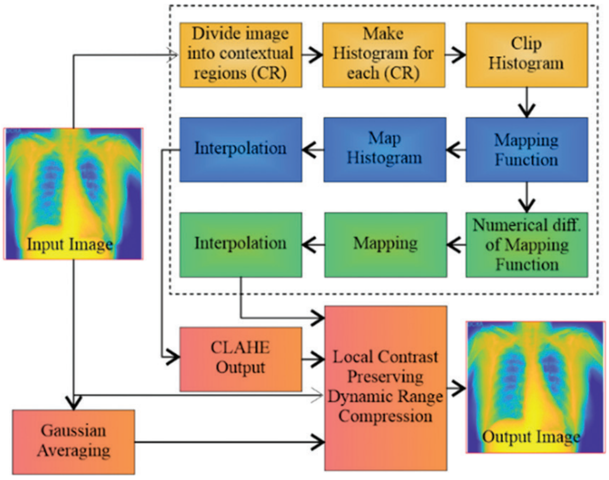

Divide image into contextual regions (CR)
Make Histogram for each (CR)
Clip Histogram
Interpolation
Map Histogram
Mapping Function
Input Image
Interpolation
Mapping
Numerical diff. of Mapping Function
CLAHE Output
Local Contrast Preserving Dynamic Range Compression
Output Image
Gaussian Averaging

**Fig. 4.** Architecture of CLAHE

### 3.5. CNN (CONVOLUTIONAL NEURAL NETWORK)

A Convolutional neural network, generally called ConvNet or CNN, tends to be a deep learning algorithm that can take an input image and classify it among other images. One main thing differentiating CNN from other algorithms is that it requires much lower pre-processing than different algorithms; additionally, ConvNet has self-learning capabilities.

The general architecture of ConvNet is shown in Fig. 5 below. Here we have an RGB image as input. The element that performs the convolution operation at the very first is called the kernel; in the image, the kernel is shown with red color. The kernel will have the same depth as the input image if the image has multiple channels. The kernel will move all over the image to extract the high-level features. Then we have pooling layers, the same as the convolution layers pooling layers are responsible for reducing the spatial size of the convolved features. It reduces the computational power requirements to process extensive data. At last, fully connected layers are used to learn the non-linear combinations of features represented by the output of the convolutional layer.

### 3.6. EXPERIMENTAL ENVIRONMENT

All experimental simulations are carried out on a system with Intel i7 Processor, 8 GB RAM, and NVIDIA G-force 2 GB Graphic card. The simulation software used for all this is MATLAB 2020a. While the training process, we resized all images to 70 × 70, so all the illustrations should be constant according to size.

### 3.7. PERFORMANCE METRICS

The performance of the planned system is demonstrated in the form of a confusion matrix. Classification and validation accuracy is also used to calculate the projected model's performance.

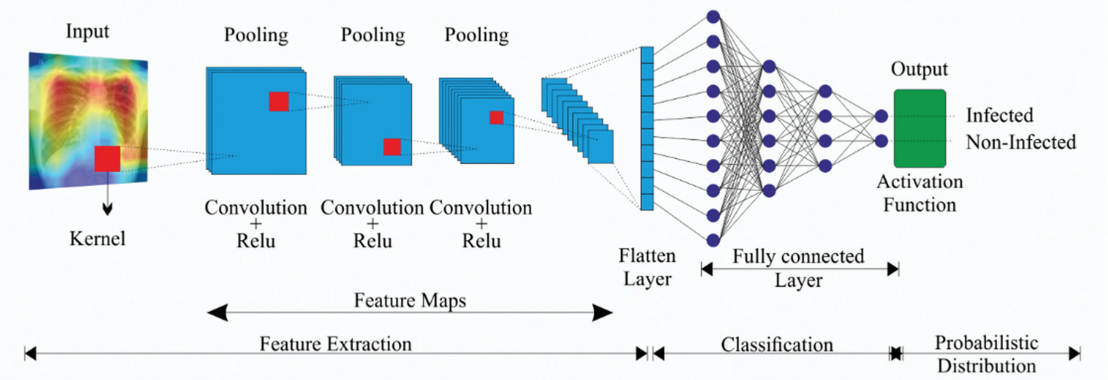


**Fig. 5.** Architecture CNN

**Classification Accuracy** is the most vital recital evaluation metric. One can calculate it as a (true positive + true negative) ratio with the total length.

$$Classification\ accuracy = \frac{TP + TN}{N}$$

**Specificity** is the ratio of true negatives with the sum of true negatives and false positives. Mathematically it can be represented as

$$Specificity = \frac{TN}{TN + FP}$$

**Precision** is the ratio of true positives with the sum of true positives and false positives. Mathematically it can be represented as

$$Precision = \frac{TP}{TP + FP}$$

**The Recall** is the ratio of true positives with the sum of true positives and false negatives. Mathematically it can be represented as

$$Recall = \frac{TP}{TP + FN}$$

**F1 Score** is the function of Precision and Recall. Mathematically it can be represented as

$$F1 = 2 \times \frac{Precision * Recall}{Precision + Recall}$$

Where **TP** = True Positive (How many times did the model correctly classify a Positive sample as Positive?)

**FP** = False Positive (How many times did the model incorrectly classify a Negative sample as Positive?)

**TN** = True Negative (How many times did the model correctly classify a Negative sample as Negative?)

**FN** = False Negative (How many times did the model incorrectly classify a Positive sample as Negative?)

**N** = Total Length = (*TP*+*TN*+*FP*+*FN*)

In the confusion matrix, correctly classified COVID-19 positive cases by the model are represented as TP, and those incorrectly classified as COVID-19 negative are represented as FP. Similarly, adequately categorized COVID-19 negative cases are categorized as TN, and wrongly classified as COVID-19 positive are termed FN.

## 4. EXPERIMENTAL RESULTS

This section provides a performance analysis of the planned system to categorize the COVID-19 infected X-ray scans. The model is trained for 50 epochs. The proposed method achieved an average accuracy of 99.8%, Precision of 92.6%, Recall of 99.81%, and F1 Score of 99.32% while determining the COVID-19 contamination among X-ray scans. To analyze the given model's effectiveness, some plots have been generated. The figures below show the confusion matrix ROC curve and the training and validation graph obtained. In the training and validation graph, spikes in blue show the obtained classification accuracy of the proposed system, and the markers in black are the validation lines.

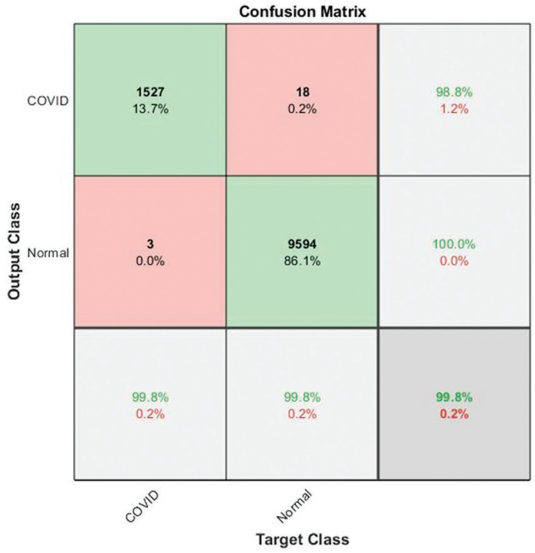


**Fig. 6.** Confusion Matrix

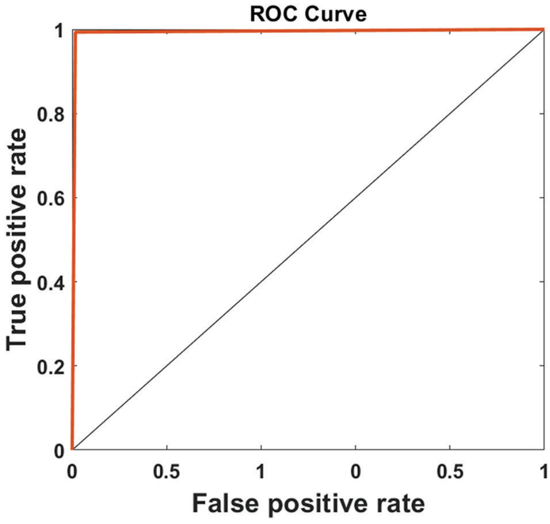


**Fig. 7.** ROC Curve

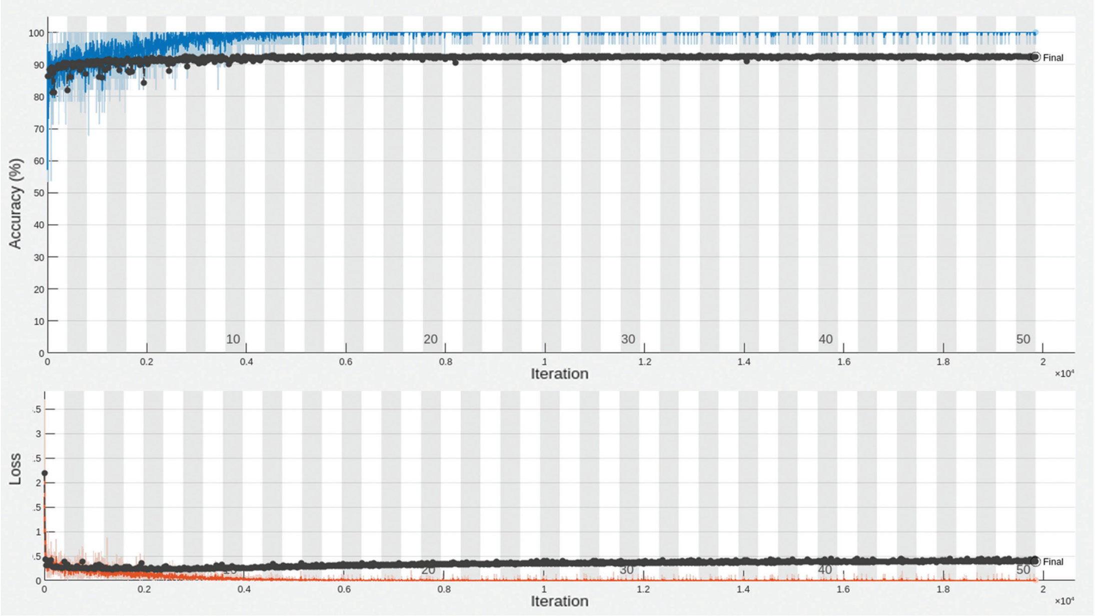


**Fig. 8.** Training and validation graph

### 4.1. K-FOLD CROSS VALIDATION

Additionally, the given model is tested with K-fold cross-validation. It is a statistical practice to analyze the performance of machine learning algorithms. This entire dataset is divided into k no of folds, and the performance is analyzed when new data is given. Here we have chosen three as the value of K to perform the analysis. We also tried with the values 4 and 5, but the results are almost similar, so we have decided to go with the minimum value of 3. The results obtained from k-fold validation are as follows. Where the three folds have given the accuracy of 99.7%, 99.8%, and 99.9%, respectively, if we take the mean of these values, we will get the same result as our model, which is 99.80% accuracy.

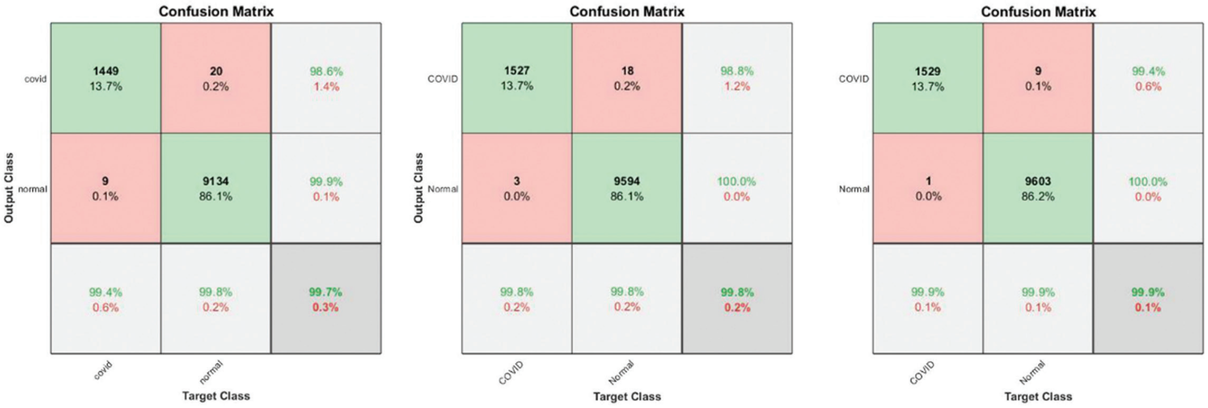


**Fig. 9.** Confusion matrix (K-Fold cross-validation)

As we can see from fig. 9 above, If we take the mean of these values, the three folds have given the accuracy of 99.7%, 99.8%, and 99.9%, respectively.

$$Average\ Accuracy\ = \frac{1}{K}\sum_{k=0}^{3}(99.7 + 99.8 + 99.9)$$

$$Average\ Accuracy\ = 99.80\%$$

As we can see, the final result is the same as the obtained result from a proposed model in terms of accuracy; now, we can say the proposed model is validated through k fold validation process.

### 4.2. COMPARISON WITH OTHER MODELS

In this section, we compare the proposed model with other existing models. The table below shows the comparison of different algorithms, such as PDCOVIDNET [10], VGG16 [18], ResNet50 [21], and HOG+CNN, with the proposed algorithm in terms of accuracy.

**Table 2.** Comparison with other existing algorithms

| Method | Accuracy | Specificity | Precision | Recall | F1 |
|---|---|---|---|---|---|
| PDCCOVIDNet | 96.58 | NE | 96.58 | 96.59 | 96.58 |
| VGG 16 | 93.8 | NE | 93.84 | 93.86 | 93.83 |
| ResNet50 | 94.74 | NE | 92.67 | 92.15 | 92.12 |
| HOG+CNN[20] | 99.49 | 95.7 | NE | NE | NE |
| Proposed CLAHE+CNN | 99.81 | 99.81 | 98.83 | 99.81 | 99.32 |

*Where NE is not evaluated

PDCOVIDNET [13] in Dec 2020 used small convolution in a similar stack to capture and stretch the required properties to obtain a recognition accuracy of 96.58%, a precision of 96.58%, and a recall of 96.59%, and F1 Score of 96.58%. VGG16 [12] model as consisting of two stages, one is pre-processing, and the second is data dissemination and learning transfer, and lastly shows an accuracy of around 99.8%, Precision of 93.84%, Recall of 93.86%, and the F1 Score of 93.83%. ResNet50 [10] has the lowest parameters compared to all other models, with an accuracy of 94.74%, Precision of 92.6%, Recall of 92.15%, and the F1 Score of 92.12%. HOG+CNN [20] by Noor-A-Alam achieved an accuracy of 99.49% and a Specificity of 95.7%. The proposed system CLAHE+CNN has achieved the highest Accuracy of 99.81%, Precision of 92.6%, Recall of 99.81%, and F1 Score of 99.32%.

### 4.3. LIMITATIONS

Although the proposed algorithm has achieved a very high accuracy of 99.81%, every technique has some limitations. The proposed system works only on indexed images that can be converted into RGB color format. This was challenging for authors to find a large image dataset with all the images with the same graphical properties. The proposed technique is a promising tool in healthcare to classify COVID-19 infection.

## 5. CONCLUSION

We have presented a deep learning model by combining CLAHE and CNN to detect COVID-19 infection using chest X-ray images. CLAHE is used before CNN to enhance the contrast of X-ray images so the CNN model can classify the X-ray images more precisely. This research developed an intelligent system to identify COVID-19 infection using chest X-rays Images with great accuracy of 99.8% and low complexity. This is very encouraging that X-ray images are used to detect COVID-19 contamination at this level. This proposed system was more accurate than the results obtained from personal isolation techniques like HOG and CNN. Additionally, the given approach is validated with the same accuracy by using the k-fold authentication procedures. For future work, we will develop contactless image-capturing methods for front-end healthcare workers.

## 6. REFERENCES

[1] F. Wu, et al. "A new coronavirus associated with human respiratory disease in China", Nature, Vol. 579, 2020, pp. 265-269.

[2] W. Guan, Z. Y. Ni, Y. Hu, W. H. Liang, C. Q. Ou, J. X. He, L. Liu, C. L. Lei, "Clinical characteristics of coronavirus disease 2019 in China", New England Journal of Medicine, Vol. 382, No. 18, 2020, pp. 1708-1720.

[3] N. Chen, M. Zhou, X. Dong, J. Qu, F. Gong, Y. Han, Y. Qiu, J. Wang, Y. Liu, Y. Wei, L. Zhang, "Epidemiological and clinical characteristics of 99 cases of 2019 novel coronavirus pneumonia in Wuhan, China: a descriptive study", The Lancet, Vol. 395, No. 10223, 2020, pp. 507-513.

[4] C. Wang, P. W. Horby, F. G. Hayden, G. F. Gao, "A novel coronavirus outbreak of global health concern", The Lancet, Vol. 395, No. 10223, 2020, pp. 470-473.

[5] N. Zhu, D. Zhang, W. Wang, X. Li, B. Yang, J. Song, et al., "A novel coronavirus from patients with pneumonia in China", New England Journal of Medicine, Vol. 382, 2020, pp. 727-733.

[6] Q. Li, X. Guan, P. Wu, X. Wang, L. Zhou, Y. Tong, R. Ren, K. K. S. Leung, E. H. Y. Lau, J. Y. Wong, et al., "Early transmission dynamics in Wuhan, China, of novel coronavirus-infected pneumonia", New England Journal of Medicine, Vol. 382, 2020, pp. 1199-1207.

[7] L. Wang, A. Wong, "COVID-Net: A Tailored Deep Convolutional Neural Network Design for Detection of COVID-19 Cases from Chest X-Ray Images", arXiv:2003.09871, 2020.

[8] A. Afzal, "Molecular diagnostic technologies for COVID-19: Limitations and challenges", Journal of Advanced Research, Vol. 26, 2020, pp. 149-159.

[9] World Health Organization, "Use of Chest Imaging in COVID-19", www.who.int/publications/i/ item/ use-of-chest-imaging-in-COVID-19 (accessed: 2021)

[10] K. He, X. Zhang, S. Ren, J. Sun, "Deep Residual Learning for Image Recognition", arXiv:1512.03385, 2015.

[11] A. Krizhevsky, S. Ilya, G. E. Hinton, "Imagenet classification with deep convolutional neural networks", Communications of the ACM, Vol. 60, No. 6, 2017, pp. 84-90.

[12] K. Simonyan, A. Zisserman, "Very deep convolutional networks for large-scale image recognition", arXiv:1409.1556, 2014.

[13] N. K. Chowdhury, M. M. Rahman, M. A. Kabir, "PD-COVIDNet: A parallel-dilated convolutional neural network architecture for detecting COVID-19 from chest X-ray images", Health Information Science and Systems, Vol. 8, 2020, pp. 1-14.

[14] I. U. Khan, N. Aslam, "A Deep-Learning-Based Framework for Automated Diagnosis of COVID-19 Using X-ray Images", Information, Vol. 11, No. 9, 2020.

[15] S. Minaee, R. Kafieh, M. Sonka, S. Yazdani, G. J. Soufi, "Deep-COVID: Predicting COVID-19 from chest X-ray images using deep transfer learning", Medical Image Analysis, Vol. 65, No. 101794, 2020.

[16] B. Sekeroglu, I. Ozsahin, "Detection of COVID-19 from Chest X-Ray Images Using Convolutional Neural Networks", SLAS TECHNOLOGY: Translating Life Sciences Innovation, Vol. 25, No. 6, 2020, pp. 553-565.

[17] N. E. M. Khalifa, M. H. N. Taha, G. Manogaran, M. Loey, "Retracted article: a deep learning model and machine learning methods for the classification of potential coronavirus treatments on a single human cell", Journal of Nanoparticle Research, Vol. 22, No. 11, 2020, pp. 1-13.

[18] A. T. Sahlol, D. Yousri, A. A. Ewees, M. A. A. Al-Qaness, R. Damasevicius, M. A. Elaziz, "COVID-19 image classification using deep features and fractional-order marine predator's algorithm", Scientific Reports, Vol. 10, No. 1, 2020, pp. 1-15.

[19] M. M. Rahman, S. Nooruddin, K. M. A. Hasan, N. K. Dey, "HOG+CNN Net: Diagnosing COVID-19 and Pneumonia by Deep Neural Network from Chest X-Ray Images", SN Computer Science, Vol. 2, No. 5, 2021.

[20] N. A. Alam, M. Ahsan, Md. A. Based, J. Haider, M. Kowalski, "COVID-19 detection from chest X-ray images using Feature fusion and deep learning", Sensors, Vol. 21, No. 4, 2021.

# Support Vector Regression Machine Learning based Maximum Power Point Tracking for Solar Photovoltaic systems

Original Scientific Paper

**P. Venkata Mahesh**
Research Scholar, Department of Electronics and Instrumentation Engineering,
Annamalai University, Chidambaram, Tamilnadu-608002, India. &
Assistant Professor, Department of Electrical and Electronics Engineering,
RVR & JC College of Engineering, Guntur, Andhra Pradesh-522019, India.
vnktmahesh@gmail.com

**S. Meyyappan**
Assistant Professor, Department of Electronics and Instrumentation Engineering,
Annamalai University, Chidambaram, Tamilnadu-608002, India. &
Assistant Professor, Department of Instrumentation Engineering,
Madras Institute of Technology, Chennai, Tamil Nadu-600044, India.
meys.narayan@gmail.com

**RamaKoteswaraRao Alla**
Associate Professor, Department of Electrical and Electronics Engineering,
RVR & JC College of Engineering, Guntur, Andhra Pradesh-522019, India.
ramnitkkr@gmail.com

***Abstract*** *– Photovoltaic panels use the sun's radiation on their surface to convert solar energy into electricity. This process is dependent on the temperature of the surface and the intensity of the sun's radiation. To escalate the energy transformation, the solar system must be functioned at its maximum power point (MPP). Every maximum power point tracking (MPPT) technique has a distinct mechanism for tracking maximum power point (MPP). The support vector machine (SVM) regression algorithm is used in this work to develop a novel method for tracking the MPP of a PV panel. The solar panel technical parameters were used to prepare the data for training and testing the SVM model. The SVM algorithm predicts the PV panel's maximum power and relevant voltage for specific irradiation and temperature. The duty cycle of the boost converter corresponding to the maximum power was evaluated using the predicted values. The result of the simulation shows that the proposed control strategy forces the solar panel to work near the predicted MPP. The SVM regression control strategy gives the MPP tracking efficiency of more than 94% for the solar PV system despite variable climatic conditions during its stable state operation. In addition, a comparative analysis of the proposed method was carried out with the existing approaches to confirm the effective tracking of the proposed technique.*

***Keywords****: Boost converter, MPPT, Photovoltaic system, Regression machine learning, Support vector machine*

## 1. INTRODUCTION

The environmental harm produced by conventional power sources may be mitigated using solar energy. Photovoltaic generation systems (PVGS) convert solar energy into electricity. However, since the PVGS is not worked at maximum power point (MPP), it is strongly advised to drive the system at its MPP to enhance the energy conversion efficiency. This is achieved through a process known as maximum power point tracking (MPPT). The MPPT uses an algorithm to compel the PVGS to work at MPP. There are several MPPT approaches published in the literature. Each technique has its own set of strengths and weaknesses and its own method of tracking the MPP. The conventional methods are the Perturb and Observe (P&O) [1] and incremental conductance (IC) [2] methods, mathematical methods such as curve fitting [3] and beta MPPT [4], measurement-based methods such as look-up table [5] and current sweep [6], constant parameter methods such as fractional open circuit voltage [7] and fractional short circuit current [8] methods, trial and error methods such as gradient descent method [9] and variable inductance method [10], optimization techniques like genetic algorithm [11], ant colony optimization [12], practical swarm optimization [13], gray wolf optimization [14], and cuckoo search optimization [15], intellectual methods like an artificial neural network [16], fuzzy logic control [1], and ANFIS [1,9] are listed in the literature.

The need for clean, affordable, and sustainable energy is expanding rapidly, and technology is actively seeking methods to meet this need [17, 18]. The maximum power extraction from solar PV system is challenging task under partial shading conditions [19-21]. Artificial intelligence (AI) and machine learning (ML) have emerged as significant technological solutions. These cutting-edge technologies can forecast the future, enhance the present, and examine the past. This indicates that most of the current problems may be resolved using AI and ML [22]. Machine learning for MPPT typically eliminates the need for a controller. MPPT was implemented in the literature utilizing support vector machine learning in conjunction with a Proportional Integral Derivative (PID) controller [23], reinforcement learning [24], and a random forest technique [25]. The ML algorithm (MLA) may predict the unknown information if the model is trained, tested, and validated using existing information. Typically, the data for training, testing, and validating the machine learning model are chosen in the ratios of 60:20:20. Sum squared error (SSE), root mean square error (RMSE), and $R^2$ are three metrics that may use to assess the prepared model's prediction ability. For the calculation of RMSE, SSE, and $R^2$, the following equations Eq. (1), (2), and (3) [26, 27] are used.

$$RMSE = \left[\frac{1}{n_s}\sum_{k=1}^{n_s}\left(Y_{A,k} - Y_{P,k}\right)^2\right]^{1/2} \tag{1}$$

$$SSE = \sum_{k=1}^{n_s}\left(Y_{A,k} - Y_{P,k}\right)^2 \tag{2}$$

$$R^2 = 1 - \frac{\sum_{k=1}^{n_s}\left(Y_{A,k}-Y_{P,k}\right)^2}{\sum_{k=1}^{n_s}\left(Y_{A,k}-Y_{Avg}\right)^2} \tag{3}$$

where, $Y_A$ is the real data, $Y_P$ is the data predicted, the total samples number is $n_s$, and the real values average is $Y_{Avg}$. The $R^2$ is between 0 and 1 which gives the model prediction potential, and for the best suited model, the $R^2$ is near to 1. Similarly, the $SSE$ and $RMSE$ quantifies the error among $Y_P$ and $Y_A$. The model with the strongest ability to predict is therefore represented by $RMSE$ and $SSE$ that are close to zero.

A power electronic converter is necessary to transmit the maximum amount of power from PVGS to the load. In literature, DC-DC converters such as the boost [2-6], buck-boost [7, 8], buck [10], and SEPIC [14] are employed. In addition, an inverter [9] can also be used to drive the ac loads or to supply the grid. This study proposes a unique method for tracking the MPP of a solar module using support vector machine regression learning. The suggested approach's efficacy was evaluated in contrast to classic MPPT algorithms such as P&O, IC methods, and intelligent control techniques such as ANN, FLC. The comparison has been done by considering time domain specifications of power response such as tracking speed, settling value, and overshoot, etc.

The rest of  paper is organized as follows, the system description, which includes the PV module with technical parameters, boost converter, and support vector machine regression algorithm, is provided in Section-2; the methodology comprises collecting data, preparing the model, and PV panel working with support vector machine regression control approach have been provided in Section-3; simulation result with discussions of the proposed method are provided in Section-4, the proposed approach is compared with the existing P&O, IC, ANN and FLC methods in Section-5. The paper is concluded in Section-6.

## 2. DESCRIPTION OF SYSTEM

### 2.1. PV MODULE AND BOOST CONVERTER

Solar cells convert sunlight into electrical energy through photoelectric effect. Multiple solar cells connected to form a solar PV module. From the solar cell's single diode equivalent [28, 29] model the mathematical representation of solar module is in Eq.(4).

$$I_m = I_{PH} - I_0\left(e^{\frac{V_m+I_m N_S R_S}{n N_S V_T}} - 1\right) - \frac{V_m+I_m N_S R_S}{N_S R_{sh}} \tag{4}$$

where the solar module current is $I_m$ and $I_{PH}$ indicates the light generated current. The saturation current of the diode is $I_0$, $V$ is the module voltage, the ideal factor of pn-diode is n (1 ≤ $n$ ≤ 2), the thermal voltage is $V_T$, and $N_s$ is number of series cells. The resistances $R_{sh}$ and $R_s$ are the module shunt and series resistances respectively.

A 10W solar panel with 21.50V open circuit voltage, 0.62A short circuit current, 0.57A current and 17.50V voltage at MPP is used in this work. The current-voltage (I-V) and power-voltage (P-V) characteristics are provided in Fig.1.

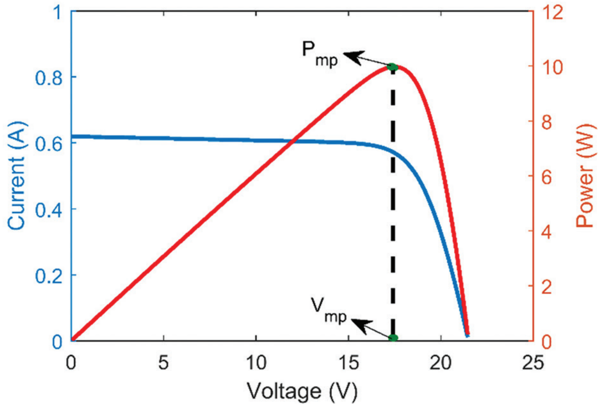


**Fig. 1.** I-V and P-V Characteristics of solar module at 1000w/m² and 25 °C

A dc-dc boost converter with pulse width modulation (PWM) control [29, 30] shown in Fig.2 is employed in this work. The power transferred to load from input source was controlled by using the duty cycle ($D$) of the switch. The inductor ($L$) enhances the input voltage to the necessary output value.

The input and output capacitors ($C_i$ & $C_o$) both help to lower the voltage ripple content.

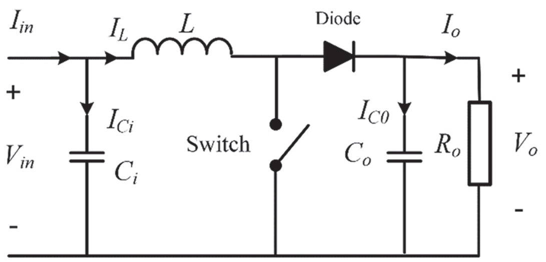


**Fig. 2.** DC-DC boost converter [29]

## 2.2. SUPPORT VECTOR MACHINE REGRESSION ALGORITHM

Support vector machines (SVM) were initially built to classify binary issues and were expanded to include the classification and regression of multiclass problems. In the training data set, by estimating the linear or nonlinear relationship between a given input and its associated output, the support vector machine regression (SVMR) technique [31] predicts the output based on the input. As a result, the developed SVMR model may be used to predict outcomes based on supplied inputs. The goal of support vector regression with ε-intense loss function is to determine the optimal hyperplane with the shortest distance between all data points. Suppose a training data set with N samples are denoted as $(x_i, y_i)$, $i = 1, 2,\ldots, N$, where $x_i$ represents the input and $y_i$ represents the output. The optimal hyperplane approximates the training points as closely as possible while reducing the prediction error. The linear hyperplane function is defined as $f(x) = \beta x+b$, where $x$ denotes a point on the plane, $\beta$ specifies the hyperplane's inclination in space, and b is the bias that determines the distance of the hyperplane from the origin, as shown in Fig. 3.

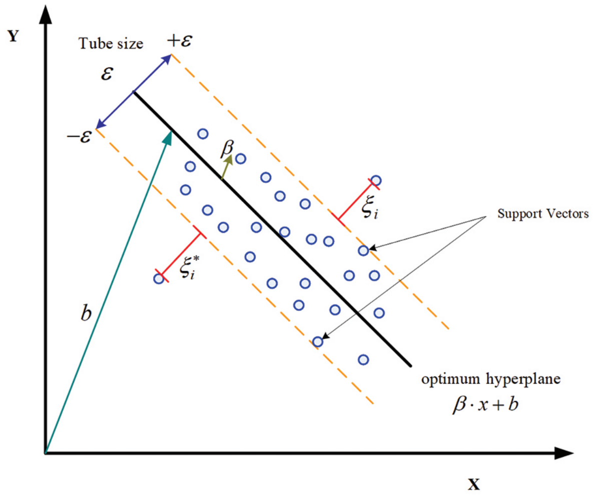


**Fig. 3.** SVM for linear regression problem on two dimensional space [31]

In the specified ε-insensitive loss function, SVMR looks for an ideal hyperplane that can predict y without errors. In other terms, the distance between any data point and the ideal hyperplane is smaller than ε. Where ε represents the radius of the tube. SVMR uses a ε- intensive loss function to compute linear regression in a high-dimensional feature space while minimizing model complexity by reducing the value of $||\beta||^2$. The ε- intensive loss function is a function that is used to optimize generalization boundaries that are close to actual value and are located at a particular distance by ignoring errors. As a result, SVMR is defined as the solution to the optimization problem [31] given in Eq.(5) and Eq.(6).

$$Minimize\ \frac{1}{2}\parallel \beta \parallel^2 + C\sum_{i=1}^{N}(\xi_i - \xi_i^*) \tag{5}$$

$$Subject\ to \begin{cases} y_i - (\beta\,.x_i + b) \le \varepsilon + \xi_i \\ (\beta\,.x_i + b) - y_i \le \varepsilon + \xi_i^* \\ \xi_i \ge 0 \\ \xi_i^* \ge 0 \end{cases} \tag{6}$$

The slack variables $\xi_i$ and $\xi_i^*$ ($i$ =1, 2,…, $N$) will measure the deviation of the training samples outside the ε-insensitive zone, and the penalty parameter or the regularization constant is C which determines the trade-off between the model complexity and the training error. If the data has a non-linear shape, SVMR uses a non-linear transformation function ($K(x_i, x)=\phi(x_i).\phi(x)$) called a kernel function. This kernel is for mapping the input pattern to a high-dimensional feature space to identify the ideal hyperplane that minimizes discriminating errors in the training data. Next, a linear model is constructed in this feature space. As a result, the SVMR function for approximating nonlinear training data is as follows,

$$f(x) = \beta \times \phi(x) + b = \sum_{i=1}^{N}(\alpha_i - \alpha_i^*)\ K(x_i, x) + b \tag{7}$$

$$\beta = \sum_{i=1}^{N}(\alpha_i - \alpha_i^*)x_i \tag{8}$$

The kernel function in linear form is given by Eq.(9).

$$K(x_i, x) = x_i' \times x \tag{9}$$

The complementarity constraints by Karush Kuhn Tucker are optimization boundaries required to find the optimal solutions. These conditions are in Eq.(10) for Linear SVMR.

$$\left.\begin{array}{r} \alpha_i(\varepsilon + \xi_i - y_i + \beta\,.x_i + b) = 0 \\ \alpha_i^*(\varepsilon + \xi_i^* + y_i - \beta\,.x_i - b) = 0 \\ \xi_i(C - \alpha_i) = 0 \\ \xi_i^*(C - \alpha_i^*) = 0 \end{array}\right\} \tag{10}$$

These circumstances address all perceptions rigorously inside the epsilon edge team having $\alpha_i$=0 and $\alpha_i^*$=0. An observation is called a support vector if either $\alpha_i$ or $\alpha_i^*$ is not zero. The difference between two support vectors Lagrange multipliers ($\alpha_i$ - $\alpha_i^*$) is stored by the parameter α for a trained SVM model. The support vector's properties and bias store $x_i$ and $b$, respectively. The type of kernel function and its parameters will decide the prediction performance of SVMR. The kernel functions are linear, radial basis, polynomial, and sigmoid [32]. Here, a linear kernel function is preferred as the data is almost linearly separable and is faster in training with fewer parameters to optimize.

## 3. METHODOLOGY

There are two phases in the proposed strategy. Obtaining data from the PV module specifications and creating the SVMR model comes in primary phase. The secondary is to employ the prepared SVMR model for MPPT. The power at MPP ($P_{mp}$) and the corresponding voltage at maximum power ($V_{mp}$) depends on irradiance ($I_r$) and temperature ($T$), so the T and $I_r$ are used as input features in the prediction of $P_{mp}$ and $V_{mp}$. The prepared SVMR models predict the $P_{mp}$ and $V_{mp}$ of the PV panel. The predicted $P_{mp}$ and $V_{mp}$ are used to compute the converter's duty cycle (D) such that the PV module works at the predicted MPP.

### 3.1. COLLECTING THE DATA & PREPARING THE MODEL

$I_r$, $T$, $P_{mp}$, and $V_{mp}$ are the data needed for training and testing of the model. Solar panel parameters were used to gather the data. Matlab/Simulink software used to train the SVMR models. The flowchart in Fig. 4 depicts the process for gathering data and building a machine learning (ML) model.

### 3.2. SVMR MPPT CONTROL STRATEGY

For the input features $I_r$ and $T$, the trained ML model predicts the $P_{mp}$ and $V_{mp}$. The $R_{mp}$ resistance, which corresponds to MPP, is evaluated using the predicted values $P_{mp}$ and $V_{mp}$ as in Eq.(11) [30]. In Fig. 5, $R_{mp}$ will be replicated across node-p and node-q by controlling $D$ of the boost converter. According to Figs. 4 and 7, the resistance ($R_{pq}$) between nodes p and q is zero when $D$ is zero. As $D$ grows, $R_{pq}$ rises and will reach $R_o$ when $D$ is one. The parameter $D$ in $R_{mp}$ and load resistance ($R_o$) is in Eq.(12) [30].

$$R_{mp} = \frac{V_{mp}^2}{P_{mp}} \tag{11}$$

$$D = 1 - \sqrt{\frac{R_{mp}}{R_0}} \tag{12}$$

The extreme and least values for load resistance are calculated using the method recommended by Razman Ayop et al. in [30]. The boost converter's design procedure is explained by Muhammad H. Rashid [33]. Equation (13) gives the boost converter inductance, and Eq.(14) provides the capacitance, respectively [33].

$$L = \frac{V_{ip}\times(V_{op}-V_{ip})}{f_{sw}\times\Delta I\times V_{op}} \tag{13}$$

$$C = \frac{I_{op}\times(V_{op}-V_{ip})}{f_{sw}\times\Delta V\times V_{op}} \tag{14}$$

where $V_{ip}$ denotes input voltage, $V_{op}$ denotes output voltage, $f_{sw}$ indicates frequency of switching, $\Delta I$ represents current ripple, and $\Delta V$ represents voltage ripple. The control strategy diagram for the solar PV module with the SVM regression ML is shown in Fig. 5.

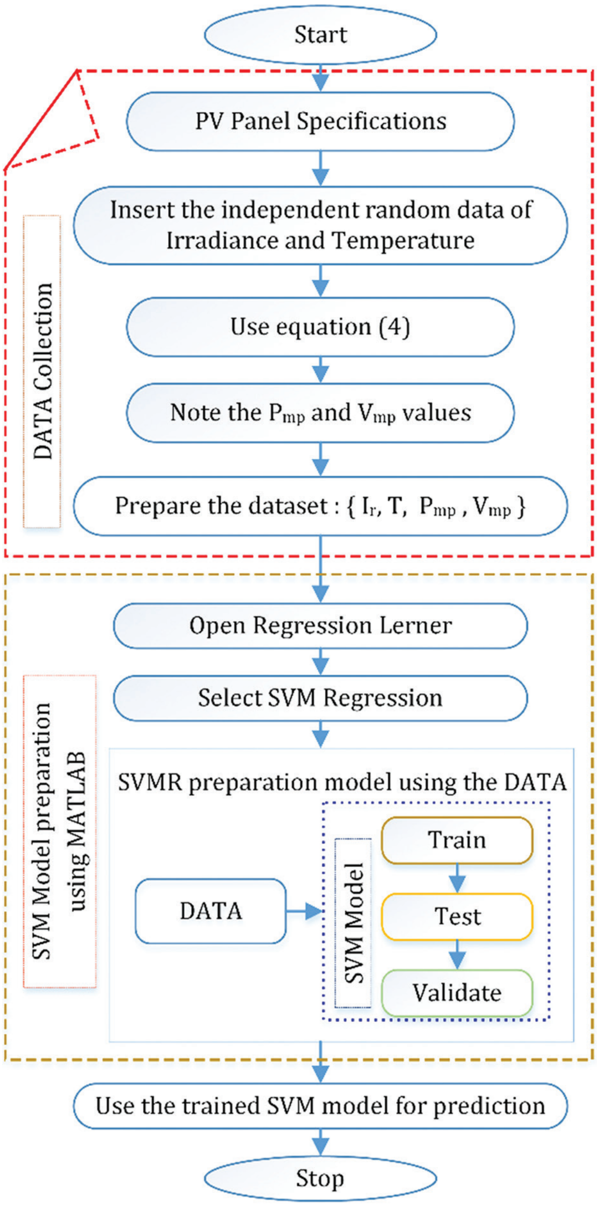

Start
PV Panel Specifications
Insert the independent random data of Irradiance and Temperature
Use equation (4)
Note the Pmp and Vmp values
Prepare the dataset : { Ir, T, Pmp , Vmp }
DATA Collection
Open Regression Lerner
Select SVM Regression
SVMR preparation model using the DATA
DATA
SVM Model
Train
Test
Validate
SVM Model preparation using MATLAB
Use the trained SVM model for prediction
Stop

**Fig. 4.** Data collecting and ML model building procedure as a flowchart

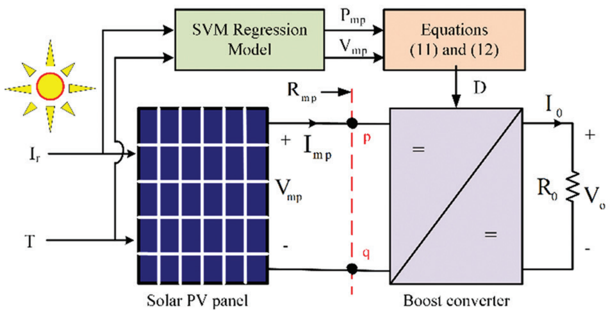


**Fig. 5.** Control strategy block diagram with SVM regression ML & dc-dc boost converter

## 4. SIMULATION RESULTS OF PROPOSED SVMR MPPT CONTROL STRATEGY

With the aid of solar panel technical parameters, the data from the PV panel has been collected in the suggested method specified in section-3. The pairwise relation and correlation among the data is given in [29]. To test the tracking performance of the proposed technique in the presence of variables $I_r$ and $T$, the simulation was run in four intervals of 0.5 seconds. For each interval either $I_r$ or $T$ are changed while keeping the other fixed. This variation is shown in Table -1.

**Table 1.** Input parameters of the PV panel for various intervals

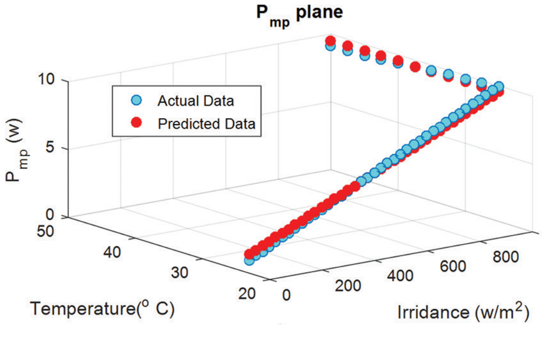
| Parameter | Interval-1 | Interval-2 | Interval-3 | Interval-4 |
|---|---|---|---|---|
| Time (sec) | 0 to 0.5 | 0.5 to 1 | 1 to 1.5 | 1.5 to 2 |
| $I_r$ (W/m$^2$) | 450 | 450 | 950 | 950 |
| $T$ (°C) | 25 | 35 | 35 | 25 |

The simulation values used here in study are, PV power ($P$) = 10W, $f_{sw}$ = 5 kHz, ripple voltage allowed ($\Delta V$) = 1 %, ripple current allowed ($\Delta I$) = 5 %, $L$ = 34 mH, $C_o$ = 68 μF, $R_0$ = 300 Ω, and $C_i$ = 1000 μF.

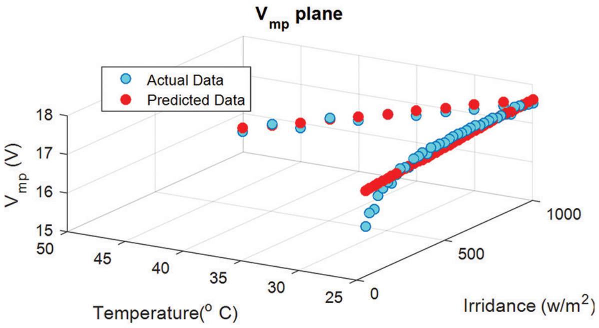
**Table 2.** Parameters of created the SVMR models with linear kernel

| Parameter | SVMR-1 (Pmp plane) | SVMR-2 (Vmp plane) |
|---|---|---|
| Bias | 0.4568 | 19.1963 |
| ε | 0.4224 | 0.0397 |
| β | [0.0091 -9.4161×10$^{-4}$] | [4.2608×10$^{-4}$ -0.0802] |
| No. of support vectors | 3 | 30 |
| No. of iterations | 9 | 1×106 |

The SVMR models (SVMR-1 and SVMR-2) are created with $I_r$ and T input features. The output predicted response for SVMR-1 is $P_{mp}$ and for SVMR-2 is $V_{mp}$. The parameters of the created models are in Table-2. The actual and predicted data by the developed SVMR models are given in Fig. 6. Fig. 6a shows a small residual in prediction on the $P_{mp}$ plane. On the other side Fig. 6b shows that for low $I_r$ and $T$, the prediction error is high, and for the rest is minor on the $V_{mp}$ plane.



(a)



(b)

**Fig. 6.** Predicted and actual data by a) SVMR-1 on $P_{mp}$ plane b) SVMR-2 on $V_{mp}$ plane

Fig. 7 indicates the solar panel and load V, I, and power (P) responses with the developed SVMR models. These results illustrate a small oscillation in the transient response if there is a variation in T and fluctuations with large amplitude if Ir is varied. Figure 8 shows the tracking efficiency and comparison of the predicted and working PV power. It can be observed that the proposed methodology tracks the accurate MPP in the stable state.

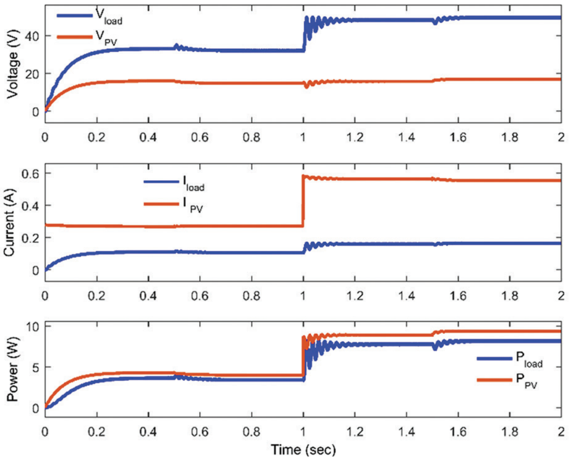


**Fig. 7.** The load and solar panel $V$, $I$ and $P$ responses with SVMR models

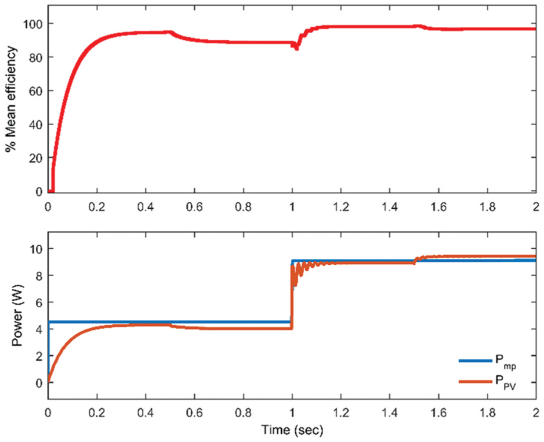


**Fig. 8.** Mean efficiency (%), $P_{mp}$ and $P_{pv}$ waveforms with SVMR models

## 5. PERFORMANCE COMPARISON PROPOSED METHOD WITH EXISTING METHODS

In this section, the results of the proposed control strategy are compared with the classical methods like perturb and observe (P&O) and incremental conductance (IC) and intelligent methods like artificial neural network (ANN) and fuzzy logic control (FLC).

### 5.1. WITH P&O METHOD

The P&O algorithm [1] controls the duty cycle ($D$) of the converter depending on the PV panel's present voltage and power values. The predicted maximum power $P_{mp}$ by the SVMR model, proposed SVMR strategy ($P_{svmr}$), and P&O method responses are compared in Fig. 9. The $P_{p\&o}$ response has continuous oscillations near the MPP. In contrast, the proposed SVMR methodology response is not having any oscillations in the steady state. Therefore, the SVMR method operates the solar panel almost nearer to MPP, even in variable $I_r$ and $T$ presence.

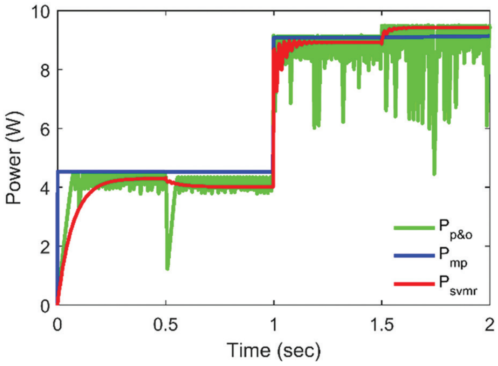


**Fig. 9.** Comparative plot for $P_{mp}$, $P_{svmr}$ and $P_{p\&o}$

### 5.2. WITH IC METHOD

The IC method [2] controls the converter's D depending on the voltage and current values of the PV panel. The predicted $P_{mp}$ by the SVMR model, IC algorithm ($P_{IC}$), and proposed SVMR strategy ($P_{svmr}$) responses are compared in Fig. 10. The $P_{IC}$ response has continuous oscillations near the MPP. On the other hand, the proposed SVMR methodology operates the solar panel nearer to MPP with no fluctuations under variable climatic conditions in the steady state.

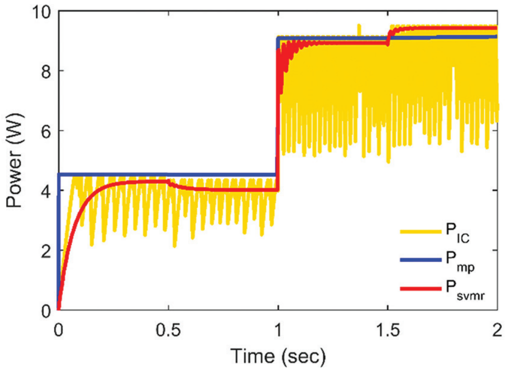


**Fig. 10.** Comparative plot for $P_{mp}$, $P_{svmr}$ and $P_{IC}$

### 5.3. WITH ANN METHOD

The proposed control strategy results are compared with the perceptron type ANN MPPT [26, 29]. The ANN was trained with the same data used for SVMR model training. The ANN model's inputs are $I_r$, $T$, and outputs are $P_{mp}$, $V_{mp}$. Ten hidden layer and two output layer neurons make up the ANN architecture [29]. The data were decomposed to training data, validating data, and testing data for the ANN model in 60%, 20%, and 20%, respectively.

In Fig. 5, the SVMR ML model is replaced with the trained ANN model for MPPT. The $P_{mp}$ predicted by the SVMR model, ANN algorithm ($P_{nn}$), and proposed SVMR strategy ($P_{svmr}$) are compared in Fig. 11. The ANN algorithm works at MPP for low values of $I_r$. But if there is a huge change in the value of $I_r$ the ANN algorithm has large magnitude continuous oscillations, and for high values of $I_r$, the power response has small fluctuations near MPP in the steady state. The proposed SVMR approach provides the operation of the PV panel nearly at MPP with a small residual value under variable $I_r$ and $T$ in the steady state.

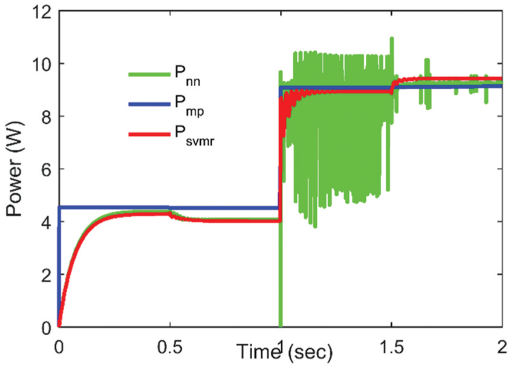


**Fig. 11.** Comparative plot for $P_{mp}$, $P_{svmr}$ and $P_{nn}$

### 5.4. WITH FLC METHOD

Fuzzy logic control (FLC) [1, 27] handles the system's nonlinearities in a better way, no need a precise mathematical model and also works with defective inputs. The variation in $D$ ($\Delta D$) is FLC output. The duty ratio for the converter is determined by Eq.(18). Fig. 12 shows the triangular membership functions of FLC. The variables negative big & small (NB & NS), zero (ZE), and positive big & small (PB & PS) are allotted to membership functions with fuzzy subsets. Table-3 provides the rule base of FLC.

$$D(k+1)=D(k)+\Delta D \qquad (18)$$

The predicted $P_{mp}$ by the SVMR model, $P_{svmr}$, and FLC method ($P_{flc}$), are compared in Fig. 13. The FLC response is similar to that of the SVMR method. But the FLC performance is based on the designed rule base, which needs humanoid experience and expertise. In Fig.13 (the portion in zoom) it is seen that if there is a huge increment in the $I_r$, there is a short duration overshoot with the FLC method. On the other hand, the SVMR method does not have it.

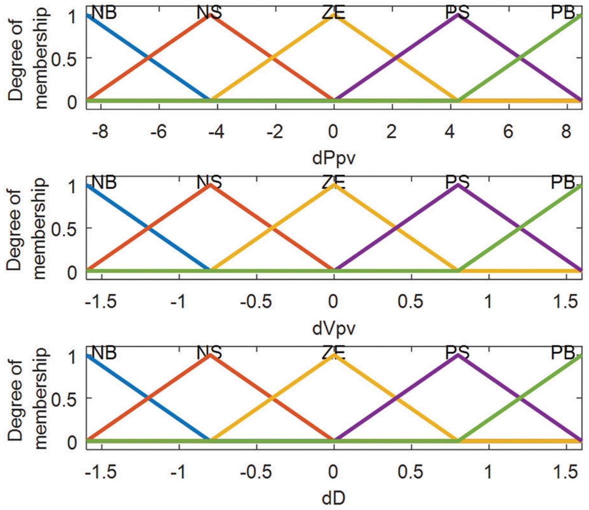


**Fig. 12.** $\Delta P_{pv}$, $\Delta V_{pv}$, and $\Delta D$ membership functions

**Table 3.** Fuzzy rule base

| **Output** ***ΔD*** | | **Input-2** ***ΔVpv*** | | | | |
|---|---|---|---|---|---|---|
| | | PB | PS | ZE | NS | NB |
| **Input-1** ***ΔPpv*** | NB | NS | NB | NB | PB | PS |
| | NS | NS | NS | NS | PS | PS |
| | ZE | ZE | ZE | ZE | ZE | ZE |
| | PS | PS | PS | PS | NS | NS |
| | PB | PB | PS | PS | NB | NS |

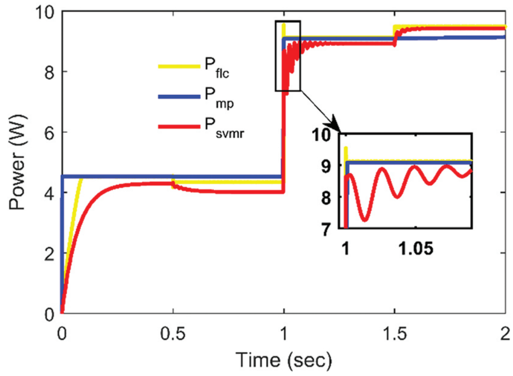


**Fig.13.** Comparative plot for $P_{mp}$, $P_{svmr}$ and $P_{flc}$

The predicted $P_{mp}$ by the SVMR model, $P_{svmr}$, and FLC method ($P_{flc}$), are compared in Fig. 13. The FLC response is similar to that of the SVMR method. But the FLC performance is based on the designed rule base, which needs humanoid experience and expertise. In Fig.13 (the portion in zoom) it is seen that if there is a huge increment in the $I_r$, there is a short duration overshoot with the FLC method. On the other hand, the SVMR method does not have it.

### 5.5. POWER RESPONSE COMPARISON DURING 0 TO 0.5 SEC (INTERVAL-1)

The SVMR model dynamic power response was compared with a few models in literature as a graphical in Fig. 14 and numerically as time-domain values in Table-4 during the time interval-1. Fig. 14 demonstrates that, in the steady state, the IC and P&O techniques have oscillatory responses while the other methods do not exhibit them.

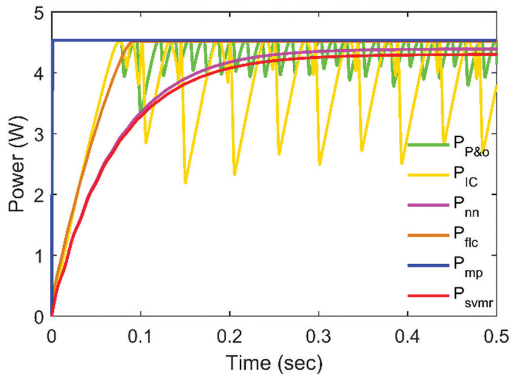


**Fig. 14.** PV power response comparison for various methods (interval-1)

Table-4 shows that, as compared to P&O, the proposed SVMR model response has settled approximately half as fast with a superior settling power of 3.8960 W and no overshoot. Compared to incremental conductance method, the SVMR model response has settled almost in half time with a better final value and with no overshoot. Regarding settling time, final value, and overshoot, the SVMR model response beats the P&O and IC techniques. The SVMR model power response numerical values are nearly similar to the intellectual method ANN. The FLC model power response is superior in numerical during 0 to 0.5 sec, but the FLC response depends on the strength of the rule base, which requires human experience and expertise. This comparative analysis shows that the proposed SVMR control strategy is good at chasing the MPP for PV systems under variable weather situations.

**Table 4.** MPP tracking response numerical comparison for various approaches

| Parameter | SVMR | P&O | IC | ANN | FLC |
|---|---|---|---|---|---|
| Rise Time (sec) | 0.1558 | 0.0519 | 0.0470 | 0.1541 | 0.0690 |
| Peak Time (sec) | 0.5 | 0.4989 | 0.1327 | 0.5 | 0.5 |
| Peak value (W) | 4.3288 | 4.5211 | 4.5211 | 4.4195 | 4.5512 |
| Settling Min. (W) | 3.8960 | 3.3161 | 2.1899 | 3.9777 | 4.1186 |
| Settling Time (sec) | 0.2846 | 0.5 | 0.4992 | 0.2762 | 0.0868 |
| Undershoot (%) | 0 | 0 | 0 | 0 | 0 |
| Overshoot (%) | 0 | 9.1727 | 18.8405 | 0 | 0 |

## 6. CONCLUSION

In this work, a new SVMR machine learning-based approach for MPPT of the solar panel is used in association with a PWM control boost converter. The mean efficiency value was determined to be greater than 94 per cent in steady state to confirm the efficacy of the SVMR algorithm. The SVMR approach has produced better MPPT outcomes than traditional perturb and observe and incremental conductance algorithms, intellectual prediction artificial neural network and fuzzy logic control algorithms, and even under dynamic climate. Furthermore, the simulation results demonstrate greater accuracy in tracking and working the system at MPP with the proposed SVMR control strategy in the steady state.

## 7. REFERENCES

[1] A. S. Mahdi, K. Mahamad, S. Saon, T. Tuwoso, H. Elmunsyah, S. W. Mudjanarko, "Maximum power point tracking using perturb and observe, fuzzy logic and ANFIS", SN Applied Sciences, Vol. 2, No. 1, 2020, pp.1-9.

[2] L. Shang, H. Guo, W. Zhu, “An improved MPPT control strategy based on incremental conductance algorithm”, Protection and Control of Modern Power Systems, Vol. 5, No. 1, 2020, pp.1-8.

[3] C. González-Castaño, L. L. Lorente-Leyva, J. Muñoz, C. Restrepo, D. H. Peluffo-Ordóñez, "An MPPT strategy based on a surface-based polynomial fitting for solar photovoltaic systems using real-time hardware", Electronics, Vol. 10, No. 2, 2021, 206.

[4] X. Li, H. Wen, C. Zhao, "Improved beta parameter based MPPT method in photovoltaic system", Proceedings of the 9th International Conference on Power Electronics and ECCE Asia, Seoul, Korea, 1-5 June 2015, pp. 1405-1412.

[5] V. R. Kota, M. N. Bhukya, "A simple and efficient MPPT scheme for PV module using 2-dimensional lookup table", Proceedings of the IEEE Power and Energy Conference at Illinois, Urbana, IL, USA, 19-20 February 2016, pp. 1-7.

[6] M. Kaffash, M. H. Javidi, A. Darudi, "A combinational maximum power point tracking algorithm in photovoltaic systems under partial shading conditions", Proceedings of the Iranian Conference on Renewable Energy & Distributed Generation, Mashhad, Iran, 2-3 April 2016, pp. 103-107.

[7] D. Baimel, S. Tapuchi, Y. Levron, J. Belikov, "Improved fractional open circuit voltage MPPT methods for PV systems", Electronics, Vol. 8, No. 3, 2019, p. 321.

[8] H. A. Sher, A. F. Murtaza, A. Noman, K. E. Addoweesh, K. Al-Haddad, M. Chiaberge, "A new sensorless hybrid MPPT algorithm based on fractional short-circuit current measurement and P&O MPPT", IEEE Transactions on Sustainable Energy, Vol. 6, No. 4, 2015, pp. 1426-1434.

[9] K. Amara, T. Bakir, A. Malek, D. Hocine, E. B. Bourennane, A. Fekik, M. Zaouia, "An Optimized Steepest Gradient Based Maximum Power Point Tracking for PV Control Systems", International Journal on Electrical Engineering & Informatics, Vol. 11, No. 4, 2019, pp. 662-683.

[10] L. Zhang, W. G. Hurley, W. H. Wölfle, "A new approach to achieve maximum power point tracking for PV system with a variable inductor", IEEE Transactions on Power Electronics, Vol. 26, No. 4, 2010, pp. 1031-1037.

[11] S. Hadji, J. P. Gaubert, F. Krim, "Real-time genetic algorithms-based MPPT: study and comparison (theoretical an experimental) with conventional methods", Energies, Vol. 11, No. 2, 2018, p. 459.

[12] G. S. Krishnan, S. Kinattingal, S. P. Simon, P. S. R. Nayak, "MPPT in PV systems using ant colony optimisation with dwindling population", IET Renewable Power Generation, Vol. 14, No. 7, 2020, pp. 1105-1112.

[13] M. Alshareef, Z. Lin, M. Ma, W. Cao, "Accelerated particle swarm optimization for photovoltaic maximum power point tracking under partial shading conditions", Energies, Vol. 12, No. 4, 2019, 623.

[14] K. Atici, I. Sefa, N. Altin, "Grey wolf optimization based MPPT algorithm for solar PV system with SEPIC converter", Proceedings of the 4th International Conference on Power Electronics and their Applications, Elazig, Turkey,25-27 September 2019, pp. 1-6.

[15] M. I. Mosaad, M. abed el-Raouf, M. A. Al-Ahmar, F. A. Banakher, "Maximum power point tracking of PV system based cuckoo search algorithm; review and comparison", Energy Procedia, Vol. 162, 2019, pp. 117-126.

[16] L. P. N. Jyothy, M. R. Sindhu, "An Artificial Neural Network based MPPT Algorithm For Solar PV System", Proceedings of the 4th International Conference on Electrical Energy Systems, Chennai, India, 07-09 February, 2018, pp. 375-380.

[17] N. Dharani Kumar, T. A. Ramesh Kumar, A.R.K. Rao, "A Brief Review on Conventional and Renewable Power Generation Scenario in India", Proceedings of the 2nd Electric Power and Renewable Energy Conference, Lecture Notes in Electrical Engineering, Vol. 812, February 2022.

[18] S. Vunnam, M. Vanitha Sri, A.R.K. Rao, "Performance analysis of mono crystalline, poly crystalline and thin film material based 6 × 6 T-C-T PV array under different partial shading situations", Optik, Vol. 248, 2021, p.168055.

[19] R. Kandipati, T. Ramesh, "Maximum power enhancement under partial shadings using a modified Sudoku reconfiguration", CSEE Journal of Power and Energy Systems, Vol. 7, No. 6, 2021, pp. 1187-1201.

[20] T. Ramesh, K. Rajani, A. K. Panda, "A novel triple-tied-cross-linked PV array configuration with reduced number of cross-ties to extract maximum power under partial shading conditions", CSEE Journal of Power and Energy Systems, Vol. 7, No. 3, 2021, pp. 567-581.

[21] K. Rajani, T. Ramesh, "Reconfiguration of PV arrays (TCT, BL, HC) by considering wiring resistance", CSEE Journal of Power and Energy Systems, Vol. 8, No. 5, 2022, pp. 1408-1416.

[22] S. Chatterjee, R. Chaudhuri, S. Kamble, S. Gupta, U. Sivarajah, "Adoption of Artificial Intelligence and Cutting-Edge Technologies for Production System Sustainability: A Moderator-Mediation Analysis", Information Systems Frontiers, 2022, pp. 1-16.

[23] M. Takruri et al. "Maximum power point tracking of PV system based on machine learning", Energies, Vol. 13, No. 3, 2020, p. 692.

[24] Y. Chou, "Maximum Power Point Tracking of Photovoltaic System Based on Reinforcement Learning," Sensors, Vol. 19, No. 22, 2019, p. 5054.

[25] H. Shareef, A. H. Mutlag, A. Mohamed, "Random Forest-Based Approach for Maximum Power Point Tracking of Photovoltaic Systems Operating under Actual Environmental Conditions", Computational Intelligence and Neuroscience, Vol. 2017, 2017, 17 pages.

[26] K. Bingi, B. R. Prusty, "Chaotic time series prediction model for fractional-order duffing's oscillator", Proceedings of the 8th International Conference on Smart Computing and Communications, Kochi, Kerala, India, 1-3 July 2021, pp. 357-361.

[27] P. V. Mahesh, S. Meyyappan, R. K. R. Alla, "A New Multivariate Linear Regression MPPT Algorithm for Solar PV System With Boost Converter", ECTI Transactions on Electrical Engineering, Electronics, and Communications, Vol. 20, No. 2, 2022, pp. 269-281.

[28] V. Tamrakar, S. C. Gupta, Y. Sawle, "Single-diode PV cell modeling and study of characteristics of single and two-diode equivalent circuit", Electrical and Electronics Engineering: An International Journal, Vol. 4, No.3, 2015, p. 12.

[29] P. V. Mahesh, S. Meyyappan, A. R. K. Rao, "Maximum Power Point Tracking with Regression Machine Learning Algorithms for Solar PV Systems", International Journal of Renewable Energy Research, Vol. 12, No. 3, 2022, pp. 1327-1338.

[30] R. Ayop, C. W. Tan, "Design of boost converter based on maximum power point resistance for photovoltaic applications", Solar Energy, Vol. 160, 2018, pp. 322-335.

[31] B. Ergun, T. Kavzoglu, I. Colkesen, C. Sahin, "Data filtering with support vector machines in geometric camera calibration", Optics express, Vol. 18, No.3, 2010, pp. 1927-1936.

[32] H. Hong, B. Pradhan, D. T. Bui, C. Xu, A. M. Youssef, W. Chen, "Comparison of four kernel functions used in support vector machines for landslide susceptibility mapping: a case study at Suichuan area (China)", Geomatics, Natural Hazards and Risk, Vol. 8, No. 2, 2017, pp. 544-569.

[33] M. H. Rashid, "Power electronics: circuits, devices, and applications", 4th Edition, Pearson Education India, 2009.

# Design and Implementation of Real-time Fault Location Experimental System for Teaching and Training University Students

Case Study

**Ngo Minh Khoa**
Quy Nhon University,
Faculty of Engineering and Technology
170 An Duong Vuong, Quy Nhon city, Vietnam
ngominhkhoa@qnu.edu.vn

**Doan Duc Tung**
Quy Nhon University,
Faculty of Engineering and Technology
170 An Duong Vuong, Quy Nhon city, Vietnam
doanductung@qnu.edu.vn

***Abstract*** *– The fault location problem is one of the most important issues in power system operation and control. To obtain an experimental system for teaching and training electrical engineering students, this paper performed a study to design and implement a real-time fault location laboratory-scale model from practical hardware components. The impedance-based fault location method is embedded in the system to determine the distance to fault in the transmission line. Furthermore, the monitoring and controlling program is designed by the Matlab App Designer – A new professional app to create the graphical user interface and use the integrated editor quickly. Several fault types including three-phase fault, phase-to-phase fault, phase-to-ground fault are created to evaluate the performance of the fault location experimental system. The real-time measurement results which are acquired and observed on the user guide interface of the program confirm the effectiveness of the experimental system; therefore, the system can be considered a powerful tool for electrical engineering students.*

***Keywords****: fault location, impedance method, transmission line, communication protocols, Matlab App Designer*

## 1. INTRODUCTION

Power transmission lines play a crucial role in a power system. They are used to connect power stations and substations, and for connections between substations as well. They aim to efficiently transmit large amounts of power energy at high voltages [1-3]. Depending on the voltage level, the transmission line has different characteristics, i.e., long distances. Although a short circuit rarely occurs during a power system operation process, it is one of the most extremely serious errors. A short circuit is caused by many different sources and has a lot of effect on the power system [4-6], such as blackout, instability, interruption, etc.

Several methods have been developed to locate the fault in transmission and distribution lines [7-11]. The method based on reactance measurement has been proposed to locate a fault on overhead transmission lines of alternating current electrified railway [12]. Ref. [13] proposed the least-square method based on impedance to locate a fault in the ungrounded grids. In Ref. [14], the impedance-based method was modified for estimating the distance to the fault in transmission lines in the presence of the fault current limiting. On the other hand, the travelling-based fault location methods have been proposed for estimating the fault location in the distribution network, such as the unsynchronized- and synchronized-based methods [15-19], the adaptive convolution neural network-based method [20, 21], and the single-end traveling wave fault location method [17, 22, 23]. For hybrid distribution lines, the authors in [24] proposed the single-ended fault location method based on the characteristic distribution of traveling wave along the transmission line.

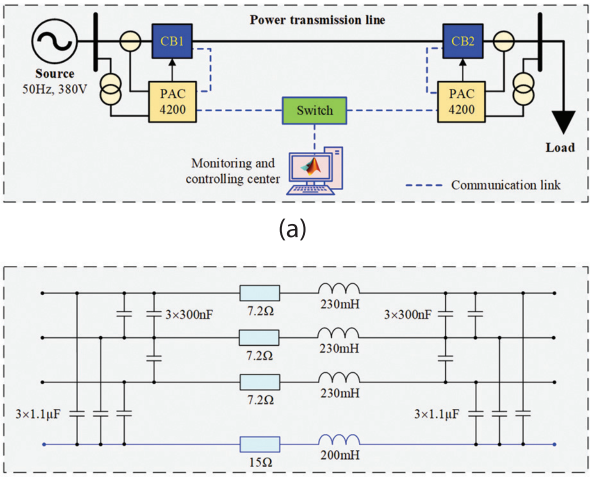
For universities, electrical engineering students need to improve their knowledge about the fault location system for power transmission lines. Due to cost and space requirements as well as the complexity of a practical power system, the university students have less chance to interact with a practical power system to study and verify the theory that they learned in the lectures. Therefore, laboratory experiments are one of the best approaches to link the theory with the hands-on skills of the students. For most students, laboratories are their first practical experiences and their excellent chances for not only achieving optimum learning experiences, but also developing valuable skills for future employment. Besides, the university can improve the quality of training of future electrical engineers by promoting their creativity and self-learning. One of the most important subjects for electrical engineering students is the relay protection in power system. The lessons of this subject are designed to help the students answering the following questions: "Which main components are included in a fault location system on transmission line?", "Which signals from the measurement devices are applied for locating fault?", "How does the fault location system respone when a fault occurs in the transmission line?", "How can the system detect and identify accurately the fault location on the transmission line?", etc. These above teaching questions can be answered via the simulation programs such as: EMTP, Matlab/Simulink programs, etc.; however, the biggest limited issue of these simulation programs is the measurement data and control signals not in real-time. To help student get more theoretical and practical knowledge about the fault location system, the main objective of this paper is to design and implement a real-time fault location laboratory-scale system for a power transmission line supplying a load via only one source. This experimental system is built using hardware devices of the authors' smart grid laboratory for university students. In addition, its software is programmed on the Matlab platform for detecting, identifying, and locating faults which occur in a power transmission line.

## 2. DESIGN AND IMPLEMENTATION OF EXPERIMENTAL SYSTEM

### 2.1. HARDWARE UNIT

In a laboratory range, with a safety low voltage 380/220 V, the paper designs an experimental system based on the hardware components for locating faults in a power transmission line as shown in Fig. 1. The hardware devices in Fig. 1a consist of an adjustable three-phase power supply, a type-Pi transmission line module, a load module, two circuit breakers, two power quality meters, a switch unit. These devices are available in the authors' smart grid laboratory, so students can easily access them to carry out various experiments for the designed system. Moreover, the personal computer, which is communicated to the devices (the two meters and two circuit breakers) via the Ethernet protocol, is used to represent the monitoring and controlling center. The algorithm for detecting, identifying, and locating faults on the transmission line is implemented on this computer and is run in real-time. The main specifications of the other hardware components in the experimental system are described as follows:

Firstly, the adjustable three-phase power supply is a voltage source which can adjust its output voltage in a range from 0 to 450V line-to-line voltage. This device is manufactured by Lucas Nuller, Germany. Its current rating is 5A and it is integrated with an overcurrent protection relay. In the fault location experimental system, this power supply is utilized to supply an appropriate voltage setpoint for several experiments. Secondly, the two solid-state switch modules are used to replace two circuit breakers in the system. These modules are also manufactured by Lucas Nuller, Germany. They are located at the two ends to protect when a fault occurs in the line. Besides, they can communicate with the control center via the Ethernet interface; therefore, their IP address will be set before they can receive the commands from the monitoring and controlling center. Thirdly, to measure, monitor, and control the experimental system, two PAC4200 power quality meters are also connected at the two ends of the line. These meters can measure all power quality parameters such as the frequency, voltage, current, power, etc. In addition, they can be communicated with the monitoring and controlling center via the Ethernet communication protocol, so every meter is also set to an appropriate IP address. In this experimental system, the two power quality meters continuously measure the parameters in real-time and send these parameters to the monitoring and controlling center to detect, identify, and locate accurately faults in the power transmission line. Observing Fig. 1, the current and voltage transformers are depicted to create the secondary measurement signals as the inputs of the PAC4200 power quality meters; however, all the hardware devices selected to establish this experimental system are in the laboratory range having the voltage and current ratings of 400V and 5A, respectively.



(a)

(b)

**Fig. 1.** a) The schematic diagram of the real-time fault location experimental system, b) The detail model of the transmission line

The three-phase power transmission line utilized in this experimental system is a module which is manufactured by Lucas Nuller, Germany.

This module is also designed for applications in the smart grid laboratory, and it is based on the type-Pi transmission line model as shown in Fig. 1b. The specifications of this module consist of the line length 300km, the resistance 7.2Ω, the inductance 230mH, the capacitance between phase to ground 1.1μF, and the capacitance between phase-to-phase 300ηF. The real-time fault location experimental system is shown in Fig. 2.

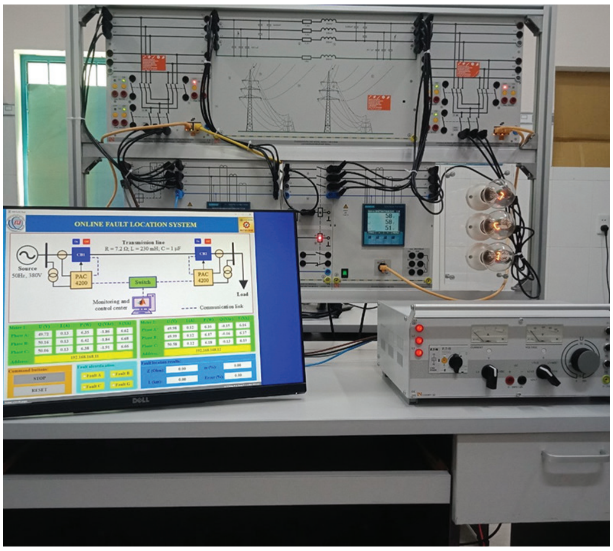
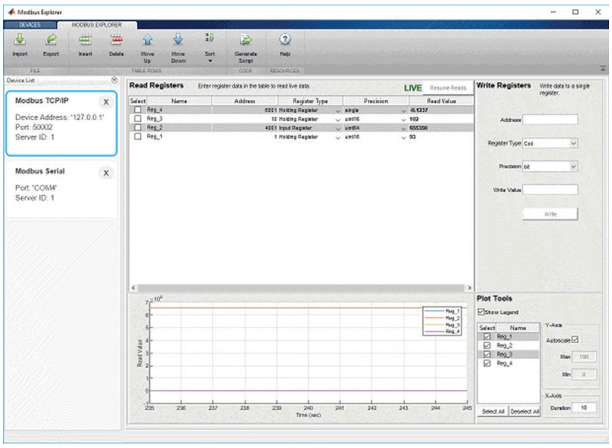


**Fig. 2.** The hardware unit of the real-time fault location experimental system

### 2.2. SOFTWARE UNIT

In this paper, the 2021a version of Matlab software was applied to design the fault location program on the transmission line. In this version, Matlab sofware is equipmented some advanced packages such as Modbus Explorer, App Designer, etc. Using the packages to test easily the connection between the hardware platforms and the software program. In addition, they can be used to design the friendly user guide for monitoring and controlling the fault location system. Moreover, the students can easily use the commands of Matlab to create a private program to test and record the responses of the hardware components in real-time.

The Modbus Communication toolbox of Matlab supports Modbus interaction via the TCP/IP or Serial RTU protocol. It can be used to communicate with the Modbus servers such as PLC, DSP, etc. as well as for reading data from a measurement device, controlling, and monitoring the temperature and humidity from the sensors. Besides, the coils and registers can be read by using the Modbus Explore toolbox that has a friendly user guide interface. Therefore, the Modbus Explorer was used to check the read and write commands from the hardware units including: the PAC4200 power quality meters and circuit breaker modules via the Ethernet network. The user guide interface of Modbus Explorer is shown in Fig. 3. The command syntaxes in Modbus Communication are used to declare, read, and write as shown in Table 1.



Modbus TCP/IP
Device Address: '127.0.0.1'
Port: 50002
Server ID: 1

Modbus Serial
Port: 'COM4'
Server ID: 1

Read Registers — Enter register data in the table to read live data.

| Select | Name | Address | Register Type | Precision | Read Value |
|---|---|---|---|---|---|
| ☐ | Reg_4 | 6001 | Holding Register | single | -0.1237 |
| ☐ | Reg_3 | 10 | Holding Register | uint16 | 102 |
| ☐ | Reg_2 | 4001 | Input Register | uint64 | 655350 |
| ☐ | Reg_1 | 1 | Holding Register | uint16 | 93 |

Write Registers — Write data to a single register.
Address:
Register Type: Coil
Precision: bit
Write Value:

☑ Show Legend

| Select | Name |
|---|---|
| ☑ | Reg_1 |
| ☑ | Reg_2 |
| ☑ | Reg_3 |
| ☑ | Reg_4 |

Y-Axis: Autoscale ☑ Max 100 Min 0
X-Axis: Duration 10

**Fig. 3.** The Modbus Explorer application to check the read and write commands

**Table 1.** The commands of Modbus Communication

| Syntax | Function |
|---|---|
| modbus | Create Modbus object |
| read | Read data from a Modbus server |
| write | Perform a write operation<br>to the connected Modbus server |
| writeRead | Perform a write then read operation on groups of<br>holding registers in a single Modbus transaction |
| maskWrite | Perform mask write operation on a holding register |
| instrhwinfo | Information about available hardware |
| clear | Remove instrument objects from Matlab workspace |

### 2.3. IMPLEMENTED ALGORITHM

The signal processing technique applied in the PAC4200 power quality meters is Discrete Fourier Transform (DFT). The voltage and current signals are sampled and processed to calculate the parameters for identifying and locating faults on the transmission line. The background of the DFT is briefed as follows [25]:

The continuous-time voltage and current signals as the inputs of the power quality meters can be defined with Fourier series representation. In addition, these discrete-time signals can be represented within finite duration in practice. An alternative transformation is called DFT for a finite-length signal, which is discretized in frequency. The frequency range of discrete-time signals is defined over the interval between –π to π. A periodic digital signal consisted of N samples is to separate the frequency components into 2π/N radians intervals by dividing the frequency-domain. Then, Fourier series representation of the discrete-time signal will consist of N frequency components [25]. The general Fourier series representation of a periodic signal (x(n)) is expressed as:

$$x(t) = \sum_{k=0}^{N-1} c_k e^{jk(2\pi/N)n} \tag{1}$$

where $N$ is the harmonic index related with the exponentials function ($e^{jk(2\pi/N)n}$) for $k = 0, 1, \ldots, N$-1, ck is the coefficients of the Fourier series.

The coefficients ck are calculated as:

$$c_k = \frac{1}{N}\sum_{N=0}^{N-1} x(n)e^{-jk(2\pi/N)n} \qquad (2)$$

The coefficients $c^k$ of Fourier series are the form of a periodic sequence of fundamental period $N$. The timedomain spectrum of a periodic signal can be represented as periodic sequence with DFT. The frequency analysis of discrete-time periodic signals (sin(nt) and cos(nt)) involves Fourier transform of the time-domain signal. DFT is defined as multiplication of $N$ samples $x(n)$ with $N$ discrete frequencies. These samples are taken at discrete frequencies ($\omega^k = 2\pi k/N$), where $k$ = 0, 1, ..., $N$-1, between $0 \le \omega \le 2\pi$. This means that $X(\omega)$ is evaluated at the successive samples by equally spaced frequencies. $X(\omega)$ is given in the following.

$$X(\omega_k) = \sum_{n=-\infty}^{\infty} x(n)e^{-jn(2\pi/N)kn} \qquad (3)$$

DFT is a mapping between N samples $x(n)$ of the timedomain into $N$ samples $X(\omega)$ of the frequency-domain. This gives the opportunity to compute DFT of the periodic and the finite-length signals. The frequency-domain spectrum of a periodic sequence can be re-obtained as the periodic signal by using inverse discrete-time Fourier transform (IDFT). IDFT can be defined by using the frequency samples of $X(k)$. It is given in the following.

$$X(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k)W_N^{-kn}, n = 0, 1, \ldots, N-1 \qquad (4)$$

IDFT shows that there is no loss information by transforming the frequency spectrum of X(k) back into the original time sequence of x(n).

**a) The button START**

When the button START is pressed, the program will perform to get the IP addresses of the two PAC4200 meters. The program then initiates the Modbus communication, reads measurement data from the registers of the two meters via the Ethernet network, sets the read values to the user guide interface of the program. If a short circuit occurs, the current at the beginning of the line will be compared with the pickup value to send the trip command to the two circuit breakers. When the button START is pressed, the program will perform the function **STARTButtonValueChanged**(app, event) as follows:

```
function STARTButtonValueChanged(app, event)
value = app.STARTButton.Value;
    if value == 1
      app.STARTButton.Text = 'STOP';
    else
      app.STARTButton.Text = 'START';
    end
    Add_1 = app.Address1.Value;
    m1 = modbus('tcpip', Add_1, 502);
    Add_2 = app.Address2.Value;
        m2 = modbus('tcpip', Add_2, 502);
        ok = 1;
        while ok == 1
  Data1 = read(m1, 'holdingregs', 2, 18, 'single');
  Data2 = read(m2, 'holdingregs', 2, 18, 'single');
  app.Ua1.Value = Data1(1);
  app.Ub1.Value = Data1(2);
  app.Uc1.Value = Data1(3);
  app.Ia1.Value = Data1(7);
  app.Ib1.Value = Data1(8);
  app.Ic1.Value = Data1(9);
  app.Sa1.Value = Data1(10);
  app.Sb1.Value = Data1(11);
  app.Sc1.Value = Data1(12);
  app.Pa1.Value = Data1(13);
  app.Pb1.Value = Data1(14);
  app.Pc1.Value = Data1(15);
  app.Qa1.Value = Data1(16);
  app.Qb1.Value = Data1(17);
  app.Qc1.Value = Data1(18);
  app.Ua2.Value = Data2(1);
  app.Ub2.Value = Data2(2);
  app.Uc2.Value = Data2(3);
  app.Ia2.Value = Data2(7);
  app.Ib2.Value = Data2(8);
  app.Ic2.Value = Data2(9);
  app.Sa2.Value = Data2(10);
  app.Sb2.Value = Data2(11);
  app.Sc2.Value = Data2(12);
  app.Pa2.Value = Data2(13);
  app.Pb2.Value = Data2(14);
  app.Pc2.Value = Data2(15);
  app.Qa2.Value = Data2(16);
  app.Qb2.Value = Data2(17);
  app.Qc2.Value = Data2(18);
  value = app.STARTButton.Value;
  if value == 0
  ok = 0;
  app.Notification.Text = '';
  clear m1 m2;
  end
  mode = app.Faultlocation.Value;
  if (max(Data1(7:9)) >= 0.35) & (mode == 1)
  ok = 0;
  app.Notification.Text = '';
  clear m1 m2;
  app.m1.Value = Data1(1)/Data1(7);
  app.STARTButton.Text = 'START';
  app.STARTButton.Value = 0;
  OffCB1ButtonPushed(app, event);
  OffCB2ButtonPushed(app, event);
  end
  end
end
```

### b) The button STOP

The buttons STOP and START are designed by a button on the user guide interface. If the button STOP is pressed, the program will stop the while loop that is running.

### c) The button RESET

The function of the button RESET is to reset all values on the user guide interface to the initial values.

### d) The button for closing the CB1

When pressing the button ON above the circuit breaker CB1, the program will initialize the Modbus communication with the IP address of the CB1: 192.168.168.15. The program will then perform the commands to write the value 1 on the coils 1 and 5; other coils will be written with the value 0. Therefore, the circuit breaker CB1 will be closed to supply the transmission line. At that time, the color of the circuit breaker CB1 will be changed to the green color. The function **OnCB1ButtonPushed**(app, event) to close the circuit breaker CB1 at the beginning of the line as follows:

```
function OnCB1ButtonPushed(app, event)
        m_cb = modbus('tcpip', '192.168.168.15', 502);
        write(m_cb, 'coils', 1, 1)
        write(m_cb, 'coils', 2, 0)
        write(m_cb, 'coils', 3, 0)
        write(m_cb, 'coils', 4, 0)
        write(m_cb, 'coils', 5, 1)
        write(m_cb, 'coils', 6, 0)
        app.LabelCB1.BackgroundColor = 'Blue';
end
```

### e) The button for tripping the CB1

When pressing the button OFF above the circuit breaker CB1, the program will initialize the Modbus communication with the IP address of the CB1: 192.168.168.15. The program will then perform the commands to write the value 1 on the coils 1 and 6; other coils will be written with the value 0. Therefore, the circuit breaker CB1 will be tripped. At that time, the color of the circuit breaker CB1 will be changed to the red color. The function OffCB1ButtonPushed(app, event) to trip the circuit breaker CB1 at the beginning of the line as follows:

```
function OffCB1ButtonPushed(app, event)
        m_cb = modbus('tcpip', '192.168.168.15', 502);
        write(m_cb, 'coils', 1, 1)
        write(m_cb, 'coils', 2, 0)
        write(m_cb, 'coils', 3, 0)
        write(m_cb, 'coils', 4, 0)
        write(m_cb, 'coils', 5, 0)
        write(m_cb, 'coils', 6, 1)
        app.LabelCB1.BackgroundColor = 'Red';
end
```

### f) The button for closing the circuit breaker CB2

When pressing the button ON above the circuit breaker CB2, the program will initialize the Modbus communication with the IP address of the CB2: 192.168.168.16. The program will then perform the commands to write the value 1 on the coils 1 and 5; other coils will be written by the value 0. Therefore, the circuit breaker CB2 will be closed to supply the load. At that time, the color of the circuit breaker CB2 will be changed to the green color. The function OnCB2ButtonPushed(app, event) to close the circuit breaker CB2 at the ending of the line as follows:

```
function OnCB2ButtonPushed(app, event)
        m_cb = modbus('tcpip', '192.168.168.16', 502);
        write(m_cb, 'coils', 1, 1)
        write(m_cb, 'coils', 2, 0)
        write(m_cb, 'coils', 3, 0)
        write(m_cb, 'coils', 4, 0)
        write(m_cb, 'coils', 5, 1)
        write(m_cb, 'coils', 6, 0)
        app.LabelCB2.BackgroundColor = 'Blue';
end
```

### g) The button for tripping the CB2

When pressing the button OFF above the circuit breaker CB2, the program will initialize the Modbus communication with the IP address of the CB2: 192.168.168.16. The program will then perform the commands to write the value 1 on the coils 1 and 6, other coils will be written by the value 0. Therefore, the circuit breaker CB2 will be tripped. At that time, the color of the circuit breaker CB2 will be changed to the red color. The function OffCB2ButtonPushed(app, event) to trip the circuit breaker CB2 at the ending of the line as follows:

```
function OffCB2ButtonPushed(app, event)
        m_cb = modbus('tcpip', '192.168.168.16', 502);
        write(m_cb, 'coils', 1, 1)
        write(m_cb, 'coils', 2, 0)
        write(m_cb, 'coils', 3, 0)
        write(m_cb, 'coils', 4, 0)
        write(m_cb, 'coils', 5, 0)
        write(m_cb, 'coils', 6, 1)
        app.LabelCB2.BackgroundColor = 'Red';
end
```

The above codes are programmed in Matlab sofware to design and implement the fault location experimental system via the algorithm as shown in Fig. 4. All measurement data from the power meters are continuously read to calculate the fault location results when a fault occurs on the transmission line. In addition, the control buttons on the user guide are programmed by the codes in each function to run the program.

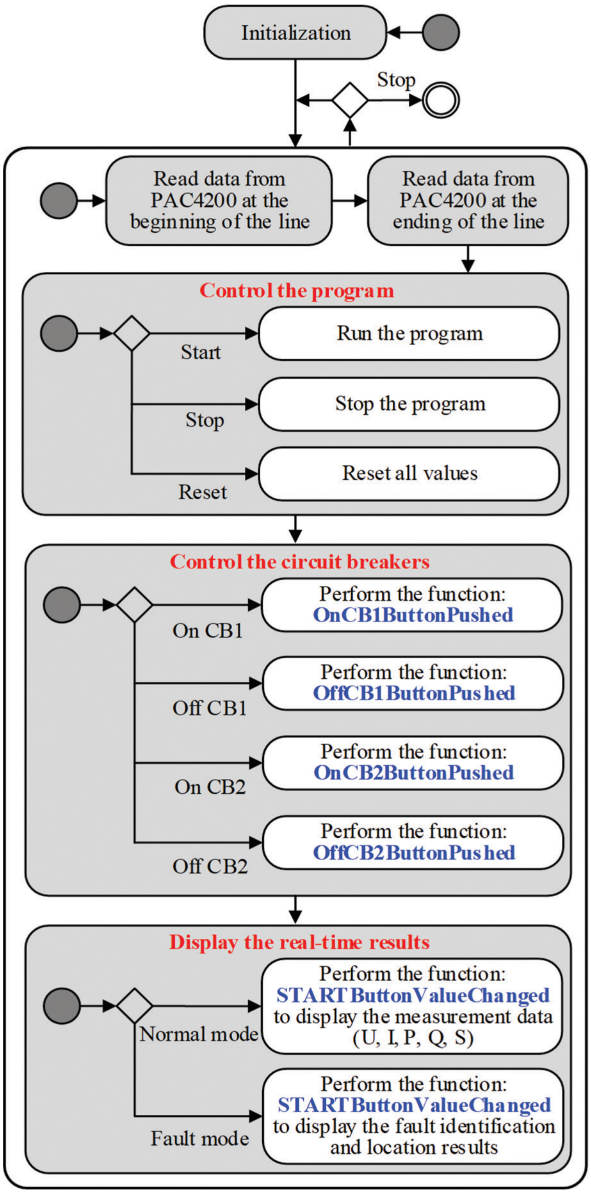

Initialization
Stop
Read data from PAC4200 at the beginning of the line
Read data from PAC4200 at the ending of the line
Control the program
Start
Run the program
Stop
Stop the program
Reset
Reset all values
Control the circuit breakers
On CB1
Perform the function: OnCB1ButtonPushed
Off CB1
Perform the function: OffCB1ButtonPushed
On CB2
Perform the function: OnCB2ButtonPushed
Off CB2
Perform the function: OffCB2ButtonPushed
Display the real-time results
Normal mode
Perform the function: STARTButtonValueChanged to display the measurement data (U, I, P, Q, S)
Fault mode
Perform the function: STARTButtonValueChanged to display the fault identification and location results

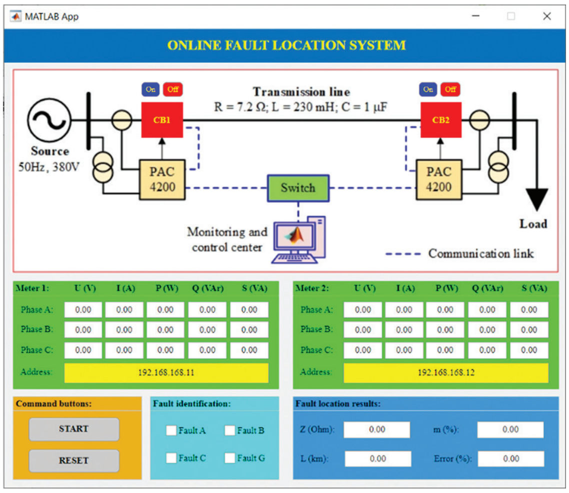
**Fig. 4.** The algorithm of the real-time fault focation experimental system

The Matlab App Designer is used to design the user guide interface of the fault location program on the transmission line. After all hardware devices are correctly connected according to the diagram schematic in Fig. 1, a computer is used as a center unit to acquire and process all the signals from the hardware devices. The user guide interface of the fault location program is designed as shown in Fig. 5. The upper side of the user guide interface shows the diagram schematic and the buttons ON and OFF for closing/tripping the two circuit breakers of the transmission line. To correctly connect the program to the hardware devices of the experimental system via the Ethernet protocol, the IP addresses of the two PAC4200 power quality meters must be exactly declared on the user guide interface. In addition, the IP addresses of the two circuit breakers are established in this program. The IP addresses of the hardware devices are connected via the Ethernet protocol as shown in Table 2.

**Table 2.** The IP addresses of the hardware devices

| No. | Hardware device | IP address |
|---|---|---|
| 1 | The PAC4200 power quality meter at the beginning of the line | 192.168.168.11 |
| 2 | The PAC4200 power quality meter at the ending of the line | 192.168.168.12 |
| 3 | The circuit breaker at the beginning of the line | 192.168.168.15 |
| 4 | The circuit breaker at the ending of the line | 192.168.168.16 |

On the user guide interface, the open and closed statuses of the circuit breakers are remotely controlled via the On and Off. Moreover, the off and on statuses are represented by the red and green colors, respectively. The measurement data including voltage, current, active power, reactive power, and apparent power from the two PAC4200 power quality meters are continuously read in real time to display on the user guide interface. Therefore, the user can easily monitor the operation of the transmission line. In the fault mode, the center processing unit will detect and send the trip command to trip simultaneously the two circuit breakers of the line. At the same time, the fault location methods will use the measurement data from the PAC4200 meters to determine the fault location and to display the experimental results on the user guide interface.



| Meter 1: | U (V) | I (A) | P (W) | Q (VAr) | S (VA) |
|---|---|---|---|---|---|
| Phase A: | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| Phase B: | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| Phase C: | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| Address: | 192.168.168.11 | | | | |

| Meter 2: | U (V) | I (A) | P (W) | Q (VAr) | S (VA) |
|---|---|---|---|---|---|
| Phase A: | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| Phase B: | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| Phase C: | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| Address: | 192.168.168.12 | | | | |

Fault location results:

| | | | |
|---|---|---|---|
| Z (Ohm): | 0.00 | m (%): | 0.00 |
| L (km): | 0.00 | Error (%): | 0.00 |

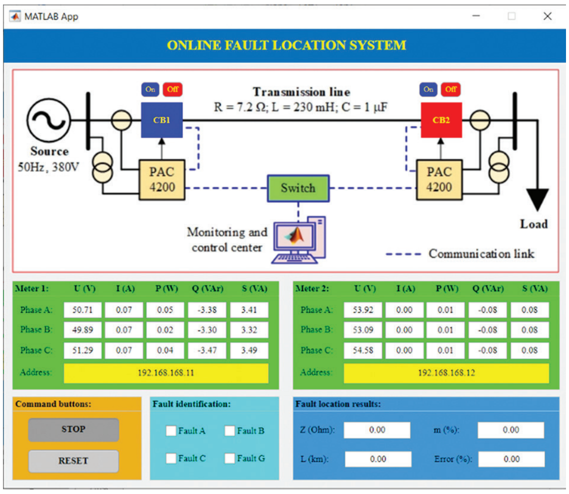
**Fig. 5.** The user guide of the program

## 3. EXPERIMENTAL RESULTS AND DISCUSSION

To carry out experiments, the hardware devices are connected according to the schematic diagram as shown in Fig. 1. The IP addresses of these power quality meters, and circuit breakers are established and communicated with the monitoring and controlling center via the Ethernet communication protocol.

### 3.1. NORMAL OPERATION MODE WITHOUT LOAD

Controlling the circuit breakers remotely at two ends of the transmission line is performed via the buttons ON and OFF on the user guide interface. When the program is started, the circuits is at the open status; therefore, the user allows to close the circuit breaker CB1 to supply power for the transmission line. The line is working without load and all measurement data in this case is monitored and displayed on the user guide interface as shown in Fig. 6. Observing on Fig. 6, it can be clearly seen that the currents at the PAC4200 power quality meter at the beginning of the line are $I_A$ = 0.07 A, $I_B$ = 0.07 A, and $I_C$ = 0.07 A for the phases A, B, and C respectively. These currents of three phases are due to the pi-model of the line, the reactive power of the line is injected into the system. As a result, the voltages, the currents, active powers, reactive powers, and apparent powers at the beginning of the line are shown in Fig. 6. Moreover, as the reactive power injected by the line, the voltages at the ending of the line are $U_A$ = 53.92 V, $U_B$ = 53.09 V, and $U_C$ = 54.58 V for the phases A, B, and C, respectively while the voltages at the beginning of the line are $U_A$ = 50.71 V, UB = 49.89 V, and $U_C$ = 51.29 V for the phase A, B, and C, respectively. Therefore, in this case the voltages at the ending of the line are higher than the voltages at the beginning of the line.



| Meter 1: | U (V) | I (A) | P (W) | Q (VAr) | S (VA) |
|---|---|---|---|---|---|
| Phase A: | 50.71 | 0.07 | 0.05 | -3.38 | 3.41 |
| Phase B: | 49.89 | 0.07 | 0.02 | -3.30 | 3.32 |
| Phase C: | 51.29 | 0.07 | 0.04 | -3.47 | 3.49 |
| Address: | 192.168.168.11 | | | | |

| Meter 2: | U (V) | I (A) | P (W) | Q (VAr) | S (VA) |
|---|---|---|---|---|---|
| Phase A: | 53.92 | 0.00 | 0.01 | -0.08 | 0.08 |
| Phase B: | 53.09 | 0.00 | 0.01 | -0.08 | 0.08 |
| Phase C: | 54.58 | 0.00 | 0.01 | -0.08 | 0.08 |
| Address: | 192.168.168.12 | | | | |

□ Fault A □ Fault B
□ Fault C □ Fault G

Fault location results:

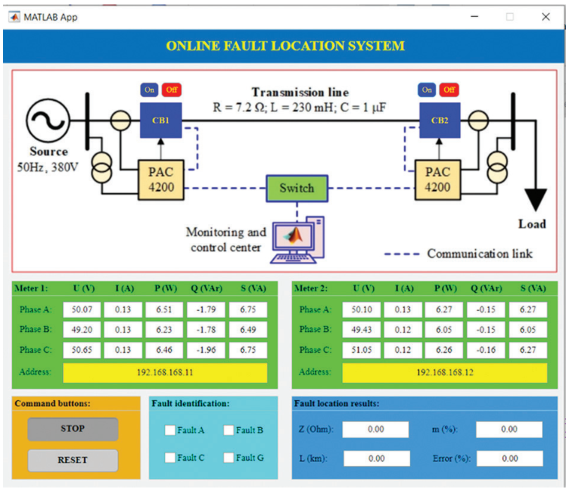
| | | | |
|---|---|---|---|
| Z (Ohm): | 0.00 | m (%): | 0.00 |
| L (km): | 0.00 | Error (%): | 0.00 |

**Fig. 6.** The operation mode of the line without load

## 3.2. NORMAL OPERATION MODE WITH LOAD

In this mode, the circuit breaker at the ending of the line is closed to supply power to the resistance load. The load is represented by three 220 V, 60 W bulbs. The experimental results of this mode are presented in Fig. 7. It is clear that the currents at the beginning of the line ($I_A$ = 0.13 A, $I_B$ = 0.13 A, and $I_C$ = 0.13 A) and at the ending of line ($I_A$ = 0.13 A, $I_B$ = 0.12 A, and $I_C$ = 0.12 A). In addition, the active, reactive, and apparent powers are also shown in Fig. 7.

To supply the load at the ending of the transmission line, the user must close the circuit breaker CB2 at the ending of the line. The resistance loads in this case study are three 220 V, 60 W bulbs. The currents will be increased depending on the resistance loads. As a result, the monitoring results in this case are displayed in Fig. 7. From Fig. 7, the currents at the beginning of the line are $I_A$ = 0.13 A, $I_B$ = 0.13 A, and $I_C$ = 0.13 A for the phases A, B, and C, respectively and the currents at the ending of the line are $I_A$ = 0.13 A, $I_B$ = 0.12 A, and $I_C$ = 0.12 A. Besides, the active powers, reactive powers, and apparent powers are also shown on the user guide interface. For the voltages, it can be clearly seen that the voltages at the beginning of the line ($U_A$ = 50.07 V, $U_B$ = 49.20 V, and $U_C$ = 50.65 V) are higher than the voltages at the ending of the line ($U_A$ = 50.10 V, $U_B$ = 49.43 V, and $U_B$ = 51.05 V).



| Meter 1: | U (V) | I (A) | P (W) | Q (VAr) | S (VA) |
|---|---|---|---|---|---|
| Phase A: | 50.07 | 0.13 | 6.51 | -1.79 | 6.75 |
| Phase B: | 49.20 | 0.13 | 6.23 | -1.78 | 6.49 |
| Phase C: | 50.65 | 0.13 | 6.46 | -1.96 | 6.75 |
| Address: | 192.168.168.11 | | | | |

| Meter 2: | U (V) | I (A) | P (W) | Q (VAr) | S (VA) |
|---|---|---|---|---|---|
| Phase A: | 50.10 | 0.13 | 6.27 | -0.15 | 6.27 |
| Phase B: | 49.43 | 0.12 | 6.05 | -0.15 | 6.05 |
| Phase C: | 51.05 | 0.12 | 6.26 | -0.16 | 6.27 |
| Address: | 192.168.168.12 | | | | |

□ Fault A □ Fault B
□ Fault C □ Fault G

Fault location results:

| | | | |
|---|---|---|---|
| Z (Ohm): | 0.00 | m (%): | 0.00 |
| L (km): | 0.00 | Error (%): | 0.00 |

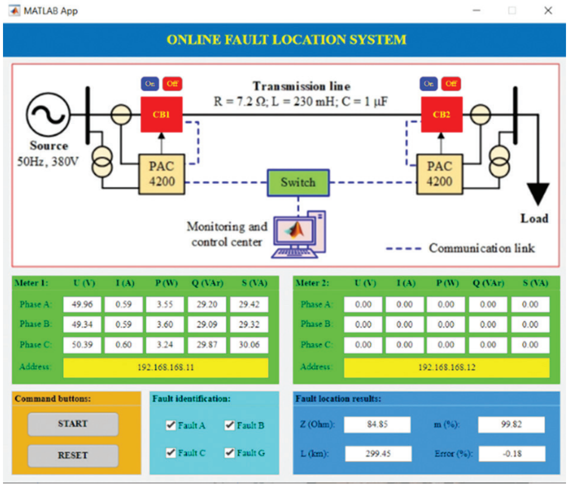
**Fig. 7.** The operation mode of the line with load

## 3.3. FAULT MODE

To establish faults in the experimental system, one side of a circuit breaker is connected to the ending of the line and the other side of the circuit breaker is connected to obtain several fault types including: ABCG, AB, AC, BC, AG, BG, and CG. As shown in Fig. 1, the schematic diagram of the real-time fault location experimental system consists of a type-Pi transmission line model. The limitation of this experimental system is only one transmission line model; therefore, the end of the line can be only applied to investigate the performance of this system. In addition, it is assumed that the faults are solid faults with the fault resistance of $R_F$ = 0Ω. The experimental results for these faults are shown in the figures below.

Fig. 8 shows the monitoring, identifying, and locating experimental results when the three-phase fault ABCG at the ending of the line with the fault resistance of RF = 0Ω. It can be clearly seen that all measurements at the ending of the line are zero because the voltages and currents equal zero. However, the currents at the beginning of the line are the three-phase fault currents ($I_A$ = 0.59 A, $I_B$ = 0.59 A, and $I_C$ = 0.59 A). These current values are higher than the pickup value Iset = 0.25 A; therefore, the program detects the fault and then sends the signal to trip two circuit breakers of the line (CB1 and CB2 are changed to the open status). In addition, the three-phase fault ABCG is clearly shown in the frames of fault identification and the fault location results on the user guide interface are at the location with $m$ = 99.82%, $L$ = 299.45 km, and Error = -0.18%.

In this paper, two unbalanced faults including the BC fault and AG fault are created and shown in Fig. 9 and Fig. 10, respectively. The fault location results are 97.32% for the BC fault and 100.09% for the AG fault. On the other hand, the identification results and the status of the circuit breakers are displayed accurately on the user guide interface for each fault.



| Meter 1: | U (V) | I (A) | P (W) | Q (VAr) | S (VA) |
|---|---|---|---|---|---|
| Phase A: | 49.96 | 0.59 | 3.55 | 29.20 | 29.42 |
| Phase B: | 49.34 | 0.59 | 3.60 | 29.09 | 29.32 |
| Phase C: | 50.39 | 0.60 | 3.24 | 29.87 | 30.06 |
| Address: | 192.168.168.11 | | | | |

| Meter 2: | U (V) | I (A) | P (W) | Q (VAr) | S (VA) |
|---|---|---|---|---|---|
| Phase A: | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| Phase B: | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| Phase C: | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| Address: | 192.168.168.12 | | | | |

Command buttons:

Fault identification:

☑ Fault A ☑ Fault B

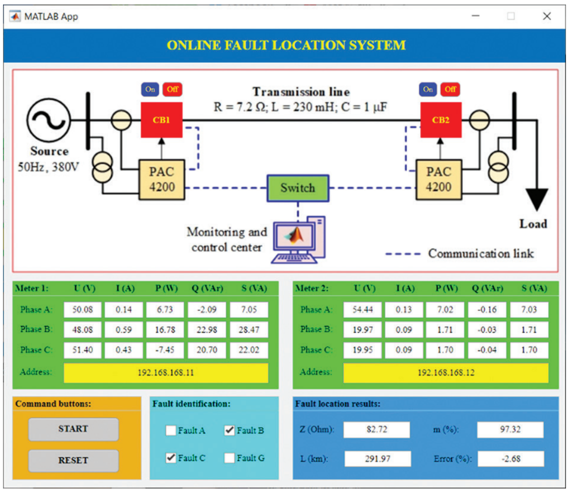
☑ Fault C ☑ Fault G

Fault location results:

Z (Ohm): 84.85 m (%): 99.82

L (km): 299.45 Error (%): -0.18

**Fig. 8.** The experimental result of the ABCG fault



| Meter 1: | U (V) | I (A) | P (W) | Q (VAr) | S (VA) |
|---|---|---|---|---|---|
| Phase A: | 50.08 | 0.14 | 6.73 | -2.09 | 7.05 |
| Phase B: | 48.08 | 0.59 | 16.78 | 22.98 | 28.47 |
| Phase C: | 51.40 | 0.43 | -7.45 | 20.70 | 22.02 |
| Address: | 192.168.168.11 | | | | |

| Meter 2: | U (V) | I (A) | P (W) | Q (VAr) | S (VA) |
|---|---|---|---|---|---|
| Phase A: | 54.44 | 0.13 | 7.02 | -0.16 | 7.03 |
| Phase B: | 19.97 | 0.09 | 1.71 | -0.03 | 1.71 |
| Phase C: | 19.95 | 0.09 | 1.70 | -0.04 | 1.70 |
| Address: | 192.168.168.12 | | | | |

Command buttons:

Fault identification:

☐ Fault A ☑ Fault B

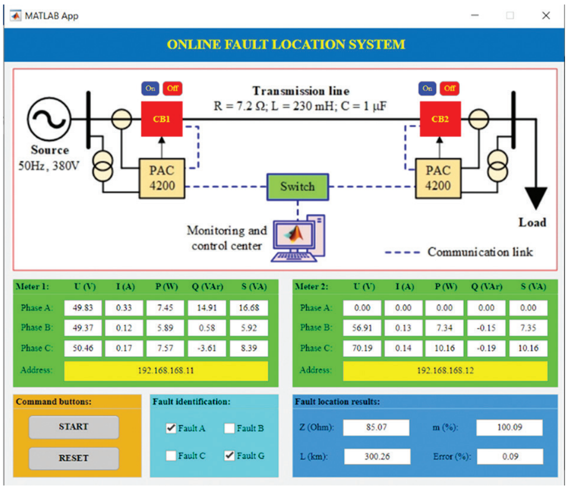
☑ Fault C ☐ Fault G

Fault location results:

Z (Ohm): 82.72 m (%): 97.32

L (km): 291.97 Error (%): -2.68

**Fig. 9.** The experimental result of the BC fault



| Meter 1: | U (V) | I (A) | P (W) | Q (VAr) | S (VA) |
|---|---|---|---|---|---|
| Phase A: | 49.83 | 0.33 | 7.45 | 14.91 | 16.68 |
| Phase B: | 49.37 | 0.12 | 5.89 | 0.58 | 5.92 |
| Phase C: | 50.46 | 0.17 | 7.57 | -3.61 | 8.39 |
| Address: | 192.168.168.11 | | | | |

| Meter 2: | U (V) | I (A) | P (W) | Q (VAr) | S (VA) |
|---|---|---|---|---|---|
| Phase A: | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| Phase B: | 56.91 | 0.13 | 7.34 | -0.15 | 7.35 |
| Phase C: | 70.19 | 0.14 | 10.16 | -0.19 | 10.16 |
| Address: | 192.168.168.12 | | | | |

Command buttons:

Fault identification:

☑ Fault A ☐ Fault B

☐ Fault C ☑ Fault G

Fault location results:

Z (Ohm): 85.07 m (%): 100.09

L (km): 300.26 Error (%): 0.09

**Fig. 10.** The experimental result of the AG fault

The fault location results for the different fault types are also carried out in this paper and they are shown in Table 3. These experimental results confirm that the designed system can help electrical engineering students to practice with all fault types.

**Table 3.** The results for the different fault types

| Fault type | Z (Ω) | m% | L (km) | Error (%) |
|---|---|---|---|---|
| ABCG | 84.85 | 99.82 | 299.45 | -0.18 |
| AB | 85.02 | 100.02 | 300.07 | 0.02 |
| BC | 82.72 | 97.32 | 291.97 | -2.68 |
| AC | 85.49 | 100.57 | 301.71 | 0.57 |
| AG | 85.07 | 100.09 | 300.26 | 0.09 |
| BG | 84.45 | 99.36 | 298.07 | -0.64 |
| CG | 84.85 | 99.83 | 299.48 | -0.17 |

## 4. CONCLUSION

The paper presented an idea to design and implement a real-time fault location experimental system in a laboratory range. The hardware devices are connected to build an experimental system in which the type-pi three-phase power transmission line is a main device. Furthermore, the real-time measurement data is continuously monitored at the monitoring and controlling center via the Ethernet communication protocol. The software program with the friendly user guide interface is designed to detect, identify, and locate faults that occur in the power transmission line. The main function of the experimental system is to train the electrical engineering students in how to understand the fault location issue in real power systems.

For future works, the authors will develop the experimental system by additional functions such as varying the fault locations as well as the fault resistances along the line. Machine learning and artificial intelligence methods will be applied in the software unit of the experimental system to help students approach these methods.

## 5. REFERENCES:

[1] S. Das, S. Santoso, A. Gaikwad, M. Patel, “Impedance-based fault location in transmission networks: theory and application”, IEEE Access, Vol. 2, 2017, pp. 537-557.

[2] Y. J. Deng, C. M. Wang, S. Zhang, W. Z. Han, “A fault location algorithm for shunt-compensated lines under dynamic conditions”, International Journal of Electrical Power & Energy Systems, Vol. 143, 2022, pp. 108387.

[3] R. Fan, Y. Liu, R. Huang, R. Diao, S. Wang, “Precise Fault Location on Transmission Lines Using Ensemble Kalman Filter”, IEEE Transactions on Power Delivery, Vol. 33, No. 6, 2018, pp. 3252-3255.

[4] F. V. Lopes, E. J. S. Leite Jr, J. P. G. Ribeiro, A. B. Piardi, A. V. Scheid, G. Zat, R. G.F. Espinoza, "Single-ended multi-method phasor-based approach for optimized fault location on transmission lines", Electric Power Systems Research, Vol. 212, 2022, p. 108361.

[5] C. M. Furse, M. Kafal, R. Razzaghi, Y.-J. Shin, "Fault diagnosis for electrical systems and power networks: A review", IEEE Sensors Journal, Vol. 21, No. 2, 2020, pp. 888-906.

[6] A. N. Sheta, G. M. Abdulsalam, A. A. Eladl, "Online tracking of fault location in distribution systems based on PMUs data and iterative support detection", International Journal of Electrical Power & Energy Systems, Vol. 128, 2021, p. 106793.

[7] A. Gopalakrishnan, M. Kezunovic, S. M. McKenna, D. M. Hamai, "Fault location using the distributed parameter transmission line model", IEEE Transactions on Power Delivery, Vol. 15, No. 4, 2000, pp. 1169-1174.

[8] S. S. Gururajapathy, H. Mokhlis, H. A. Illias, "Fault location and detection techniques in power distribution systems with distributed generation: A review", Renewable and Sustainable Energy Reviews, Vol. 74, 2017, pp. 949-958.

[9] Y. Jia, Y. Liu, B. Wang, D. Lu, Y. Lin, "Power Network Fault Location with Exact Distributed Parameter Line Model and Sparse Estimation", Electric Power Systems Research, 2022, p. 108137. (in press)

[10] S. M. Saad, N. El Naily, F. A. Mohamed, "Investigating the effect of DG infeed on the effective cover of distance protection scheme in mixed-MV distribution network", International Journal of Renewable Energy Development, Vol. 7, No. 3, 2018, pp. 223-231.

[11] A. Swetapadma, S. Chakrabarti, A. Y. Abdelaziz, "Feasibility study of intelligent fault location estimation methods for double-circuit transmission lines", International Transactions on Electrical Energy Systems, Vol. 31, No. 12, 2021, p. e13198.

[12] Z. Han, S. Li, S. Liu, S. Gao, "A reactance-based fault location method for overhead lines of AC electrified railway", IEEE Transactions on Power Delivery, Vol. 35, No. 5, 2020, pp. 2558-2560.

[13] T. Namas, I. Džafić, "Least square method for impedance based fault location in ungrounded networks", Proceedings of the 2nd Global Power, Energy and Communication Conference, Izmir, Turkey, 20-23 October 2020, pp. 274-278.

[14] J. Barati, A. Doroudi, "Novel modified impedance-based methods for fault location in the presence of a fault current limiter", Turkish Journal of Electrical Engineering & Computer Sciences, Vol. 26, No. 4, 2018, pp. 1881-1893.

[15] K. Kalita, S. Anand, S. K. Parida, "A novel non-iterative fault location algorithm for transmission line with unsynchronized terminal", IEEE Transactions on Power Delivery, Vol. 36, No. 3, 2021, pp. 1917-1920.

[16] C. A. Apostolopoulos, C. G. Arsoniadis, P. S. Georgilakis, V. C. Nikolaidis, "Fault location algorithms for active distribution systems utilizing two-point synchronized or unsynchronized measurements", Sustainable Energy, Grids and Networks, Vol. 32, 2022, p. 100798.

[17] F. Xu, X. Dong, "A novel single-ended traveling wave fault location method based on reflected wave-head of adjacent bus", Proceedings of the 12th IET International Conference on Developments in Power System Protection, Copenhagen, Denmark, 31 March - 03 April 2014, pp. 1-5.

[18] J. Xie, G. Jin, Y. Wang, X. Ni, X. Liu, "New Algorithm for 2-terminal Transmission Line Fault Location Integrating Voltage Phasor Feature and Phase Angle Jump Checking", Electric Power Systems Research, Vol. 209, 2022, p. 10797.

[19] M. Majidi, M. Etezadi-Amoli, "A new fault location technique in smart distribution networks using synchronized/ nonsynchronized measurements", IEEE Transactions on Power Delivery, Vol. 33, No. 3, 2017, pp. 1358-1368.

[20] J. Liang, T. Jing, H. Niu, J. Wang, "Two-terminal fault location method of distribution network based on adaptive convolution neural network", IEEE Access, Vol. 8, 2020, pp. 54035-54043.

[21] M.-R. Mosavi, A. Tabatabaei, "Traveling-wave fault location techniques in power system based on wavelet analysis and neural network using GPS timing", Wireless Personal Communications, Vol. 86, No. 2, 2016, pp. 835-850.

[22] R. L. A. Reis, F. V. Lopes, W. L. A. Neves, D. Fernandes Jr., C. M. S. Ribeiro, G. A. Cunha, "An improved single-ended correlation-based fault location technique using traveling waves", International Journal of Electrical Power & Energy Systems, Vol. 132, 2021, p. 107167.

[23] S. Sawai, R. N. Gore, O. D. Naidu, "Novel traveling wave phase component-based fault location of transmission lines", Proceedings of the IEEE International Conference on Power Electronics, Drives and Energy Systems, Jaipur, India, 16-19 December 2020, pp. 1-5.

[24] H. Shu, X. Liu, X. Tian, "Single-Ended Fault Location for Hybrid Feeders Based on Characteristic Distribution of Traveling Wave Along a Line", IEEE Transactions on Power Delivery, Vol. 36, No. 1, 2021, pp. 339-350.

[25] S. M. Kuo, B. H. Lee, "Real-time digital signal processing implementations applications and experiments with the TMS320C55x", John Wiley & Sons, 2001.

# INTERNATIONAL JOURNAL OF ELECTRICAL AND COMPUTER ENGINEERING SYSTEMS

Published by Faculty of Electrical Engineering, Computer Science and Information Technology Osijek,
Josip Juraj Strossmayer University of Osijek, Croatia.

## About this Journal

The International Journal of Electrical and Computer Engineering Systems publishes original research in the form of full papers, case studies, reviews and surveys. It covers theory and application of electrical and computer engineering, synergy of computer systems and computational methods with electrical and electronic systems, as well as interdisciplinary research.

## Topics of interest include, but are not limited to:

- Power systems
- Renewable electricity production
- Power electronics
- Electrical drives
- Industrial electronics
- Communication systems
- Advanced modulation techniques
- RFID devices and systems
- Signal and data processing
- Image processing
- Multimedia systems
- Microelectronics
- Instrumentation and measurement
- Control systems
- Robotics
- Modeling and simulation
- Modern computer architectures
- Computer networks
- Embedded systems
- High-performance computing
- Parallel and distributed computer systems
- Human-computer systems
- Intelligent systems
- Multi-agent and holonic systems
- Real-time systems
- Software engineering
- Internet and web applications and systems
- Applications of computer systems in engineering and related disciplines
- Mathematical models of engineering systems
- Engineering management
- Engineering education

### Paper Submission

Authors are invited to submit original, unpublished research papers that are not being considered by another journal or any other publisher. Manuscripts must be submitted in doc, docx, rtf or pdf format, and limited to 30 one-column double-spaced pages. All figures and tables must be cited and placed in the body of the paper. Provide contact information of all authors and designate the corresponding author who should submit the manuscript to https://ijeces.ferit.hr. The corresponding author is responsible for ensuring that the article's publication has been approved by all coauthors and by the institutions of the authors if required. All enquiries concerning the publication of accepted papers should be sent to ijeces@ferit.hr.

The following information should be included in the submission:

- paper title;
- full name of each author;
- full institutional mailing addresses;
- e-mail addresses of each author;
- abstract (should be self-contained and not exceed 150 words). Introduction should have no subheadings;
- manuscript should contain one to five alphabetically ordered keywords;
- all abbreviations used in the manuscript should be explained by first appearance;
- all acknowledgments should be included at the end of the paper:
- authors are responsible for ensuring that the information in each reference is complete and accurate. All references must be numbered consecutively and citations of references in text should be identified using numbers in square brackets. All references should be cited within the text;
- each figure should be integrated in the text and cited in a consecutive order. Upon acceptance of the paper, each figure should be of high quality in one of the following formats: EPS, WMF, BMP and TIFF;
- corrected proofs must be returned to the publisher within 7 days of receipt.

### Peer Review

All manuscripts are subject to peer review and must meet academic standards. Submissions will be first considered by an editor-in-chief and if not rejected right away, then they will be reviewed by anonymous reviewers. The submitting author will be asked to provide the names of 5 proposed reviewers including their e-mail addresses. The proposed reviewers should be in the research field of the manuscript. They should not be affiliated to the same institution of the manuscript author(s) and should not have had any collaboration with any of the authors during the last 3 years.

### Author Benefits

The corresponding author will be provided with a .pdf file of the article or alternatively one hardcopy of the journal free of charge.

#### Units of Measurement

Units of measurement should be presented simply and concisely using System International (SI) units.

### Bibliographic Information

Commenced in 2010.
ISSN: 1847-6996
e-ISSN: 1847-7003

Published: semiannually

### Copyright

Authors of the International Journal of Electrical and Computer Engineering Systems must transfer copyright to the publisher in written form.

### Subscription Information

The annual subscription rate is 50€ for individuals, 25€ for students and 150€ for libraries.

### Postal Address

Faculty of Electrical Engineering,
Computer Science and Information Technology Osijek,
Josip Juraj Strossmayer University of Osijek, Croatia
Kneza Trpimira 2b
31000 Osijek, Croatia

# IJECES Copyright Transfer Form

(Please, read this carefully)

This form is intended for all accepted material submitted to the IJECES journal and must accompany any such material before publication.

**TITLE OF ARTICLE** (hereinafter referred to as "the Work"):

COMPLETE LIST OF AUTHORS:

The undersigned hereby assigns to the IJECES all rights under copyright that may exist in and to the above Work, and any revised or expanded works submitted to the IJECES by the undersigned based on the Work. The undersigned hereby warrants that the Work is original and that he/she is the author of the complete Work and all incorporated parts of the Work. Otherwise he/she warrants that necessary permissions have been obtained for those parts of works originating from other authors or publishers.

Authors retain all proprietary rights in any process or procedure described in the Work. Authors may reproduce or authorize others to reproduce the Work or derivative works for the author's personal use or for company use, provided that the source and the IJECES copyright notice are indicated, the copies are not used in any way that implies IJECES endorsement of a product or service of any author, and the copies themselves are not offered for sale. In the case of a Work performed under a special government contract or grant, the IJECES recognizes that the government has royalty-free permission to reproduce all or portions of the Work, and to authorize others to do so, for official government purposes only, if the contract/grant so requires. For all uses not covered previously, authors must ask for permission from the IJECES to reproduce or authorize the reproduction of the Work or material extracted from the Work. Although authors are permitted to re-use all or portions of the Work in other works, this excludes granting third-party requests for reprinting, republishing, or other types of re-use. The IJECES must handle all such third-party requests. The IJECES distributes its publication by various means and media. It also abstracts and may translate its publications, and articles contained therein, for inclusion in various collections, databases and other publications. The IJECES publisher requires that the consent of the first-named author be sought as a condition to granting reprint or republication rights to others or for permitting use of a Work for promotion or marketing purposes. If you are employed and prepared the Work on a subject within the scope of your employment, the copyright in the Work belongs to your employer as a work-for-hire. In that case, the IJECES publisher assumes that when you sign this Form, you are authorized to do so by your employer and that your employer has consented to the transfer of copyright, to the representation and warranty of publication rights, and to all other terms and conditions of this Form. If such authorization and consent has not been given to you, an authorized representative of your employer should sign this Form as the Author.

Authors of IJECES journal articles and other material must ensure that their Work meets originality, authorship, author responsibilities and author misconduct requirements. It is the responsibility of the authors, not the IJECES publisher, to determine whether disclosure of their material requires the prior consent of other parties and, if so, to obtain it.

- The undersigned represents that he/she has the authority to make and execute this assignment.
- For jointly authored Works, all joint authors should sign, or one of the authors should sign as authorized agent for the others.
- The undersigned agrees to indemnify and hold harmless the IJECES publisher from any damage or expense that may arise in the event of a breach of any of the warranties set forth above.

| | |
|---|---|
| ______________________________ | ______________________________ |
| **Author/Authorized Agent** | **Date** |

**CONTACT**
**International Journal of Electrical and Computer Engineering Systems (IJECES)**
Faculty of Electrical Engineering, Computer Science and Information Technology Osijek
Josip Juraj Strossmayer University of Osijek
Kneza Trpimira 2b
31000 Osijek, Croatia
Phone: +38531224600,
Fax: +38531224605,
e-mail: ijeces@ferit.hr