Hindawi
Security and Communication Networks
Volume 2021, Article ID 9924478, 17 pages
https://doi.org/10.1155/2021/9924478

Research Article

An Efficient Three-Phase Fuzzy Logic Clone Node Detection Model

Sachin Lalar,[1] **Shashi Bhushan**,[2] **Surender Jangra**,[3] **Mehedi Masud**,[4] **and Jehad F. Al-Amri**[5]

[1]*Department of Computer Science and Engineering, I. K. Gujral Punjab Technical University, Kapurthala, Punjab, India*
[2]*Department of Computer Science and Engineering, Amity University, Patna, India*
[3]*Department of Computer Application, Guru Tegh Bahadur College, Bhawanigarh, Punjab, India*
[4]*Department of Computer Science, College of Computers and Information Technology, Taif University, P. O. Box 11099, Taif 21944, Saudi Arabia*
[5]*Department of Information Technology, College of Computers and Information Technology, Taif University, P. O. Box 11099, Taif 21944, Saudi Arabia*

Correspondence should be addressed to Mehedi Masud; mmasud@tu.edu.sa

Received 18 March 2021; Revised 7 April 2021; Accepted 16 April 2021; Published 26 April 2021

Academic Editor: Vijay Kumar

Wireless sensor networks have been deployed in the open and unattended environment where the attacker can capture the sensors and create the replica of captured nodes. As the clone nodes have been considered legitimate nodes, clone nodes can initiate different network attacks. We have designed a three-phase clone node detection method named fuzzy logic clone node detection (FLCND). The first phase of FLCND checks whether any node is missing from the network or not. In the next phase, FLCND finds out whether any missing node has arisen in the network in a stipulated time. If any missing node is alive, there is a possibility the node may be cloned. The information of suspected nodes is entered into the Hot-List, which has been maintained in the network. Phase III uses the suspected list and finds out the possibility of clone node using fuzzy logic. Two different scenarios have been simulated in NS2 to evaluate FLCND. The simulation result shows that the proposed method increases the packet delivery ratio (PDR) and reduces packet loss, end-to-end delay, and energy consumption. The simulation results illustrate that the FLCND method reduces the average power consumption by 27% and increases the detection rate by 46% compared to the existing techniques.

1. Introduction

Wireless sensor networks (WSNs) have small, low-cost, and resource-limited sensor nodes that have frequently been used in numerous surveillance functions. The sensor node is an active device that has a processor, memory, low-power supply, radio link, and actuators [1]. WSNs are susceptible to many types of attacks due to the network's open nature [2]. These attacks are classified into two types: application-based attacks and application-independent attacks. Application-based attacks target any network functions, such as data aggregation, localization, and routing [3]. This paper focuses on a clone node (replication) attack, which is recognized as an independent application attack. In some applications, sensor networks are deployed in an open and unattended environment where an attacker can access and capture the sensors. The attacker creates the replica of captured nodes by collecting the information, such as key and encrypted content, and places the clone nodes inside the network [4]. Adversaries insert these duplicate nodes into the tactical network position and commence more internal attacks. The clone node attack can happen either in a static wireless sensor network (SWSN) or in a mobile wireless sensor network (MWSN). In the former type of WSN, the sensor position is fixed in the network, whereas in MWSN, a sensor can change its position. Sensor nodes can move and exchange information with other sensor nodes in mobile sensor networks [5]. If any network communication channel is weak, the mobile node can be connected to the lost communication channel and improve channel efficiency. Mobility performs an essential factor in the sensor network [6].

An example of a mobile sensor network for wildfire tracking is shown in Figure 1. The motion sensor will preserve a certain distance from the fire and provide updated information to the firefighters. Similarly, if the flame spreads, the motion sensor can track and send the information to the base station. In this example, the sensor node has been replicated and inserted into various positions in the network. These clone nodes may produce false information regarding the fire. Mobile WSNs are vulnerable to clone node attacks. Clones can affect network performance if they cannot remove/detect from the network [7]. Some techniques have been proposed to detect clone node attacks in SWSN [8–21], but these methods do not apply to mobile WSNs. In this paper, we will propose a new clone node detection method for MWSN.

The clone node can also change its position in MWSN, so node replication attacks in MWSNs are more challenging to resolve. Attackers can use these mobile replicated nodes to initiate more covert attacks [22]. The discovery strategy may be used to check whether sensor nodes are found in their original position. However, sensor nodes appear at different locations at different times. Node replication attacks are dangerous in MWSN if it has not been eliminated from the network. It will prompt us to find the solution to detect a replicated node in MWSN. The attacker launches the clone node attack in three steps. In the first step, the attackers steal the sensor node from the network. The next step will generate the clone of the stolen node and then place it in the network. After that, the clone nodes can produce a different type of attack in the network. If we maintain the missing node information in the network, when replicated nodes are inserted back into the network, it will be detected.

A new method, FLCND, is proposed. It works in three phases with the step of generating the clone nodes. Initially, the proposed algorithm finds the node which is missing from the network. After that, the proposed algorithm finds out whether any missing node is to come alive in the network. If any missing node is alive, there is a possibility the node may be cloned. The suspected node's information is entered into the suspected list, which is maintained within the network. Phase III uses the suspected list and finds out the clone node by applying fuzzy logic. In the fuzzy method, the parameters are speed, packet delivery ratio (PDR), false input value, residual power, and delay, which are processed as fuzzy logic input and depend on the outcome module; the clone node is detected from the network.

There are the following contributions that are as follows:

(i) The paper proposes an FLCND-based distributed clone node detection method

(ii) The proposed method can increase the packet delivery ratio and reduce packet loss, energy consumption, and end-to-end delay

(iii) The proposed method does not increase the additional communication cost while increasing the detection rate compared to EDD, XED, HO, and CBCD methods

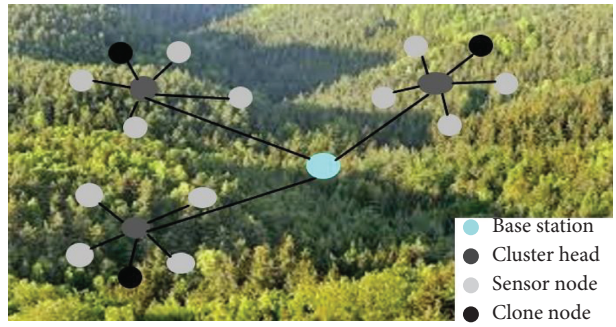

Figure 1: Clone node attack example.

The remainder of the paper is organized as follows. Section 2 reviews existing detection schemes for identifying mobile network cloned nodes. Section 3 describes the system and the attacking model. Section 4 explains the proposed method, fuzzy logic clone node detection (FLCND). Section 5 describes the simulation of the proposed method and the comparison of FLCND with the existing methods. Finally, the paper is concluded in Section 6.

2. Related Work

Different techniques have been invented to detect clone nodes in MWSN, which can be divided into two parts: centralized and distributed. In the centralized MWSN system, all mobile nodes receive information about the clone nodes and transmit the information to the base station, which makes the final decision about the detection of the clone node. On the other hand, the distribution system for identifying a clone node is locally identified by the node [23, 24].

Chia [25] designed the clone node detection method using location information. In this method, each sensor will interchange log lists with neighboring nodes to avoid unauthorized operations. Each node sustains a table that stores information about the nodes used to detect the clone nodes. When monitored nodes meet with each other and exchange recorded information about their IDs, they may find the clone node's conflicting information. Each node acts as a normal node as well as a monitoring node. However, for this method, each node must store a message of each monitored node. The storage overhead of the sensor nodes is high.

Ho et al. [26] proposed a detection scheme based on a probability ratio test. This method's idea is based on the speed of movement not exceeding the maximum speed set in the network for a mobile node. In contrast, clone nodes move much faster than normal nodes as the new clone node's measurement speed appears high to the node's configured maximum speed. When the node speed exceeds the configured speed value, the node's probability value as the clone node is increased. When using SPRT, if the speed is equal to or lower than the maximum speed of the configured system, the null hypothesis is used. If the alternative hypothesis is accepted, then the duplicate node is removed from the network. However, SPRT relies on the base station,

which has limitations such as rapid power loss of nodes near the base station and a single point of failure.

Chia et al. [27] proposed a new method, XED, to detect clone node attacks in MWSNs. The idea behind XED is that, in a nonreplicating network, sensor node, A, encountered another sensor node, B, and A sends a random number, r, to B. When node A meets Bnode again, A will ask for a random number, r, to determine if it is already a matching node. Based on these observations, a "strategy for learning and challenge" has been proposed. The sensor node generates a random number. When sensor nodes want to communicate, they will exchange the generated random number. Each node maintains a table containing the generated received random number and node ID. For a pair of nodes that have already been matched, perform the above steps to replace the random number with the new one.

Yu et al. [28] projected two methods, i.e., EDD and SEDD, to identify replication attacks. It works on an approach that node T that encounters node B must be limited to a number for a given time interval. Each node has the potential to detect duplicates. In EDD, the first phase has calculated the parameters and threshold, which is used to distinguish the actual nodes from the duplicates. In an online phase, each meeting of the node is computed by the node. In EDD, we can see that each node must maintain the list S, resulting in $O(n)$ storage overhead. The basic idea of SEDD is to monitor the subset of nodes instead of all nodes. The number of monitored nodes will be equal to the SEDD program's storage, so the storage overhead is diminished in it.

Deng et al. [29] proposed two schemes, ULTSE and MDLSD, to identify mobile WSN node replication vulnerabilities. As with any agreement, the witness will communicate over the network after receiving the time location statement. The basic idea is to use motion properties. When a node communicates with others, it will track time and location requirements. In other words, if the request for time location is tracked, the witnesses receive and reflect the communication, if they are outside, immediately when the status request witnesses are not sent, but the witness finds the status request for UTLSE multiple locations requiring each of the location claims. The data observed by the node described in the position claim extension are introduced by the method of saving only in the position of the MTLSD claim.

Deng and Xiong [30] projected a new protocol for detecting mobile replicated nodes. Bloom filter and polynomial-based key predelivery schemes are used to find the clone node. The base station finds how many time keys are used. This method runs in four steps: node initialization, pairing, side creation, and discovery. Before setting up the network, symmetric polynomial keys are formed for each node generated by the key server. Each node generates a statement periodically, which contains the ID and the number of keys used. The report was forwarded to the base station. The base station calculates each node's Bloom filter and collects the number of pairs of keys used. Nodes that exceed the limit of the key count are considered clone nodes.

Wang and Shi [31] used the mobile node as a patrol to find distributed clones in various areas of the network. Two detection mechanisms for fixed and mobile systems have been proposed, which include patrol methods. The proposed method identifies duplicates using fixed sensors; if more than two sensors in the same location have equal node ID, then sensors of the same ID will be considered clones. When using a patrol sensor, when the mobile sensor moves at a momentum that exceeds the specified maximum speed, it is considered an attacker node.

Lou et al. [32] proposed a node cloning attack detection protocol for mobile WSN, called single-hop detection (SHD). The node's neighborhood is distinguished by a list of one-hop neighbors available in the regular WSN. Neighboring nodes will be known when the sensor node communicates with other nodes. Each node must sign its neighbor list. When getting a fingerprint complaint from a nearby claim sensor, the receiving node determines that the monitoring node is a clone node.

Shaukat et al. [33] proposed a hybrid method to detect clones in MWSN-based danger theory in the human immune system. The fundamental strategy is to determine the cloned node by the observed abnormal behavior of mobile nodes in the MWSN.

Cheng et al. [34] proposed the NI-LEACH protocol, an improved version of the LEACH protocol. The authors influenced the power consumption of the data transmission and improved the clone node's detection efficiency.

Dong et al. [35] presented a new distributed clone detection protocol known as LSCD. The protocol projects the discovered path of the witness node in which the distance between any two detection paths should be less than the length of the tracking path. The clone detection is also performed in non-hot spot areas and maintains high energy levels that improve energy efficiency and network lifetime.

Anthoniraj and Razak [36] proposed a cluster-based clone detection method, CBCD. In this method, the network is divided into clusters, and each cluster has a cluster head. When the cloned node moves from one cluster to the other, it is identified by the cluster head. Rajesh and Shanmugam [37] proposed the RE-GSASA method in which the authors investigate the simulated model based on GSA to identify clone attack nodes in the network.

Sankar and Roy [38] proposed a CND algorithm based on a Cuckoo filter. The algorithm considers the maximum similarity statements of collaboration spectrum realization decision. The authors enhanced QoS using SDN-based algorithms and located the clones domestically and geographically with a low-cost authentication system.

Conti et al. [39] proposed two clone node detection methods known as HIP and HOP to identify the cloned node in MWNs, which uses the local information and node mobility. The nodes maintain the neighbor information and update the location claim after r number of rounds. The nodes compare their location claims with location claims received from the neighbor. In the HIP, the node compares its location claim only with its neighbor whereas in the HOP, the node compares the received location claim with the other

neighbors. The limitation of this algorithm is that it has high communication, computing, and storage cost.

Manickavasagam and Padmanabhan [40] proposed a new algorithm in mobile WSN to detect the clone nodes. The algorithm is based on the concept that different physical resources are proliferating when multiple clone nodes transmit data with the same source node ID. The algorithm uses the source number in each transmission of the message. If the intermediate node encounters any out-of-order message, it will check whether the source ID is cloned or not. The algorithm's limitation is that it has high communication and memory overload.

Jamshidi et al. proposed a new algorithm in [41] to detect the cloned nodes in a mobile WSN in which watchdog nodes use the learning agent. The watchdog nodes monitor the movement of nodes as well as network traffic. The watchdog changes the status of the learning agent after each monitoring round. The algorithm detects the cloned node by checking the status of the learning agent. The algorithm suffers from low detection and high communication rate when the network consists of a large number of sensor nodes. Jamshidi et al. proposed another watchdog-based algorithm in [42], which uses the node speed to determine the clone nodes in MWSN. If the watchdog node determines that a node is moving faster than a certain limit, the node is considered a replica node. The disadvantages of the algorithm are slow speed, high memory, and computing cost in a dense network.

Jamshidi et al. suggested one more clone node detection algorithm in [43], which uses the mobility model. The sensor node will meet with the same node in each monitoring round. If the node number is higher than a probability value, the node is considered a clone node. The algorithm works in three steps, and in the first step configuration of the watchdog, nodes are there. In the second step, each watchdog monitors the network traffic and records the observation process to estimate the probability value. Watchdog finds the replication node using the probability value calculated in the second step. The proposed method's limitation is that the cloned node cannot be detected if some calculation error is on the probability value. The communication cost is also high.

Anitha et al. [44] proposed three methods, i.e., exponential moving average-based replica detection (EMABRD), SACOP, and FZKA methods, to detect the cloned nodes in MWSN. The main work of the EMARBD algorithm is to compare the actual energy consumption and estimated energy consumption of the sensor node to identify the replication node. A SACOP-based algorithm calculates the trust value of a sensor node from the recommendations of its neighbors. FZKA algorithm relies on fingerprints to identify the clone nodes. The first level is used to verify each node's unique fingerprint, and the second level is used to verify each node's authenticity without sending a personal value. SACOP has a higher clone detection rate as compared to EMABRD and FZKA.

Many of the mobile network's early detection algorithms rely on node mobility and node-to-node communication, which reduces detection if the node moves slowly. This paper is a distributed replica detection program inspired by Ho et al. [26]. Related research of clone node detection in MWSN can be found in [45, 46].

3. System Model

This section explains the network and attack model for the proposed method.

3.1. Network Model. Each mobile sensor node has assigned a unique node ID. We have assumed that the network has a node replica, replicating with the same ID of a node [47, 48]. Each sensor communicates symmetrically and has an information radius. The network is deployed and used the random way motion model. We have assumed that the network is divided into different clusters, and each cluster has a cluster head. The sensor node belongs to anyone cluster. Cluster heads maintain various parameters, i.e., speed, residual energy, delay, packet delivery ratio, and the suspected node's false input value. It has been assumed that all nodes in the MWSN have the same initial energy and the same transmission power. $V_{\max}$ is the upper limit of the speed of node movement. During the simulation, each node begins to move from the starting point to the selected random object point in the simulation area. Table 1 mentions the notations used in the paper [49].

3.2. Attack Model. It has been assumed that the attacker can compromise sensors in the network, and the attacker can execute only a clone node attack. Clone nodes can be set up anywhere in the hostile network [50]. We can only copy legitimate node. It has been assumed that no node with a new node ID cannot insert into the network. Besides, we can use an identity-based public key to allow such nodes to be recognized.

4. Fuzzy Logic-Based Clone Node Detection (FLCND) Scheme

This section explains the new node replication detection method in MWSN. As we know, the clone node attack consists of three main steps as follows:

(a) Attacker first captures the legitimate node from the network

(b) Attacker creates the clone by extracting the information from the captured node and then deploying the network's clones

(c) Then, clone nodes can launch the different attacks inside the network

FLCND method works in three phases, and in each phase, the FLCND method works towards detecting the cloned node in WSN. The FLCND method is divided into three phases, which is explained as follows:

Phase I: find the missing node from the network

The base station initiates the detection phase after deployment of the sensor network. When an attacker

Table 1: Notations and their meanings.

Notations	Meaning
n	Total number of sensor nodes
k	Total number of cluster heads in the network
r_0	Threshold
E_{tp}	Transmitter energy
R	Distance between transmitter and receiver
E_{Ele}	The energy required to operate the transceiver
E_f	Transmitter energy for free space
E_m	Multipath transmitter energy
CH	Cluster head
E_{msg}	The energy required to transmit hello message
H	Number of cluster heads in the node transmitter range
$E_{\text{msg\_CH}}$	The energy required by the cluster head to transmit hello message
CH_t	Relay node
CH_r	Receiver cluster head
$E_{\text{Clone\_Detect}}$	Energy required to detect clone node
E_{SE}	The initial energy of the sensor node
N_F	Number of packets received by a sensor node
N_R	Number of packets received by neighbor nodes
$V_{\max}$	Maximum speed of sensor node

steals any node from the network and creates the replica of legitimate nodes, the complete process of replication will take time, which will be greater than the node's sleeping time. In WSN, the sensor node uses the sleeping time to save the energy/battery. In this phase, each node will check the presence of its neighbor. If any node is not missing from its position or does not give a response after sleeping time, the node will store the suspected node's information and send the information of that node to the cluster head. We will use that information in the second phase for further processing.

Phase II: create the Hot-List

In phase II, the node will check whether any missing node is coming alive or not. If any missing node is alive, that node may be a clone node [51, 52]. The sensor broadcasts the message containing the node ID in the network. The receiving sensors will determine whether the same ID exists in their neighbors or not. If any node ID is presented in the network, then the clone node has been detected from WSN. The information of clone nodes has been sent to the base station for further processing. If the same ID is not in the network, then the information of suspected nodes is entered into the network's Hot-List. There may be a possibility to add the cloned node later in the network. We will use the information entered in the Hot-List in the next phase.

Phase III: fuzzy-based clone node detection

Phase III of the proposed method finds the cloned node using fuzzy logic (Algorithm 1). The identification of clone nodes is predicted based on five parameters such as speed (SP), residual energy (RE), delay (DL), packet delivery rate (PDR), and false input value (FIV). The proposed method FLCND based on the fuzzy system will determine whether the node is a clone or not [53, 54]. The proposed method assumes that each node in MWSN has the same initial energy and transmission range. The four basic components are required to implement the FLCND method are shown in Figure 2 and explained below.

(1) Information: the sensor node sends all the information of suspected nodes from Hot-List in the form of a hello message. This communication occurs between nodes within the framework containing the parameters SP, RE, DL, PDR, and FIV.
(2) Data collection: the cluster head identifies and generates a list based on the hello message. The Hot-Lists, along with parameters, are stored in the database, as is the information for other nodes.
(3) Fuzzy interference system: while collecting data, the information of each suspected node is analyzed using the state of the parameter set. These parameters determine if a clone node exists on the network.
(4) Intra- and intercluster communication: after detecting the cloned node, the clone node's information has been sent to other cluster heads and base stations for further processing. After sending the information, the cloned node will be removed from the sink node's sensor network.

The flowchart of the fuzzy logic-based clone node detection scheme is shown in Figure 3.

4.1. Estimation of Metrics. This section calculates the fuzzification value of each input parameter. First, we will calculate the energy consumption of the sensor node.

4.1.1. Energy Model Analysis. The first step of the analysis is to find the sensor node's energy consumption during the transmission of data. Then, the residual energy is calculated by subtracting the energy consumption from the node's initial energy. Different methods have been proposed to

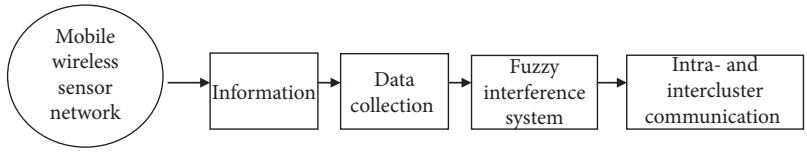

Figure 2: Information flow of the proposed method FLCND.

```
Algorithm: proposed clone node detection algorithm for FLCND
Phase I and II
(1) Begin
(2)   for each node n of the network
(3)     n = Encrypt (ID_i, L_i)//initialize each node with ID and Location
(4)   for each node n of the network
(5)     n[neighbor] = {id_j, L_j, Time_j }//each node finds its neighbor
(6)   For each node x
(7)     Check the response from its neighbor
(8)     if any node x does not respond
(9)       Wait for sleep time
(10)        If response does not come
(11)          Wait for Xn time
(12)          If response comes
(13)            Check the clone of x
(14)            if clone present
(15)              Send information to BS and initiate the trigger revocation procedure
(16)            else
(17)              Addxin the suspected list
(18)          else
(19)            Node xwill be dead and send information to BS and go to step 6
(20)      else
(21)            go to step 6
Phase III
(22)  N = total suspected nodes
(23)  W = alive sensor node in the current round
(24)  for each node [N]
(25)      Cluster head receives message from neighbor of N
(26)    node[N].Info and calculate input parameter: node[N].RE, node[N].PDR node[N].SP, node[N].DL, node[N].FIV
//analysis through fuzzy inference system (FIS)
(27)    node[N].probability = FIS (node[N].RE, node[N].PDR node[N].SP, node[N].DL, node[N].FIV)
(28)    If node[N].probability == High
(29)            node[N].state = Clone
(30)            Advertise Clone_Message and initiate the trigger revocation procedure
(31)    else
(32)            go to step 24
(33) End
```

Algorithm 1: Phase III of the proposed method FLCND.

reduce the energy consumption in WSN [55, 56]. The energy consumption is found out by using a first-order radio model, as mentioned in [57, 58]. When the distance between the transmitter and receiver is less than the threshold r_0, then the data directly communicate between nodes. Otherwise, it will use the multipath fading channel. Equation (1) expresses the transmitter energy (E_{tp}) required for sending an l-bit packet at a distance r between transmitter and receiver. E_{Ele} needs the energy to operate the transceiver, which depends on factors, i.e., digital encoding and modulation, ε_f is the transmitter energy for free space, and ε_m stands for multipath transmitter energy.

$$Etp(l,r) = lE_{\text{Ele}} + l\varepsilon r^{\beta}, \tag{1}$$

$$Etp(l,r) = \begin{cases} lE_{\text{Ele}} + l\varepsilon_f r^2, & r < r_0 \\ lE_{\text{Ele}} + l\varepsilon_m r^4, & r \geq r_0. \end{cases} \tag{2}$$

The threshold r_0 is calculated according to the following formula:

$$r_0 = \sqrt{\frac{\varepsilon_f}{\varepsilon_m}}. \tag{3}$$

The energy consumed by a node after receiving the message is given by

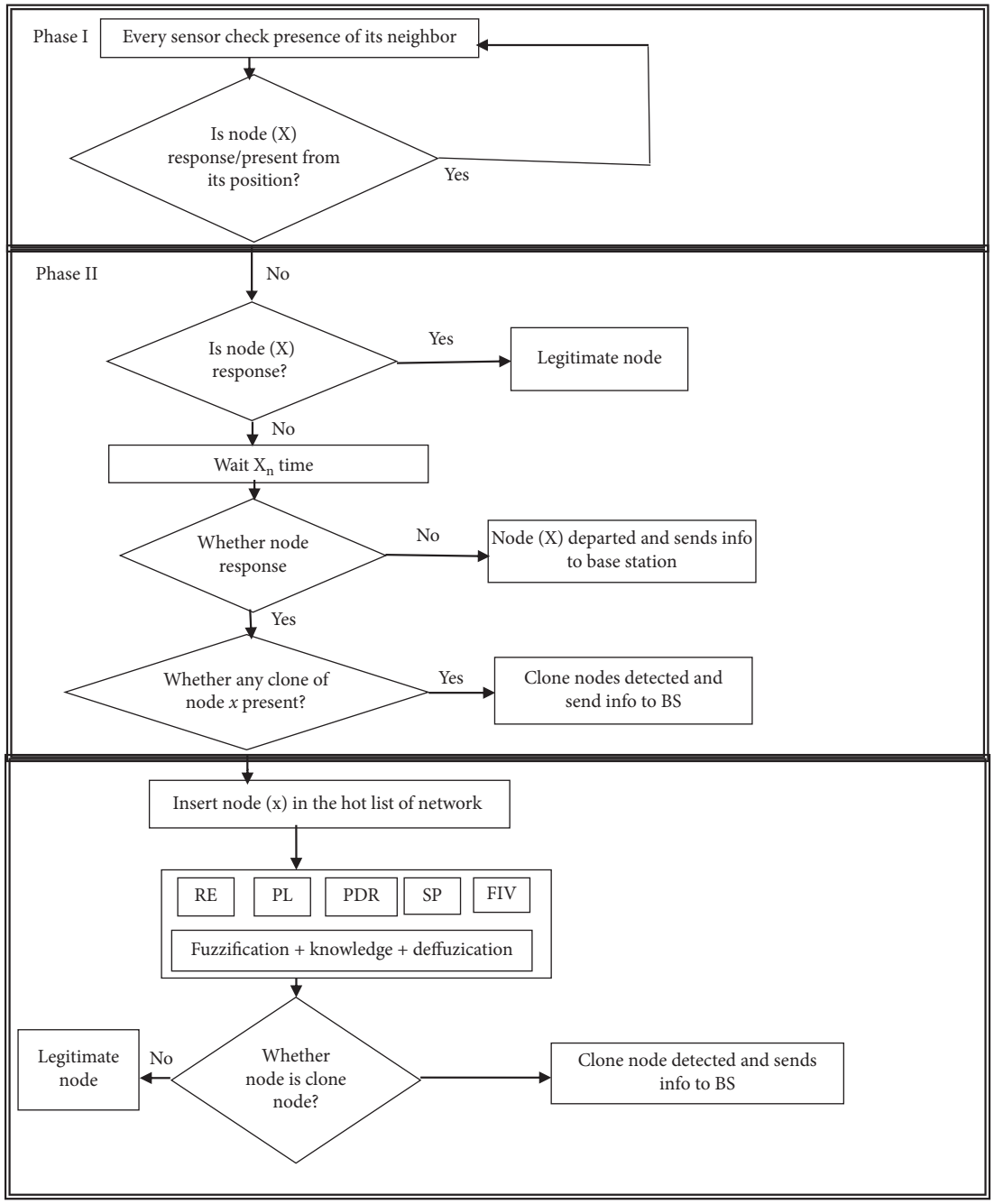

Figure 3: Flowchart of the proposed method FLCND.

$$Erp(l,r) = lE_{\text{Ele}}. \tag{4}$$

Furthermore, the detection process's energy consumption is divided into two phases: clone detection phase and data transmission phase. First, we will compute the energy required to detect the cloned node. When any node is suspected on any node, it will send the information to the cluster head (CH). The nodes will select the CH by sending the hello message among neighbor nodes. The energy required to transmit the hello message by CH is provided by equation (5). The first part calculates the energy required for transmitting the message. The next part represents the energy required to receive a message from other nodes (h):

$$E_{\text{msg}} = ml\left(E_{\text{Ele}} + l\varepsilon_f D_t^2\right) + \frac{h\pi D_t^2}{A^2} lE_{\text{Ele}}. \tag{5}$$

In equation (5), h indicates the number of CH in the range of CH_r, where A is the network region and it also refers to the energy consumed by the CH to transmit the hello message to the other cluster head.

$$E_{\text{msg\_CH}} = kl\left(E_{\text{Ele}} + l\varepsilon_f D_t^2\right) + \frac{n\pi D_t^2}{A^2} klE_{\text{Ele}}. \quad (6)$$

Similarly, equations (5)–(7) state the non-CH energy consumption where n is total nodes and k is cluster heads.

$$E_{\text{msg\_mem}} = (n-k)\left(E_{\text{Ele}} + l\varepsilon_f D_t^2\right) + \frac{k\pi D_t{}^2}{A^2} lE_{\text{Ele}}. \quad (7)$$

When data have been received from the non-CH node, the cluster head (CH) is aggregated, compressed, and sent to the BS or another cluster head. The data are forwarded to the next cluster head or BS that depends on the threshold. For example, if a data packet is transmitted to the CH and its distance from the BS is smaller than TH-BS, the data packet is directly transmitted to the BS. Otherwise, CH forwards to a chosen/relay node from its neighbor. Suppose CH_t as relay/forwarding node. As the free space propagation model has been using, CH_t will directly interact with the BS. The energy consumed by CH_r and CH_t can be given by

$$\begin{aligned} E_{\text{IM}} &= E_{tp}[l, r(\text{CH}_r, \text{CH}_t)] + E_{rp}[l, r(\text{CH}_t, \text{BS})] + E_{rp} \\ &= l\left[E_{\text{Ele}} + \varepsilon_f r^2(CH_r, CH_t)\right] + l\left[E_{\text{Ele}} + \varepsilon_f r^2(\text{CH}_r, \text{BS})\right] + lE_{\text{Ele}} \\ &= l\varepsilon_f\left(r^2(\text{CH}_r, \text{CH}_t) + r^2(\text{CH}_r, \text{BS})\right) + 3lE_{\text{Ele}}. \end{aligned} \quad (8)$$

Obviously,$r^2(\text{CH}_r, \text{CH}_t) + r^2(\text{CH}_r, \text{BS})$ plays a huge role in the total energy consumed during data transmission. Therefore, energy requires more for transmission when the distance is large. Thus, the entire detection phase of total energy consumption is given by

$$E_{\text{Clone\_Detect}} = E_{\text{msg}} + 2E_{\text{msg\_ch}} + E_{\text{msg\_mem}}. \quad (9)$$

Each node transmits the data to its CH during the data transfer phase, which is given by

$$E_{\text{msg\_data}} = l(n-k)\left(E_{\text{Ele}} + \varepsilon_f D^2\right). \quad (10)$$

Therefore, the estimation of the remaining energy of each node (n) using data communication is given by

$$E_{\text{res}} = E_{\text{SE}} - E_{\text{clone\_Detect}} - E_{\text{msg\_data}}, \quad (11)$$

where E_{SE} = initial energy node, $E_{\text{Clone\_Detect}}$ = energy consumed during detection, and $E_{\text{msg\_data}}$ = energy required for transmission of data.

4.1.2. Packet Delivery Ratio (PDR). PDR is the ratio packet forwarded by a node to receive from the neighboring nodes.

$$\text{PDR} = \frac{N_F}{N_R}, \quad (12)$$

where N_F is the number of packets sent by the node and N_R is the total packets received from its neighbor nodes. The fuzzification of N is based on the following equation:

$$\text{PDR}_f = \left\{ \text{PDR}\, N_R < N_F 1 - \text{PDR}\, N_R = N_F - \frac{1}{\text{PDR}} N_R > N_F \right. . \quad (13)$$

4.1.3. Delay (DL). It is the delay by a suspected node to the delay by its neighboring nodes.

$$\text{DL} = \frac{D_F}{D_R}, \quad (14)$$

where D_F is a delay caused by the node and D_R is the delay by the neighboring nodes. The fuzzification of DL is given by

$$\text{DL}_f = \left\{ \text{DL}\, D_R < D_F 1 - \text{DL}\, D_R = D_F - \frac{1}{\text{DL}} D_R > D_F. \right. \quad (15)$$

4.1.4. False Input Data (FIP). It is the ratio of the number of invalid inputs forwarded by a node to the number of packets forwarded by the neighbor nodes. The fuzzification of FIP is given by

$$\text{FIP}_f = \text{FIP}. \quad (16)$$

4.1.5. Speed (SP). This parameter is used to measure the speed of the node. The following equation gives the fuzzification of SP:

$$\text{SP}_f = \text{SP}. \quad (17)$$

4.2. Fuzzy Inference System. The fuzzy interference system's first step consists of fuzzifications that determine the appropriate uncertainty for the input parameters. Figures 4(a)–4(f) show the input and output variables' members, respectively.

The knowledge base consists of the evaluation of rules and the integration of rule results. In the rule evaluation, fuzzy rules were applied to the inputs and obtained the output. Then result aggregation is performed, as shown in Table 2. We have considered the five parameters to find the cloned node. When any clone node is inserted into the network, the node's speed and residual energy will be higher than with the existing node in the network. When any node wants to launch an attack in the network, the PDR of that node will be low, and the delay will be high. The attacker node will give the false sensing value in the network. If more than two states meet the above condition, then a node's probability as a clone node is high. When any two conditions satisfy, then the probability of clone node is medium. Otherwise, the node will consider as a normal node.

The ideal conditions of the node to become a clone node will be the following:

(1) Speed (SP) of the node will be higher than that of the normal node
(2) Delay (DL) will be high
(3) Packet delivery ratio (PDR) will be low
(4) Residual energy (RE) will be high
(5) False input value (FIV) will be high

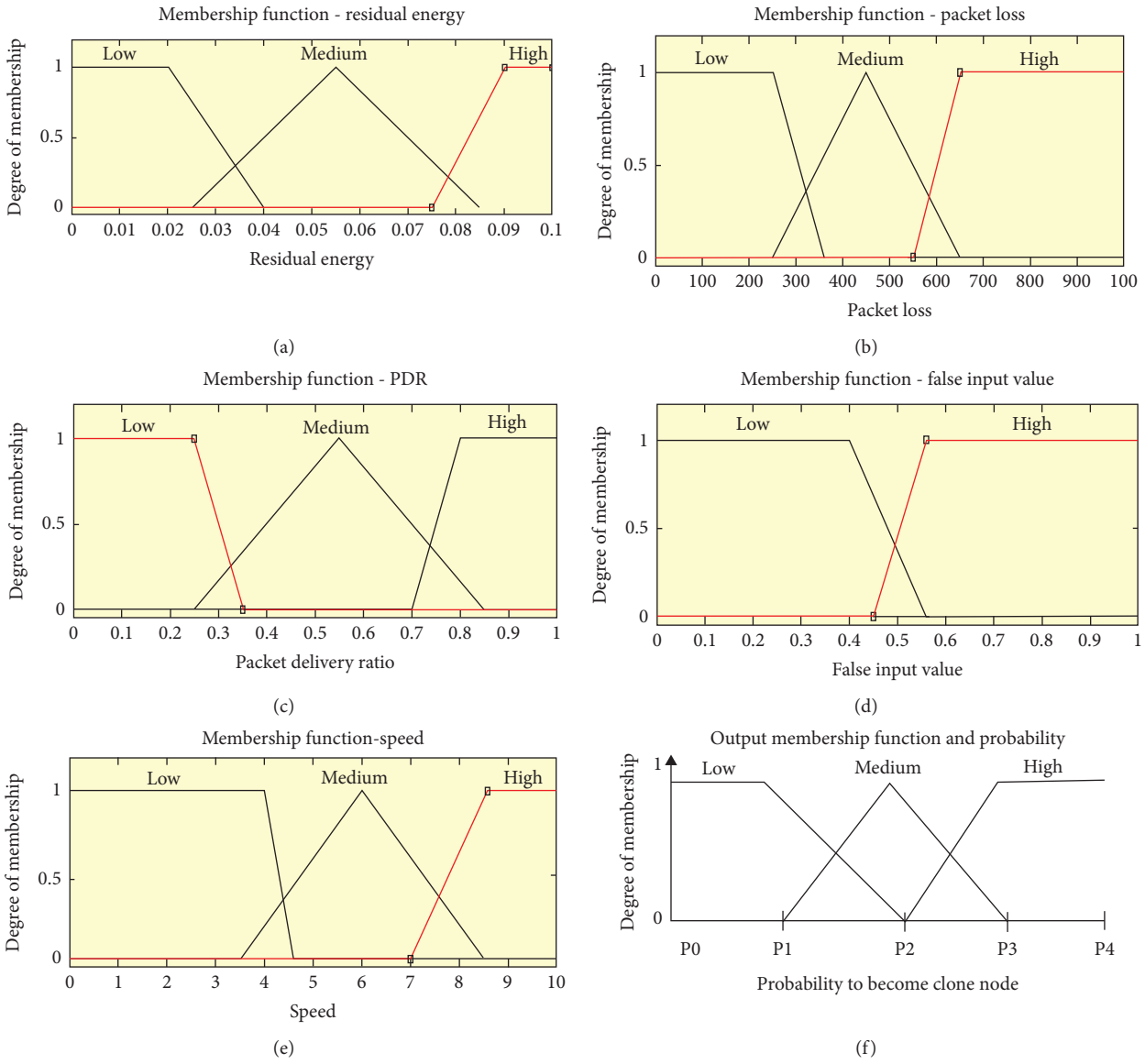

FIGURE 4: Proposed method input variable for membership function (MF). (a) MF and residual energy. (b) MF and delay. (c) MF and packet delivery ratio. (d) MF versus false input value. (e) MF versus speed. (f) Output variables, membership function, and probability.

The value of P_0, P_1, P_2, P_3, and P_4 is taken as 0, 0.3, 0.6, 0.8, and 1 in Figure 4(f). There are three output values, i.e., low, medium, and high, obtained from the output membership function and probability as symbolized in Figure 4(f). The probability of cloned nodes considers the medium value as best because the false detection will be low. If the proposed method considers the high probability value, the chance of false-positive detection will be high, which decreases the effectiveness of the proposed method.

5. Experiment Result and Performance Evaluation

The suggested FLCND method has been analyzed using Network Simulator (NS2). The simulation network consists of 100 mobile sensor nodes that have been propagated in a 750×750-meter area with a simulation time of 10 seconds. Table 3 summarizes the simulation parameters. Two different simulation cases have been implemented to verify the efficiency of the proposed method. In the first case, the standard network without the proposed method has been simulated, in which the attacker node exists in the network that will lodge different types of attacks inside the network. The second case consists of the same network, but the proposed method has been used in it. The network also has a clone node inside the network.

We evaluate the performance of the network in both cases using four performance parameters: PDR, packet loss, end-to-end (E-E) delay, and residual power .

(a) PDR: the second parameter is PDR, which measures the packet's ratio arriving at the receiver node to the number of packets sent. The comparison of the PDR

Table 2: Aggregation of fuzzy rules.

	Inputs					Output
	False input data	Speed	PDR	Delay	Residual energy	Probability to become clone node
1	Low	High	Low	High	Low	Medium
2	High	High	Low	High	Low	High
3	Low	High	Low	Low	Low	Low
4	High	High	Low	Low	Low	Medium
5	Low	Low	High	Low	Low	Low
6	High	Low	High	Low	Low	Medium
7	Low	High	High	Low	Low	Medium
8	High	High	High	Low	Low	High
9	Low	Low	Low	High	Low	Low
10	High	Low	Low	High	Low	Medium
11	Low	High	Low	High	Low	Medium
12	High	High	Low	High	Low	High
13	Low	Low	High	High	Low	Medium
14	High	Low	High	High	Low	High
15	Low	High	High	High	Low	High
16	High	High	High	High	Low	High
17	Low	Low	Low	Low	High	Low
18	High	Low	Low	Low	High	Medium
19	Low	High	Low	Low	High	Medium
20	High	High	Low	Low	High	High
21	Low	Low	High	Low	High	Medium
22	High	Low	High	Low	High	High
23	Low	High	High	Low	High	High
24	High	High	High	Low	High	High
25	Low	Low	Low	High	High	Medium
26	High	Low	Low	High	High	High
27	Low	High	Low	High	High	High
28	High	High	Low	High	High	High
29	Low	Low	High	High	High	High
30	High	Low	High	High	High	High
31	Low	High	High	High	High	High
32	High	High	High	High	High	High

Table 3: Simulation parameters.

Parameters	Values
Packet size	512 bytes
Simulation time	10 s
Traffic	CBR
Number of nodes	100
Mobility module	Random waypoint
Transmission range	150 meters
Speed	10–40 m/s

of both scenarios is shown in Figure 5. The green line in the figure represents the PDR of the 1st scenario where the proposed method has not been used in the network and replicate nodes are present. The red line in the figure represents the PDR of the 2nd scenario where the proposed method has been implemented in a sensor network when the attacker node is in the network, and the packet delivery rate is low compared to the 2nd scenario where the proposed method has been implemented. The 2nd scenario using the FLCND method delivers 57% more packets compared to the 1st scenario.

(b) End-to-end delay: the average time of a packet sent to the destination node from the source node. The comparison of the end-to-end delay for both scenarios is shown in Figure 6. The first scenario consists of the network containing the cloned nodes without the proposed method, and the 2nd scenario is the network having the clone nodes with the proposed method. The green line in the figure indicates the end-to-end delay in the first scenario. The E-E delay is high in this case at the attacker node. The red line in the figure shows the E2E delay for the second scenario. The E-E delay is low compared to the first scenario as clone nodes are detected by the proposed method and cannot affect the network's performance. Therefore, we can say that the clone node attack does not affect the second scenario of the network by using the proposed method.

(c) Packet loss: it is represented by the packet which does not reach the target node. Figure 7 shows a comparison of the packet loss in both situations. In the first case, the packet loss is higher due to clone nodes, while in the second scenario, the packet loss rate is lower due to the detection of clone nodes.

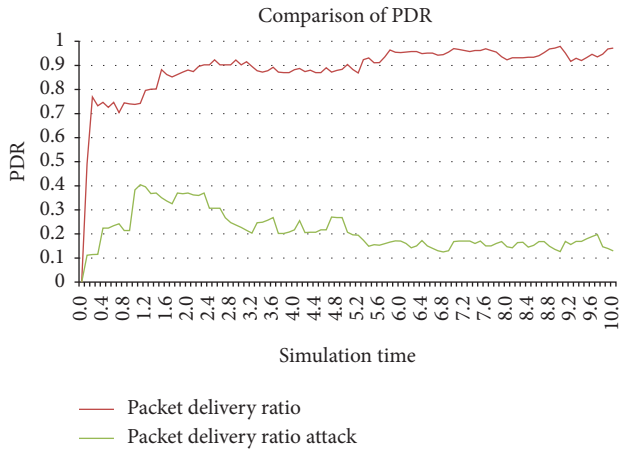

Figure 5: Comparison of packet delivery ratio.

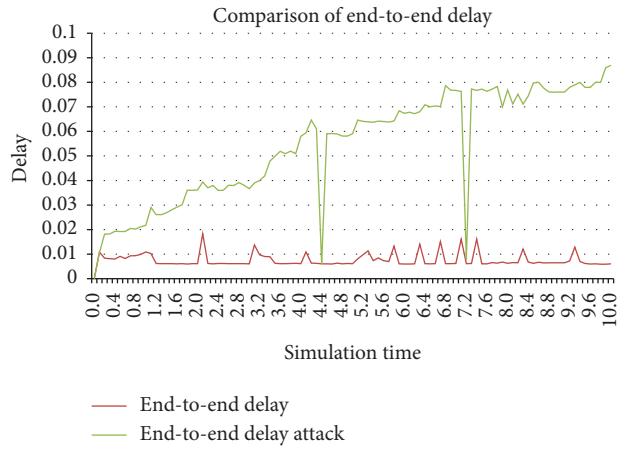

Figure 6: Comparison of end-to-end delay.

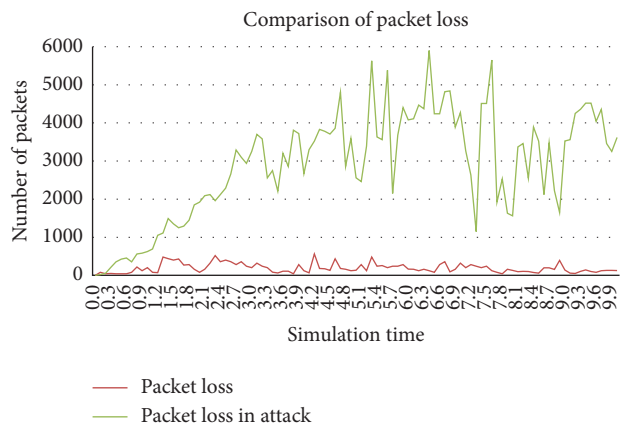

Figure 7: Comparison of packet loss.

Based on this result, we can conclude that the cloned node does not affect the network using the FLCND method.

(d) Residual energy: the next parameter is the residual energy [51, 56, 58], which is calculated as the total energy minus the power consumed by the node during data transmission. When the proposed method is not implemented in the network containing replicated nodes, the green line in the figure represents the residual energy in Figure 8. The red line in the figure shows the residual energy of the proposed method. However, the scenario using the FLCND method consumes 37% less energy than the 1st scenario. Therefore, the proposed method is also optimal in energy consumption.

5.1. Comparison of FLCND with Existing Methods. To calculate the proposed solution's effectiveness, we compared the work of FLCND with existing methods HO, XED, CBCD, and EDD. All five methods have been simulated by using an NS2 simulator by varying the speed and number of sensors from 20 to 200 and 10 m/s to 40 m/s. We have compared the FLCND method with HO, XED, CBCD, and EDD methods in total energy consumption, detection rate, and false-negative rate. Figures 9–11 demonstrate the result of the comparison of the proposed method with existing methods.

5.1.1. Energy Consumption. First, we have compared the total energy consumption of the detection methods by varying the total sensor from 20 to 200. We know that when any sensor transfers or processes the data using any detection method, the node will consume some energy. The lower energy cost indicates the higher efficiency of the detection method. Figures 9(a)–9(d) show the energy consumption of five clone node detection methods, where the number of nodes varies from 20 to 200, and the speed has changed from 10 to 40 m/s. The figures conclude that the total energy consumed of the proposed FLCND method has less among HO, XED, CBCD, and EDD methods. We calculate the total energy consumed for each case and determine the FLCND consumes 41% lower energy to the HO method, 27% lower than the XED method, 46% lower than the CBCD method, and 54% lower than to EDD method. The less energy consumption in FLCND may be due to its approach for the detection of clone nodes. The detection method of the FLCND method focuses only on those nodes which are suspected, whereas that of the HO, XED, CBCD, and EDD methods focuses on the complete network.

5.1.2. Clone Detection Rate. The second parameter is the detection rate of clone nodes. The detection rate of a clone node is calculated as the number of clone nodes detected from the total number of clone nodes existing in the network and then multiply it by 100. Figures 10(a)–10(d) show the detection rate of HO, XED, CBCD, EDD, and FLCND methods by varying the number of nodes and the speed of nodes. The figure shows that the proposed method, FLCND, has a 67% higher detection rate than the HO method, 65% higher detection rate than the XED method, 46% higher detection rate than the CBCD method, and 53% higher detection rate than the EDD method. For generating the

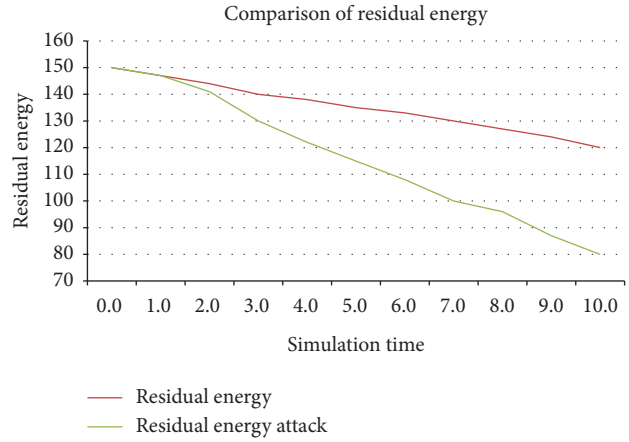

Figure 8: Comparison of residual energy.

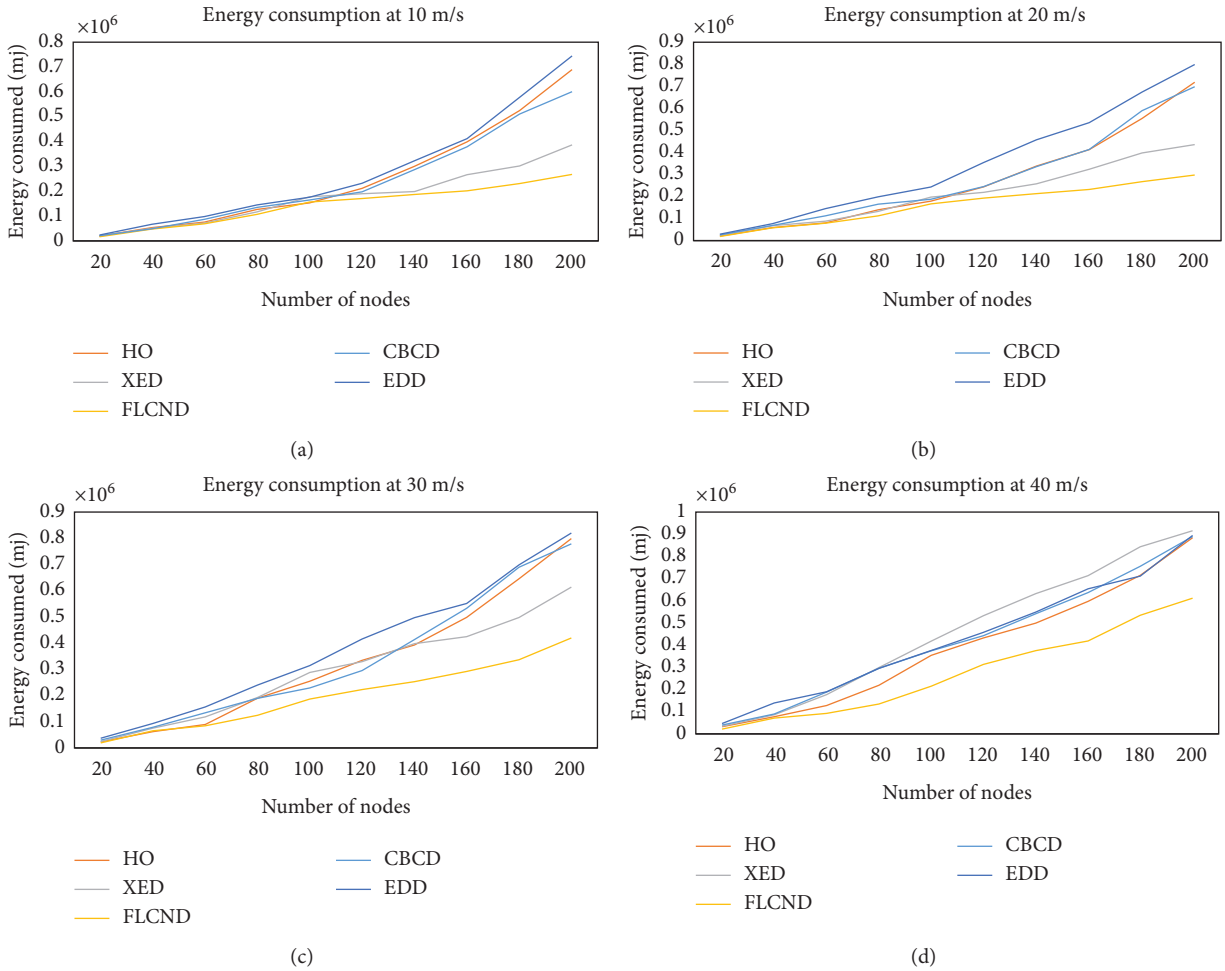

Figure 9: Comparison of total energy consumed at a speed of (a) 10 m/s, (b) 20 m/s, (c) 30 m/s, and (d) 40 m/s.

cloned node, the legitimate node must be stolen from the network. FLCND method finds the missing node from the network so that it may be the reason for the higher detection rate in FLCND.

5.1.3. False-Positive Rate. The next parameter is the wrongly detected clones known as false positive. Figures 11(a)–11(d) show the false-positive detection rate of XED, FLCND, HO, CBCD, and EDD methods

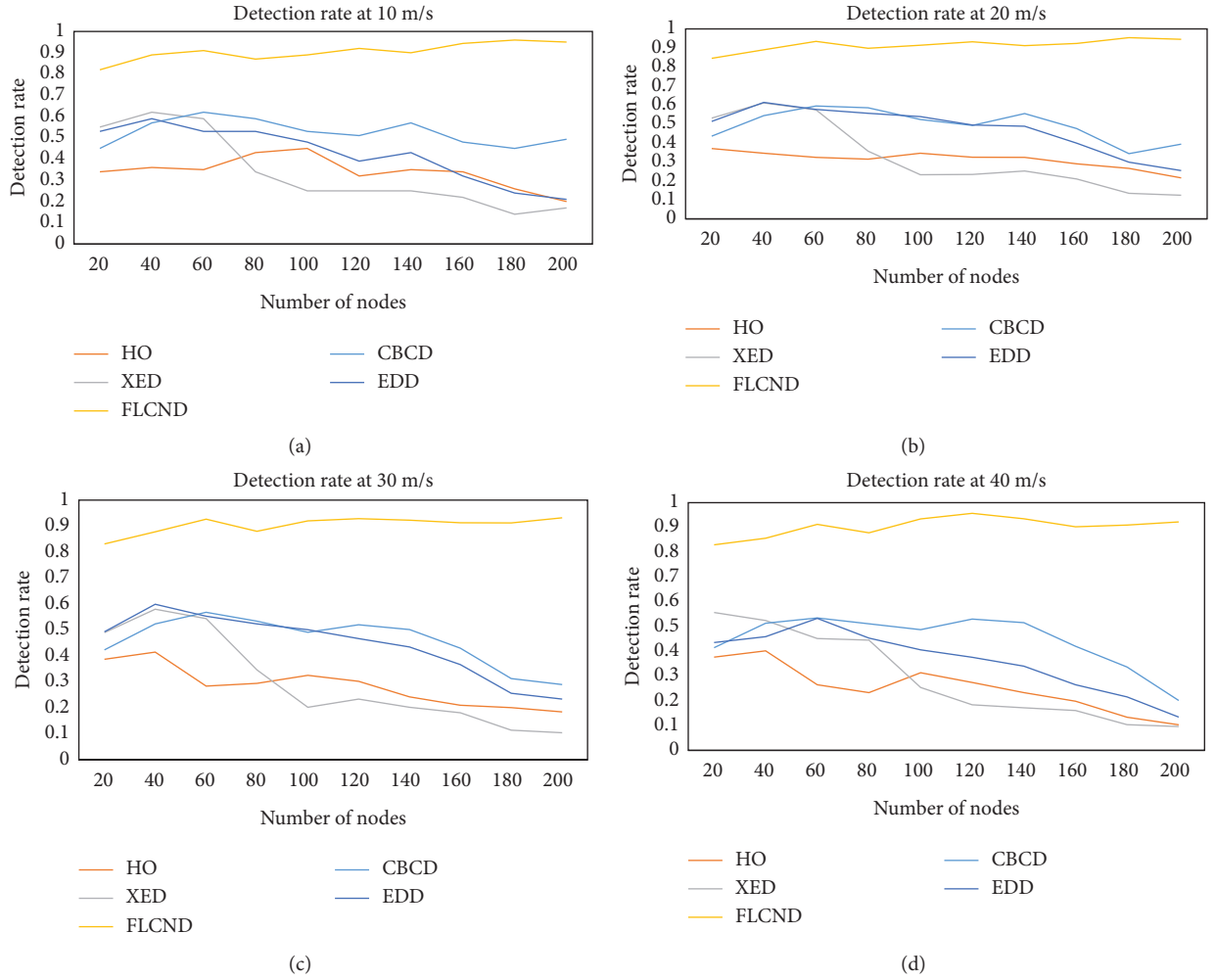

Figure 10: Comparison of detection rate at a speed of (a) 10 m/s, (b) 20 m/s, (c) 30 m/s, and (d) 40 m/s.

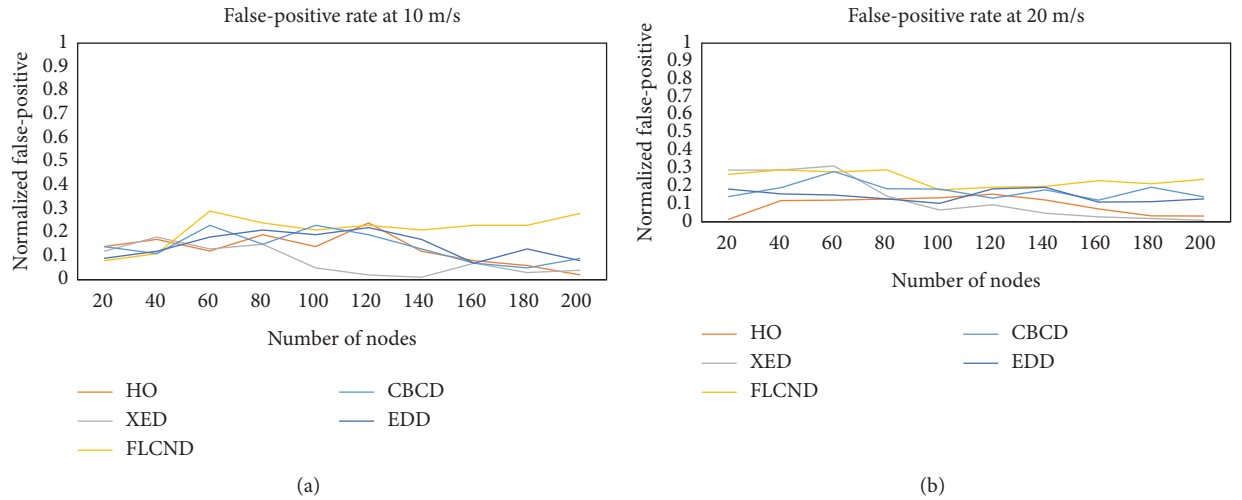

Figure 11: Continued.

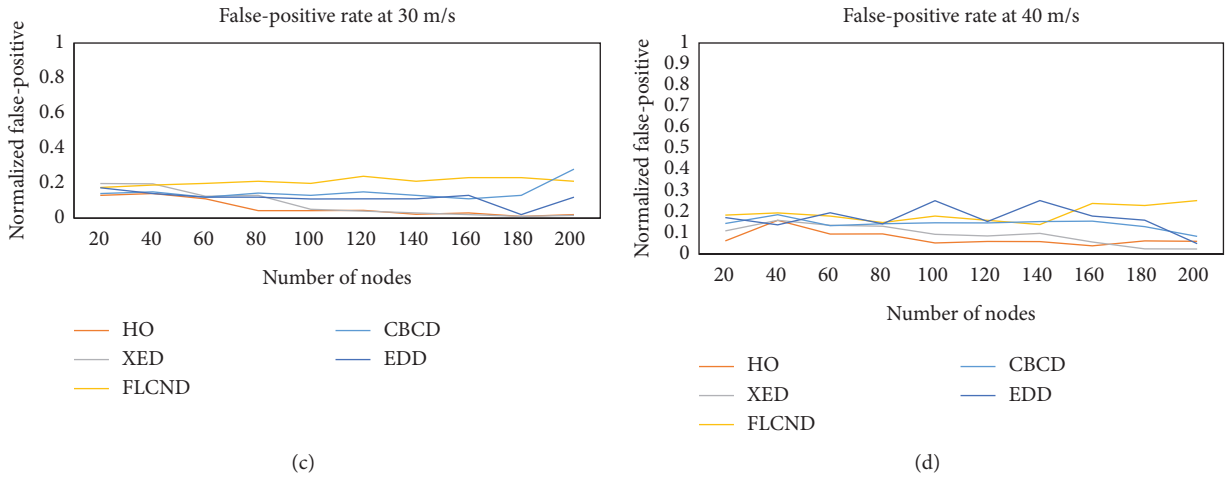

Figure 11: Comparative analysis of false-positive rate at a speed of (a) 10 m/s, (b) 20 m/s, (c) 30 m/s, and (d) 40 m/s.

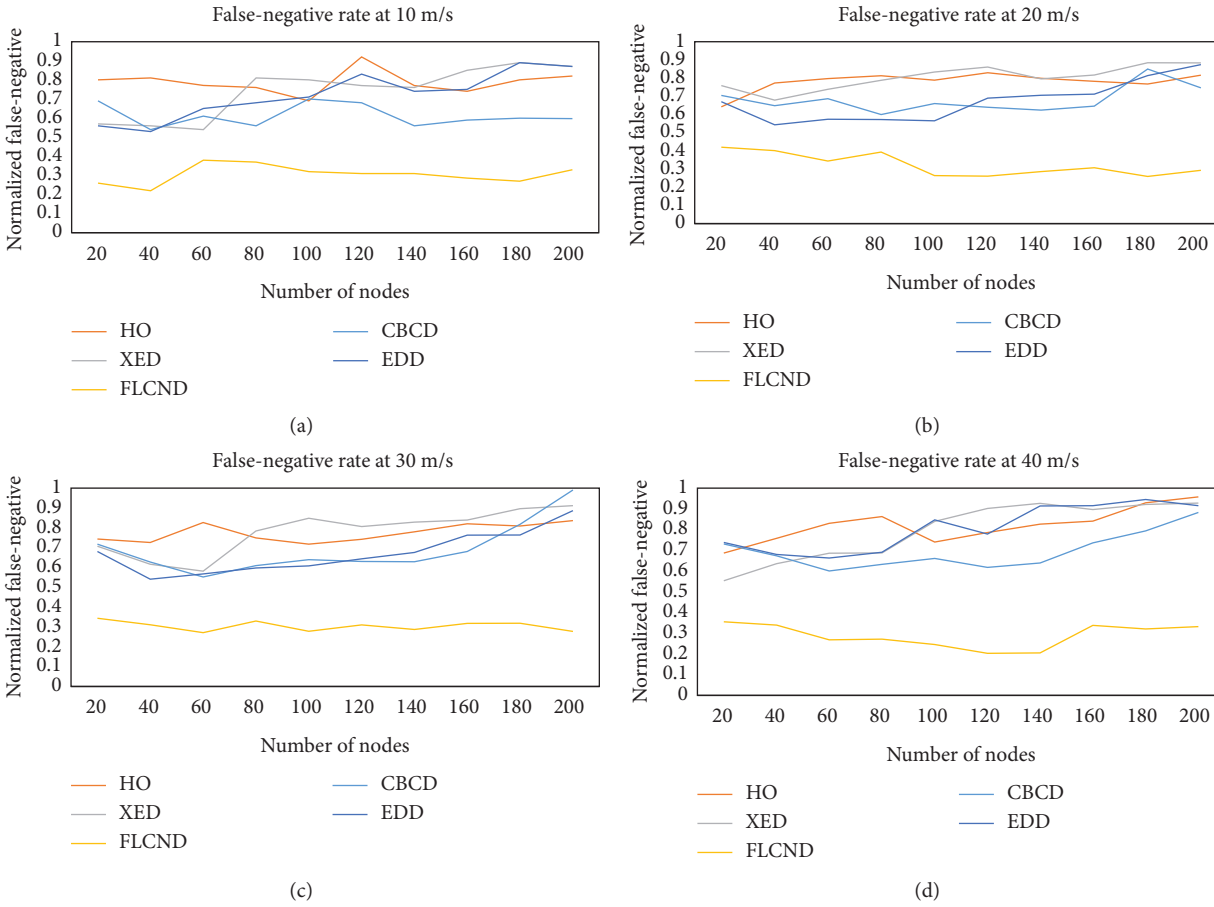

Figure 12: Comparative analysis of false-negative rate at a speed of (a) 10 m/s, (b) 20 m/s, (c) 30 m/s, and (d) 40 m/s.

concerning the number of sensor nodes and speed. In all cases, the false detection rate of the proposed FLCND approach is 21% greater than the XED approach, 19% greater than the HO approach, 16% greater than the CBCD approach, and 15% of the EDD method. As the detection rate of FLCND is higher than other existing methods, the false-positive rate of FLCND is also higher.

5.1.4. False-Negative Rate. The next parameter is the false-negative detection rate in which the clone mobile node is not identified as a clone. Figures 12(a)–12(d) show the false detection rate of XED, FLCND, HO, CBCD, and EDD methods concerning the number of sensor nodes and speed. In all cases, the false detection rate of the proposed FLCND approach is 31% less than the XED

approach, 28% less than the HO approach, 29% less than the CBCD approach, and 29% of the EDD method.

6. Conclusion

An FLCND method to detect the cloned node is designed. The parameters of FLCND are falsely input value, speed, PDR, delay, and residual power, which are processed as fuzzy logic inputs, and based on the outcome, a clone node is detected in WSN. The FLCND method's performance was evaluated using PDR, packet loss, E-E delay, and residual energy parameters in the NS2 simulator. Two different scenarios have been implemented in NS2, where the first scenario is a normal network which is having clone nodes, and the second case consists of the proposed method and the cloned nodes in the network. After comparing the results, the FLCND method consumes less energy and high packet delivery rate. We can conclude that the cloned node does not affect the network due to the FLCND method. We have also compared the proposed method with EDD, HO, CBCD, and XED in terms of total energy consumed, false-negative rate, and detection rate. FLCNDS consumes less than 27% energy with each method. FLCND has a 67% higher detection rate than the HO method, 65% higher detection rate than the XED method, 46% higher detection rate than the CBCD method, and 53% higher detection rate than the EDD method. The false-negative detection rate of the proposed FLCND approach is less than 28% of each method. We have found that the FLCND has less energy consummation and a better detection rate compared to XED, HO, CBCD, and EDD methods. In the future, we will simulate the proposed algorithm by changing the number of nodes from 1000 to 10000. We will evaluate the FLCND method's performance with other parameters and compare it with other existing methods.

In the near future, we will utilize various metaheuristic techniques to enhance the results further. Also, the proposed model will be tested on other kinds of wireless technologies.

Data Availability

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

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of this paper.

Acknowledgments

The authors would like to acknowledge the support of Taif University Researchers Supporting Project number (TURSP-2020/211), Taif University, Taif, Saudi Arabia.

References

[1] A. Rajput and V. B. Kumaravelu, "Scalable and sustainable wireless sensor networks for agricultural application of Internet of things using fuzzy c-means algorithm," *Sustainable Computing: Informatics and Systems*, vol. 22, pp. 62–74, 2019.

[2] M. Naghibi and H. Barati, "EGRPM: energy efficient geographic routing protocol based on mobile sink in wireless sensor network," *Sustainable Computing: Informatics and Systems*, vol. 25, Article ID 100377, 2020.

[3] C. Karlof and D. Wagner, "Secure routing in wireless sensor networks: attacks and countermeasures," *Ad Hoc Networks*, vol. 1, no. 2-3, pp. 293–315, 2003.

[4] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.

[5] X. Du and Y. Xiao, "Chapter 17: a survey on sensor network security," in *Wireless Sensor Networks and Applications. Signals and Communication Technology*, Y. Li, M. T. Thai, and W. Wu, Eds., Springer, Boston, MA, USA, 2008.

[6] K. Chao, M. Jo, T. Kwon, H. H. Chen, and D. H. Lee, "Classification and experimental analysis for clone detection approaches in wireless sensor networks," *IEEE Systems Journal*, vol. 7, no. 1, pp. 26–35, 2012.

[7] C. Hartung, J. Balasalle, and R. Han, "Node compromise in sensor networks: the need for secure systems," Technical Report CU-CS-990-05, Department of Computer Science, University of Colorado at Boulder, Boulder, Colorado, 2005.

[8] B. Parno, A. Perrig, and V. Gligor, "Distributed detection of node replication attacks in sensor networks," in *Proceedings of IEEE Symposium on Security and Privacy*, pp. 49–63, Oakland, CA, USA, May 2005.

[9] R. Brooks, P. Y. Govindaraju, M. Pirretti, N. Vijaykrishnan, and M. T. Kandemir, "On the detection of clones in sensor networks using random key predistribution," *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*, vol. 37, no. 6, pp. 1246–1258, 2007.

[10] H. Choi, S. Zhu, and T. F. Porta, "SET: detecting node clones in sensor networks," in *Proceedings of the Third International Conference on Security and Privacy in Communications Networks*, pp. 17–21, Washington, WA, USA, October 2007.

[11] M. Conti, R. D. Pietro, L. V. Mancini, and A. Mei, "A randomized, efficient and distributed protocol for the detection of node replication attacks in wireless sensor networks," in *Proceedings of ACM International Symposium on Mobile Ad Hoc Networking and Computing, MobiHoc'07*, pp. 80–89, Montréal, Canada, September 2007.

[12] B. Zhu, V. G. K. Addada, S. Setia, S. Jajodia, and S. Roy, "Efficient distributed detection of node replication attacks in sensor networks," in *Proceedings of the Twenty-Third Annual Conference in Computer Security Applications*, pp. 257–267, Miami Beach, FL, USA, December 2007.

[13] M. Conti, R. D. Pietro, L. V. Mancini, and A. Mei, "Emergent properties: detection of the node-capture attack in mobile wireless sensor networks," in *Proceedings of the 1st ACM Conference on Wireless Network Security, WiSec'08*, Alexandria, VA, USA, January 2008.

[14] S. Lalar, S. Bhushan, and N. A. Surender, "Clone detection using fuzzy logic in static wireless sensor network," *International Journal of Vehicle Information and Communication Systems*, vol. 5, no. 3, pp. 334–353, 2020.

[15] Z. Zhang, S. Luo, H. Zhu, and Y. Xin, "A clone detection algorithm with low resource expenditure for wireless sensor networks," *Journal of Sensors*, vol. 2018, Article ID 4396381, 16 pages, 2018.

[16] M. Conti, R. Di Pietro, L. Mancini, and A. Mei, "Distributed detection of clone attacks in wireless sensor networks," *IEEE*

Transactions on Dependable and Secure Computing, vol. 8, no. 5, pp. 685–698, 2011.

[17] C. M. Yu, C. S. Lu, and S. Y. Kuo, "CSI: compressed sensing-based clone identification in sensor networks," in *Proceedings of the IEEE International Conference on Pervasive Computing and Communications*, pp. 290–295, Lugano, Switzerland, March 2012.

[18] Y. Zeng, J. Cao, S. Zhang, S. Guo, and L. Xie, "Random-walk based approach to detect clone attacks in wireless sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 5, pp. 677–691, 2010.

[19] W. B. Jaballaha, M. Conti, G. File, M. Mosbah, and A. Zemmari, "Whac-a-mole: smart node positioning in clone attack in wireless sensor networks," *Computer Communications*, vol. 119, pp. 66–82, 2018.

[20] S. Lalar, S. Bhushan, and Surender, "An efficient tree-based clone detection scheme in wireless sensor network," *Journal of Information and Optimization Sciences*, vol. 40, no. 5, pp. 1003–1023, 2019.

[21] N. Muhammad, S. Fazli, Z. Khan et al., "A systematic review on clone node detection in static wireless sensor networks," *IEEE Access*, vol. 8, pp. 65450–65461, 2020.

[22] J.-W. Ho, M. Wright, and S. K. Das, "Fast detection of replica node attacks in mobile sensor networks using sequential analysis," in *Proceedings - IEEE INFOCOM*, pp. 1773–1781, Rio de Janeiro, Brazil, April 2009.

[23] W. T. Zhu, J. Zhou, R. H. Deng, and F. Bao, "Detecting node replication attacks in mobile sensor networks: theory and approaches," *Security and Communication Networks*, vol. 5, no. 5, pp. 496–507, 2011.

[24] W. T. Zhu, J. Zhou, R. H. Deng, and F. Bao, "Detecting node replication attacks in wireless sensor networks: a survey," *Journal of Network and Computer Applications*, vol. 35, no. 3, pp. 1022–1034, 2012.

[25] M. C. Chia, "Efficient and distributed detection of node replication attacks in mobile sensor networks," in *Proceedings of the 70th IEEE Vehicular Technology Conference VTC. Fall 2009*, Anchorage, AK, USA, September 2009.

[26] J.-W. Ho, M. Wright, and S. K. Das, "Fast detection of mobile replica node attacks in wireless sensor networks using sequential hypothesis testing," *IEEE Transactions on Mobile Computing*, vol. 10, no. 6, pp. 767–782, 2011.

[27] M. Y. Chia, S. L. Chun, and Y. K. Sy, "Mobile sensor network resilient against node replication attacks," in *Proceedings of the 5th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON'08)*, pp. 597–599, San Francisco, CA, USA, June 2008.

[28] C. M. Yu, C. Lu, and S. Kuo, "Efficient and distributed detection of node replication attacks in mobile sensor networks," in *Proceedings of the 70th IEEE Vehicular Technology Conference Fall (VTC '09-Fall)*, pp. 1–5, Anchorage, AK, USA, September 2009.

[29] X. Deng, Y. Xiong, and D. Chen, "Mobility-assisted detection of the replication attacks in mobile wireless sensor networks," in *Proceedings of the IEEE 6th International Conference on Wireless and Mobile Computing, Networking and Communications*, pp. 225–232, Niagara Falls, Canada, October 2010.

[30] X.-M. Deng and Y. Xiong, "A new protocol for the detection of node replication attacks in mobile wireless sensor networks," *Journal of Computer Science and Technology*, vol. 26, no. 4, pp. 732–743, 2011.

[31] L.-M. Wang and Y. Shi, "Patrol detection for replica attacks on wireless sensor networks," *Sensors*, vol. 11, no. 3, pp. 2496–2504, 2011.

[32] Y. Lou, Y. Zhang, and S. Liu, "Single hop detection of node clone attacks in mobile wireless sensor networks," *Procedia Engineering*, vol. 29, pp. 2798–2803, 2012.

[33] H. R. Shaukat, F. Hashim, and A. Sali, "Danger theory based node replication attacks detection in mobile wireless sensor network," in *Proceedings of the IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE)*, Penang, Malaysia, April 2014.

[34] G. Cheng, S. Guo, Y. Yang, and F. Wang, "Replication attack detection with monitor nodes in clustered wireless sensor networks," in *Proceedings of the 34th IEEE International Performance Computing and Computing Conference*, pp. 1–8, Nanjing, China, December 2015.

[35] M. Dong, K. Ota, L. T. Yang, A. Liu, and M. Guo, "LSCD: a low-storage clone detection protocol for cyber-physical systems," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 35, no. 5, pp. 712–723, 2016.

[36] J. Anthoniraj and A. Razak, "CBCD: cluster based clone detection in mobile wireless sensor networks," *Indian Journal of Science and Technology*, vol. 9, no. 31, pp. 1–10, 2016.

[37] D. Rajesh and A. Shanmugam, "A hyper heuristic localization based cloned node detection technique using GSA based simulated annealing in sensor networks," *Cognitive Computing for Big Data Systems Over IoT*, vol. 14, pp. 307–335, 2018.

[38] P. C. Sankar and M. Roy, "Lightweight cloned-node detection algorithm for efficiently handling SSDF attacks and facilitating secure spectrum allocation in CWSNs," *IET Wireless Sensor Systems*, vol. 8, no. 3, pp. 121–128, 2018.

[39] M. Conti, R. Di Pietro, and A. Spognardi, "Clone wars: distributed detection of clone attacks in mobile WSNs," *Journal of Computer and System Sciences*, vol. 80, no. 3, pp. 654–669, 2014.

[40] V. Manickavasagam and J. Padmanabhan, "A mobility optimized SPRT based distributed security solution for replica node detection in mobile sensor networks," *Ad Hoc Networks*, vol. 37, pp. 140–152, 2016.

[41] M. Jamshidi, S. Sheikh Abooli Poor, N. Nasih Qader, M. Esnaashari, and R. M. Mohammad, "A lightweight algorithm against replica node attack in mobile wireless sensor networks using learning agents," *IEIE Transactions on Smart Processing & Computing*, vol. 8, no. 1, pp. 58–70, 2019.

[42] M. Jamshidi, M. Esnaashari, A. M. Darwesh, and M. R. Meybodi, "Using time-location tags and watchdog nodes to defend against node replication attack in mobile wireless sensor networks," *International Journal of Wireless Information Networks*, vol. 27, no. 1, pp. 102–115, 2020.

[43] M. Jamshidi, S. Abooli, A. Arghavani, M. Esnaashari, A. A. Shaltooki, and M. R. Meybodi, "A simple, lightweight, and precise algorithm to defend against replica node attacks in mobile wireless networks using neighboring information," *Ad Hoc Networks*, vol. 100, 2020.

[44] S. Anitha, P. Jayanthi, and V. Chandrasekaran, "An intelligent based healthcare security monitoring schemes for detection of node replication attack in wireless sensor networks," *Measurement*, vol. 167, 2021.

[45] S. Chen, Z. Pang, H. Wen, K. Yu, T. Zhang, and Y. Lu, "Automated labeling and learning for physical layer authentication against clone node and sybil attacks in industrial wireless edge networks," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 3, pp. 2041–2051, 2021.

[46] T. Dimitriou, E. A. Alrashed, M. H. Karaata, and A. Hamdan, "Imposter detection for replication attacks in mobile sensor networks," *Computer Networks*, vol. 108, pp. 210–222, 2016.
[47] T. Wiens, "Engine speed reduction for hydraulic machinery using predictive algorithms," *International Journal of Hydromechatronics*, vol. 2, no. 1, pp. 16–31, 2019.
[48] M. Kaur, D. Singh, and R. Singh Uppal, "Parallel strength pareto evolutionary algorithm-II based image encryption," *IET Image Processing*, vol. 14, no. 6, pp. 1015–1026, 2019.
[49] M. Khurana, R. Thalore, V. Raina, and M. K. Jha, "Improved time synchronization in ML-MAC for WSN using relay nodes," *AEU-International Journal of Electronics and Communications*, vol. 69, no. 11, pp. 1622–1626, 2015.
[50] R. Thalore, J. Sharma, M. Khurana, and M. K. Jha, "QoS evaluation of energy-efficient ML-MAC protocol for wireless sensor networks," *AEU-Journal of Electronics and Communications*, vol. 67, no. 12, pp. 1048–1053, 2013.
[51] S. Osterland and J. Weber, "Analytical analysis of single-stage pressure relief valves," *International Journal of Hydromechatronics*, vol. 2, no. 1, pp. 32–53, 2019.
[52] M. Kaur and D. Singh, "Multiobjective evolutionary optimization techniques based hyperchaotic map and their applications in image encryption," *Multidimensional Systems and Signal Processing*, vol. 32, no. 1, pp. 281–301, 2021.
[53] S. Ghosh, P. Shivakumara, P. Roy, U. Pal, and T. Lu, "Graphology based handwritten character analysis for human behaviour identification," *CAAI Transactions on Intelligence Technology*, vol. 5, no. 1, pp. 55–65, 2020.
[54] B. Gupta, M. Tiwari, and S. Singh Lamba, "Visibility improvement and mass segmentation of mammogram images using quantile separated histogram equalisation with local contrast enhancement," *CAAI Transactions on Intelligence Technology*, vol. 4, no. 2, pp. 73–79, 2019.
[55] P. P. Devi and B. Jaison, "Protection on wireless sensor network from clone attack using the SDN-enabled hybrid clone node detection mechanisms," *Computer Communications*, vol. 152, pp. 316–322, 2020.
[56] H. S. Basavegowda and G. Dagnew, "Deep learning approach for microarray cancer data classification," *CAAI Transactions on Intelligence Technology*, vol. 5, no. 1, pp. 22–33, 2020.
[57] SD. Gandham, S. Dawande, M. Prakash, and S. Venkatesan, "Energy efficient schemes for wireless sensor networks with multiple mobile base stations," in *Proceedings of the IEEE Global Telecommunications Conference, GLOBECOM*, pp. 377–381, San Francisco, CA, USA, December 2003.
[58] R. Wang, H. Yu, G. Wang, G. Zhang, and W. Wang, "Study on the dynamic and static characteristics of gas static thrust bearing with micro-hole restrictors," *International Journal of Hydromechatronics*, vol. 2, no. 3, pp. 189–202, 2019.