Research Article

Black Hole Attack Detection Using K-Nearest Neighbor Algorithm and Reputation Calculation in Mobile Ad Hoc Networks

Gholamreza Farahani

*Department of Electrical Engineering and Information Technology,
Iranian Research Organization for Science and Technology (IROST), Sh. Ehsani Rad St., Enqelab St., Parsa Sq.,
Ahmadabad Mostoufi Rd., Azadegan Highway, Tehran 3353136846, Iran*

Correspondence should be addressed to Gholamreza Farahani; farahani.gh@irost.org

Received 4 January 2021; Revised 31 July 2021; Accepted 7 August 2021; Published 20 August 2021

Academic Editor: Shehzad Ashraf Chaudhry

Copyright © 2021 Gholamreza Farahani. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The characteristics of the mobile ad hoc network (MANET), such as no need for infrastructure, high speed in setting up the network, and no need for centralized management, have led to the increased popularity and application of this network in various fields. Security is one of the essential aspects of MANETs. Intrusion detection systems (IDSs) are one of the solutions used to ensure security in this network. Clustering-based IDSs are very popular in this network due to their features, such as proper scalability. This paper proposes a new algorithm in MANETs to detect black hole attack using the K-nearest neighbor (KNN) algorithm for clustering and fuzzy inference for selecting the cluster head. With the use of beta distribution and Josang mental logic, the trust of each node will be calculated. According to the reputation and remaining energy, fuzzy inference will select the cluster head. Finally, the trust server checks the destination node. If allowed, it notifies the cluster head; otherwise, it detects the node as a malicious node in the black hole attack in each cluster. The simulation results show that the proposed method has improved the packet loss rate, throughput, packet delivery ratio, total network delay, and normalized routing load parameters compared with recent black hole detection methods.

1. Introduction

Today, in many environments, security is based on an in-depth defense approach, in which multiple layers of defense are used to prevent enemies from violating security policies. This approach assumes that even if the enemy infiltrates one of the defensive layers, he will not be able to inflict damage because the other layers will provide an adequate level of support.

A MANET is a network without infrastructure and a self-configured network of mobile devices that are connected wirelessly. Each device in a MANET has complete independence and freedom to move in any direction; therefore in many cases, the connection between each mobile device and other devices is changing.

Without the need for a fixed communication infrastructure to create a dynamic network, the importance of

MANETs in applications such as military battlefield communications, relief and emergency operations, environmental protection, taxi networks, and independent space communications is increasing. Growing demand for MANETs has raised many concerns about security issues, especially for sensitive security applications. Unlike wired networks, MANETs are inherently insecure due to shared wireless media and the lack of any central control. The unique characteristics of MANETs have created new challenges for security design [1].

MANETs, like any other type of radio-based network technology, are under threat. These threats include foreign attackers and abusers within the network. Therefore, in many technologies, various information, such as data encryption, access control, identity management, and intrusion detection, is required to protect these types of networks. Unfortunately, many known intrusion detection methods

implemented in infrastructure-based IP networks do not apply to radio communications because of many implications for the use of these technologies in radio communications and the mobility of related devices. Therefore, the bandwidth attack level becomes smarter and larger. Also, the risk of identity theft and man in the middle (MITM) attack in the network has increased.

Various intrusion detection studies have been performed for traditional wired networks. Using wired network research for wireless networks is not an easy task due to key architectural differences. Due to their vulnerabilities, MANETs present a more difficult challenge for IDS design [2].

Given the possibility of unsuccessful transmission of protocol packets, the probability of false alarms and false node charges in MANETs is very high. This probability increases with physical movement in the network, which leads to interruption of transmissions and fluctuation of routes. Also, there is no crucial location in the network where all relevant traffic can be observed and analyzed to detect malicious behavior, while these facilities are available for routers, switches, and firewalls in wired IP networks [3].

Although these mechanisms provide a powerful barrier to malicious users, an additional layer of defense called intrusion detection is often used to protect the network. The IDS focuses on detecting malicious activity, usually attackers that have successfully infiltrated the defense environment.

Wireless networks have special security requirements and problems. These problems are due to the nature and properties of wireless networks and are as follows.

- (1) Lack of infrastructure: in wireless networks, centralized and integrated structures such as routers are not necessarily available. For this reason, their security solutions are usually decentralized, distributed, and based on the cooperation of all network nodes.
- (2) Use of wireless link: there are no common defense lines in a wireless network like in wired networks (for example, firewall as the front line of defense). An attacker can target any node from any direction without the need for physical access to the link.
- (3) Multihopping: in most wireless routing protocols, the nodes themselves act as routers (especially in ad hoc networks), and packets have several different hops. Therefore, every node cannot be trusted for a task like routing.
- (4) Node autonomy in relocation: mobile nodes in a wireless network are difficult to track due to relocation, especially in large networks.

Other natural features of a wireless network that are sources of security problems include the lack of a fixed topology and limited resources such as power, processor, and memory. The primary function of the IDS is to detect intrusion from audit data collected from the network. Based

on detection techniques, IDSs in MANETs can be classified into the following categories.

- (A) Anomaly detection: in this case, the natural behaviors of users are compared with the data taken. Any activity that deviates from the base is considered a possible intrusion. This information is then passed to the system administrator.
- (B) Misuse detection: the system maintains a known pattern of attacks and compares these patterns with the captured data. Each matched pattern is treated as an intrusion. Then, the proper response is initiated [4].

Prevention-based methods may not eliminate intrusion because there are always some weaknesses in the system. In MANETs, a malicious node can launch a denial of service (DoS) attack or disrupt the routing mechanism by generating error routing messages. For these types of attacks, intrusion detection can serve as a second wall of defense and has paramount importance in high-security networks [1]. In recent years, ad hoc networks have been increasingly used in military and civilian works. Some features in ad hoc networks cause the attacker to have many opportunities to disrupt these networks. These features include network structure dynamics, node mobility, nodes to trust each other for easy communication together, power limitation, lack of central management to study behaviors and functions, and lack of clear defense lines. Features of the MANET that make it difficult to use are as follows [5]:

- (i) Dynamic topology: this means that nodes can move freely in or out of the network. As a result, network maintenance is difficult, and route discovery must be repeated.
- (ii) Self-organization: nodes are capable of organizing internal network functions without the aid of a central administrator.
- (iii) Multihop: it is not possible to communicate directly between two nodes that are far apart. Hence, the intermediate nodes are used to send the packets from source to destination.
- (iv) Heterogeneity: different wireless devices can join MANET and take part in data communication.
- (v) Energy: mobile nodes have limited battery life since they are wireless devices.
- (vi) Limited resources: just like limited energy, resources are also limited and are shared by other devices.
- (vii) Scalability: nodes can freely join or leave the network. Therefore, there is no criterion for the number of nodes in a MANET.

The black hole attack is one of the most dangerous active attacks in ad hoc networks. The black hole node acts by responding to all path request packets, pretending to have the best path to the destination node, and then destroying all received packets.

In this paper, an anomaly-based IDS is presented that will identify black hole attack using KNN clustering and fuzzy inference to select cluster heads. Also, beta distribution and Josang mental logic will use to calculate the trust of each node. In continuation, in Section 2, IDS, along with the black hole attack, will explain. Then, Section 3 describes the ad hoc on demand distance vector (AODV) routing algorithm briefly. Related works are expressed in Section 4. In Section 5, the proposed method is presented. Simulation and conclusion are presented in Sections 6 and 7, respectively.

2. IDS

The main function of IDS is to detect intrusions from audit data (check log files related to an event on each system) collected from the network. The three main functions of IDSs are monitoring and evaluation, detection, and response.

IDSs can be classified as network-based or host-based. Network-based IDSs are not suitable for MANETs because they require monitoring or data collection that passes through the network hardware interface. Host-based IDSs rely on data generated by users or programs located on the host. The software of such IDSs, which are installed individually on all systems and operate separately, is a good candidate for MANETs [1]. Due to the complexities of the MANET, a typical IDS is not suitable for this new environment. Therefore, researchers have focused on developing new IDSs or improving existing systems.

2.1. Vulnerability of MANET. Vulnerability means the possibility of being attacked by some harmful sources or destroying the normal functioning of the network. This section focuses on some security issues and the attacks that result from them. The reasons for some of these attacks are as follows.

First, a MANET is a wireless network, where, unlike wired networks where an adversary must gain physical access to the network wires or pass through several lines of defense at firewalls and gateways, attacks on a wireless network can come from all directions and target any node [6].

Second, MANET nodes are independent nodes that move freely along the network. Therefore, any weak node can be hijacked by malicious nodes, and its behavior can change like a malicious node and damage the entire network.

Third, there is no centralized management across the MANET. As a result, the network cannot be monitored, and it is very difficult to monitor malicious nodes. Also, when the network size is increasing, it is almost impossible to track the behavior of malicious nodes. Nodes in the network trust and cooperate and exchange information. Malicious nodes take advantage of this principle of trust and disrupt the entire network. Fourth, MANET nodes have no power supply requirements and are battery operated. Malicious nodes use this constraint to continuously send packets to the destination node, reduce the power supply energy amount, or

delegate heavy tasks such as calculations to the destination node to discharge the battery [7].

2.2. Attacks on MANETs. Attacks on MANETs challenge mobile infrastructure in nodes that can easily join or leave dynamic requests without a static routing path. A pattern of different attacks on each layer of the network is described as follows [8]:

Application layer: reputation, data corruption, and malicious code

Transfer layer: session hijacking and SYN flooding

Network layer: Sybil, flooding, black hole, gray hole, worm hole, and link spoofing

Data link layer/medium access control (MAC): malicious behavior, selfish behavior, active, and passive

Physical layer: interference, traffic jamming, and eavesdropping

2.2.1. Black Hole Attack. One of the attacks in MANETs is the black hole attack. It only occurs when a malicious node claims to have the shortest path from source to destination and, after creating this illusion, destroys all received packets. An example of this type of attack is given in Figure 1.

As shown in Figure 1, node A is the source node and wants to send information to the destination node I. The possible path to send this data is $A \rightarrow B \rightarrow G \rightarrow I$; however, node D here acts as a destructive node, claiming to have the shortest path from source to destination. It sends the incorrect response to the route request (RREQ) sent by A. The malicious node, therefore, destroys all the data packets it receives. Different aspects are combined in other types of black hole attack, and a few destructive nodes collaborate, and their destructive behavior affects the whole network.

One solution to prevent black hole attack is to interrupt routing. In this method, it is recommended to select more than one route from source to destination. It is recommended that at least three paths be selected from source to destination in each condition. Initially, the source node sends the packet containing the RREQ message to the destination and the intermediate nodes. After receiving the request from the source, nodes that can establish a route from the source to the destination notify the source with a route replay (RREP) message. When the source receives the RREP packet, it is stored in its buffer to receive two more paths. Then, all the packets are placed in the buffer, and after analysis, the appropriate route is selected. The source node selects the safest route based on the number of nodes and avoids the black hole attack [9].

3. AODV Routing Algorithm

Different routing algorithms have been proposed for sending packets in computer networks, one of the most famous of which is AODV. AODV is a demand-based routing protocol, in which all routers are discovered only when needed. AODV aims to reduce the number of

FIGURE 1: Black hole attack.

messages transmitted within the network by discovering the path according to demand.

AODV supports unicast, multicast, and broadcast communications. This protocol can repair the route locally if the path link is lost. AODV can support many nodes in the network. This protocol is ready to leave or join nodes arbitrarily in the network. The path discovery process in this protocol is performed when the source node does not have a valid path to the destination. During the routing process, the following four messages are exchanged:

- (i) RREQ
- (ii) RREP
- (iii) Route error (RERR)
- (iv) Route reply acknowledgment (RREP-ACK)

RREQ message will generate to create a route from source to destination, which with flood method detects the route in the network. This routing request packet includes the IP address of the source node, current sequence number, IP address of the destination node, and last known sequence number. After completing the routing request packet submission step, the RREP is sent from the destination node in the reverse direction to the source node.

Intermediate nodes can respond to the RREQ packet, only if the sequence number of the destination they maintain is greater than or equal to the number in the header of the route request packet. When the intermediate nodes forward the route request packets to their neighbors, they keep the address of the neighbor who delivered the first copy of the packet in their routing tables.

This stored information is then used to construct the reverse path (for RREP routing response). The RREP message is sent when the node is either the destination or has a path to the destination node. For this purpose, AODV uses symmetric links. After receiving the RREP, the source sends the information. If multiple RREP messages are received, the source node selects the shortest path. If the source node is moved, the path discovery process is performed again. When a node is destroyed in an active path or the intermediate node moves, a RERR path error message is generated to notify surrounding nodes that this link does not exist, and then the path rediscovery begins. The hello message is also used to maintain the connectivity of a node [10]. These messages are sent via the UDP/IP protocol. Details of the route discovery process in the AODV protocol are shown in Figure 2.

FIGURE 2: The path discovery process in the AODV routing protocol.

The advantages of the AODV protocol are minimal control overhead, minimal processing overhead, multihop routing capability, dynamic topology maintenance, and loop-free routing process [11].

4. Related Works

This section presents relevant research on intrusion detection in MANETs to detect black hole attack. Table 1 describes the most recent related methods to detect black hole attack, including the author's name, year of publication, used technique, routing protocol, and limitations.

5. The Proposed Method

In this paper, to solve the problem of detecting black hole attack in MANETs, the KNN clustering technique with fuzzy inference will be used. A general schematic of the proposed method (Algorithm 1) is presented in Figure 3. This system detects attack by the data it receives from the nodes. For this purpose, the sent packets are followed by the nodes. Then, with the use of the KNN clustering, the neighborhoods of the nodes are calculated.

The nodes that are in a range are placed in a cluster. Each node calculates the level of trust in the nodes around it and, according to these values, exchanges information with neighboring nodes. Then, among the nodes in each cluster, the node whose value is more than a threshold can be nominated for cluster head. In continuation, with the help of fuzzy inference, from the candidate nodes, the node with the most reliable neighbors at the desired energy level is selected as the cluster head.

After forming the network and calculating the clusters, cluster heads and nodes will trust each other. Then, closed routing in the network will start, and malicious nodes will be identified. The definition of fuzzy logic rules should be such that, on the one hand, it does not increase the complexity of the system. On the other hand, it increases the accuracy of the system in detecting attack. If a new node enters the network, its trust is calculated according to Algorithm 2, as shown in Figure 4.

In the continuation of this section, the details of the proposed method are given.

TABLE 1: Existing techniques along with their limitations to detect black hole attack in MANET.

Author (year)	Technique	Routing protocol	Limitations
Poongodi and Karthikeyan (2016) [12]	Localized secure architecture for MANET (LSAM)	AODV	(i) It increases routing overhead
Kumar and Chadha (2016) [13]	Fuzzy logic generic algorithm	AODV	(i) The methodology is unclear
Shahabi et al. (2016) [14]	Behavior of nodes	AODV	(i) Failed in collaborative gray hole attack
Kumar and Somasundaram (2016) [15]	Cryptokey based on Diffie-Hellman	Cryptokey based black hole detection and avoidance protocol (CBHDAP)	(i) Fails to detect gray hole attack
Arathy and Sminesh (2016) [16]	Average destination sequence number (ADSN)	AODV	(i) The method fails when smart gray holes send an average sequence number
Dumne and Manjaramkar (2016) [17]	Cooperative bait detection scheme (CBDS)	AODV and DSR	(i) Sensitive to select the threshold value to detect a malicious hole
Nitnaware and Thakur (2016) [18]	Mitigation algorithm to avoid and prevent genuine nodes from malicious attack	Dynamic MANET on demand (DYMO)	(i) Sensitive to select the threshold value to detect a black hole
Tamilselvi and Ganesh Babu (2017) [19]	Bait request	DSR	(i) The delay is significant if the route is long
Arulkumaran and Gnanamurthy (2017) [20]	Fuzzy logic	AODV	(i) It fails to detect smart gray holes
Panos et al. (2017) [21]	Dynamic threshold cumulative sum	AODV	(i) Only detects black hole attack
Dorri (2017) [22]	Extended data routing information	AODV	(i) Fails to detect smart gray holes
Dhende et al. (2017) [23]	Neighbor node opinion	AODV	(i) Cannot handle mobility and routing overhead
Kamel et al. (2017) [24]	Trust value	AODV	(i) Fails to detect gray holes
Kollati and Somasundaram (2017) [25]	Integrated bloom filter in watchdog algorithm	AODV	(i) Complex and unclear methodology
Gurung and Chauhan (2018) [26]	Mitigating black hole effects through detection and prevention (MBDP)	AODV	(i) High routing overhead (ii) It fails to detect smart gray holes
Tiruvakadu and Pallapa (2018) [27]	Honeypot with the aid of attack tree and history database	AODV	(i) It increases routing overhead (ii) The delay is significant
Khamayseh et al. (2018) [28]	Usage of the observer node to detect the delivery of the transmission of data packets to the next hop	AODV	(i) It increases the overhead that affects the delay of packets in the case of sparse networks
Hammamouche et al. (2018) [29]	The reputation of nodes and multihop acknowledgment	AODV and OLSR	(i) Sensitive to select the threshold value to detect a black hole (ii) Fails to detect gray holes

FIGURE 3: Overview of the proposed method (Algorithm 1).

FIGURE 4: Algorithm implementation steps after insertion of a new node (Algorithm 2).

5.1. KNN Algorithm. Among the classification algorithms in data mining, the KNN algorithm is less complicated in theory. KNN's goal is to put the closest values in a cluster. The KNN algorithm is very effective in various fields of pattern recognition, cancer diagnosis, text classification, and so on. KNN is a lazy learning or instance-based method. This algorithm labels a sample of data based on its nearest K neighborhood. The similarity score of each node is used as the weight of the neighboring node cluster. The score it receives for the similarity of each instance to the cluster is considered as its weight [30]. As shown in Figure 5, this weight has been used effectively to calculate the distance among neighbors [31].

One of the important properties of the KNN algorithm is that it refers to a single or multidimensional feature vector and calculates the Euclidean distance. A two-dimensional vector is used to represent the nodes in the proposed algorithm. Therefore, the distance of the nodes from each other is calculated according to the Euclidean relation (1), and the clusters are formed.

$$d(x_i, x_j) = \sqrt{\sum_{i,j} (x_i - x_j)^2}, \quad (1)$$

where x_i and x_j are network nodes.

With the use of the KNN classification, which is based on learning, the performance of the proposed method for determining the number of attackers will improve. Then, the new node is compared with a known node that is like it.

Each node represents a point in an N -dimensional space. In this way, all known nodes are stored in the N -dimensional pattern space. Then, when N unknown nodes enter the system, the KNN starts searching the pattern space for K known nodes that are close to the unknown node. To cluster KNN, an unknown node is mapped to the most common

FIGURE 5: Use of KNN for clustering.

cluster of KNNs. Since the choice of parameter K in this algorithm is very important, its value can change from one to the square root of the size of the training set [32].

5.2. Fuzzy Logic. The fuzzy logic technique is simple to implement and produces accurate output by eliminating ambiguities. In classical logic, elements are labeled as 0 or 1, while, in fuzzy set theory, the set of values is between 0 and 1. Fuzzy logic can make rational decisions in an environment with imprecision, uncertainty, and incomplete information. Therefore, using data collected in environments that contain such properties makes it an ideal method to be applied in scenarios with real, continuous-valued elements [33].

Wu and Banzhaf [34] recommend the validity of using fuzzy logic for a network anomaly detection system (NADS) for two reasons:

- (1) Intrusion detection problem involves many numeric attributes in collected audit data and various derived statistical measures. Building models directly on numeric data causes high detection errors.
- (2) The security itself includes fuzziness because the boundary between the normal and abnormal is not well defined.

The fuzzy inference system aims to map an input to output, applying fuzzy reasoning in the process. The following are the steps of the fuzzy inference system [33].

5.2.1. Fuzzification of Inputs. The first step in fuzzy inference systems is to receive inputs and determine their degree of membership to each of the fuzzy sets through membership functions. The output of this step is a fuzzy degree that determines the amount of input membership in the fuzzy input set. This output is always a number between zero and one.

5.2.2. Apply the Fuzzy Operators. After the fuzzification of inputs, the degree of accuracy of each hypothetical part will be determined. If the hypothetical part has several parts, fuzzy operators are used to combine the degree of accuracy of the parts and produce a number as the degree of accuracy

of the hypothetical part. The number obtained from this process goes to the output function.

5.2.3. Apply the Implication Method. Before applying the implication method, one weight must be determined for each rule. Each rule weights the range of zero to one. This weight is applied to the value obtained from the hypothetical part. After estimating the appropriate values for the weights of each rule, the implication method is implemented. The input of the process indicates a number, and its output is a fuzzy set.

5.2.4. Output Aggregation. The final decision in the fuzzy system is made based on all the rules. Therefore, the outputs from the different rules must be aggregated. Aggregation is the process by which fuzzy sets providing the output of each rule are combined into a fuzzy set.

5.2.5. Defuzzification. The input of the defuzzy process is a fuzzy set, and its output is a number.

The reasons for the importance of using a fuzzy inference system and applying this method in intrusion detection are as follows:

- (1) Ability to work in two areas of classification and clustering
- (2) Ability to interpret and monitor the expert in the process of learning and updating the fuzzy system, including fuzzy sets and its database rules
- (3) Ability in of intelligent systems in issues that face uncertainty

5.3. Trust Plan. Before explaining the trust plan, in Section 5.3.1, the assumptions of the trust plan for the network are described.

5.3.1. Assumptions. Before network establishment, a trusted authority is responsible for the following items.

Each network node has a unique identity (Id_x is the unique identifier of node x). A random number called S is selected as the main secret of the network that only the network nodes are aware of it. For each network node, such as x , a key based on node ID (K_x) is generated using the following:

$$K_x = S.H(Id_x), \quad (2)$$

where $H(.)$ is a hash function. Data cannot be retrieved from a hash value because these types of algorithms are entirely one-way (nonreversible) and are mostly used to speed up the search process and ensure the correctness of data during transmission.

The hash operation converts the input data stream into a small summary. Given that, in (2), the hard problem of discrete logarithms is used, the enemy, even if he is somehow aware of the values Id_x and K_x algorithm of the hash function, cannot get the value of the main secret of the node.

Each node, like x , is preloaded with a unique identifier and key before a trusted authority develops the network.

A trust server certificate should be used, which is the trust whose public key is known to all valid nodes. First, the keys are created and then exchanged through the relationship between the trust and each node.

Each node must request a certificate from the trust server before entering the ad hoc network. Once each node has securely verified its authenticity for the trust server, it receives only one certificate. All nodes must maintain the new certificates received from the trust server. These certificates are used to verify the authenticity of the node to other nodes during the exchange of routing messages.

5.3.2. Method. In the proposed method, the initial clustering is carried out by the KNN algorithm. Each node after the transaction with another node, depending on how the opposite node behaves during the transaction and the quality of service received, evaluates that transaction and considers it as a positive (p) or negative (n) transaction. Then, at certain intervals, each node calculates the amount of trust in that node according to the number of positive and negative transactions stored for each neighboring node. The proposed scheme uses the beta distribution and Josang mental logic to calculate trust. Therefore, before explaining the proposed method for calculating direct trust between two nodes, these two methods and how to use them in calculating trust have been described.

(1) *Beta Distribution.* In statistics and probability theory, the beta distribution is a family of the continuous probability distribution. It is usually used for random variables with a continuous value in the range $[0, 1]$ and is defined by two parameters α and β . According to the properties of beta distribution, the desired mathematical expectation is expressed as follows:

$$E(b|\alpha, \beta) = \frac{\alpha}{\alpha + \beta}. \quad (3)$$

The reasons for using the beta distribution are as follows:

- (1) In the trust analysis models, considering that the values of all three trusts (general, recent, and final trust) and the value of reputation (trust result) are numerical between 0 and 1. Therefore, Beta distribution is used.
- (2) Considering that the nodes in the network interact with each other and these interactions can be positive or negative, the beta distribution has been used to model this issue, which has considered parameters for these interactions.
- (3) This statistical model has been successful in generating probability from uncertain binary data [35].

To evaluate trust using beta distribution, according to equation (4), the number of interactions with a positive result (p) is attributed to α parameter, and the number of interactions with a negative result (n) is attributed to β parameter.

$$\begin{aligned}\alpha &= p + 1, \\ \beta &= n + 1.\end{aligned}\quad (4)$$

Finally, trust is calculated as follows:

$$T = E(b|\alpha, \beta) = \frac{\alpha}{\alpha + \beta} = \frac{p + 1}{p + n + 2}. \quad (5)$$

(2) *Josang Mental Logic*. Jøsang et al. [36] addressed uncertain belief representation by using subjective logic, in which an opinion regarding belief is represented as a triple (b, d, u) , where b , d , and u denote the degrees of belief, disbelief, and uncertainty, respectively, and $b + d + u = 1$.

Then, another parameter is entered as the base rate (a), which is in the range $[0, 1]$ to the previous representation and forms a quadruple (b, d, u, a) . The base rate determines the extent to which uncertainty has contributed to the mathematical expectation of the level of trust [36]. The mathematical expectation of the trust level (a value that is ultimately estimated as the trust level of the triple (b, d, u)) is considered as follows:

$$T = b + au. \quad (6)$$

The values of b , d , and u can be calculated in different ways. These values are calculated as follows:

$$\begin{aligned}b &= \frac{p}{p + n + 1}, \\ d &= \frac{n}{p + n + 1}, \\ u &= \frac{1}{p + n + 1}.\end{aligned}\quad (7)$$

Among the reasons for using Josang mental logic are the following:

- (1) This trust model can identify the level of trust well
- (2) Using this mental logic in which the nodes' belief in the trust is expressed, a false claim is identified

(3) *Calculation of Direct Trust between Two Nodes*. In the proposed method, the calculation of trust is carried out in three steps:

- (1) T_{Total} : calculation of total trust in terms of the beta distribution. In this step, total trust is calculated based on beta distribution and considers all the interactions between the two nodes. The value of all transactions is deemed the same.
- (2) T_{Last} : calculation of the last trust according to Josang mental logic. The last trust shows the outcome of trust in the last interactions. To calculate the trust level of the last transactions, in addition to the positive or negative evaluation of the transaction, a coefficient is given based on the elapsed time of the transaction. Then, at certain intervals, the last trust is calculated based on the last few transactions. The weighting coefficient of each transaction is calculated according to the following:

$$\alpha_i = \frac{1}{\lambda_i}, \quad (8)$$

where i is the unit of time past the transaction, λ_i is the forget coefficient and can be changed according to the network conditions, and α_i is the transaction coefficient that i time unit passes from the transaction. Given that, over a long period, the effect of the transaction coefficient becomes very low and close to zero, a time window is provided for calculating these coefficients. The size of the time sliding window can be changed according to the node's trust conditions and values. If the node's final trust value is below the threshold or the node behaves badly in most of its last interactions, the sliding window size is reduced. Therefore, node behavior is detected sooner. After assigning the appropriate coefficients for the last transactions, the amount of the last trust is calculated based on Josang's mental logic.

- (3) T_{Final} : in this step, the final trust is calculated based on the results of the previous two steps with the help of fuzzy logic.

A fuzzy system consists of three parts: fuzzification, inference engine, and defuzzification.

- (A) Fuzzification: this section maps each input value to the corresponding fuzzy set. As a result, it assigns an integer or degree of membership to each fuzzy set. Figure 6 graphically shows the membership functions of three fuzzy sets.

Hence, the degree of membership of each feature to the formed fuzzy set is expressed as follows:

$$\mu_R(F_i), \mu_L(F_i) \in \{\text{low, medium, high}\}. \quad (9)$$

- (B) Inference engine: the inference engine contains a database of different rules and methods for inferring rules that process fuzzy values. The rule database is a series of IF-THEN rules that relate fuzzy input variables to the fuzzy output variables by language variables. Each of the rules is described by a fuzzy set and implicit fuzzy operators OR, AND, and so on. Mamdani fuzzy system [37] is a simple law-oriented method that does not require complex calculations and can use IF-THEN rules to control systems. To calculate the degree of trust between two nodes, if a reasonable probability is obtained for T_{Total} , but the probability obtained is not suitable for T_{Last} , then the trust is equal to unreliable. Also, if the probability obtained is unsuitable for T_{Total} and the probability obtained is appropriate for T_{Last} , the trust obtained will be slightly less reliable; therefore, according to these hypotheses, the desired laws are formed. The Mamdani inference system uses fuzzy sets as the result of the law, and the output of each law is nonlinear and fuzzy. It is also different from other inference systems in terms of the defuzzification method. In the Mamdani inference algorithm,

FIGURE 6: Graphic representation of the membership functions of three fuzzy sets.

logical results are expressed with a relatively simple structure. They are mostly used in systems that can interpret rules and decision support systems. In this part, the outputs of all the rules are combined to form a fuzzy composite set.

Since decision-making is based on all the rules in fuzzy inference systems, the rules must be able to be combined in some way to make a decision. Aggregation is how all output sets of each rule are combined into a single fuzzy set. The aggregation process input is a list of output functions cut by the implication process for each rule, and its output is a fuzzy set. There are different methods for aggregation, the most important of which is maximization and addition. In this paper, addition method is used. The reason for using the addition method is that the addition method considers the sum of the rules, while, in the maximum method, it considers the rule that has the maximum value and ignores the rest of the rules.

- (C) Defuzzification: it finds an exact output value from the solution fuzzy space. The rules for calculating the final trust in this paper are shown in Table 2.

As mentioned, total trust is calculated according to the beta distribution, and the last trust is based on Josang mental logic. Using the amount of trust calculated for each node of cluster and cluster heads, the final level of node trust to each other will be calculated according to Table 2.

This process of calculating trust is also carried out in the second level of the network between the cluster heads and the main station. The only difference is that, due to the greater importance of communication at this level, the sliding window size is considered smaller to determine the effect of transaction sooner. Also, if the main station identifies a cluster head as destructive or its level of trust drops to some extent, it will quickly delete and add it to the blacklist.

(4) *Reputation Calculation*. The reputation of each node is the result of the trust of other nodes in the desired node. Nodes within each cluster periodically send trust values to each node in the trust table to the cluster head. The cluster head node also updates the node's reputation value in its trust

TABLE 2: Fuzzy rules for calculating the final trust.

T_{Total}	T_{Last}	T_{Final}
High	High	Exactly trusted
High	Rather high	Trusted
High	Medium	Rather trusted
High	Low	Untrusted
High	Very low	Untrusted
Rather high	High	Trusted
Rather high	Rather high	Trusted
Rather high	Medium	Rather trusted
Rather high	Low	Untrusted
Rather high	Very low	Untrusted
Medium	High	Rather trusted
Medium	Rather high	Rather trusted
Medium	Medium	Rather trusted
Medium	Low	Untrusted
Medium	Very low	Untrusted
Low	High	Rather untrusted
Low	Rather high	Rather untrusted
Low	Medium	Rather untrusted
Low	Low	Untrusted
Low	Very low	Untrusted
Very low	High	Rather untrusted
Very low	Rather high	Rather untrusted
Very low	Medium	Untrusted
Very low	Low	Untrusted
Very low	Very low	Untrusted

table when it receives a certain number of opinions about the node. The degree of influence of each node in aggregating opinions depends on the reputation of the node in the cluster head trust table and is calculated based on the following:

$$\begin{aligned}
 Rep_x &= \alpha_y T_{y \rightarrow x} + \alpha_k T_{k \rightarrow x} + \alpha_l T_{l \rightarrow x} + \dots \\
 &= \sum_{i=\{y,k,l,\dots\}, i \neq x}^{cnt} \alpha_i T_{(i \rightarrow x)}, \quad \text{where } \sum_{i=1}^{cnt} \alpha_i = 1,
 \end{aligned} \tag{10}$$

where Rep_x specifies the reputation value of node x , $T_{(i \rightarrow x)}$ the trust value of node i to x , i is the set of nodes that have expressed their opinion about node x , cnt is the total number of nodes that commented on node x , and α_i is the weight factor of each opinion and shows the effect of each node's opinion in calculating the reputation of node x . The weight factor of node y is calculated as (11).

$$\alpha_y = \frac{Rep_y}{\sum_{i=\{y,k,l,\dots\}}^{cnt} Rep_i}. \tag{11}$$

The cluster head uses the calculated reputation values in addition to updating the reputation of each node in important processes such as aggregating information and selecting a new cluster head. Also, nodes within the cluster can use these values as indirect trusts. The cluster head node periodically, based on its reputation table, prepares a list of unreliable nodes and distributes them in the cluster so that other nodes are more careful in communicating with them.

(5) *Selection of a New Cluster Head*. The selection of the cluster head is usually carried out periodically. Unless, for

any reason, such as a dramatic drop in energy levels, this choice is made sooner. The current cluster head carries out cluster head selection. In the proposed plan, the cluster head is selected according to the following two steps:

Step 1. Screening: as mentioned, the nodes and their Euclidean distance from each other are first calculated using the KNN algorithm, and the nodes are placed inside the cluster. Then, according to its reputation table, from the nodes within the cluster, the nodes whose reputation value is more than the threshold are selected. These nodes will enter the second step as candidate nodes for the cluster head. The threshold of trust can be changed according to the network conditions and is usually more than 0.5.

Step 2. Cluster head selection based on fuzzy logic: given that the purpose of this paper is to improve network security and energy has always been one of the most important parameters in MANETs, the two parameters of remaining energy of the node and the maximum number of trustable neighbors are considered as fuzzy system inputs.

(6) *Detection of Malicious Nodes.* Once the trusted network is formed, and the clusters, cluster heads, and trust of nodes to each other are calculated, the packet routing in the network begins, and malicious nodes are identified. The steps of this algorithm are as follows:

- (1) Nodes are randomly distributed in the network, and each node receives a unique ID.
- (2) The nodes are clustered using the KNN algorithm and fuzzy inference, and for each cluster C_i , a cluster head CH_i is selected.
- (3) The source node S_i sends the path request to the cluster head CH_i to find the destination.
- (4) The cluster head CH_i first checks the nodes of its cluster to find the destination D_i . If the desired node is not found, it sends the request to the other cluster heads to find the shortest path.
- (5) The node that is introduced as the next destination sends a reply to the cluster head.
- (6) The cluster head sends the ID of the node that gave the reply to the trust.
- (7) The trust server checks the destination node ID and, if allowed, notifies the cluster head. Otherwise, the trust server adds it to its blacklist and notifies other cluster heads to update their blacklist. Then, step 4 is repeated.
- (8) End.

The packet routing and black hole attack detection in the proposed algorithm is shown in Figure 7.

6. Simulation

First, the network is simulated, then the network parameters will be evaluated, and the simulation results will be

presented. Then, the results of the proposed method are compared with the results of the trust based technique [38], three-layered ANN for classification and SVM as the supervised learning model [39], neurofuzzy inference system (ANFIS), particle swarm optimization (PSO) [40], and fuzzy trust approach to detect black hole attack based on a certificate authority, energy auditing, packet veracity check, and trust node to improve the performance of AODV [41]. Finally, the results will be discussed and analyzed.

The parameters of the simulated network for the proposed algorithm and other compared methods are shown in Table 3.

The parameters for performance evaluation of the proposed method are as follows:

- (1) Packet lost rate (PLR): in a transmission interval, the PLR can be calculated as follows:

$$PLR(\%) = \frac{N^{tx} - N^{rx}}{N^{tx}} \times 100, \quad (12)$$

where N^{tx} and N^{rx} are the total number of transmitted and received packets, respectively.

- (2) Throughput (TH): it is defined as follows:

$$TH = \frac{N^{rx}}{T}, \quad (13)$$

where T is the simulation time.

- (3) Packet delivery ratio (PDR): the PDR can be calculated as follows:

$$PDR(\%) = \frac{N^{rx}}{N^{tx}} \times 100. \quad (14)$$

- (4) Total network delay (TND) (T_{total}): it is the total delay of the entire network in all cases. This amount is obtained by calculating the delay of packets that arrived at the destination and not lost packets.
- (5) Normalized routing load (NRL): it is the number of routing packets transmitted per number of data packets delivered at the destination as follows:

$$NRL = \frac{N^{rsx}}{N^{dx}}, \quad (15)$$

where N^{rsx} is the number of transmitted routing packets, and N^{dx} is the number of delivered data packets.

6.1. Results. The desired network is simulated in the following cases:

- (A) The network does not have any malicious nodes
- (B) Existence of malicious node and nonuse of the proposed IDS
- (C) Existence of malicious node and use of the proposed IDS

FIGURE 7: General process of packet routing and black hole attack detection in the proposed method.

TABLE 3: The simulation parameters.

Parameters	Value
Simulator	NS-3.31
Network area	800 m × 800 m
Number of nodes	20, 40, 60, 80, 100
Mobility model	Random walk mobility (Bonn motion)
Routing protocol	AODV
Simulation time	600 s
MAC type	IEEE 802.11
Traffic type	Constant bitrate (CBR)
Agent	UDP
Packet size	512 bytes
Mobility	0.5–1.0 m/s
Pause time	5–20 s
Transmission range	250 m
Antenna	Omni antenna

For the previously mentioned cases, to ensure the result accuracy, the simulated network runs five times, and then the average of results is calculated.

Table 4 presents the parameters of the simulated network without attack.

According to Table 4, the NRL in the network with 80 nodes is the lowest. Also, the TH in the network with 60 nodes has the highest value. T_{total} in the network with 20 nodes has the lowest value, and the PDR in the network with 80 nodes has the highest value and the lowest PLR.

Table 5 shows the results related to the network simulation in the presence of a malicious node. In the case where a network has 20 nodes, the NRL has the lowest value. The TH has a maximum value at the 80 nodes. T_{total} at 20 nodes has the lowest value. The PDR in the network with 80 nodes has the highest value and the lowest PLR.

TABLE 4: Performance parameters of the simulated network without attack.

Number of nodes	PLR (%)	TH (kbps)	PDR (%)	T_{total} (ms)	NRL
20	8.88	173.81	91.12	9.25	2.94
40	7.69	152.98	92.31	17.31	2.86
60	9.77	187.90	90.23	15.03	2.86
80	7.66	160.37	92.34	24.48	1.60
100	8.28	152.19	91.72	23.80	2.87

TABLE 5: Performance parameters of the simulated network in the presence of a malicious node.

Number of nodes	PLR (%)	TH (kbps)	PDR (%)	T_{total} (ms)	NRL
20	71.30	106.10	28.70	16.31	8.11
40	67.62	101.29	32.38	24.54	10.73
60	69.63	99.63	30.37	31.21	11.02
80	67.07	136.69	32.93	48.20	13.09
100	68.14	133.08	31.86	74.87	13.79

Table 6 presents the results of the simulations performed by the proposed method in the presence of a malicious node. According to Table 6, the NRL in 100 nodes has the lowest routing load in the network. Also, 60 nodes in the network have the highest TH. The lowest T_{total} occurs at 20 nodes in the network, and the PDR on the network with 40 nodes is the highest value and the lowest PLR.

Finally, the average improvement percentage of the proposed method over black hole attack without using the proposed method is shown in Table 7.

As shown in Table 7, all parameters are improved with the proposed method during the black hole attack, although the most improvement has occurred in the PDR parameter.

TABLE 6: Performance parameters of the simulated network with the proposed method in the presence of the malicious node.

Number of nodes	PLR (%)	TH (kbps)	PDR (%)	T_{total} (ms)	NRL
20	9.36	176.53	90.64	2.88	3.53
40	7.30	150.91	92.70	13.44	4.02
60	9.92	192.14	90.08	12.35	3.32
80	7.64	162.31	92.36	11.28	3.39
100	8.71	164.72	91.29	14.09	3.22

TABLE 7: Average improvement percentage of the proposed method over black hole attack case.

Parameter	The improvement over black hole attack (%)
T_{total}	72.31
TH	46.78
PDR	192.54
PLR	87.51
NRL	69.19

6.2. Comparison of the Performance Parameters with Other Methods. In this section, the data obtained for each of the parameters that are mentioned in Section 6.1 will be compared with the recently proposed methods for black hole attack detection.

6.2.1. PLR. Whatever the PLR is less, the data transfer rate is more, and the network performance is better. Figure 8 shows that the network without attack had the lowest PLR compared to other methods. In the case of a black hole attack, the amount of PLR using the proposed method compared to the attack case is reduced for each number of nodes. Also, the proposed method is closer to the normal condition compared with other methods.

6.2.2. TH. Figure 9 shows the graph obtained from the results of the average throughput. It can be seen that, as expected, the average network throughput of the proposed method is better than the normal condition (no black hole attack) and also better than other methods in [38–41].

As shown in Figure 9, the network throughput of the proposed method in the presence of a malicious node is higher than normal conditions in all cases except 40 nodes.

6.2.3. PDR. Figure 10 shows the results of the PDR obtained for 20, 40, 60, 80, and 100 nodes. As it is clear, the PDR of the proposed method is very close to the normal condition. Also, the PDR of the network during the black hole attack is very less than the usage of the other methods [38–41].

6.2.4. TND. Figure 11 shows the TND of the entire network for different methods where the network has been simulated.

Since the lower delay of the entire network makes the higher data transfer rate and the better network performance, the proposed method to detect malicious nodes should not exceed the expected delay of the network.

FIGURE 8: Comparison of PLR for different methods.

FIGURE 9: Comparison of network throughput for different methods.

FIGURE 10: Comparison of PDR for different methods.

FIGURE 11: Comparison of total network delay for different methods.

According to Figure 11, as expected, the TND of the normal condition in any number of nodes is less than the presence of a malicious node. In the case that the number of nodes is low, the total delay of the network in all methods is almost close to each other. However, with increasing the number of nodes to more than 20 nodes, the delay difference between different methods increases slowly. The performance of the proposed method is slightly better than other methods.

6.2.5. NRL. The more stable the network topology and the less it changes, the less the need to do the routing process is because the routing tables in the nodes are less changed and more stable. As a result, time and network resources are spent transmitting the data packets instead of routing packets. Figure 12 shows a graph of the NRL for all methods, in which the network is simulated.

According to Figure 12, as expected, if the network does not have any attack, the NRL of the network is less. The proposed method has the best NRL values between others for all numbers of nodes and is closer to the normal condition.

6.3. Discussion. In this section, to evaluate the proposed method, the results are compared with other methods in [38–41].

In [38], with the use of a technique based on trust, black hole attack will detect. The main advantage of the proposed method over [38] is that, in the proposed method, in addition to the determination of trust of nodes to each other with the calculation of reputation, its clustering will carry out with the KNN algorithm. Therefore, the proposed method can better separate the untrusted nodes than [38]. Also, in the proposed method, using beta distribution, Josang mental logic, and fuzzy inference will help better calculate reputation.

Reference [39] uses the three-layered ANN for classification and SVM as the supervised learning model. The

FIGURE 12: Comparison of NRL for different methods.

simulation results of the proposed method show that usage of the KNN algorithm for clustering and beta distribution, Josang mental logic, and fuzzy inference for calculation of node trust will perform better black hole attack detection in the network. Also, Figures 8–12 have shown that the usage of the ANN + SVM works better than trust-based technique [38] in all parameters on average.

In [40], ANFIS and PSO are used to detect black hole attack. As it is clear from Figures 8–12, ANFIS + PSO can perform better than ANN + SVM [39] and trust-based technique [38], but the proposed method still has a better performance in all parameters than ANFIS + PSO.

Finally, the proposed method is compared with the fuzzy trust approach [41] that is based on a certificate authority, energy auditing, packet veracity check, and trust node to detect black hole attack. Although this method has the best performance in comparison to [38–40], where the main reason is the usage of fuzzy logic for the determination of trust of nodes in the network, the results in Figures 8–12 have shown that the proposed reputation calculation performs better than the fuzzy trust approach [41] in all TH, TND, NRL, PDR, and PLR parameters.

In Table 8, the average improvement percentage of the proposed method for different parameters including TND, TH, PDR, PLR, and NRL is compared with [38–41]. As shown in Table 8, the proposed method performs better than all compared methods, especially in the PLR parameter.

In most of the previous researches [38–40], to separate the malicious node from the nonmalicious node, a threshold value has been used, which in case of similarity of the malicious and nonmalicious node behavior system may fail. Therefore, to distinguish properly in these conditions, the fuzzy inference can be a good option, as shown in Table 8. The initial clustering by the KNN algorithm causes each node within the cluster to calculate the degree of trust to the nodes in its neighborhood.

Also, the proposed method could be used for future infrastructures such as cognitive radio enabled 5G-based internet of things (IoT) [42]. The proposed algorithm that uses the KNN for clustering and reputation to detect black

TABLE 8: Average improvement percentage of the proposed method over other methods.

Parameter	Trust-based technique [38]	ANN + SVM [39]	ANFIS + PSO [40]	Fuzzy trust approach [41]
TND	21.50	21.50	20.87	11.71
TH	9.42	4.75	4.35	3.79
PDR	8.83	5.53	4.56	4.03
PLR	46.35	35.81	31.70	29.20
NRL	43.47	23.09	16.63	16.62

hole attack with high accuracy in wireless networks can be used in 5G, which is a subset of wireless networks.

7. Conclusion

The main purpose of this paper is to provide an efficient method to detect a black hole attack. According to the results, in a black hole attack, the network parameters such as TH, TND, NRL, PDR, and PLR have been improved. The proposed method includes the KNN algorithm for clustering and beta distribution, Josang mental logic, and fuzzy inference to calculate trust. Then, the reputation of nodes will be calculated in the trust server. Finally, the cluster head will distribute a list of unreliable nodes in the cluster periodically based on the reputation table.

The proposed method can be developed or combined with other methods and used to identify other attacks. Also, in the nodes clustering section, other decision-making methods such as SVM, neural network, decision tree, and naive Bayes methods can be used.

Data Availability

All relevant data are included within the article.

Conflicts of Interest

The author declares no conflicts of interest.

Acknowledgments

This work was supported by IROST.

References

- [1] F. R. Yu and H. Tang, "Distributed node selection for threshold key management with intrusion detection in mobile ad hoc networks," *Wireless Networks*, vol. 16, no. 8, pp. 2169–2178, 2010.
- [2] M. Elboukhari and M. Azizi, "Intrusion detection systems in mobile ad hoc networks: a survey," in *Proceedings of the Workshop on Codes, Cryptography and Communication Systems (WCCCS)*, El Jadida, Morocco, November 2014.
- [3] V. Anjana Devi and R. S. Bhuvaneswaran, "Agent based cross layer intrusion detection system for MANET," *Advances in Network Security and Applications*, vol. 196, pp. 427–440, 2011.
- [4] T. Poongothai and K. Duraiswamy, "Intrusion detection in mobile AdHoc networks using machine learning approach," in *Proceedings of the International Conference on Information Communication and Embedded Systems*, Chennai, India, February 2014.
- [5] A. Vij and V. Sharma, "Security issues in mobile adhoc network: a survey paper," in *Proceedings of the International Conference on Computing, Communication and Automation*, Greater Noida, India, April 2016.
- [6] Y. Zhang and W. Lee, "Security in mobile ad-hoc networks," *Ad Hoc Networks*, Springer, in *Proceedings of 4th International Conference, ADHOC-NOW 2005*, pp. 249–268, October 2005.
- [7] S. Kumar and K. Dutta, "Security issues in mobile ad hoc networks: a survey," *Security, Privacy, Trust, and Resource Management in Mobile and Wireless Communications*, IGI Global, Hershey, PA, USA, pp. 176–221, 2014.
- [8] M. Enshaei, Z. Mohd Hanapi, and M. Othman, "A review: mobile ad hoc networks challenges, attacks, security, vulnerability and routing protocols," *International Journal on Communications Antenna and Propagation (IRECAP)*, vol. 4, no. 5, pp. 168–179, 2014.
- [9] F. H. Tseng, L. D. Chou, and H. C. Chao, "A survey of black hole attacks in wireless mobile ad hoc networks," *Human-centric Computing and Information Sciences*, vol. 1, no. 4, pp. 1–16, 2011.
- [10] V. Rishiwal, S. K. Agarwal, and M. Yadav, "Performance of AODV protocol for H-MANETs," in *Proceedings of the International Conference on Advances in Computing*, April 2016.
- [11] B. Issac, K. Hamid, and C. E. Tan, *Analysis of Demand Driven Ad Hoc Routing Protocols on Performance and Mobility*, pp. 136–141, IWTS, Shah Alam, Malaysia, 2006.
- [12] T. Poongodi and M. Karthikeyan, "Localized secure routing architecture against cooperative black hole attack in mobile ad hoc networks," *Wireless Personal Communications*, vol. 90, no. 2, pp. 1039–1050, 2016.
- [13] R. Kumar and R. Chadha, "Mitigation of black hole attack using generic algorithms and fuzzy logic," *International Journal of Engineering Sciences & Research Technology*, vol. 5, no. 6, pp. 818–826, 2016.
- [14] S. Shahabi, M. Ghazvini, and M. Bakhtiarian, "A modified algorithm to improve security and performance of AODV protocol against black hole attack," *Wireless Networks*, vol. 22, no. 5, pp. 1505–1511, 2016.
- [15] K. V. Kumar and K. Somasundaram, "An effective CBHDAP protocol for black hole attack detection in MANET," *Indian Journal of Science and Technology*, vol. 9, no. 36, 2016.
- [16] K. S. Arathy and C. N. Smimesh, "A novel approach for detection of single and collaborative black hole attacks in MANET," *Procedia Technology*, vol. 25, pp. 264–271, 2016.
- [17] P. R. Dumne and A. Manjaramkar, "Cooperative bait detection scheme to prevent collaborative blackhole or grayhole attacks by malicious nodes in MANETs," in *Proceedings of the International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)*, Noida, India, September 2016.
- [18] D. Nitnaware and A. Thakur, "Black hole attack detection and prevention strategy in DYMO for MANET," in *Proceedings of the International Conference on Signal Processing and Integrated Networks*, Noida, India, February 2016.

- [19] P. Tamilselvi and C. Ganesh Babu, "An efficient approach to circumvent black hole nodes in manets," *Cluster Computing*, vol. 22, no. S5, pp. 11401–11409, 2017.
- [20] G. Arulkumaran and R. K. Gnanamurthy, "Fuzzy trust approach for detecting black hole attack in mobile adhoc network," *Mobile Networks and Applications*, vol. 24, pp. 386–393, 2017.
- [21] C. Panos, C. Ntantogian, S. Malliaros, and C. Xenakis, "Analyzing, quantifying, and detecting the blackhole attack in infrastructure-less networks," *Computer Networks*, vol. 113, pp. 94–110, 2017.
- [22] A. Dorri, "An EDRI-based approach for detecting and eliminating cooperative black hole nodes in MANET," *Wireless Networks*, vol. 23, no. 6, pp. 1767–1778, 2017.
- [23] S. Dhende, S. Musale, S. Shirbahadurkar, and A. Najan, "SAODV: black hole and gray hole attack detection protocol in MANETs," in *Proceedings of the International Conference on Wireless Communications*.
- [24] M. B. M. Kamel, I. Alameri, and A. N. Onaizah, "STAODV: a secure and trust based approach to mitigate blackhole attack on AODV based MANET," in *Proceedings of the Advanced Information Technology, Electronic and Automation Control Conference*.
- [25] V. K. Kollati and S. Somasundaram, "IBFWA: integrated Bloom Filter in Watchdog Algorithm for hybrid black hole attack detection in MANET," *Information Security Journal: A Global Perspective*, vol. 26, no. 1, pp. 49–60, 2017.
- [26] S. Gurung and S. Chauhan, "A dynamic threshold based approach for mitigating black-hole attack in MANET," *Wireless Networks*, vol. 24, no. 8, pp. 2957–2971, 2018.
- [27] D. S. K. Tiruvakadu and V. Pallapa, "Honey-pot based black-hole attack confirmation in a MANET," *International Journal of Wireless Information Networks*, vol. 25, no. 4, pp. 434–448, 2018.
- [28] Y. M. Khamayseh, S. A. Aljawarneh, and A. E. Asaad, "Ensuring survivability against Black Hole Attacks in MANETS for preserving energy efficiency," *Sustainable Computing: Informatics and Systems*, vol. 18, pp. 90–100, 2018.
- [29] A. Hammamouche, M. Omar, N. Djebari, and A. Tari, "Lightweight reputation-based approach against simple and cooperative black-hole attacks for MANET," *Journal of Information Security and Applications*, vol. 43, pp. 12–20, 2018.
- [30] S. Taneja, C. Gupta, K. Goyal, and D. Gureja, "An enhanced k-nearest neighbor algorithm using information gain and clustering," in *Proceedings of the International Conference on Advanced Computing & Communication Technologies*, Roh-tak, India, February 2014.
- [31] A. R. Yusof, N. I. Udzir, and A. Selamat, "An evaluation on KNN-SVM algorithm for detection and prediction of DDoS attack," in *Proceedings of the International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, Morioka, Japan, August 2016.
- [32] A. B. Hassanat, M. A. Abbadi, G. A. Altarawneh, and A. A. Alhasanat, "Solving the problem of the k parameter in the KNN classifier using an ensemble learning approach," *International Journal of Computer Science and Information Security*, vol. 12, no. 8, pp. 33–39, 2014.
- [33] A. H. Hamamoto, L. F. Carvalho, L. D. H. Sampaio, T. Abrão, and M. L. Proença, "Network anomaly detection system using genetic algorithm and fuzzy logic," *Expert Systems with Applications*, vol. 92, pp. 390–402, 2018.
- [34] S. X. Wu and W. Banzhaf, "The use of computational intelligence in intrusion detection systems: a review," *Applied Soft Computing*, vol. 10, no. 1, pp. 1–35, 2010.
- [35] W. T. L. Teacy, J. Patel, N. R. Jennings, and M. Luck, "Coping with inaccurate reputation sources: experimental analysis of a probabilistic trust model," in *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems*, Utrecht, Netherlands, July 2005.
- [36] A. Jøsang, G. Elizabeth, and M. Kinatder, "Simplification and analysis of transitive trust networks," *Web Intelligence and Agent Systems*, *International Journal*, vol. 4, no. 2, pp. 139–161, 2006.
- [37] R. Czabanski, M. Jezewski, and J. Leski, *Introduction to Fuzzy Systems, Theory and Applications of Ordered Fuzzy Numbers*, pp. 23–43, Springer Open, Switzerland, 2017.
- [38] M. Goswami, P. Sharma, and A. Bhargava, "Black hole attack detection in MANETs using trust based technique," *International Journal of Innovative Technology and Exploring Engineering*, vol. 9, no. 4, pp. 1446–1451, 2020.
- [39] S. Pandey and V. Singh, "Blackhole attack detection using machine learning approach on MANET," in *Proceedings of the International Conference on Electronics and Sustainable Communication Systems*.
- [40] H. Moudni, M. Er-roudi, H. Mouncif, and B. E. Hadadi, "Black hole attack detection using fuzzy based intrusion detection systems in MANET," *Procedia Computer Science*, vol. 151, pp. 1176–1181, 2019.
- [41] G. Arulkumaran and R. K. Gnanamurthy, "Fuzzy trust approach for detecting black hole attack in mobile adhoc network," *Mobile Networks and Applications*, vol. 24, no. 2, pp. 386–393, 2019.
- [42] A. U. Khan, G. Abbas, Z. H. Abbas, M. Waqas, and A. K. Hassan, "Spectrum utilization efficiency in the cognitive radio enabled 5G-based IoT," *Journal of Network and Computer Applications*, vol. 164, no. 102686, pp. 1–16, 2020.