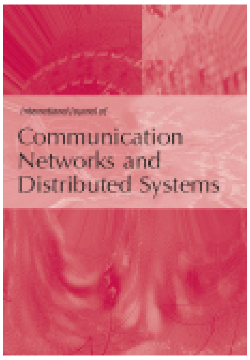



**International Journal of Communication Networks and Distributed Systems**

ISSN online: 1754-3924 - ISSN print: 1754-3916
https://www.inderscience.com/ijcnds

---

# Cluster-based multiple malicious node detection using honeypot-AODV in MANETs

H.K. Sampada, K.R. Shobha

**DOI:** 10.1504/IJCNDS.2024.10053453

**Article History:**

| | |
|---|---|
| Received: | 08 June 2022 |
| Accepted: | 17 November 2022 |
| Published online: | 30 November 2023 |

# Cluster-based multiple malicious node detection using honeypot-AODV in MANETs

## H.K. Sampada*

Department of Electronics and Communication Engineering,
Atria IT,
Bangalore, India
Email: sampada.hk@atria.edu
*Corresponding author

## K.R. Shobha

Department of Telecommunication Engineering,
MSRIT,
Bangalore, India
Email: shobha_shankar@msrit.edu

**Abstract:** Security and scalability are two major research areas in the field of mobile ad-hoc networks (MANETs). The existing solutions for security and scalability are majorly used for static networks, e.g., sensor networks. The focus of the present work is to detect and remove the multiple malicious black holes (MBH) and multiple malicious grey hole (MGH) nodes from the dynamic networks, e.g., MANETs. The proposed solution increases network security. An efficient weight-based clustering technique is used to enhance the stability and load balancing of the network. Cluster head (CH) is selected based on the maximum weight factor. The weight of the node is based on three factors: constancy factor ($C_x$) trust value ($T_y$) and link factor ($L_z$). Weightage values for the parameters can be prioritised and tested for consistency using analytic hierarchy process (AHP) algorithm. Each CH executes honeypot-AODV (H-AODV) to find the MBH and MGH nodes in its network.

**Keywords:** mobile ad hoc networks; MANETs; honeypot-AODV; H-AODV; modified-AODV; M-AODV; clustering; malicious blackhole/grayhole attack; MBH/MGH.

**Reference** to this paper should be made as follows: Sampada, H.K. and Shobha, K.R. (2024) 'Cluster-based multiple malicious node detection using honeypot-AODV in MANETs', *Int. J. Communication Networks and Distributed Systems*, Vol. 30, No. 1, pp.1–29.

**Biographical notes:** H.K. Sampada completed her MTech in Digital Communication and Networking in 2004 from SJC Institute of Technology (SJCIT), Chikballapur, under Visvesvaraya Technological University (VTU). She is currently pursuing her PhD under VTU, in the Department of ETC, MSRIT, Bengaluru. She is currently working as an Assistant Professor in the Department of ECE, Atria IT, Bengaluru. Her research interests are in security issues in mobile ad-hoc networks, VANETs, IoT and data science. She has presented her research papers in several international conferences and journals.

K.R. Shobha received her ME degree in Digital Communication Engineering from Bengaluru University, Karnataka, India, and PhD from Visveswaraya Technological University. She is currently working as an Associate Professor in the Department of Electronics and Telecommunication Engineering, MS Ramaiah Institute of Technology, Bengaluru. Her research areas include mobile ad hoc networks, IoT and cloud computing. Se has more than 25 papers publications to her credit. She is a Senior IEEE member serving as Secretary of IEEE Sensor Council, Bengaluru Section. She is also an active member of IEEE Communication Society and WiE under IEEE Bengaluru Section.

## 1 Introduction

Mobile ad hoc networks (MANETs) are a class of self-healing and highly flexible networks. They are used in different applications like disaster rescue, battlefield communications and device to device communications. Understanding the performance of these networks is necessary to implement and commercialise them. With the advent of wireless devices like mobile phones, laptops and personal digital assistance, the growth of ad hoc networks is going to be iconic in terms of their applications in almost all fields.

MANETs are part of infrastructure-less wireless networks. They are classified as fixed networks and mobility-oriented networks. Wireless sensor networks (WSNs) are an example of fixed networks that are not dynamic in nature.

MANETs are part of wireless ad hoc networks with mobility and dynamic topology. They are also described as on the fly networks, distributed and decentralised networks. The network gets established as soon as the nodes are in each other's communication range. Each node has several roles to play: individual host, intermediate host and gateway (GW). Each node in the network is a host. Intermediate nodes (INs) will help to route packets from source nodes (SNs) to destination nodes. GW node helps in routing the packets through the internet. MANETs facilitate networks to add or remove nodes. Due to mobility, the network topology is unstable and due to limited node battery, network lifetime is limited. This results in several challenges in scalability, endurance, security and performance of the network. Mobility models (Garg and Verma, 2017; Bai and Helmy, 2004; Sichitiu, 2009) predominantly impact the performance of the network. The scalability and endurance of the network can be improved based on energy-efficient routing protocols and resource management techniques. Existing energy-efficient routing protocols are reactive like dynamic source routing (DSR), ad hoc on-demand distance vector routing protocol (AODV) and hybrid protocols like zone routing protocol (ZRP). These in turn lead to overhead due to flooding during the route discovery process. Resource management techniques like clustering enhance the network stability and scalability. Security is another noticeable constraint as the nodes are dynamic and the network is decentralised.

To give mobility to the nodes there are several mobility models in MANETs.

Figure 1 shows the various mobility models in MANETs. Specifically, to analyse the performance of the protocol designed for MANETs.

Each model emulates the movement of nodes like in real-life scenarios. Thus, while emulating the underlying protocol it is important to choose a suitable mobility model.

For MANETs the preferred mobility models are random as MANETs move with high random mobility in a random direction with random velocity.

**Figure 1** MANET mobility models

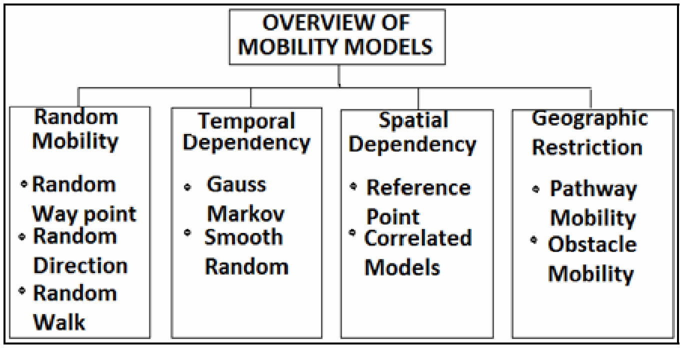
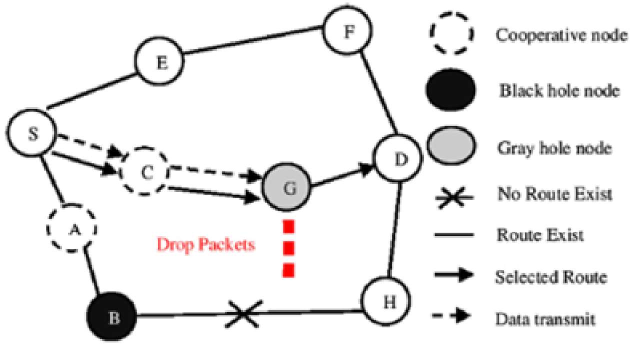



**Figure 2** Multiple malicious blackhole/greyhole attack (see online version for colours)




Other types of mobility models are temporal dependency type: node movement depends on the movement history of the nodes, spatial dependency type: nodes move in correlation with each other and geographic restriction-based: nodes move within a restricted geographical area like in streets, pathways and are mainly used in vehicular ad hoc networks (VANETs)

To improve resource management in MANETs, clustering (Sood and Kanwar, 2014; Agarwal and Motwani, 2009) is a promising technique that enhances the stability and scalability of the network. It helps in resource management by distributing the load evenly among all nodes in the network. Clustering techniques can be classified as mobility-based, topology-based and energy-based methods. Clustering (Rahman et al., 2020) has two phases, the cluster formation and cluster maintenance phase. In the cluster formation phase, the nodes that are in each other's communication range from clusters and one of them is elected as a cluster head (CH), based on maximum battery, stability and number of neighbours. In the cluster maintenance phase, the health of the cluster is preserved.

Nodes in the network use routing protocols to communicate. Routing protocols are classified as reactive, proactive and hybrid routing protocols. Proactive protocols know the entire network topology by doing the regular route exchange messages between the neighbours. Reactive protocols discover the routes only on demand, when a source wants to communicate with the destination. The most commonly used reactive routing protocol is AODV. Hybrid protocols are a mixture of both. The routing protocols communicate between each other assuming all nodes in the network are trusted nodes. This assumption makes these routing protocols highly vulnerable to attackers who disrupt the entire communication process and QoS of the network. Like any wireless network, MANETs Nadeem and Howarth (2013) are highly vulnerable to various attacks and threats. Vulnerability is more in MANETs as there is no centralised control over the network. They are exposed to both active and passive attacks. Active attacks disrupt working of the network and degrade the performance of the network, they are easy to detect and difficult to prevent. Whereas passive attacks do not harm the system but, may reveal important information like source address, destination addresses message length, etc. Passive attacks are simple to prevent but very tough to detect. The most common active attacks in MANETs are blackhole (BH) attacks and greyhole (GH) attacks. BH nodes simply drop the packets sent from the SNs or INs, which increase the number of retransmissions in the network posing a serious threat to the networks quality of service (QoS). GH attacks drop only selected packets from the IN or the SNs and hence are difficult to detect. These malicious attackers can act in single or in multiple coordinations. There are several ways of detecting the BH nodes namely baiting (Devasthali and Kadam, 2017), neighbour trust-based, and acknowledgement-based, encryption-based, sequence number based and so on.

In our approach, in order to detect and avoid black hole and GH nodes, modifications are done in the AODV routing protocol. The new modified AODV (M-AODV) is called honeypot-AODV (H-AODV). In H-AODV, a trap is laid by the CH node to lure the attackers and remove them from the network. Figure 2 shows the scenario of multiple malicious blackhole/greyhole attack (MBH/MGH) (Nadeem and Howarth, 2013; Sharma and Bisen, 2016; Gurung and Chauhan, 2019). Node *S* is source, node *D* is destination, *B* is the BH node, *G* is the GH node and all other nodes are INs. SN initiates route request (RREQ) through its neighbours. Neighbour nodes forward RREQ as they have no route to destination mentioned. The BH is one of the IN in the network. The main objective of the BH node is to drop all packets that arrived. Many such nodes are called multiple malicious BH nodes. If they are in coordination with each other, they are called cooperative blackhole (CBH) nodes. GH nodes function similarly, with the difference that they drop selected packets.

The following are the main contributions of the work:

- A novel weight-based clustering algorithm is implemented. Three parameters are used for weight calculation, i.e., constancy factor, trust value, and distance-based link factor. These parameters ensure network stability and security.
- The mathematical model of the analytic hierarchy process (AHP) is proposed and implemented. The proposed model is a decision-making algorithm as there are multiple parameters involved for CH selection.
- H-AODV is developed to detect attackers locally and then the ALARM packet removes the malicious nodes from the entire network.
- Comparison analysis of AODV, M-AODV, and H-AODV is done in the present work. Results reveal that H-AODV performs better in terms of throughput, PDR, packet drop, routing overhead, end to end delay.

- The performance of H-AODV is also analysed for different mobility models e.g., random waypoint, random direction, and random walk mobility models. However, results show that the H-AODV performs better for the random waypoint mobility model.

The paper is divided into five subsections. Section 2 focuses on related work and describes the various clustering techniques, their advantages and disadvantages, various energy-efficient clustering techniques and various cluster-based coordinated attack detection and prevention techniques. Section 3 discusses the proposed work and describes the various parameters included for the CH election and their calculation. Followed by the AHP mathematical model to calculate the weightage values for deriving weight equation for CH selection. Further, in the same section the cluster formation, CH election, cluster maintenance and multiple malicious BH/GH node detection is explained in detail. Section 4 is about simulation parameters, results and discussion. Section 5 is conclusion with future work.

## 2 Literature review

Many research scholars have worked to mitigate the BH and GH attacks (Gaber and Azer, 2022) in MANETs. Most work (Aluvala and Rajasekhar, 2022; Ravi et al., 2022; Kumar, 2022; Shukla and Joshi, 2022; Annepu and Jayaprasad, 2022; Kathole and Chaudhari, 2022; Yadav et al., 2021; Ali, 2020) is on single BH/GH attack detection which fails if the attackers are in cooperation with each other. Solutions proposed for cooperative attacks are for static environments with assumptions that the SN and the DN are clean nodes which may not be true in all cases. Resources used and time consumed for detection is also high which degrades the QoS of the network. MANET node capabilities are different from other ad hoc networks like VANET (Al Dener and Orman, 2022), WSNs and mesh networks in terms of battery and processing capabilities. Hence the researchers must implement simple yet efficient algorithms for attack detection in dynamic MANET environments.

Simple and efficient BH/GH detection techniques (Bharti et al., 2022; Chawhan et al., 2022; Abood et al., 2020) for MANETs are threshold-based (Yamini et al., 2022), anomaly detection-based techniques (Legashev and Grishina, 2022) and cluster-based methods, as the detection computation and control overhead (Nadeem and Howarth, 2013) are less in such techniques. Sequence number-based approach (Raj and Swadas, 2008) explains the use of sequence number of the reply packets detects the BH and GH nodes. The BH node usually puts the highest and random sequence number for the packet so that the packet looks like the latest packet. The sequence number is compared with the threshold sequence number possible in the network, to detect BH attacks. If the sequence number exceeds threshold value this behaviour is considered malicious and an ALARM is sent to the whole network. This adds additional overhead to the network as the sequence number and threshold value are updated regularly. As an improvement of the above paper (Shrestha et al., 2020) proposes arbitrary maximum sequence number approach. When a SN node receives a sequence number more than the DN sequence number (more than the assumed arbitrary sequence number), it resends the RREQ with the new but same sequence number. For this if the RREQ received has a high sequence number again it will discard the packet. The SN node will discard the entire path traced during the process.

Saha et al. (2012) proposed an energy efficient administrator-based secure routing in MANET, here the network is separated into four types of nodes, common nodes, associate nodes, administrator nodes and watchdog nodes. The common nodes are just general nodes (nodes with no additional function), associate nodes are the neighbour nodes (participate as the INs), administrator nodes do the routing and watchdog nodes will watch the associate nodes malicious activity and find out the total packets transmitted to total packets forwarded. If the ratio is beneath the threshold, the nodes are detected as BH nodes. Once the malicious nodes are listed the admin nodes disclose these nodes to the entire network. This method, cannot detect false positives and false negatives. In anomaly-based detection techniques, malicious behaviour by any node is detected by either SN, destination node or IDS node. In Huang et al. (2003), destination node detects the anomaly detection. A new concept called the cross-feature analysis helps to get the correlation pattern in traffic. This approach has a training and testing phase. The training phase is fed with an extensive dataset available. In the testing phase, actual logs are used. But the problem with the approach is QoS parameters are not very promising as the method is complicated.

Sharma and Sheetlani (2020) long yet effective GH/BH node removal process using neighbour node detection, followed by local, global detection and removal process. Initiator node (IN) maintains a routing table (RT) having FROM, TO, RTS/CTS ratio, check bit entry. IN node will send a RREQ using AODV and keeps a check on its RT. In this table a ‘1’ indicates a YES and a ‘0’ indicates a NO. The node with more ‘0’s’ and high RTS/CTS ratio is a malicious node. Then the local and global detection is done using cooperative nodes and the further probe packets to find the nodes over the entire network. A global alarm packet helps in removing the rouge nodes from the network. Though the process is long the QoS parameters like PDR are improved.

Terai et al. (2020) target the cooperative BH/GH (CBH/CGH) attack called smart attack. In this BH nodes will predict the threshold sequence number and avoid from the detection process. To overcome these least square methods is used, in order to collect the sequence number and the time stamp and detect the BH nodes. Attack success rate is less but effective method to remove smart attackers. Khalaf et al. (2020) the dual-cooperative bait detection scheme (D-CBDS) helps in detecting CBH/CGH attackers. Has proactive baiting, where one of the nodes will launch the baiting process to find the attackers. The selected node can be an attacker. So, to detect such cases reactive baiting is used. Two neighbouring nodes are used for bait detection. This is implemented for various routing algorithms and the performance is found satisfactory.

A novel dynamic source routing algorithm (Chang et al., 2015) for GH/BH attack detection is used. It is a mixture of both proactive and reactive routing algorithms. But the algorithm has issues in solving the cooperative attacks. This is resolved using a cooperative bait detection algorithm that reverses traces of the route reply from the destination, to detect the anomalous activity. The scheme exceeds the performance of the dynamic source routing algorithm and other benchmark algorithms like best-effort routing algorithms, in terms of overhead and PDR. These techniques are best suited for stable environments like WSNs and can address single BH/GH node detection in the network. Further, these techniques believe that the SN and the destination nodes are always secure nodes in the network. This issue is solved by clustering techniques. In clustering, priority for security is not given to any node in the network, each node is normal node with all security capabilities.

Clustering provides a way of efficient resource allocation and ensures stability by providing a hierarchical routing environment in MANETs (Ullah, 2020). Clustering divides the entire network into small groups. Each cluster has a CH, cluster members and a GW for inter-cluster communication. Clustering (Yadav and Mishra, 2022; Susan et al., 2023) enhances security in the network thus ensuring better QoS in the network. Lightweight cooperative black hole detection is a cluster-based method (Saurabh et al., 2017) in which every member of the cluster pings the number of packets received/dropped to the CH node once. This is to ensure any abnormal behaviour in the network. The CH informs the entire network if any fault is detected. This improves the PDR, delay and energy efficiency. Another approach by Raman et al. (2018), is to build a secure MANETs against cooperative BH attacks using cluster-based routing. In this approach, cluster formation is done using a weight-based clustering algorithm with the highest weighted node as the CH node. Node weights are calculated based on the residual battery of the node. Every node compares its weight with the weight of the neighbouring node within two hops, for CH selection. The largest weighted node declares itself as the CH and announces others to join its network. If a node does not announce itself as the CH within the specified time it is detected as the BH node. If the members do not join the cluster within the specified time, they are listed as BH nodes. In the route maintenance stage, each node exchanges the link-state information to maintain the cluster health. If a node moves from the cluster for more than two hops then, the node is removed from that cluster. Local topology changes are regularly updated to the CH. The simulation results for this approach show good performance in terms of PDR and reduced overhead for cooperative black hole (CBH) nodes detection. One of the effective mechanisms (Gaikwad and Ragha, 2015) to detect malicious coordinated BH attacks is 'further route request' (FRR) which can handle a single black hole attack in MANETs. Using FRR the IN confirms route sent from the destination node. FRR are the control messages sent by the INs to confirm the node authenticity. But this enhances the routing overhead of the overall network and works only for a single BH attack. To overcome overhead, another mechanism to detect the CBH attack is by means of cooperative security agents. A data routing information table and the 'from', 'to', 'thro' table is sent as the input to the security agent. The agents cross-check the information from the INs. Further, the malicious node notification is sent to all other nodes in the network. Results show an overall improvement in the QoS parameters of the network. Routing overhead in the network increases due to additional control packets added and this, in turn, reduces the energy efficiency of the network. An energy-efficient detection technique with clustering is 'REAct' (Kozma and Lazos, 2009). This protocol has three types of nodes deployed namely SN, destination node and audit node. Audit nodes are selected by the CH node. Only if the destination node detects the decrease in the PDR and notifies to its CH node the detection process is triggered. Due to this, the network overhead reduces drastically. Meanwhile, the SN starts the audit process. The destination node submits its audit report to audit node and SN to validate this report. The audit node and SNs compare the reports and create a list of the blacklisted nodes and notify others in the network. A more automated approach is a decision-making model that is the hidden Markov model (Kalkha et al., 2019) for detecting the shortest path between SN and the DN. It has detection model called the Viterbi model to detect malicious behaviour. The system also has an attack model where the cooperative black hole nodes are deployed and a detection model based on the hop count and the sequence number, which can detect the cooperative black hole nodes. Simulation results show a satisfactory output in terms of network QoS parameters. Under hierarchical clustering, the network is divided into virtual clusters and routing clusters. This new approach (Mohandas et al., 2019) has security-based and weight-based cryptographic algorithms for security in virtual clusters. Routing clusters take care of the energy levels, mobility and lifetime of the network. The dual cluster model provides secure communication with optimal power consumption. A novel cluster-based (Jamaesha and Bhavani, 2019) swarm algorithm can predict the next hop location of the node. Based on the swarm information the new location of the node is predicted to know the connectivity of the link. The neighbour trust value is used to predict the malicious behaviour of the nodes. The overall network parameters like the throughput, control overhead is also improved. The data transmission is secure as the elliptic curve cryptographic methods are used. The network energy efficiency is affected due to the heavy cryptographic techniques used. Fuzzy interference (Farahani, 2021) for CH selection and K-nearest neighbour for clustering is used. Josang logic is used by the server node to calculate the trust of a node, thus notify the network about the BH nodes. Simulations show improvement in PDR, delay and routing overhead compared to the other recent detection methods. Considering the advantages of clustering techniques, a novel dynamic cluster-based multiple malicious BH/GH detection algorithms is used in our work.

## 3 Proposed work

The proposed work uses a novel scheme to mitigate the multiple MBH and MGH attacks in a dynamic environment of mobile nodes. Most of the research work done till now try to mitigate the coordinated attacks in static environments. But dynamic environments are closer to the practical approach. The scheme is described as follows First and foremost a new algorithm, weight-based clustering is implemented to efficiently split the network to clusters. The CH nodes are elected based on the weight of the node. A node with the highest weight among the neighbours is elected as CH node. Node weight is calculated using three parameters, first is the constancy factor, i.e., the relative stability of the node in the network to assure network lifetime. The second parameter is the trust value, the packet dropping behaviour of a node is calculated so that the BH or GH node is not elected as the CH node. The third parameter is the distance-based link factor which helps in deciding the link lifetime between the CH node and member nodes of cluster, to guarantee the connectivity between the CH node and the members in each cluster. Second, since there are three parameters involved in weight calculation, giving the right weightage to each parameter is important. To achieve this AHP algorithm is implemented to do the weightage calculation. The CH initiates the BH detection algorithm H-AODV in each cluster. MBH/MGH nodes are removed from each cluster by their respective CHs using the ALARM packet. Thus alarm the network regarding their presence and remove them from the network by eliminating the detected nodes from the routing table of individual nodes.

## 3.1 CH selection

We have implemented new CH selection algorithm using the AHP method. The algorithm helps in calculating the weightage values for each of the parameters used for the CH selection. The weight of individual node is calculated and the node with highest weight is selected as CH node. The parameters used for CH selection in our scheme are constancy factor ($C_x$), trust value ($T_y$), distance-based link factor ($L_z$). In the following section, we describe each of these parameters in detail.

The constancy factor ($C_x$) is the first parameter. It is evaluated based on the movement of a node to its neighbours. It indicates whether a node is moving slow, fast or very fast relative to its neighbours. The next parameter is the trust value. As MANETs are highly vulnerable to attacks, the presence of a malicious node can be detected by measuring the trust value of a node. Trust value is calculated depending on the neighbour behaviour. If the neighbour is dropping more packets trust is less else trust is more. The last parameter is the distance-based link factor which indicates which node is surrounded by more neighbours with a minimum distance. Based on the above parameters a node with high stability, trust and more neighbours are selected as the CH node.

### 3.1.1 Constancy factor ($C_x$)

This is a very important factor because if a node that is moving fast and less stable is chosen as the CH then the communication between the CH and its members does not continue for a long time. Hence a node that moves relatively slow and relatively more stable is chosen as a CH node. The constancy value is calculated based on the similarity theory.

**Figure 3** Change in node '*x*' mobility

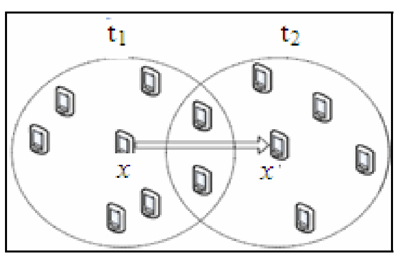
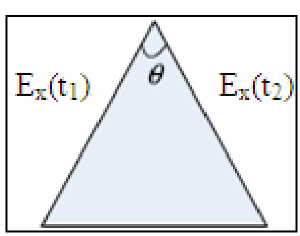



**Figure 4** Similarity graph between two vectors (see online version for colours)



The similarity theory helps in calculating the constancy factor of a node. Figure 3 shows the similarity theory of node $x$. When a node $x$ moves from time $t_1$ to time $t_2$ we can see that the neighbours change. If the node $x$ at time $t_1$ is represented by vector $E_x(t_1)$ and at $t_2$ as $E_x(t_2)$, then the average similarity between $E_x(t_1)$ and $E_x(t_2)$ is

$$C_x = \frac{1}{n-1}\left\{\sum_{j=1}^{n-1} \cos \Theta j\right\} = \frac{1}{n-1}\sum_{j=1}^{n-1}\left[\frac{E_x(t_j), E_x(t_{j+1})}{E_x(t_j)\big|E_x(t_{j+1})}\right] \tag{1}$$

here $\Theta_j$ as shown in Figure 4 is the angle between $E_x(t_1)$ and $E_x(t_2)$. If $\Theta_j$ is smaller, the stability will be more and if $\Theta_j$ is larger, the stability will be less between $t_j$ and $t_{j+1}$. If the similarity is more, the neighbour change is less and if the similarity is less the change in the neighbours is more, i.e., the node is moving fast. Hence, $C_x$ represents the constancy of node x which is calculated using equation (1).

MANET has a dynamic topology hence the algorithm must be able to support this change in topology. On the contrary, the CH node should experience a small topology change and a node that moves slowly is chosen as the CH node so that the cluster stability is maintained and less frequent cluster changes are seen.

### 3.1.2 Trust value ($T_y$)

Computation of trust value is very important in detecting malicious activity in any network. BH nodes main misbehaviour is dropping of the packets sent from the neighbouring nodes. The node trust is calculated based on communication details of the node in the past. The value indicates how many packets are forwarded by total number of packets received by the node. The value of $T_y$ from equation (2) should be more as the total packets forwarded by the node should be more.

$$T_y = \frac{No.\ of\ \ packets\ \ forwarded\ by\ node\ y}{No.\ of\ \ packets\ to\ be\ \ forwarded\ by\ node\ y} \tag{2}$$

$$T_y = \frac{N_y^{act}}{N_y} = \frac{N_y^{out} - N_y^{src}}{N_y^{in} - N_y^{dest}} \tag{3}$$

where

$N_y^{act} = No.\ of\ \ packets\ actually\ \ forwarded\ by\ node\ y$

$N_y = No.\ of\ \ packets\ to\ be\ \ forwarded\ by\ node\ y$

$N_y^{out} = No.\ of\ \ packets\ that\ are\ output\ \ from\ node\ y$

$N_y^{src} = No.of\ \ packets\ with\ node\ y\ as\ the\ source\ node$

$N_y^{in} = No.\ of\ \ packets\ that\ \ go\ into\ node\ y$

$N_y^{dest} = No.\ of\ \ packets\ with\ node\ y\ as\ the\ destination\ node$

Equation (3) gives the ability of a node to forward packets. A node can be a source, destination node or IN.

### *3.1.3 Distance-based link factor ($L_z$)*

The link between the neighbours is evaluated based on the distance of a node from other nodes. Node $x$ is said to be the neighbour of node $y$ if node $x$ is inside the communication range of $y$. While choosing the CH a node with a large neighbours and lesser distance from neighbours is chosen. If CH is chosen with only one condition that is with large neighbours and distance is not taken into consideration, there may be connectivity issues and the cluster may not very stable. The third important parameter for choosing CH is the distance-based connectivity between the nodes represented by equation (4).

$$L_z = \frac{N}{D} \tag{4}$$

where $N$ is total number of neighbours in the neighbourhood represented by equation (5)

$$N = \sum_{x=1, y \neq x}^{n} \{dist(x, y) < R\} \tag{5}$$

where $dist(x, y)$ is distance between node $x$ and node $y$.

$R$ is the maximum transmission range of any node.

$D$ = distance between the node and its neighbours is calculated using equation (6)

$$D = \sum\nolimits_{x=1, y \neq x}^{n} \sqrt{\left(a_x - a_y\right)^2 + \left(b_x - b_y\right)^2} \tag{6}$$

where $a_x$, $a_y$ and $b_x$, $b_y$ are coordinates of node $x$ and $y$ respectively.

### *3.1.4 Weight calculation*

Once the network is deployed, each node calculates its weight. To communicate with each other, each node first finds its neighbour node by exchanging the ‘hello’ packets. These ‘hello’ packets contain the details of a node like the NODE_ID (node identity), ID_MEMBER (if the node is a member, if a node is CH, then ID_CH), NODE_SPEED (speed of node), NODE_WEIGHT (weight of the node) and NODE_STATE (clustered or un-clustered state).

Each node enters all the above details in its routing table, compares the weight of each neighbouring node with its weight, and the node with highest weight is elected as the CH node. The weight of each node is calculated for the CH selection using equation (7).

$$W = \{\propto C_x + \beta T_y + \gamma L_z\} \tag{7}$$

where $\alpha$, $\beta$, $\gamma$ are the weightage values to calculate the final weight for each node.

After calculating all three parameters based on node movement, node trust and the number of neighbours the overall weight calculation is done. Since multiple parameters are used for the weight calculation, a relative weightage is calculated for each parameter. The AHP mathematical model which helps in measuring the relative weightage for the above three parameters that is constancy factor, trust value, distance-based link factor is adapted here.

### 3.1.5 AHP mathematical model

- AHP (Li et al., 2010) is a simple yet powerful tool that resolves multiple criteria decision-making problems.
- The comparisons with multiple criteria are made using an absolute scale of judgments that characterise the priority concerning for an attribute.
- The decisions may be inconsistent. How to measure inconsistency, improve the judgments and get better consistency is a concern of the AHP.

To decide CH election in a planned way the calculation of weightage value is very important, the methodology followed for calculations are demonstrated below using four steps:

Step 1 Consider weightage as $\alpha$, $\beta$, $\gamma$ for constancy factor, trust value, distance-based link factor respectively.

Step 2 Decision hierarchy structure.

**Table 1** Fundamental scale of relative importance

| *Intensity of importance* | *Definition* |
|---|---|
| 1 | Equal |
| 2 | Weak |
| 3 | Moderate |
| 4 | Extra moderate |
| 5 | Strong |
| 6 | Extra strong |
| 7 | Very strong |
| 8 | Very strong |
| 9 | Extreme |

**Figure 5** AHP Hierarchy for choosing the weightage value (see online version for colours)

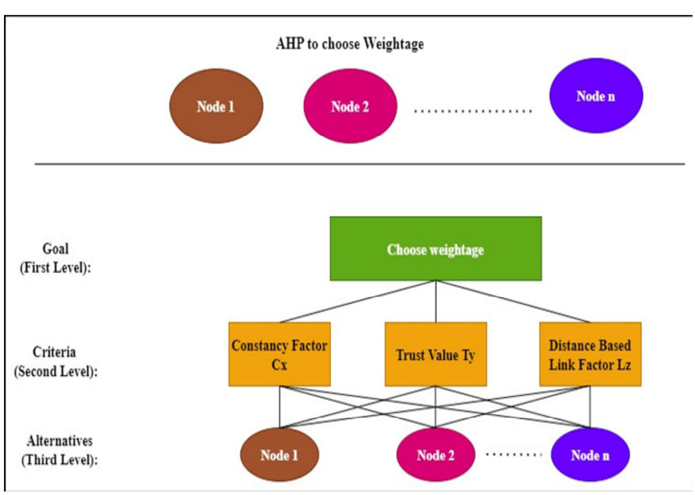


Figure 5 shows arranging a problem as hierarchy. The final goal of selecting an appropriate weightage is placed at the top of the hierarchy. The succeeding level $N$ representing main criteria are termed as a secondary goal. The three secondary goals are $C_x$, $T_y$ and $L_z$. Finally, the weight value of the second level is considered in calculating the overall weight for the selection of CH.

Step 3 Build a set of pairwise comparison matrices called the criteria matrix ($B$) as shown in equation (8). The three decision making parameters are constancy factor $C_x$, trust value $T_y$, distance-based link factor $L_z$

$$B = [b_{ij}] = \begin{bmatrix} C_x \\ T_y \\ L_z \end{bmatrix} \begin{bmatrix} C_x & T_y & L_z \end{bmatrix} \tag{8}$$

The value $b_{ij}$ denotes the level of preference of $i^{th}$ criteria on $j^{th}$ criteria. Data referred from the fundamental scale (1 to 9) in Table 1.

- Construct a reciprocal matrix of matrix $B$ using equation (9)

$$B_r = \begin{bmatrix} 1 & b_{C_xT_y} & b_{C_xL_z} \\ \dfrac{1}{b_{C_xT_y}} & 1 & b_{T_yL_z} \\ \dfrac{1}{b_{C_xL_z}} & \dfrac{1}{b_{T_yL_z}} & 1 \end{bmatrix} \tag{9}$$

  Construct a normalised vector of $B_r$, that is $B_{norm}$ using equation (10).

$$B_{norm} = \left[ \frac{1}{k} \frac{b_{ij}}{\sum_{i=1}^{k} b_{ij}} \right] \tag{10}$$

  where $k$ is the number of criteria, $k = 3$.

- Calculate the weightage of each parameter using equation (11).

$$W_i^T = [w_j] = \left[ \frac{1}{k} \sum_{j=1}^{k} \frac{1}{\sum_{i=1}^{k} b_{ij}} \right] = [\propto \quad \beta \quad \gamma] \tag{11}$$

Step 4 Consistency check

A consistency check is used to confirm the judgment errors if any in the weightage calculation. Initially, the consistency index (CI) is calculated using equation (12).

$$CI = \left[ \frac{\lambda - n}{n - 1} \right] \tag{12}$$

where $n$ is the number of elements to be compared in criteria matrix $B$, $n = 3$ for this work, as three parameters are considered. $\lambda$ can be calculated using equation (13)

$$\lambda = \left[ \frac{\sum_{i=1}^{n} \mu_t}{n} \right] \tag{13}$$

where $\mu_i$ is the consistency vector calculated using equation (14).

$$\mu_i = \left[ \frac{\sum_{j=1}^{n} W_j b_{ij}}{W_i} \right] \tag{14}$$

Finally, the consistency ratio CR is the ratio of CI and random index (RI). Value of RI can be referred from the standard Table 2.

$$CR = \frac{CI}{RI} \tag{15}$$

If $CR < 0.1$ then the judgement is correct else the algorithm must be rerun for new judgment values. Table 3 shows different combinations of weight values using the AHP algorithm The best results for $\propto$, $\beta$, $\gamma$ in equation (11) are obtained for the 3rd-row values in Table 3.

**Table 2** RI table

| Number | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| RI | 0 | 0 | 0.58 | 0.90 | 1.12 | 1.24 |

**Table 3** Best results for $\propto$, $\beta$, $\gamma$

| $C_x$ | $T_y$ | $L_z$ | $\alpha$ | $\beta$ | $\gamma$ |
|---|---|---|---|---|---|
| 1 | 2 | 6 | 58 | 34 | 8 |
| 1 | 3 | 7 | 64 | 28 | 8 |
| 1 | 3 | 6 | 60 | 30 | 10 |
| 1 | 4 | 6 | 66 | 26 | 8 |
| 1 | 4 | 9 | 70 | 24 | 6 |

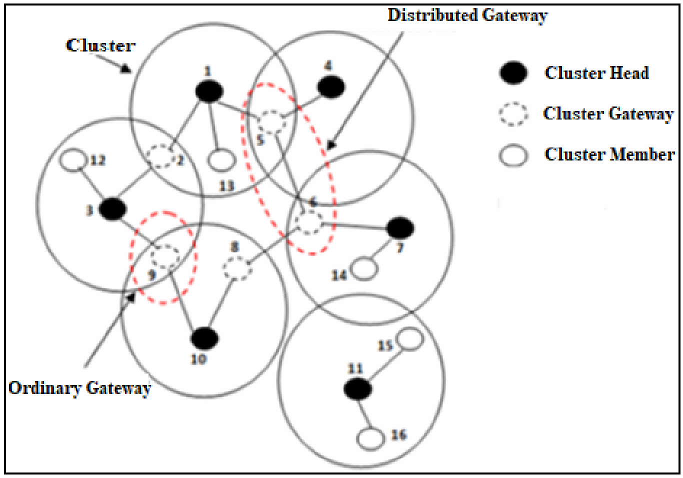
## 3.2 Clustering

Figure 6 shows the cluster architecture. The network is divided into small groups with one of the group members in each group selected as the CH based on the weight calculation. There are four types of nodes, cluster member, CH, cluster GW, distributed cluster GW (DGW). Once the CH is elected the process of clustering commences. The process of clustering has two phases, the CH formation phase and the CH maintenance phase.

### 3.2.1 Cluster formation phase

At the outset, all the nodes are in the UN_CLUSTERED STATE as shown in Algorithm 1. Every node broadcasts a 'hello' packet as self-advertisement and to know its neighbours. In the 'hello' packet various parameters like the NODE_ID, NODE_STATE, NODE_SPEED, NODE_VELOCITY and WEIGHT are communicated.

**Figure 6** Cluster architecture (see online version for colours)




Initially, ID_CH is NULL. Once the 'hello' packet is received the node stores the other nodes weight and recalculates its weight. Node with the highest weight value is chosen as the CH. Then the node changes its NODE_ID to NODE_CH.

**Algorithm 1** CH selection

**For** *all nodes i*

- *Each node broadcast HELLO*
- *HELLO includes ID, ID_MEMBER, SPEED, WEIGHT and STATE*
- *Each node initially is in UNCLUSTERED state*
- *All HELLO is added to the routing table WEIGHT comparison*
- *Node with highest WEIGHT announces itself as the ID_CH*

**End For**

After receiving the weight value from the neighbouring nodes, each node computes this in its routing table and recalculates its new weight (Algorithm 1). After comparing the weights, the node selects the largest weight-bearing node as the CH node. The state of node changes from an un-clustered state to clustered state. If the node finds its weight as the highest, it elects itself as the CH node and changes the ID_CH with its ID.

**Algorithm 2** Cluster join algorithm

**For** *all nodes i*

Only if LLT ≥ threshold between node and CH node can join a cluster

**If** (*a node receives HELLO from 2 or more CHs* (*overlapped zone*)) && (*both LLT values are* ≥ *threshold it will join both the clusters*)

*Call itself gateway node*

**Elseif** (*only one of the LLT is* ≥ *threshold it will join that cluster*) && (*none of the LLT is* ≥ *threshold it will be called an ORPHAN node*)

*And will be in UNCLUSTERED state*

**End If**

**If** (*2 nodes from different clusters can hear each other* (*non-overlapped zone*))

- *These 2 nodes will declare themselves as the scattered GW nodes*
- *Broadcast this message within their clusters*

**End if**

**End For**

---

As shown in Algorithm 2: for any node to join a cluster, link lifetime needs to be calculated between the CH node and the other nodes that want to form a cluster. Link lifetime (LLT) is the prediction of the link between the CH and the neighbour nodes. For a stable cluster, LLT should be at least equal to the predefined threshold (MIN_LLT). It is calculated as follows:

$$Link\ life\ time = \frac{|\Delta v_{ij}| \times R - \Delta v_{ij} \times \Delta d_{ij}}{(\Delta v_{ij})^2} \tag{16}$$

where

$\Delta v_{ij}$ velocity difference between nodes $i$ and $j$

$R$ maximum communication range

$\Delta d_{ij}$ distance between nodes $i$ and $j$.

As we can see in equation (16) the link lifetime depends on the velocity, distance and maximum communication range. Only if LLT between the node and the CH is exceeds the threshold (MIN_LLT) the node can join the CH and change its state to CLUSTERED state. Else its state remains as UNCLUSTERED.

If a node can receive ‘hello’ messages from more than one CH, one of the decisions listed below are taken:

1 If one of the LLT values is more than MIN_LLT, the node joins the CH with Maximum LLT and changes its state to CH_MEMBER.

2 If none of the LLT values is more than the MIN_LLT, the node does not join any of the clusters and remains in the UNCLUSTERED state.

3 If a node can hear hello from both the CH and both the LLT are more than the MIN_LLT, a node will be the member of both the clusters and will call itself a GW node and changes the state to GW_NODE. ID_CH will be the ID of both the CH nodes.

If there are no GW nodes, CH will send the GW_REQUEST message to its members. If a node other than the one belonging to this CH can hear the request, it will send the reply indicating that it can act as the Distribute GW Node (DIS_GW). GW nodes and the distributed GW nodes play prominent role in the information dissemination in network.

### 3.2.2 Cluster maintenance phase

The cluster maintenance phase has two subphases namely cluster re-elect and member-change. The details of the algorithm used for cluster maintenance is as shown in Algorithm 3.

- *In cluster re-elect:* If two CHs are in the direct communication range and can listen to each other's 'hello' messages, then the weight of each CH is calculated and compared. The CH whose weight is more and has LLT greater than the MIN_LLT (threshold) will announce itself as the NEW_CH. Once this is heard in the periodic 'hello' messages that are exchanged between the nodes the NEW_CH will attract the other CH members. If the nodes cannot meet the required condition, then they cannot join any of the clusters and will be ORPHAN nodes.

  Among the ORPHAN nodes, node with highest weight is selected as the CH and all other nodes will be joining as members of that cluster based on the LLT criteria. Thus, all the nodes in the network are made members in at least one of the clusters.

- *In member-change:* Because of the dynamic environment member change from one cluster to another will happen from time to time. LLT is used as a criterion for the members to change from one cluster to another. If a node A part of C1, comes in the overlapped area of two clusters say C1 and C2. It will receive HELLO from both the CH1 and CH2 CHs of the respective clusters. If the LLT between node A and C1, node A and C2 both exceed the LLT_MIN then node A will set itself as the GW node between C1 and C2 clusters. If node A finds that the LLT between itself and C2 is greater than LLT_MIN. It will join the new cluster.

**Algorithm 3** Cluster maintenance

Works in II phases: Cluster re-election and member handover

**If** (*2 CHs meet*; *who has the larger weight will be the new CH*)

- *The new CH will broadcast its ID*
- *Members who can meet the required LLT will join the new CH*
- *Other members will again go to ORPHAN state*

**End If**

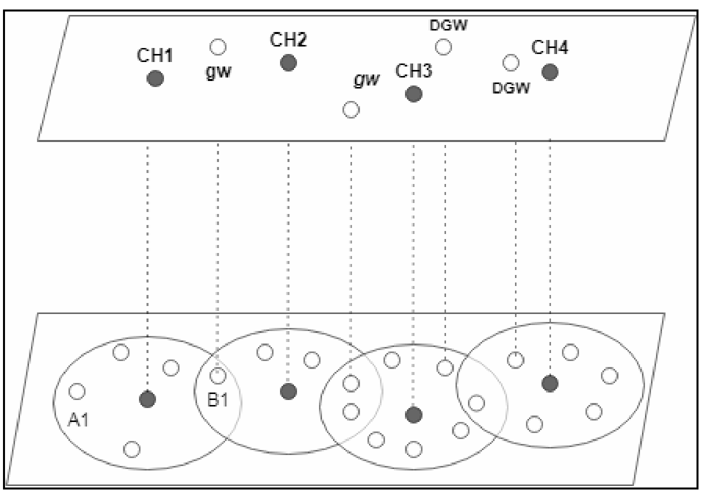
## 3.3 Honeypot-AODV

MANETs are prone to attacks that disrupt the working of the network. Two such attacks that are addressed very effectively in the paper are multiple malicious BH/GH attacks. Both are active attacks that disrupt the network performance of throughput, PDR, network lifetime. To prevent these attacks a very efficient cluster-based H-AODV with enhanced network performance is designed and executed. Honey-pot is a concept in computer security which is set to detect malicious activity in the network. They are a legitimate part of the network like the information and data that is of great value for the attackers. This is like laying a bait so that the attackers fall prey in the bait/honey-pot. H-AODV is explained in detail in Algorithm 4. Each CH will choose any random (that does not exist in the network) address as bait address or destination address and broadcast the RREQ to its entire member. Check for the RREP from the members. Members who send RREP are listed as the BH/GH nodes. This list is circulated to the entire network through the GW nodes and distributed GWs using the alarm packet. ALARM packet will have a list of malicious nodes to be removed from the network. All nodes will remove the BH/GH nodes from their routing table. H-AODV will be run after every T seconds. If the nodes do not reply then there are no malicious nodes in the network and regular AODV will be performed for route discovery and data transmission and reception.

### *3.3.1 Detection phase*

In detection phase, all CHs perform H-AODV. In the regular AODV, there is route discovery phase and route maintenance phase. In H-AODV the CH will send a route request to all its cluster members as shown in Figure 8, with the source address as the CH address and the destination address will be a honey-pot fake address to trap the malicious nodes. Each cluster member will look at the destination address and will not reply as they do not have the route to the destination address. The BH and GH nodes will reply to RREQ sent by CH node with an RREP, claiming they have the shortest route to the destination. The main difference in black hole and GH nodes is that the BH nodes will respond to all the RREQs but GH nodes will respond only to selective RREQs. Malicious nodes respond to all REEQ and this characteristic is encashed in H-AODV to trap them.

**Figure 7** MANET scenario with clustering




**Algorithm 4** Cooperative BH detection (H-AODV)

---

**For** *all CHs*

- *After every T seconds run H-AODV with DEST address as bait address*

**If** (*Reply from the other nodes*)

- *List all the nodes with RREPs as the malicious nodes*
- *Make the BH node list*
- *Send alarm packets to the entire network through GW nodes, scattered*
- *GW nodes and sink node*

- *All the nodes in the network will remove the listed nodes from their routing table*

**Else**

- *Regular AODV protocol to find the route from source to destination*
- *Data transmission and reception*

**End If**

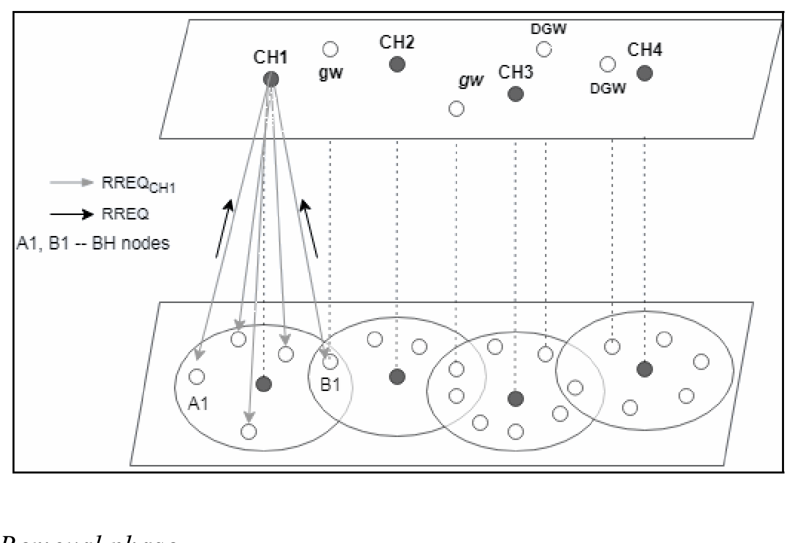
**End For**

Figure 7 shows MANETs with the clustering. Each network has nodes, SNs, destination nodes, CH, GW nodes and distributed GW nodes. H-AODV works in two phases namely the detection phase and the removal phase.

**Table 4** Simulation parameters used for the results

| | |
|---|---|
| Simulation time | 200 s |
| Simulation area | 3,000 m by 3,000 m |
| Number of nodes | 50, 100, 150 |
| Node speed | Random (0–30 m/s) |
| Packet size | 512 bytes |
| Traffic type | CBR (UDP) |
| Mobility model | RWP, RW, RD |
| Radio range | 250 m |

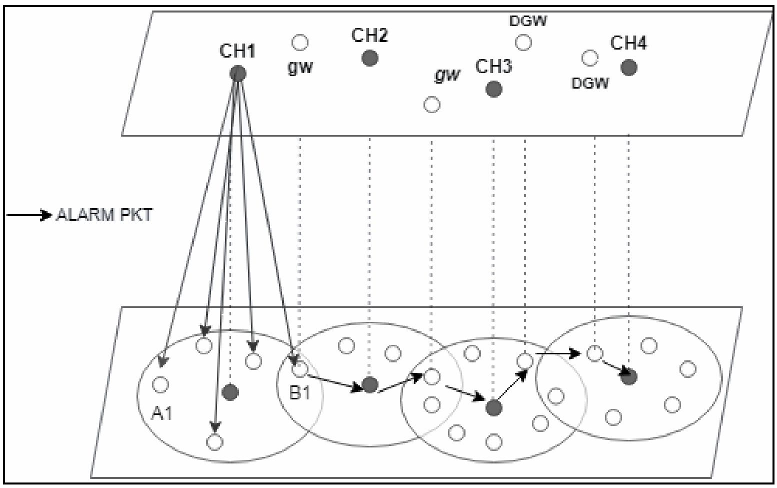
**Figure 8** MANET with MBH/MGH nodes




### 3.3.2 Removal phase

In the removal phase, after the BH and GH nodes respond to the CH nodes RREQ with an RREP, the CH nodes blacklist all such nodes as shown in Figure 9. Then the CH nodes prepare an ALARM packet to be sent to the network. The ALARM packets contain the blacklisted nodes. This packet is then circulated in the entire network through the CH nodes, GW nodes and the DGW nodes. By using H_AODV, the blacklisted nodes are sent to all the nodes through the ALARM packet. All the nodes will simply remove these malicious nodes from their routing table and that is how the BH and GH nodes are removed from the entire network. A clear indication of this is seen in the results plotted, with an enhancement in the performance of the network.

**Figure 9** BH nodes detection and prevention by CH nodes




## 4 Results and discussion

Many investigations are done on how to remove the malicious BH and GH nodes from the network. Most of existing solutions are proposed for single BH attack detection in static environments. Many solutions are restricted for a particular scenario like the RWP mobility model only and with few prior assumptions like the SN or the destination, a node can never be a malicious node. Few techniques are suggested for multiple BH/GH node detection and removal but with the increased cost and time for the detection and are not very effective in dynamic environments. Hence, we have designed a new cluster-based algorithm H-AODV for the detection and removal of both multiple MBH/MGH nodes in a dynamic environment. The various QoS parameters of the network are measured to show the efficacy of H-AODV for end-end delay, packet drop, routing overhead, PDR, throughput.

Simulation scenarios carried out in the paper are:

1 Check the performance of H-AODV with varying percentages of malicious nodes from 5% to 15%.

2 In terms of network scalability for 50 nodes, 100 nodes, and 150 nodes with M-AODV (Sampada and Shobha, 2019) and AODV.

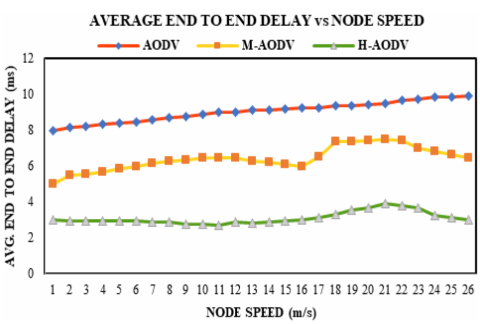
3 For different random mobility models like RWP, RW and RD.

Since the work on dynamic MANET environment is less the comparison of H-AODV is done with M-AODV which is M-AODV in a dynamic environment, along with regular AODV.

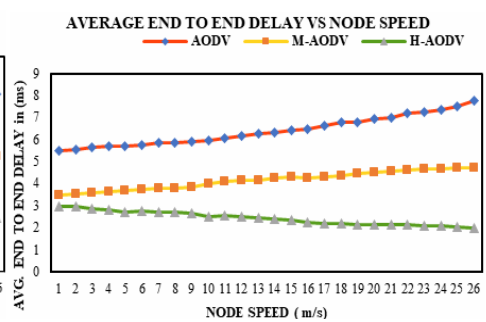
A brief explanation of M-AODV is as follows:

Modified-AODV: there are different ways in which AODV can be modified to enhance the performance of the network. One such modification is M-AODV. As we know AODV uses HELLO packets to find its neighbours. When an SN wants to find a route to DN, it will find its neighbours using HELLO received from the neighbouring nodes. Then the RREQ packet is sent to all the neighbours. The ‘HELLO’ packet used for neighbour discovery is modified by adding two parameters ‘TRUST’ and ‘willingness’. The willingness parameter indicates the residual battery in the node. Initially, all nodes will have the Willingness value of 7. As time passes due to the nodes participating in the communication and due to movement, its value starts reducing. TRUST indicates whether the node had earlier participated in any communication. Its value ranges from 0 to 1. Only nodes whose willingness is greater than 3 and trust is at least 0.5 are considered as neighbours selected for communication and other nodes are neglected even if they are in any node’s communication range. Because of these two criteria performances of the network is improved though a tolerable routing overhead is added.

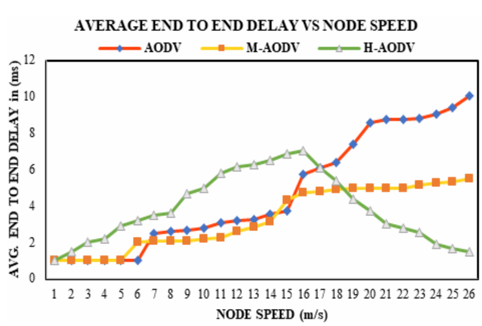
**Figure 10** (a) Average end to end delay vs. node speed (50 nodes) (b) Average end to end delay vs. node speed (100 nodes) (c) Average end to end delay vs. node speed (150 nodes) (see online version for colours)




(a)



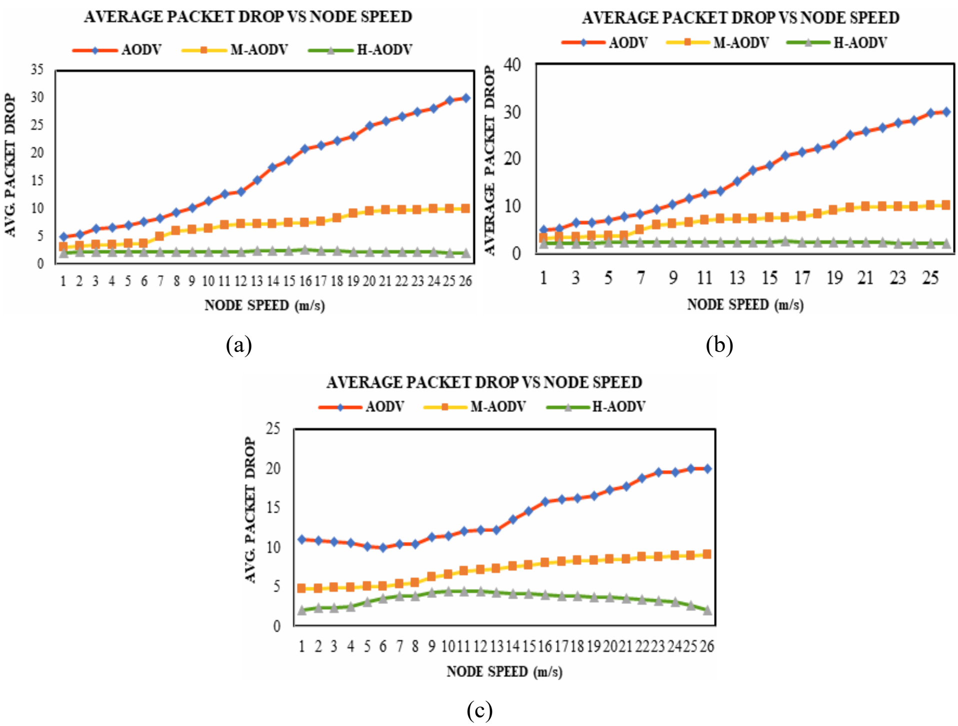

(b)




(c)

Overall results of the paper are explained in two scenarios. In the first scenario, all three algorithms (AODV, M-AODV and H-AODV) are tested with random way point mobility model.

**Figure 11** (a) Average packet drop vs. node speed (50 nodes) (b) Average packet drop vs. node speed (100 nodes) (c) Average packet drop vs. node speed (150 nodes) (see online version for colours)



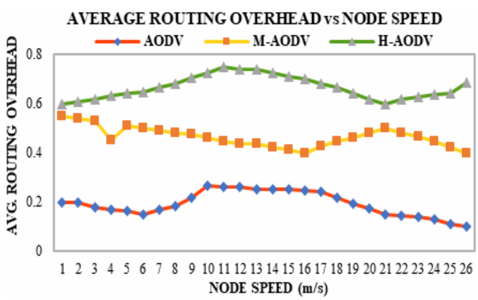

Results from Figures 10 to 14 show the various QoS parameters for reliability and stability of the network. Simulations are performed using the NS-2.35 simulator. Figure 15 show results of various mobility models.

The comparison of average end to end delay w.r.t. node speed is shown in Figure 10. Scalability of network is varied from 50, 100 and 150 nodes. Results show that the end-to-end delay is decreased when the node speed is increased.

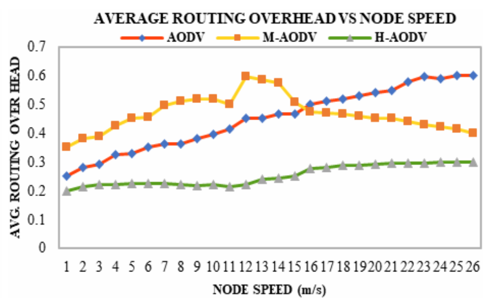
The reason is that the presence of the BH nodes is increasing the delay in the packet transmission and reception. Further there is substantial improvement of the algorithm H-AODV compared to M-AODV and AODV with BH nodes in terms of delay. Scalability is improved as H-AODV removes the BH/GH nodes through the CH nodes, by regularly using the baiting process after every T seconds. As shown in Figure 10(c), delay is almost constant till the node speeds are 20 m/s for a high scalability network which is very good in any practical scenario.

The comparison of average packet drop vs. node speed is shown in Figure 11. Results show that H-AODV performs better than AODV and M-AODV. Packet loss occurs due to packet drop by malicious nodes. The nodes are battery operated and therefore during network congestion battery of the devices gets exhausted. So, nodes with a poor battery will not be able to handle the data traffic effectively. M-AODV is a modification in AODV where willingness and trust are added in the hello packet. Willingness will tell the remaining battery in the node and trust will check the participation of the node in any previous communications, although m-AODV cannot detect the GH attack. However, it can detect the BH nodes to a certain extent. So, the performance of M-AODV is better than AODV. H-AODV is promising as it can detect and remove the BH/GH nodes from the network. As shown in Figure 11(a), small-sized network performs better in terms of packet loss as the percentages of malicious nodes are only 5%. However, for larger networks, H-AODV performs with better efficiency and consistency as shown in Figures 11(b) and 11(c).

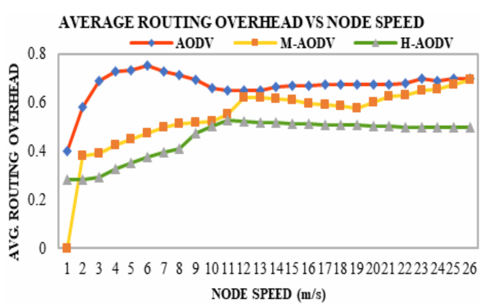
**Figure 12** (a) Average routing overhead vs. node speed (50 nodes) (b) Average routing overhead vs. node speed (100 nodes) (c) Average routing overhead vs. node speed (150 nodes) (see online version for colours)




(a)



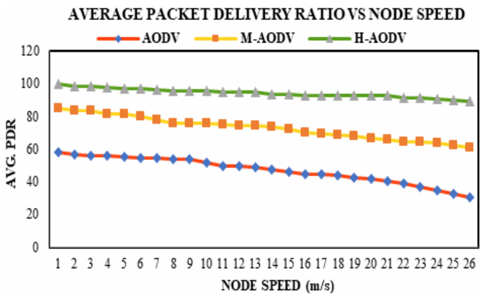

(b)



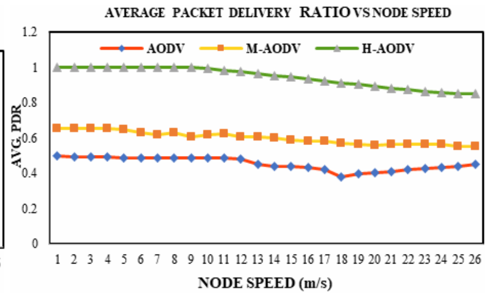

(c)

Comparison of average overhead vs. node speed is shown in Figure 12. Routing overhead is total routing packets required in the network for communication. As shown in Figures 12(a) and 12(b), for 50 nodes and 100 nodes, the overhead is less in H-AODV as compared to AODV and M-AODV. This is because no extra routing packets used in H-AODV and only changes are made is in the HELLO packet. The routing packets are used for clustering process and ALARM packets n the network. Performance is reduced with the 150 nodes because the number of clusters and the number of BH nodes are also increased. The overhead for AODV and M-AODV is more because of the higher number of retransmissions (due to the presence of malicious nodes).

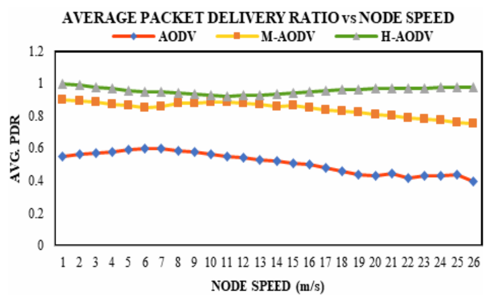
**Figure 13** (a) Average PDR vs. node speed (50 nodes) (b) Average PDR vs. node speed (100 nodes) (c) Average PDR vs. node speed (150 nodes) (see online version for colours)




(a)




(b)



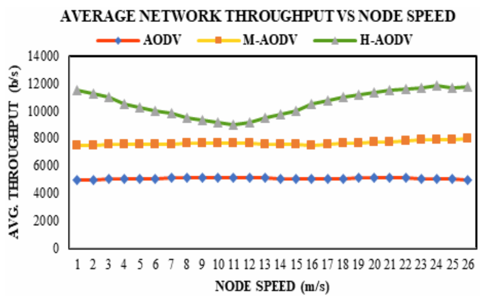

(c)

Comparison of average PDR vs. node speed is shown in Figure 13. Results show that the PDR for H-AODV is almost 100 per cent for all the node speeds. This is because the network is divided into the clusters. Results show that average packet drop in the network due to the malicious activity is reduced and the PDR is improved. Also, HODV performs better as compared to the AODV and M-AODV.

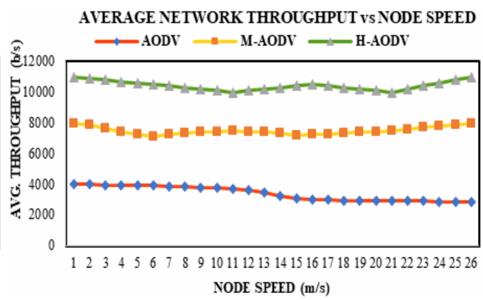
Comparison of average throughput vs. node speed is shown in Figure 14. Performance of H-AODV is improved compared to AODV and M-AODV for all the speeds as H-AODV run by all the CH nodes after every '*t*' seconds. In case of H-AODV, the malicious nodes are eliminated as soon as they enter in the network. Also, they cannot re-enter the network again as the ALARM packets are sent throughout the network. In addition, all the nodes in the network will remove the malicious nodes from their routing table. It is revealed in Figure 14(c) that the throughput of the larger networks is better for all speeds w.r.t. low and medium networks. It shows that H-AODV is highly secure and scalable.

In the second scenario, different mobility models are analysed, i.e., random way point, random walk and random direction. In the previous scenario we have already analysed different QoS parameters. Therefore, in this scenario, PDR is analysed for mobility models. The comparison of PDR vs. node speeds for various mobility models is shown in Figure 15. Simulations were carried out for all three algorithms AODV, M-AODV and H-AODV for all the three mobility models with varying percentages of BH/GH nodes in network. Performance of H-AODV is better compared to M-AODV and AODV in the case of all three mobility models. PDR vs. node speeds is shown in Figure 15 for 150 nodes, 10 BH nodes, and 5 GH nodes. From the results, it is evident that the performance of random waypoint mobility model is better than random walk and random direction. As the node speed is varied the performance is reduced but the overall performance of random waypoint is better.

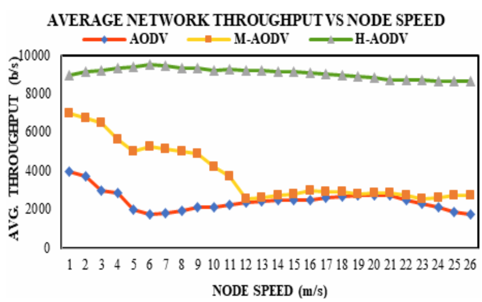
**Figure 14** (a) Average throughput vs. node speed (50 nodes) (b) Average throughput vs. node speed (100 nodes) (c) Average throughput vs. node speed (150 nodes) (see online version for colours)




(a)

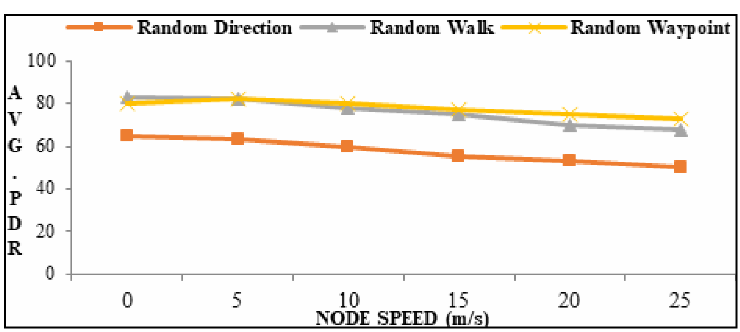



(b)




(c)

**Figure 15** Node speed vs. PDR (see online version for colours)




Hence results (Figures 10 to 15) showcase the improved performance of the H-AODV protocol as compared to M-AODV and AODV in dynamic simulation environments. The H-AODV supports network scalability, reliability (attack detection and removal with varying percentages of malicious nodes in the network). The algorithm performs satisfactorily with different mobility models that are similar to the real-world scenarios.

## 5 Conclusions and future work

In the present work, an effective cluster-based multiple MBH and GH prevention and detection algorithm are proposed. First, an efficient weight-based clustering algorithm is implemented using the AHP algorithm for weight calculation. The H-AODV carried out by each CH is an effective approach of laying the honeypot for the malicious nodes to fall into prey. The algorithm can handle the multiple malicious GHs and BH attackers efficiently. The simulations and results indicate the performance improvement of H-AODV as compared to M-AODV and AODV in terms of throughput, PDR, delay, routing overhead, and packet drop. Further, the influences of H-AODV for various mobility models are also tested. H-AODV performs well with all the three random mobility model approaches in MANETs, i.e., random waypoint, random walk, and random direction. From the obtained results, it can be concluded that the performance of random waypoint is better compared to other techniques.

The result analysis shows that the M-AODV algorithm improves the battery efficiency of the nodes and H-AODV takes care of the security of the network. Therefore, a hybrid algorithm with a combination of the two algorithms can be designed. Also, the algorithm can be made more secure by using some cross-layer approaches.

## References


Abood, M.S., Mahdi, H.F., Hamdi, M.M., Ibrahim, O.J., Mohammed, R.Q. and Ahmed, S.F. (2020) 'Black/gray holes detection tools in MANET: comparison and analysis', *2020 IEEE 7th International Conference on Engineering Technologies and Applied Sciences (ICETAS)*, pp.1–8, DOI: 10.1109/ICETAS51660.2020.9484203.

Agarwal, R. and Motwani, M. (2009) 'Survey of clustering algorithms for MANET', *International Journal on Computer Science and Engineering*, Vol. 1, No. 2, pp.98–104.

Al Dener, M.S. and Orman, A. (2022) 'STLGBM-DDS: an efficient data balanced DoS detection system for wireless sensor networks on big data environment', in *IEEE Access*, Vol. 10, pp.92931–92945, DOI: 10.1109/ACCESS.2022.3202807.

Ali, S. (2020) 'An enhanced virtual private network authenticated ad hoc on-demand distance vector routing', *Advances in Decision Sciences, Image Processing, Security and Computer Vision, Learning and Analytics in Intelligent Systems*, Springer, Cham, Vol. 3 [online] https://doi.org/10.1007/978-3-030-24322-7_25.

Aluvala, S. and Rajasekhar, K. (2022) 'Secure routing in MANETS using adaptive cuckoo search and entropy based signature authentication', *Wireless Pers. Commun.* [online] https://doi.org/10.1007/s11277-022-10008-5.

Annepu, A. and Jayaprasad, M. (2022) 'A secure data transmission using AODV and hash function for MANET', *Distributed Computing and Optimization Techniques. Lecture Notes in Electrical Engineering*, Springer, Singapore, Vol. 903, [online] https://doi.org/10.1007/978-981-19-2281-7_7.

Bai, F. and Helmy, A. (2004) *A Survey of Mobility Models* [online] http://nile.usc.edu/~helmy/important/Modified-Chapter1-5-30-04.pdf (accessed 5 January 2022).

Bharti, M., Rani, S. and Singh, P. (2022) 'Security attacks in MANET: a complete analysis', *2022 6th International Conference on Devices, Circuits and Systems (ICDCS)*, pp.384–387, DOI: 10.1109/ICDCS54290.2022.9780760.

Chang, J.M., Tsou, P.C., Woungang, I., Chao, H.C. and Lai, C.F. (2015) 'Defending against collaborative attacks by malicious nodes in MANETs: a cooperative bait detection approach', *IEEE Systems Journal*, Vol. 9, pp.65–75.

Chawhan, M.D., Karmarkar, K., Almelkar, G., Borkar, D., Kulat, K.D. and Neole, B. (2022) 'Identification and prevention of gray hole attack using IDS mechanism in MANET', *2022 10th International Conference on Emerging Trends in Engineering and Technology – Signal and Information Processing (ICETET-SIP-22)*, pp.1–6, DOI: 10.1109/ICETET-SIP-2254415.2022.9791594.

Devasthali, A.C.S. and Kadam, S. (2017) 'Cooperative bait detection scheme in MANETs', *IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI)*, Chennai, pp.679–683, DOI: 10.1109/ICPCSI.2017.8391799.

Farahani, G. (2021) 'Black hole attack detection using K-nearest neighbor algorithm and reputation calculation in mobile ad hoc networks', *Hindawi Security and Communication Networks*, Article ID 8814141, 15pp [online] https://doi.org/10.1155/2021/8814141.

Gaber, M.M. and Azer, M.A. (2022) 'Blackhole attack effect on MANETs' performance', *2022 2nd International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC)*, pp.397–401, DOI: 10.1109/MIUCC55081.2022.9781680.

Gaikwad, V.M. and Ragha, L. (2015) 'Security agents for detecting and avoiding cooperative blackhole attacks in MANET', *IEEE*.

Garg, S. and Verma, A.K. (2017) 'Simulation and comparison of AODV variants under different mobility models in MANETs', in Vishwakarma, H. and Akashe, S. (Eds.): *Computing and Network Sustainability. Lecture Notes in Networks and Systems*, Vol. 12, Springer, Singapore [online] https://doi.org/10.1007/978-981-10-3935-5_34.

Gurung, S. and Chauhan, S. (2019) 'A survey of black-hole attack mitigation techniques in MANET: merits, drawbacks, and suitability', 27 February, *Wireless Networks*, Vol. 26, pp.1981–2011, Springer Nature.

Huang, Y., Fan, W., Lee, W. and Yu, P. (2003) 'Cross-feature analysis for detecting ad-hoc routing anomalies', *IEEE Conference on Distributed Computing Systems*, pp.478–487.

Jamaesha, S.S. and Bhavani, S. (2019) 'A secure and efficient cluster-based location aware routing protocol in MANET', *Cluster Comput.*, Vol. 22, Suppl. 2, pp.4179–4186, https://doi.org/10.1007/s10586-018-1703-4.

Kalkha, H., Satori, H. and Satori, K. (2019) 'Preventing black hole attack in wireless sensor network using HMM', *Procedia Computer Science*, Vol. 148, pp.552–561, DOI: 10.1016/j.procs.2019.01.028.

Kathole, A.B. and Chaudhari, D.N. (2022) 'Securing the ad hoc network data using hybrid malicious node detection approach', *Proceedings of the International Conference on Intelligent Vision and Computing (ICIVC 2021), ICIVC 2021, Proceedings in Adaptation, Learning and Optimization*, Springer, Cham, Vol. 15 [online] https://doi.org/10.1007/978-3-030-97196-0_36.

Khalaf, O.I., Ajesh, F., Hamad, A.A., Nguyen, G.N. and Le, D-N. (2020) 'Efficient dual-cooperative bait detection scheme for collaborative attackers on mobile ad-hoc networks', in *IEEE Access*, Vol. 8, pp.227962–227969, DOI: 10.1109/ACCESS.2020.3045004.

Kozma, W. and Lazos (2009) 'Resource-efficient accountability for node misbehaviour in ad hoc networks based on random audits', *ACM Conference on Wireless Network Security*.

Kumar, S. (2022) 'Security enhancement in mobile ad-hoc network using novel data integrity based hash protection process', *Wireless Pers. Commun.*, Vol. 123, pp.1059–1083 [online] https://doi.org/10.1007/s11277-021-09170-z.

Legashev, L. and Grishina, L. (2022) 'Development of an intrusion detection system prototype in mobile ad hoc networks based on machine learning methods', *2022 International Russian Automation Conference (RusAutoCon)*, pp.171–175, DOI: 10.1109/RusAutoCon54946.2022.9896238.

Li, G., Huang, X-m. and Han, H-j. (2010) 'Characteristics-AHP model to determining attribute weights based on topics', *International Conference on Computer Application and System Modeling (ICCASM 2010)*, Taiyuan, pp.V7-489–V7-492, DOI: 10.1109/ICCASM.2010.5619045.

Mohandas, R., Krishnamoorthi, K. and Sudha, V. (2019) 'Energy sensitive cluster level security selection scheme for MANET', *Wireless Pers. Commun.*, Vol. 105, pp.973–991, https://doi.org/10.1007/s11277-019-06131-5.

Nadeem, A. and Howarth, M.P. (2013) 'A survey of MANET intrusion detection & prevention approaches for network layer attacks', in *IEEE Communications Surveys & Tutorials*, Fourth Quarter, Vol. 15, No. 4, pp.2027–2045, DOI: 10.1109/SURV.2013.030713.00201.

Rahman, T., Ullah, I., Rehman, A. and Naqvi, R. (2020) 'Clustering schemes in MANETs: performance evaluation, open challenges, and proposed solutions', *IEEE Access*, p.1, DOI: 10.1109/ACCESS.2020.2970481.

Raj, P.N. and Swadas, P.B. (2008) 'DPRAODV: a dynamic learning system against blackhole attack in AODV based MANET', *International Journal of Computer Science Issues*, Vol. 2, pp.4–59.

Raman, P.S., Shankar, K. and Ilayaraja, M. (2018) 'Securing cluster-based routing against cooperative black hole attack in mobile ad hoc network', *International Journal of Engineering & Technology*, Vol. 7, No. 1.9, pp.6–9.

Ravi, S., Matheswaran, S., Perumal, U. et al. (2022) 'Adaptive trust-based secure and optimal route selection algorithm for MANET using hybrid fuzzy optimization', *Peer-to-Peer Netw. Appl.* [online] https://doi.org/10.1007/s12083-022-01351-2.

Saha, H.N., Bhattacharyya, D. and Bhattacharyya, D. (2012) 'Energy efficient administrator based secure routing in MANET', *Advances in Computer Science, Eng. & Appl.*, AISC, Springer-Verlag, Berlin, Heidelberg.

Saidi, A., Benahmed, K. and Seddiki, N. (2020) 'Secure cluster head election algorithm and misbehavior detection approach based on trust management technique for clustered wireless sensor networks', *Adhoc Networks Journal*, 1 November, Vol. 136, p.102956, Springer.

Sampada, H.K. and Shobha, K.R. (2019) 'Performance analysis of energy-efficient MANETs-using modified AODV (M-AODV)', in Smys, S., Bestak, R., Chen, J.Z. and Kotuliak, I. (Eds.): *International Conference on Computer Networks and Communication Technologies. Lecture Notes on Data Engineering and Communications Technologies*, Vol. 15, Springer, Singapore [online] https://doi.org/10.1007/978-981-10-8681-6_9.

Saurabh, V.K., Sharma, R. and Itare, R. (2017) 'Cluster-based technique for detection and prevention of black-hole attack in MANETS', *International Conference on Electronics, Communication and Aerospace Technology (ICECA)*.

Sharma, A. and Sheetlani, J. (2020) 'Recognition and avoidance of blackhole and grayhole attacks in MANET', *International Journal of Management*, August, Vol. 11, No. 8, pp.2204–2215 [online] https://doi.org/10.34218/IJM.11.8.2020.191.

Sharma, N. and Bisen, A.S. (2016) 'Detection as well as removal of black hole and gray hole attack in MANET', *2016 International Conference on Electrical, Electronics*, pp.3736–3739, DOI: 10.1109/ICEEOT.2016.7755409.

Shrestha, S., Baidya, R., Giri, B. and Thapa, A. (2020) 'Securing blackhole attacks in MANETs using modified sequence number in AODV routing protocol', *2020 8th International Electrical Engineering Congress (iEECON)*, pp.1–4, DOI: 10.1109/iEECON48109.2020.229555.

Shukla, M. and Joshi, B.K. (2022) 'An effective scheme to mitigate blackhole attack in mobile ad hoc networks', *Edge Analytics. Lecture Notes in Electrical Engineering*, Vol. 869, Springer, Singapore [online] https://doi.org/10.1007/978-981-19-0019-8_12.

Sichitiu, M.L. (2009) 'Mobility models for ad hoc networks', in Misra, S., Woungang, I. and Misra, S.C. (Eds.): *Guide to Wireless Ad Hoc Networks. Computer Communications and Networks*, Springer, London [online] https://doi.org/10.1007/978-1-84800-328-6_10.

Sood, M. and Kanwar, S. (2014) 'Clustering in MANET and VANET: a survey', *International Conference on Circuits, Systems, Communication and Information Technology Applications (CSCITA)*, Mumbai, pp.375–380, DOI: 10.1109/CSCITA.2014.6839290.

Susan, T.S.A., Nithya, B., Agrawal, H. and Vijitendra, D. (2023) 'K-weighted cluster head selection in wireless sensor networks', *Computer Communication, Networking and IoT. Lecture Notes in Networks and Systems*, Vol. 459, Springer, Singapore [online] https://doi.org/10.1007/978-981-19-1976-3_4.

Terai, T., Yoshida, M., Ramonet, A.G. and Noguchi, T. (2020) 'Blackhole attack cooperative prevention method in MANETs', *2020 Eighth International Symposium on Computing and Networking Workshops (CANDARW)*, pp.60–66, DOI: 10.1109/CANDARW51189.2020.00024.

Ullah, Z. (2020) 'A survey on hybrid, energy efficient and distributed (HEED) based energy efficient clustering protocols for wireless sensor networks', *Wireless Pers. Commun.*, Vol. 112, pp.2685–2713, https://doi.org/10.1007/s11277-020-07170-z.

Yadav, R.K. and Mishra, R. (2022) 'Cluster-based classical routing protocols and authentication algorithms in WSN: a survey based on procedures and methods', *Wireless Pers. Commun*., Vol. 123, pp.2777–2833 [online] https://doi.org/10.1007/s11277-021-09265-7.

Yadav, S., Kumar, R., Tiwari, N. and Bajpai, A. (2021) 'An effective approach to detect and prevent collaborative grayhole attack by malicious node in MANET', *Intelligent Systems Design and Applications, ISDA 2019, Advances in Intelligent Systems and Computing*, Springer, Cham, Vol. 1181 [online] https://doi.org/10.1007/978-3-030-49342-4_31.

Yamini, K.A.P., Stephy, J., Suthendran, K. and Ravi, V. (2022) 'Improving routing disruption attack detection in MANETs using efficient trust establishment', *Special Issue: Designing and Planning of Energy Efficient Sustainable Cities and Societies: A Smart Energy*, January–12 May, Vol. 33, No. 5, p.e4446 [online] https://doi.org/10.1002/ett.4446.