Computers, Materials & Continua
DOI:10.32604/cmc.2021.015921
Article

Tech Science Press

Genetic Algorithm Routing Protocol for Mobile Ad Hoc Network

Raed Alsaqour[1], Saif Kamal[2], Maha Abdelhaq[3,*] and Yazan Al Jeroudi[4]

[1]Department of Information Technology, College of Computing and Informatics,
Saudi Electronic University, 93499, Riyadh, Saudi Arabia
[2]Department of Computer Technology Engineering, Iraq University College (IUC), Basra, Iraq
[3]Department of Information Technology, College of Computer and Information Sciences,
Princess Nourah bint Abdulrahman University, 84428, Riyadh, Saudi Arabia
[4]Department of Mechanical Engineering, International Islamic University Malaysia,
Jalan Gombak, 53100, Selangor, Malaysia
[*]Corresponding Author: Maha Abdelhaq. Email: msabdelhaq@pnu.edu.sa

Received: 05 December 2020; Accepted: 09 January 2021

Abstract: Mobile ad hoc network (MANET) is a dynamically reconfigurable wireless network with time-variable infrastructure. Given that nodes are highly mobile, MANET's topology often changes. These changes increase the difficulty in finding the routes that the packets use when they are routed. This study proposes an algorithm called genetic algorithm-based location-aided routing (GALAR) to enhance the MANET routing protocol efficiency. The GALAR algorithm maintains an adaptive update of the node location information by adding the transmitting node location information to the routing packet and selecting the transmitting node to carry the packets to their destination. The GALAR was constructed based on a genetic optimization scheme that considers all contributing factors in the delivery behavior using criterion function optimization. Simulation results showed that the GALAR algorithm can make the probability of packet delivery ratio more than 99% with less network overhead. Moreover, GALAR was compared to other algorithms in terms of different network evaluation measures. The GALAR algorithm significantly outperformed the other algorithms that were used in the study.

Keywords: Mobile ad hoc network; location-aided routing protocol; location information; genetic algorithm

1 Introduction

Mobile ad hoc network (MANET) allows users to stay connected via wireless communication known as infrastructure-less network of mobile devices. This network has a series of mobile wireless nodes that travel in every direction and position independently [1]. In MANET, nodes are arranged in different ways; they can travel in any direction and speed, enabling contact across routing protocols with one another. In MANET, several hops are required to begin

This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

data transmission between nodes when considering limited transmission distance. Despite many proposed solutions, room for improvement of routing protocols remains [2].

The most challenging task for MANET is to identify the most efficient route between nodes because of dynamic topology changes and battery drain rate. Establishing the routes in the shortest time possible is the most crucial step because it reduces the delay of data delivery and helps achieve faster convergence [3].

MANET topology's rapid and frequent changes cause the nodes' free movement with no restriction in direction or mobility (i.e., speed and pause time). The problems that arise especially for nodes running at high speed and at different directions are data routing and packet dissemination. These issues emerge because of high mobility and regular topology changes. Wireless connections between such nodes can also break or expire. Consequently, the wireless connection must be re-established. When this occurs, the network is flooded with many control packets and route error packets.

Link breakages are frequently experienced in MANET with high node mobility. For example, one of the key limitations is that a node that is part of a path loses communication with its neighbors; it loses the ability to interact with the other nodes. The disconnected node then relays its disconnection message to the source node and consequently demands a new path exploration session. This event creates additional overhead on the network, slows the packets' delivery time, and causes the broadcasting storm issue [4,5].

This work presents an analytical study that examines the location-aided routing (LAR) protocol in MANET. This well-known location-based routing method uses the information of the location of a mobile node via the global position system technique [6]. This protocol's approach decreases the overhead cost of route discovery as it utilizes location information of the mobile host. The LAR protocol uses location information and the notion of the request and expected zones. When limited flooding of LAR occurs, a request is sent by a node to another node in a request zone, given that the coverage of the request zone includes the expected zone and its surrounding areas. The demand will cover only the request zone [2].

The rest of this paper is structured as follows. Section 2 addresses the research background and related studies. Section 3 describes the suggested GALAR strategy. Section 4 assesses and analyzes the efficiency of the GALAR. Section 5 summarizes the analysis and outlines the future work.

2 Background and Related Studies

2.1 LAR Protocol

In this study, the LAR protocol, a well-known position-based protocol, is a mechanism that seeks to minimize the overhead control message of the ad-hoc on-demand distance vector routing protocol by flooding only a portion of the network that is likely to contain the destination route.

According to [7], LAR is one of the first adopted routing protocols that consider location information when routing in MANET. The procedure involves a source node S with location details from the destination node D. As shown in Fig. 1, this process gives an estimation of the expected zone, which is the expected destination region for the presence of the destination node D. Node S is aware that node D is located (Xd, Yd) with time (t_0), and the average moving speed for D is V.

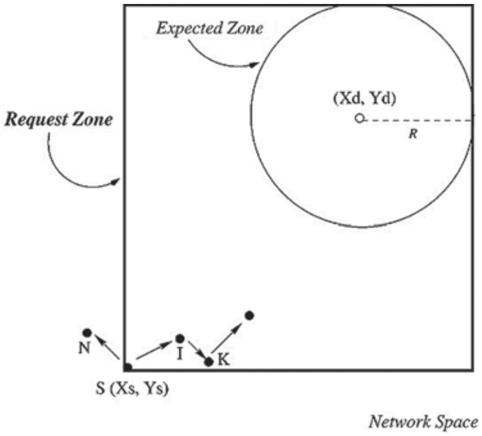


Figure 1: LAR scheme 1

In LAR, flooding happens when the forwarding packets to the nodes do not enter the destination inside the request zone. Thus, for LAR to operate, nodes must know whether they are in the request zone to either drop or begin to flood the packet. Researchers who have undertaken LAR studies suggest two separate node schemes to assess whether they are in the requested region.

The first scheme includes a sender who sends a request for a route with the rectangle coordinates in the request zone. A node that receives this request for the route may discard the request for the route if it is not within the rectangle and forward it to the endpoint if it is inside the rectangle. When the route request (RREQ) has reached its destination, it will respond with the route reply (RREP) message.

As shown in Fig. 1, when nodes I and K receive a request for a route for node D originating from node S, they can forward a request for a route given that nodes I and K are inside a rectangular request zone. Conversely, when node N receives a request for a route, it discards the request because N is beyond the rectangular request zone. However, this situation can save the routing costs. It is considered more efficient and effective than the practice of a blind search by traditional flooding algorithms for the whole network area. When the source node S fails to identify the target node D's location, the expected zone is set to the entire network region.

The second scheme does not explicitly identify the request zone when sending a request for a route. Instead, it forwards a packet depending on the distance from the destination of the sending node that is used in the RREQ. Thus, in Fig. 2, only nodes I and K forward the RREQ to their neighbors if nodes N, I, and K obtain the RREQ from node S, given that nodes I and K are closer to (Xd, Yd) than node S. As node N receives the RREQ from node S, node N discards the RREQ because node N is further from (Xd, Yd) than nodes I and K.

2.2 Overview of Genetic Algorithm (GA)

GA is a type of adaptive stochastic optimization algorithm that uses natural evolution concepts for searching and optimization [8]. It is used as an optimization method to calculate the optimal or close optimal solutions to the search's problems.

A simple and effective GA is composed of three operators, namely, reproduction, crossover, and mutation. The first process is a reproduction, which initializes the start of a population. Reproduction is the main evolutionary loop that evaluates each individual's fitness function based on the fitness score. By using one of the methods of selection, individuals from a population are chosen. The more desirable a candidate is, the better its odds of being chosen. Nature performs its function as the chosen individuals are partnered to experience the mating crossover phase to create one or more children from each pair.

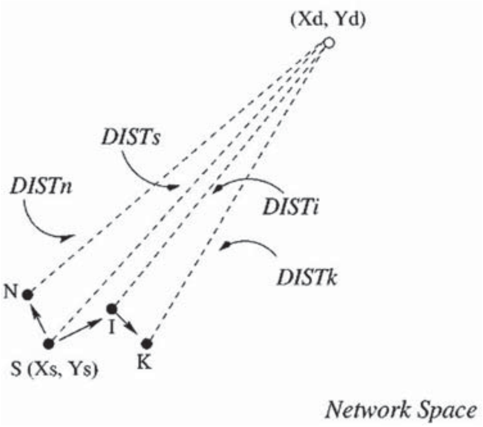


Figure 2: LAR scheme 2

The second approach is the crossover, which attempts to incorporate strategies to enable the child to inherit its parents' features. It typically consists of one parent copying half of the genotype and filling another parent with the remaining genes.

The mutation is the third phase. Each child is granted the ability to experience a mutation phase in which certain parts of their genotype spontaneously change. This stage may have the randomness factor of the method to expand the search space. As a substitute for the older population, all new children serve the new generation. The whole process is replicated before one of the stopping conditions is met.

The evolution and natural selection processes in GA are measured on the sample of candidate solutions (individuals). Each individual (offspring) in the population is a specific solution to the problem; each individual is a series of traits written in the individual's chromosome or genotype as in nature. Depending on the genotype, each individual is then evaluated on a fitness function. This function calculates the individual's score on the nature of the issue in solving a specified study problem. The problem can be either a case of minimization, where low scores are preferred or a problem of maximization, whereby high scores are preferred [9]. Algorithm 1 outlines GA's key stages.

Algorithm 1: GA [10]

```
Procedure GA
Generate the initial population of solutions randomly
Evaluate the initial population through fitness function
repeat until (A stop condition is reached) do
```

(Continued.)

```
Find a parent pair based on the fitness function
Using crossover, create two offspring
Apply mutation to the offspring
Evaluate the new offspring by the fitness function
end do
return best “parameter” values
end procedure
```

2.3 Related Studies

This section contains related works, which present diverse classes of ad hoc directing conventions and evaluations of a few areas based on a routing protocol. To generate location information, the overwhelming majority of position-based routing protocols presume a necessary process known as location service. Location services are categorized into proactive and reactive levels [11].

A mobile node that uses a proactive protocol exchanges its position data regularly with multiple mobile nodes [12]. Meanwhile, the position information is exchanged on-demand between the mobile nodes through a reactive protocol, ensuring that the location information is delivered simply upon request. The reactive location service (RLS) carries on to request the location information needed for each mobile node in the RLS [13]. The sections in area tables are cleansed in particular interims depending on the time of location information.

In [14], the authors proposed a GA-based secure and energy-aware routing (GASER) protocol for sparse MANETs. The GASER protocol selects the best route for routing packets between the source and the destination by combining GAs with other approaches, such that the selected path is the shortest. Among the other network nodes, the chosen route nodes have the most significant probability of message routing and have adequate energy to accept and then forward messages. GASER prevents the nodes from inducing network greyhole and blackhole attacks when it prefers the next hop with more likelihood of message forwarding, thereby rendering the routing protocol secure. In terms of packet delivery ratio (PDR), average residual resources, overhead ratio, and number of deceased nodes, the simulation findings revealed that GASER outperforms other protocols.

In [15], the authors suggested a GA-bacterial optimization algorithm for foraging to perform the optimum routing selection. The pathways are initialized after checking several routes to the destination node, and the GA is initiated. The location of the maximum likelihood of optimal routes, which are the initial locations of bacteria for the optimization of bacteria foraging (BFO) algorithm, is easily found by this algorithm. To compensate for the low precision of the GA, utilizing the BFO algorithm can easily determine the extreme value and the best direction. Without altering the complexity of dynamic source routing, the suggested streamlined approach strengthens the routing selection algorithm. It shows the algorithm’s integration with the desired global solution. The simulation showed that the proposed algorithm is feasible and relevant and has positive experimental results.

In [16], the authors proposed a scheme called GADA-LEACH. The scheme uses evolutionary GAs to boost cluster head selection in the legacy LEACH routing protocol in sensor networks. To facilitate coordination between the cluster head and base station, the concept of the relay node is implemented, which serves as an intermediary between the cluster head and the base station.

The findings obtained from the simulation results confirmed the proposed scheme's performance in terms of network lifespan.

The authors in [17] suggested a new LAR protocol called DALAR to enhance the efficiency of the route discovery phase. When a node sends an RREQ message, it does not unconditionally forward the message; when the message has to be retransmitted depends on the number of adjacent nodes and the location information. The simulation findings showed that the length of the constructed route in the suggested method is the same as the flooding methods currently used; however, the nodes involved in forwarding the RREQ messages are decreased by 1%–4%.

The authors in [18] proposed an algorithm to enhance the LAR protocol. Improving the use of LAR1 and LAR2 is essential to find the best routing path. The authors studied LAR1 and LAR2, ran simulations, analyzed the data, and found weaknesses in specific parameters. The authors decided to combine LAR1 and LAR2 in one algorithm, which they called LAR WHYM. They compared LAR1, LAR2, and LAR WHYM. The comparison results showed that LAR WHYM has a less end-to-end (E2E) delay than LAR1 and LAR2. Moreover, LAR WHYM is better at finding the best routing path with fewer hubs. The results also showed that LAR WHYM has a good PDR, routing overhead, throughout, and normalized routing load. The results showed that the algorithm was successful because it delivered the best performance and was proven better than LAR1 and LAR2.

In [19], the authors proposed a predictive LAR (PLAR) for mobility models in which the target of motion is known. This protocol proposes that for each node, new location services provide location information. Location information is used to predict the destination node's orientation, such as geographic coordination, current velocity, and direction of motion. Moreover, the introduction of "information lifetime" in the location service, as a major factor in determining the freshness of location data, has contributed to more precise predictions in the routing process. The simulation results revealed that the PLAR protocol overhead is less than the overhead found in LAR.

In [20], the authors considered the efficient utilization of the bandwidth in LAR (EUBLAR), which can calculate the available bandwidth of all the intermediate nodes between the source and the destination. In this proposed protocol, the authors found the minimum available bandwidth of all the intermediate nodes between the source and the destination. On the basis of the bandwidth, the authors sent the data packets over that path. EUBLAR can effectively utilize bandwidth wastage, and every single bandwidth that can be used for data transfer can be used over an entirely configured network. Thus, the quality of service (QoS) of the ad hoc network is increased in terms of bandwidth. EUBLAR helps handle one of MANET's most important performance factors (i.e., bandwidth). However, given that all the nodes were selected without considering any threshold value, it slowed down the data rate because of the increase in time to send data from the source to the destination because the bandwidth of some intermediate nodes was low. Data were sent based on the node with the minimum bandwidth.

In [21], the authors were able to map out a plan that enhances the proficiency of LAR. First, for route discovery, they selected a common baseline, which is the line between the source and the destination hub. The request packet broadcasts while considering the baseline for targeting the next broadcasting hub in the request zone. The neighboring hub with the briefest separation to the pattern is the next broadcasting hub. Moreover, they looked at the execution of the GALAR algorithm and LAR. During the simulations, they directed the control overhead, the routing lifetime, and the PDR with diverse mobility speeds. The simulation results showed that the suggested GLAR would decrease the overhead control and build up the lifespan of the route more than LAR.

In [22], the request zone was assumed to be the smallest triangle found in the source and expected zone nodes. The route increases the triangle region's angle and repeats the flooding in a given area if the first discovery of the route cannot be identified. However, the request zone becomes convenient in an area of heavy congestion due to the limited size of the request zone in this algorithm. Thus, the repetitive incremental route discovery zone increases the overhead on the nodes and the routing time for areas with fewer nodes.

In [7], by benefiting from the geographical location information to confine the flooding environment, LAR was proposed to improve the reactive execution. LAR starts with the flooding of the RREQ message to obtain the destination's geographic position. For subsequent route discoveries, this restricts the flooding of control packets to the direction in which the destination is supposed to be situated. If no destination information persists in LAR, the sender does not become an expected zone. LAR is then forced to increase the request zone to flood the entire network. In [7,17–22], different surveys were proposed to optimize LAR-1 by merely forming the point of view of reducing the request zone, where the suggested concept of the reduction of the request zone was similar in both cases.

Most approaches enhance the LAR protocol from narrowing the search region of the candidate nodes and limiting the number of candidate nodes that carry the packets. Many methods rely on the position table that contains the nodes' information location while they are in movement. This table is built by adding information about the sending node's position to the header of the packet. However, the information displayed on the information table about the node's position may not be exact because the node has moved to the current position. Thus, the earlier information of its location has expired. Therefore, any geometric-based expectation of the request and expected zones based on the single observation of the node's location (even though it is the most recent one in the packet updates) can be misleading. Given that maintaining the information displayed on the table is impossible, sending the separate packets is the updated method because this may allocate a significant part of the bandwidth.

Despite all the developments of LAR, two main limitations exist in literature approaches. First, mobility is based on the assumption that the nodes operate at a constant speed, which is invalid all the time in MANET networks. Second, either the definition of the request zone is geometrical or ignores many other contributing variables in the nodes' routing capability.

3 GALAR Algorithm

The proposed approach adds two main concepts of the LAR procedure. First, determining the destination node's current position becomes more accurate because the position information of the transmitting node is added to the packet even if the transmitting node is only a routing node. Moreover, any node that receives a packet updates either the packet location information or its location table information to the most recent one. Second, selecting the contributing variables in the routing process and their significance is performed automatically based on genetic optimization. Genetic optimization is appropriate for avoiding local minima and determining the most accurate formula for the request zone. This formula is used to select the candidate nodes from a transmitting node's coverage zone to carry a request packet to its destination for finding an optimum route for transmission between a source node and a destination node. The procedure for a developed LAR protocol consists of three sections, namely, route discovery using an RREQ packet), genetic optimization of criterion function, and GALAR algorithm operations.

3.1 Route Discovery

In GALAR, an RREQ packet includes the information provided in Fig. 3. Source node S broadcasts RREQ packets to its neighboring nodes. The nodes are selected based on a predefined threshold of a criterion function. GA determines the criterion function. Each node updates the information table and the four and five parts in the RREQ packet with the most recent location information. Moreover, the transmitting node's location information is updated in the node information table. The node decides whether they can send the packet by re-evaluating the criterion function with the updated location information. If the value is more than the provided threshold, then the node is deemed eligible and replies with an RREP packet to the transmitting node. Otherwise, the node discards the packet and checks the information of the route sequence. If the packet has been received beforehand, then the node will discard the packet without replying to the transmitting node with the RREP packet. The transmitting node counts how many RREP it has received. The packets are broadcasted with a lower value of the threshold if the number of RREP is less than the remaining RREQ. Once the destination node has received the RREQ, it will reply with the RREP.

Type	Reserved	Hop Count
Broadcast ID		
Source ID		
Destination ID		
(Source location information, timestamp)		
(Destination location information, timestamp)		
(Transmitting node location information, timestamp)		
Threshold		
Route sequence		

Figure 3: RREQ packet format

3.2 Genetic Optimization of Criterion Function

The criterion function of selecting the eligible node to carry the RREQ packet is related to six variables, as shown in Fig. 4. In the figure, *DisS* denotes the distance to the source node, *DisD* denotes the distance to the destination node, *DistBI* denotes the distance to the baseline, *DistdSDT* represents the distance between the source and the destination nodes, *NodeSpeed* denotes the node, speed and *CovRad* denotes the node coverage zone.

The six variables are used with their inverse in the criterion function. Thus, the criterion function contains 12 variables. The formula for this criterion function is

$$\begin{aligned} f(x) = {} & w_1\mathrm{DisS} + w_2\mathrm{DisD} + w_3\mathrm{DistBl} + w_4\mathrm{DistdSDT} + w_5\mathrm{NodeSpeed} + w_6\mathrm{CovRad} \\ & + \frac{w_7}{\mathrm{DisS}} + \frac{w_8}{\mathrm{DisD}} + \frac{w_9}{\mathrm{DistBl}} + \frac{w_{10}}{\mathrm{DistdSDT}} + \frac{w_{11}}{\mathrm{NodeSpeed}} + \frac{w_{12}}{\mathrm{CovRad}}. \end{aligned} \tag{1}$$

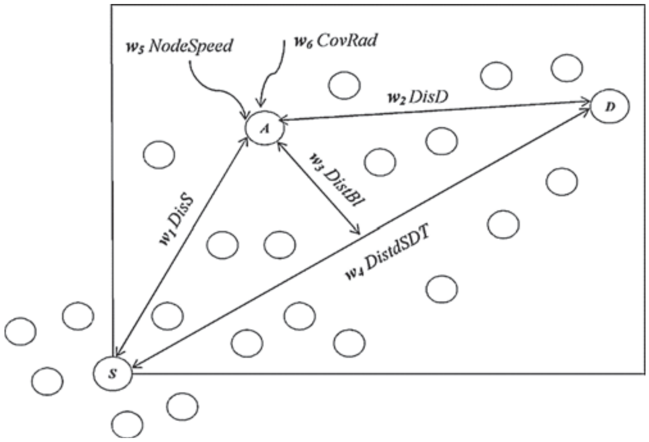


Figure 4: Criterion function variables

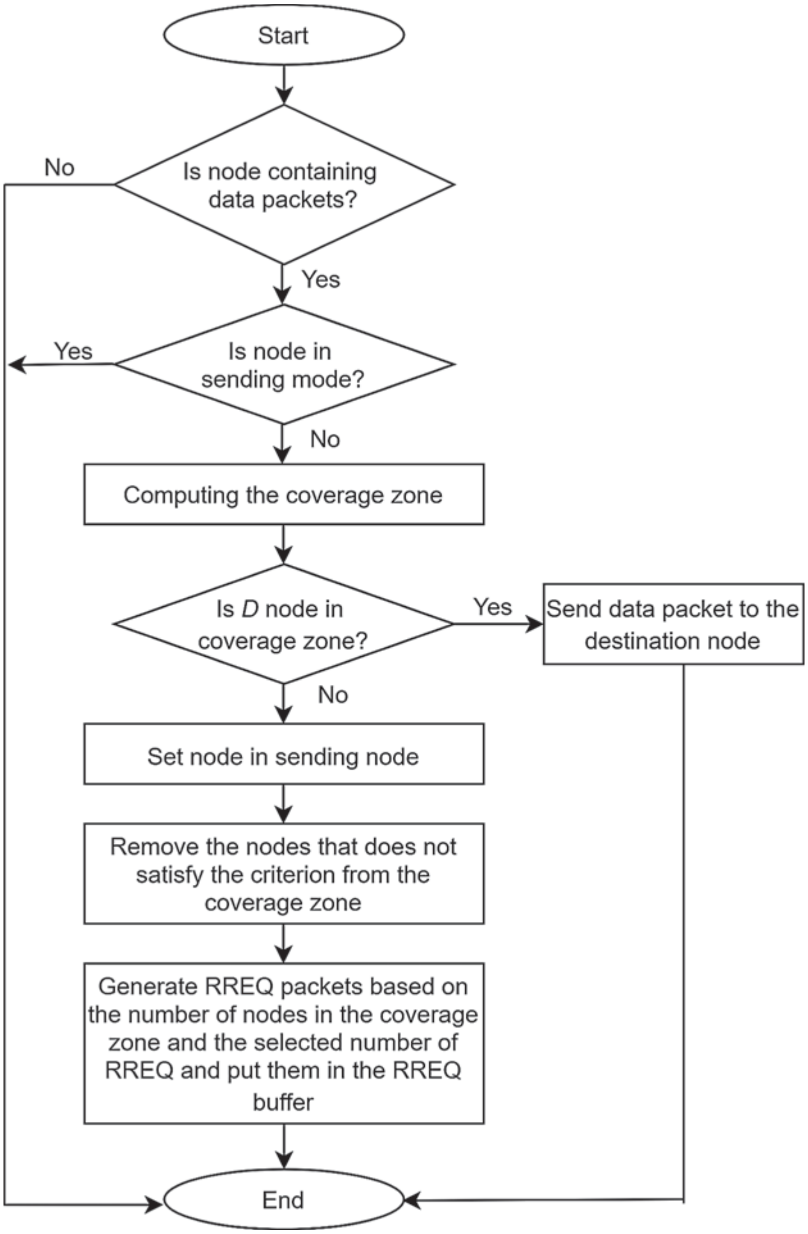

Start
Is node containing data packets?
No
Yes
Is node in sending mode?
Yes
No
Computing the coverage zone
Is *D* node in coverage zone?
Yes
Send data packet to the destination node
No
Set node in sending node
Remove the nodes that does not satisfy the criterion from the coverage zone
Generate RREQ packets based on the number of nodes in the coverage zone and the selected number of RREQ and put them in the RREQ buffer
End

Figure 5: Algorithm for sending an RREQ buffer

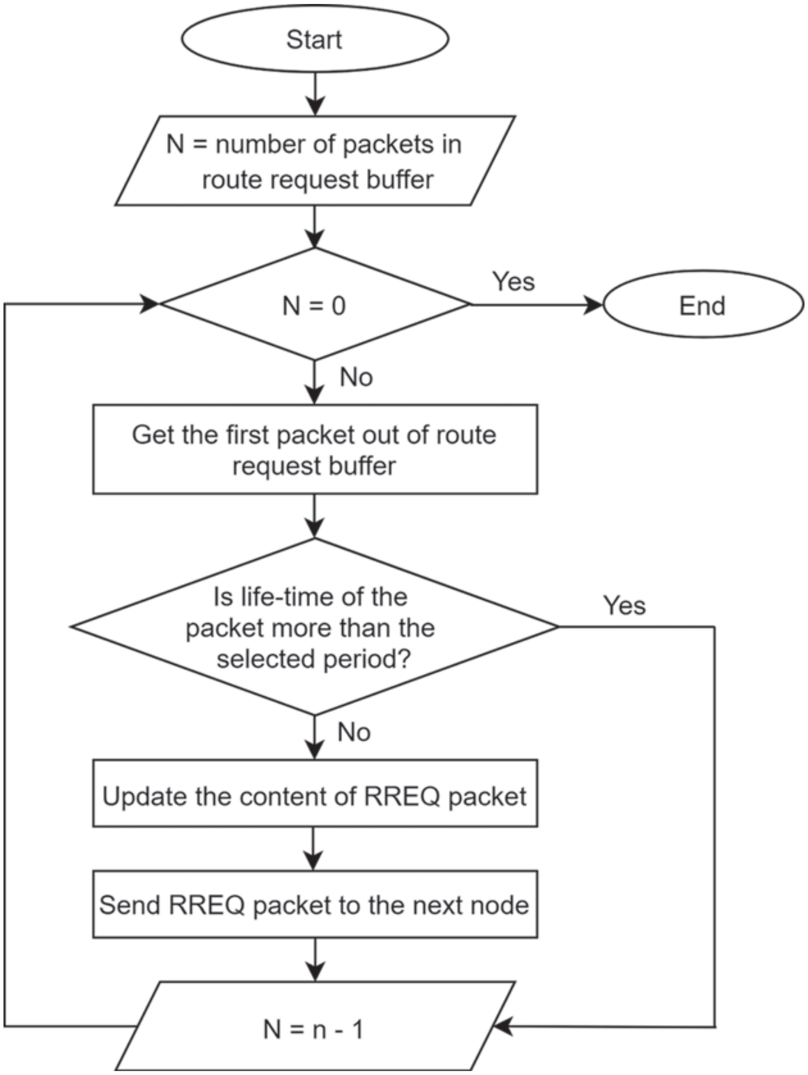

Start
N = number of packets in route request buffer
N = 0
Yes
End
No
Get the first packet out of route request buffer
Is life-time of the packet more than the selected period?
Yes
No
Update the content of RREQ packet
Send RREQ packet to the next node
N = n - 1

Figure 6: Algorithm for an RREQ buffer

The coefficients w_n are combined in a vector of chromosomes as

$$chro = (w_1, w_2, \ldots, w_{12}). \tag{2}$$

When a chromosome is enabled, the value of the criterion function is evaluated for its parameters with respect to each node in the coverage zone, and a threshold is applied to select a node from the coverage zone to be eligible for routing. To select the elite among the whole population for performing the mutation and crossover for generating the offspring, the fitness function value is calculated concerning three evaluation measures, namely, the length of the shortest route, the delay for obtaining the shortest route, and the number of routes that are generated from this criterion function. The variables are being minimized in this optimization. Thus, the elite is selected with the low value of the shortest route, its corresponding delay, and the number of routes generated from this chromosome. Once the elite is selected, 0.8/0.2 crossovers are performed, and 20% mutation is applied to maintain other solutions that may lead to the optimal one. The GA is enabled in two scenarios: the nodes are stationary (static), and the nodes are moving within 2 s interval (dynamic). Two criterion functions are found for each.

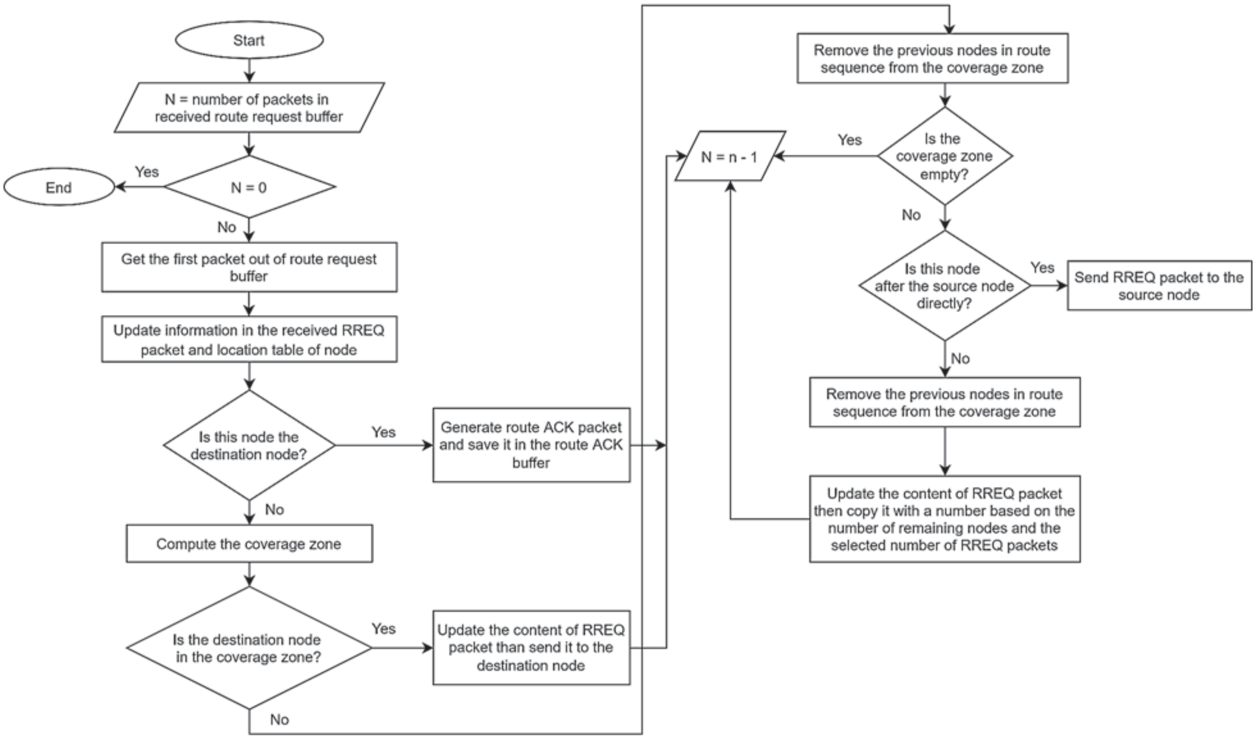

Start
N = number of packets in received route request buffer
N = 0
Yes
End
No
Get the first packet out of route request buffer
Update information in the received RREQ packet and location table of node
Is this node the destination node?
Yes
Generate route ACK packet and save it in the route ACK buffer
No
Compute the coverage zone
Is the destination node in the coverage zone?
Yes
Update the content of RREQ packet than send it to the destination node
No
Remove the previous nodes in route sequence from the coverage zone
Is the coverage zone empty?
Yes
N = n - 1
No
Is this node after the source node directly?
Yes
Send RREQ packet to the source node
No
Remove the previous nodes in route sequence from the coverage zone
Update the content of RREQ packet then copy it with a number based on the number of remaining nodes and the selected number of RREQ packets

Figure 7: Algorithm for a received RREQ buffer

If the problem has been identified, then the candidate alternatives will follow the steps of a GALAR as follows.

Step 1: Randomly generate chro (w_1, $w_2, \ldots, w_n$) coefficients for the criterion function defined in Formula 1.

Step 2: Calculate the fitness function $f(x_1)$,$f(x_2), \ldots$,$f(x_n)$ for each chro ($w_1, w_2, \ldots, w_n$). Use the fitness function to determine each string's fitness in the population (i.e., the length of the shortest route, the delay for obtaining the shortest route, and the number of routes generated from this criterion function).

Step 3: While no n offspring has been produced,

- Select two parent chromosomes (two possible solutions of the criterion function) based on the selection probability.
- The crossover operator produces two offspring: crossover and multipoint crossover probability (Pc).
- All offspring are mutated at each locus operator using the mutation given a defined mutation probability (Pm). The subsequent genes for children are incorporated into the population.
- If n is odd, then a discard function is highly likely to discard a less fit chromosome.

Step 4: Substitute the existing population with a new one.

Step 5: Return to Step 2 until a match is found for the solution or until w has several iterations.

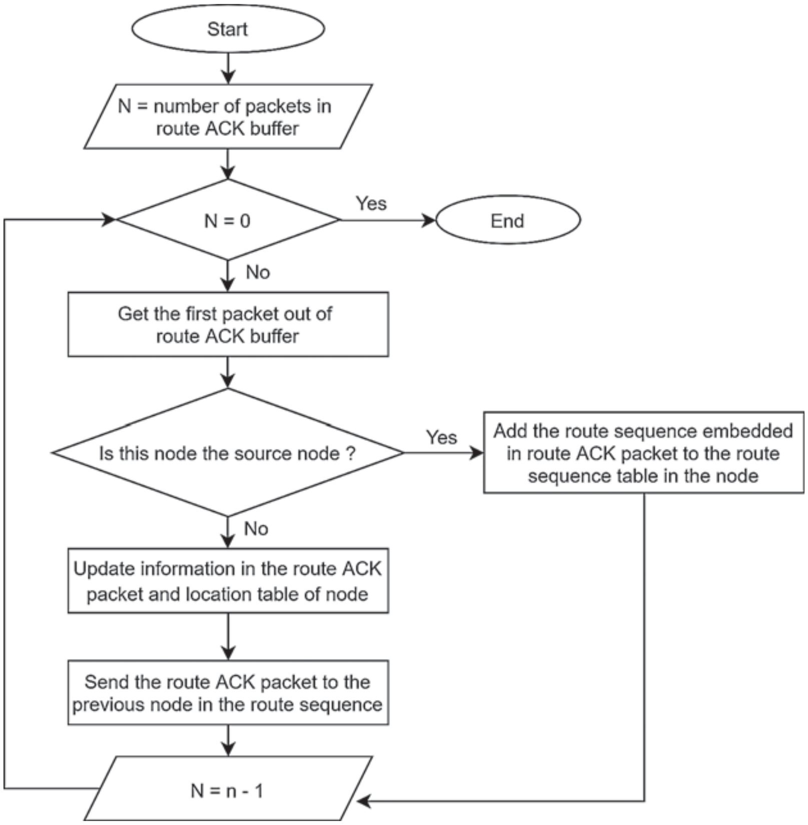

Start
N = number of packets in route ACK buffer
N = 0
Yes
End
No
Get the first packet out of route ACK buffer
Is this node the source node ?
Yes
Add the route sequence embedded in route ACK packet to the route sequence table in the node
No
Update information in the route ACK packet and location table of node
Send the route ACK packet to the previous node in the route sequence
N = n - 1

Figure 8: Algorithm for an ACK buffer

3.3 GALAR Algorithm Operations

For GALAR algorithm operations, this study has modified the mechanism of algorithms for sending an RREQ buffer, an RREQ buffer, a received RREQ buffer, and an ACK buffer. The GALAR algorithm detail operations are shown in Figs. 5–8.

4 Results and Evaluation

4.1 Simulation Model and Assumption

This section presents the simulation environment. The performance of the proposed GALAR algorithm was compared with that of LARwith, which is a development of the classical LAR by embedding the position information of the transmitting node in the RREQ packet where the information table is updated frequently about the other nodes' positions than LAR. Moreover, GALAR was compared with Heuristics, which indicates designing the criterion function based on experience about how the variables of the embedded factors should influence the routing process. The criterion function's coefficients should be guessed based on the nature of the corresponding term's variable in the criterion function and compared with the standard protocol

LAR, which has been proposed in [7]. Moreover, GALAR compared with DALAR, as a benchmark routing protocol, which has been proposed in [17]. The GALAR algorithm was simulated using MATLAB.

In this study, 10, 20, 30, 40, 50, 60, 70, 80, 90, and 100 nodes were simulated to allow for the nodes' arbitrary movement. The nodes have a fixed speed of 0.4 m/s for each node to avoid dealing with the high change of network dynamics and be suitable with the environment dimensions of 20 $m^2 \times 20$ m^2. Each node's communication range was placed at 5 m as an allowance for encountering the worst-case scenario, in which each node has a low energy and a short range of distance. Other simulation unchanged parameters are specified in Tab. 1. Finally, the result of the criterion function is presented, and detailed analysis and discussion of the result are given.

Table 1: Simulation parameters

Description	Value	Unit
Simulator	MATLAB	–
Channel type	Wireless	–
Network area size	20 × 20	m^2
Number of mobile nodes	100	nodes
Simulation time	400	s
Routing protocol	LAR	–
Node transmission range	5	m
Nodes speed	0.4	ms
Moving type	Random	–
Threshold	0.65	–
RREQ packets' lifetime	1.6	ms
Node check available routes	2	ms
Packets' generating rate	250	s

The following are the performance metrics used in the simulation experiments.

a. *PDR* is the ratio of a delivered data packet to the destination. The PDR is computed as follows:

$$PDR = \frac{\sum packets\ received\ by\ destinations}{\sum packets\ sent\ by\ sources} \times 100. \tag{3}$$

b. *Packet loss ratio (PLR)* is the ratio between the number of transmitted data packets and the number of received data packets. The E2E is calculated as follows:

$$PLR = \frac{packets\ sent\ by\ source}{packets\ received\ by\ destination} \times 100. \tag{4}$$

c. *E2E delay*. The E2E delay refers to the average time consumed in ms to transfer a data packet successfully from the source to the destination across the network. The E2E delay is calculated as follows:

$$E2E\ delay = \frac{\sum_{i=1}^{n}(R_i - S_i)}{n}, \tag{5}$$

where n is the number of data packets transmitted successfully across the network, i is the unique packet ID, R_i is the time to receive a packet with unique ID i, and S_i is the time it takes to send a packet with a unique ID i.

d. *RREQ overhead* is the ratio of the number of RREQ packets divided by the number of packets sent. The RREQ overhead is computed as follows:

$$\text{Routing overhead }(\%) = \frac{\text{No of routing packets}}{\text{No of routing packets} + \text{No of data packets sent}} * 100. \qquad (6)$$

e. *Performance index* is the performance measure that has been designed to express more than one aspect of the routing protocol performance. This index includes the delay factor, efficiency, and reliability, all of which are needed for evaluating the overall performance of the proposed GALAR algorithm.

This study enabled GA in two scenarios. The nodes were maintained at a stationary (static) position, and the nodes were made to move within a 2 s interval (dynamic). Two criterion functions were found for each. To determine which of the two functions is better; the PDR between them was compared. The results of the comparison are shown in Fig. 9. The graph shows that enabling genetic optimization in the dynamic scenario yields better PDR.

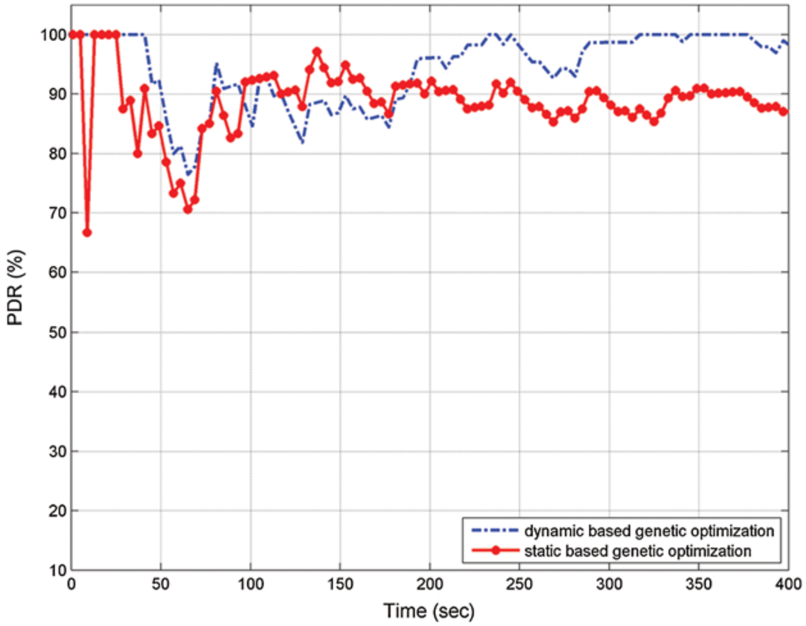


Figure 9: Comparison of PDR between dynamic-and static-based genetics

The graph shows that the network performs well when the movement of the nodes has been considered. The average value of PDR in a dynamic-node situation was 94.81%, whereas in a stationary-node situation, it was 88.14%. The efficient optimization scheme interprets this for establishing the route while tuning the criterion function in the dynamic situation. The reason is the great capability of capturing the dynamics of the environment and the moving nodes. Thus, the criterion function of a dynamic situation was used in the next experiments. The values of coefficients after tuning are as follows:

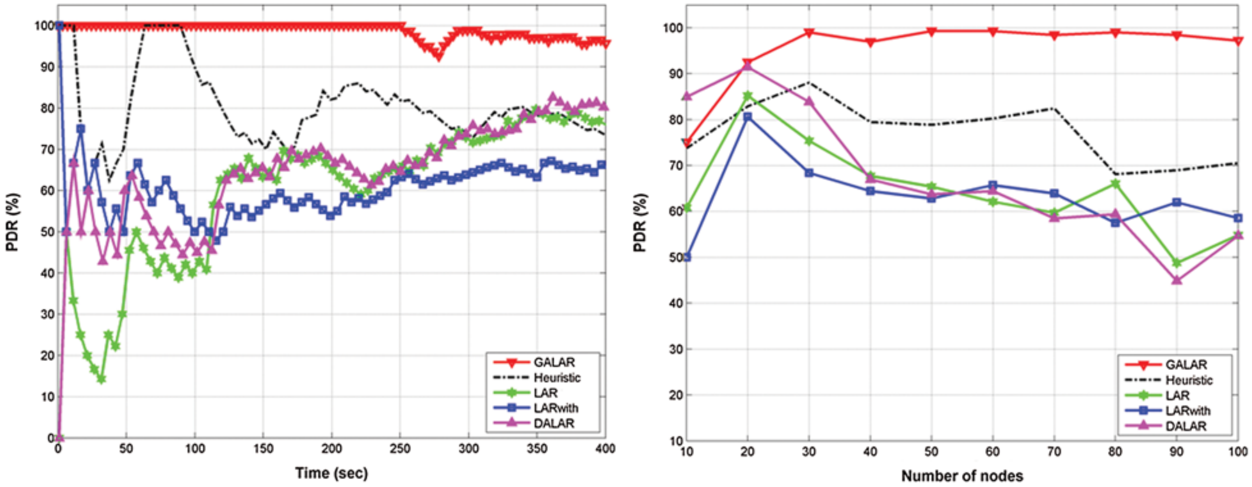
$$\text{Chro} = (w_1, w_2, \ldots, w_{12}) = (0.5128, 0.2996, 0.8899, 1.3513, 1.0201, 0.5467, 0.8880, 1.0857, 0.7611,$$
$$0.3335, 0.4428, 1.0963).$$

Experimental results indicate that the maximum weight observed was 1.3513, and the minimum weight was 0.3335. After tuning the criterion function coefficients, this study aims to prove that the proposed GALAR algorithm outperforms the traditional LAR algorithm. Thus, the PDR, PLR, E2E delay, RREQ overhead, and performance index were compared with an enabling GA for tuning the weights, GALAR.

4.2 PDR

Fig. 10a shows the PDR in the GALAR algorithm with time simulation. The graphs in the figure show that the proposed GALAR algorithm outperformed the other methods. The algorithm also improved the PDR of the network. To test the efficiency and consistency and achieve a certain level of validity of the proposed GALAR algorithm, the authors made a round of 10 experiments while changing the number of nodes and computing the average of PDR, the average required time for constructing a route between a source node and a destination node, and the average required time from generating a new data packet to the sending the packet. The proposed GALAR algorithm achieved an average PDR value of 99.87%, whereas Heuristics achieved an average PDR value of 88.34%. LAR, LARwith, and DALAR achieved an average PDR value of 73.23%, 65.65%, and 74.32%, respectively.

Fig. 10b shows the PDR in the GALAR algorithm with the number of nodes. The proposed GALAR algorithm achieved an average PDR value of 99.76%. By contrast, Heuristics achieved an average PDR value of 82.98%. LAR, LARwith, and DALAR achieved an average PDR value of 84.82%, 88.90%, and 80.73%, respectively.



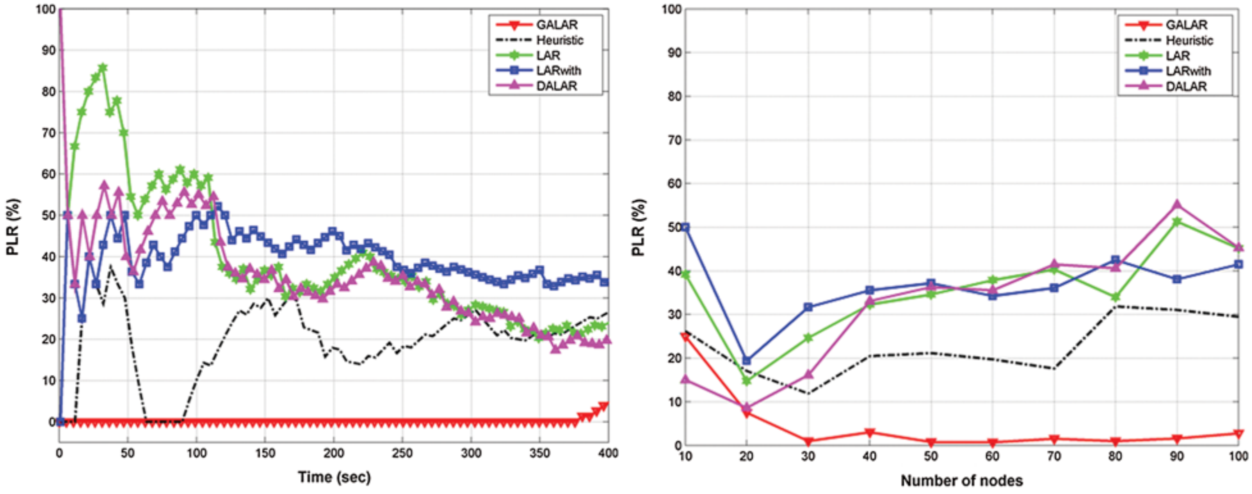
(a) (b)

Figure 10: PDR via (a) time and (b) number of nodes

4.3 PLR

Fig. 11a shows the PLR in the GALAR algorithm with time simulation. The figure shows that the proposed GALAR outperformed the other methods again. It also improved the PLR of the network. The proposed GALAR algorithm achieved an average PLR value of 0.12%, whereas Heuristics achieved an average PLR value of 11.65%. LAR, LARwith, and DALAR achieved an average PLR value of 26.76%, 34.34%, and 25.67%, respectively.

Fig. 11b shows the PLR in the GALAR with the number of nodes. The proposed GALAR achieved an average PLR value of 0.2362%, whereas Heuristics achieved 17.01%. LAR, LARwith, and DALAR achieved an average PLR value of 15.18%, 11.09%, and 19.26%, respectively. This superiority of the proposed GALAR algorithm over the other protocols is interpreted because the GALAR is the only protocol that considers all factors that contribute to efficient routing.



(a)

(b)

Figure 11: PLR via (a) time and (b) number of nodes

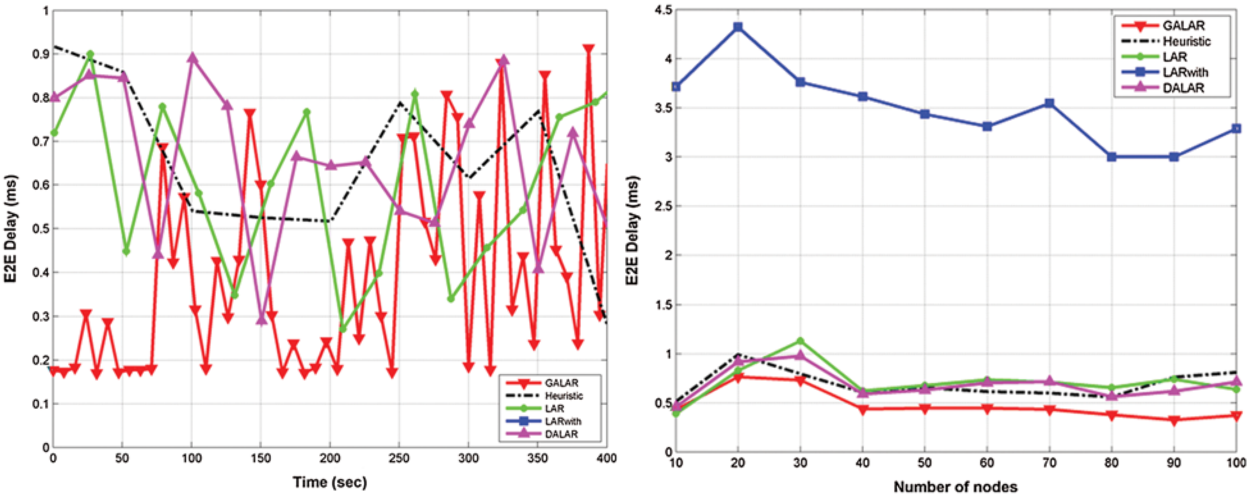
4.4 E2E Delay

The proposed GALAR algorithm acquired the least time to reach the data packet to the destination. The time at which the first packet was transmitted from the source was subtracted from the time where the first data packet arrived at the destination.

Fig. 12a shows the average E2E delay in the GALAR algorithm with time simulation. The figure shows how the proposed GALAR algorithm outperformed the other methods, and the average E2E delay of the network improved. The proposed GALAR algorithm achieved an average E2E delay value of 0.4 ms, whereas that of Heuristics, LAR, LARwith, and DALAR was 0.6, 0.5, 0.7, and 0.56 ms, respectively.

Fig. 12b shows the average E2E delay in the GALAR algorithm with the number of nodes. The proposed GALAR algorithm achieved an average E2E delay value pf 0.4 ms, whereas Heuristics, LAR, LARwith, and DALAR achieved an average E2E delay value of 0.8, 0.7, 3.7, and 0.65 ms, respectively.

This delayed reduction in the GALAR algorithm is explained by the fact that the GALAR algorithm selects the most suitable node to be a candidate for routing the packets. The efficiency in routing is explained by the criterion function optimized based on a dynamic scenario while considering all factors in successful routing. The selection process caused no computational complexity.



(a) (b)

Figure 12: E2E delay via (a) time and (b) number of nodes

4.5 RREQ Overhead

To prove that this GALAR algorithm reduces the network load, the RREQ overhead needed to achieve the connection between nodes was calculated for 10 cases. Figs. 14a and 14b show that the network load with packets in a LAR protocol case increases approximately linearly with the number of nodes because LAR depends on flooding. However, the proposed GALAR algorithm depends on the targeted search that reduces the network load regardless of the number of nodes.

Fig. 13a shows the RREQ overhead in the GALAR with time simulation. The proposed GALAR noticeably improved the RREQ overhead of the network. GALAR algorithm achieved an RREQ overhead value of 0.93%, whereas that of Heuristics, LAR, LARwith, and DALAR was 0.991%, 0.993%, 0.995%, and 0.997%, respectively.

Fig. 13b shows the RREQ overhead in the GALAR algorithm with the number of nodes. The proposed GALAR algorithm achieved an RREQ overhead value of 0.85%. Conversely, Heuristics, LAR, LARwith, and DALAR achieved an RREQ overhead value of 0.90%, 0.92%, 0.93%, and 0.95%, respectively.

4.6 Performance Index

To evaluate the overall performance of the proposed protocol, the authors calculated the new performance metrics, namely, performance index. The results of the performance index are shown in Figs. 14a and 14b.

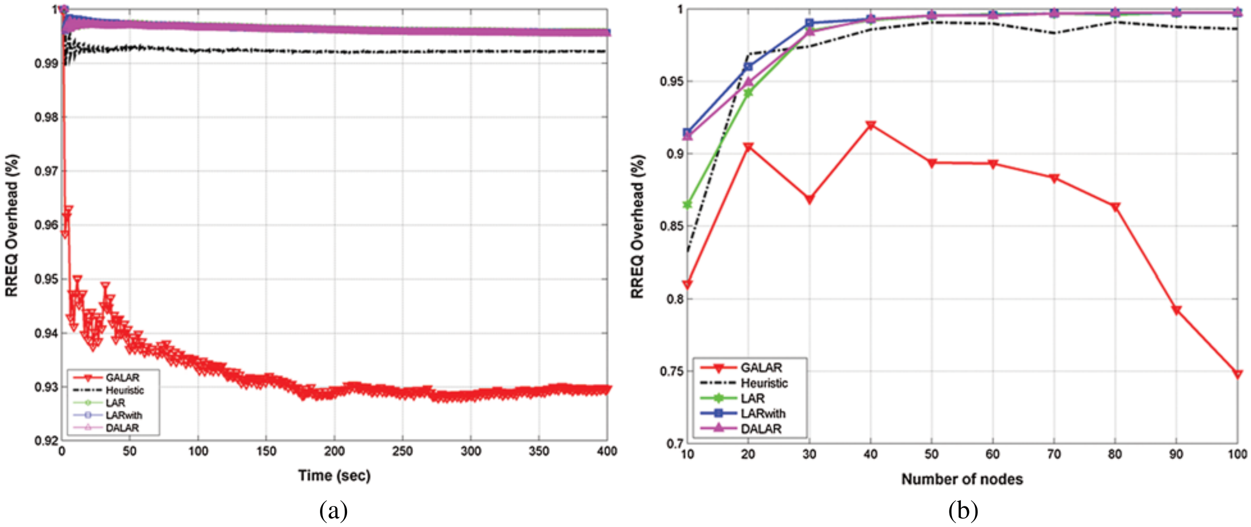


Figure 13: RREQ overhead via (a) time and (b) number of nodes

Fig. 14a shows the performance index in GALAR with time simulation. The GALAR algorithm significantly improved the performance index of the network. The proposed GALAR algorithm achieved a performance index value of 65, whereas Heuristics, LAR, LARwith, and DALAR achieved a performance index value of 42, 43, 45, and 35, respectively.

Fig. 14b shows the performance index in GALAR with the number of nodes. The GALAR algorithm achieved a performance index value of 75, whereas that of Heuristics, LAR, LARwith, and DALAR was 55, 60, 50, and 45, respectively.

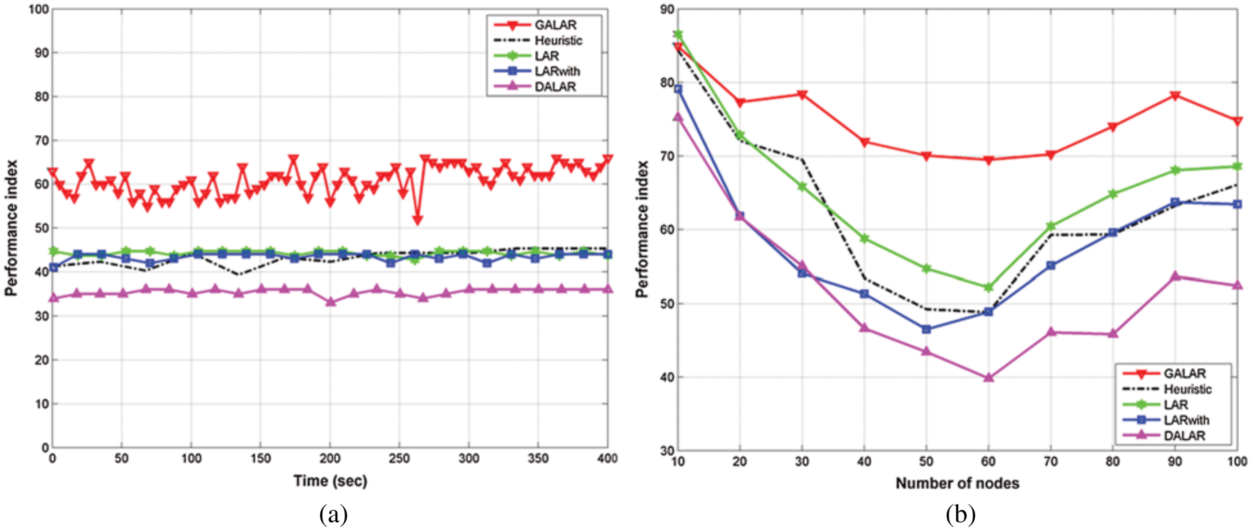


Figure 14: Performance index via (a) time and (b) number of nodes

5 Conclusion and Recommendations

This work analyzed the performance estimation of LAR protocols, considering the use of location information to enhance ad hoc network routing protocols using two scenarios. First, the likelihood of achieving high accuracy in determining the current position of the destination node was achieved by adding the position information of the transmitting node to the packet even when the transmitting node is only a routing node. Moreover, any node that received a packet updated either the packet location information or its location table information to the most recent one. Second, selecting the contributing variables in the routing process and its significance was performed automatically based on genetic optimization.

Genetic optimization is appropriate for avoiding the local minima and determining the most accurate formula for the request zone. This formula selected the candidate nodes from a transmitting node's coverage zone to carry a request packet to its destination to find an optimum route of transmission between a source node and a destination node. Location routing protocols can be improved significantly using an optimization scheme of node selection criteria to pass the packets to their final destination.

This study has proven that evolutionary genetic optimization can map the request zone to a node selection criterion. Moreover, this selection has converted routing from being geometric region-oriented in past research to being node-oriented. This approach proves and supports the concept that the request zone has nonlinearity due to different protocol natures, environments, and node dynamics. Thus, using an analytical definition of a request zone is a key factor in reducing overhead in the network, improving QoS, and addressing other delay measures. The simulation results show that the GALAR algorithm makes a packet delivery that reaches more than 99% with less RREQ, which poses a good performance with few overheads in the network. Moreover, GALAR was compared with other protocols concerning different evaluation measures, and the GALAR algorithm significantly outperformed the other methods.

As a future work, the energy level of the nodes should be considered in the selection criterion process. Nodes with less energy should be avoided to maintain the long lifetime of the network. Characterizing the influence of nodes' mobility and dynamics on the performance of the protocol would also be helpful. Thus, more studies may be able to investigate the process of improving the protocols to handle several issues.

Funding Statement: This research was funded by the Deanship of Scientific Research at Princess Nourah bint Abdulrahman University through the Fast-track Research Funding Program.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] N. Gupta and R. Gupta, "Route-discovery optimization in LAR: A review," in *Proc. of the Int. Conf. on Soft Computing for Problem Solving (SocProS 2011) December 20–22*, New Delhi, Springer, pp. 877–884, 2012.

[2] G. Chavan and V. Srikanth, "Zone based effective location aided routing protocol for MANET," in *Int. Conf. on Advances in Information Technology and Mobile Communication*, Berlin, Heidelberg, Springer, pp. 404–407, 2012.

[3] R. Vallikannu, A. George and S. Srivatsa, "Autonomous localization based energy saving mechanism in indoor MANETs using ACO," *Journal of Discrete Algorithms*, vol. 33, no. 2, pp. 19–30, 2015.

[4] M. B. Khalaf, A. Y. Al-Dubai and G. Min, "New efficient velocity-aware probabilistic route discovery schemes for high mobility Ad hoc networks," *Journal of Computer and System Sciences*, vol. 81, no. 1, pp. 97–109, 2015.

[5] M. B. Talawar and D. Ashoka, "Link failure detection in MANET: A survey," in *Modern Approaches in Machine Learning and Cognitive Science: A Walkthrough Studies in Computational Intelligence*, vol. 885. Cham: Springer, pp. 169–182, 2020.

[6] L. P. Verma, A. Mathuriya and R. Manwall, "Enhance location based power aware routing protocol in ad-hoc network," *International Journal of Computers & Technology*, vol. 2, no. 3, pp. 81–85, 2012.

[7] Y. B. Ko and N. H. Vaidya, "Location-aided routing (LAR) in mobile ad hoc networks," *Wireless Networks*, vol. 6, no. 4, pp. 307–321, 2000.

[8] S. Mirjalili, "Genetic algorithm," in *Evolutionary Algorithms and Neural Networks. Studies in Computational Intelligence*, vol. 780. Cham: Springer, pp. 43–55, 2019.

[9] S. Karakatič and V. Podgorelec, "A survey of genetic algorithms for solving multi depot vehicle routing problem," *Applied Soft Computing*, vol. 27, pp. 519–532, 2015.

[10] A. Assiri, "Anomaly classification using genetic algorithm-based random forest model for network attack detection," *Computers, Materials & Continua*, vol. 66, no. 1, pp. 767–778, 2021.

[11] T. Camp, J. Boleng and L. Wilcox, "Location information services in mobile ad hoc networks," in *Proc. of the 2002 IEEE Int. Conf. on Communications*, New York, NY, USA, pp. 3318–3324, 2002.

[12] M. Abolhasan, T. Wysocki and E. Dutkiewicz, "A review of routing protocols for mobile ad hoc networks," *Ad Hoc Networks*, vol. 2, no. 1, pp. 1–22, 2004.

[13] M. Käsemann, H. Füßler, H. Hartenstein and M. Mauve, "A reactive location service for mobile ad hoc networks," *Technical Report TR-02-014*, vol. 2, pp. 1–13, 2002.

[14] D. Kukreja, D. K. Sharma, S. K. Dhurandher and B. R. Reddy, "GASER: Genetic algorithm-based secure and energy aware routing protocol for sparse mobile ad hoc networks," *International Journal of Advanced Intelligence Paradigms*, vol. 13, no. 1–2, pp. 230–259, 2019.

[15] D. G. Zhang, S. Liu, X. H. Liu, T. Zhang and Y. Y. Cui, "Novel dynamic source routing protocol (DSR) based on genetic algorithm-bacterial foraging optimization (GA-BFO)," *International Journal of Communication Systems*, vol. 31, no. 18, pp. e3824, 2018.

[16] T. Bhatia, S. Kansal, S. Goel and A. Verma, "A genetic algorithm based distance-aware routing protocol for wireless sensor networks," *Computers & Electrical Engineering*, vol. 56, no. 8, pp. 441–455, 2016.

[17] S. C. Huang and H. Y. Chang, "A density-aware location-aided routing protocol for wireless ad-hoc networks," in *2014 Tenth Int. Conf. on Intelligent Information Hiding and Multimedia Signal Processing*, Kitakyushu, Japan, pp. 670–673, 2014.

[18] I. Khider, Y. Eltigani and W. Ismael, "Integrated location-aided routing scheme for mobile ad hoc networks," *MIT International Journal of Electronics and Communication Engineering*, vol. 3, no. 2, pp. 59–62, 2013.

[19] R. Aboki, E. Shaghaghi, P. Akhlaghi and R. M. Noor, "Predictive location aided routing in mobile ad hoc network," in *2013 IEEE 11th Malaysia Int. Conf. on Communications*, Kuala Lumpur, Malaysia, pp. 57–61, 2013.

[20] M. Sharma, M. Kaur and K. Singh, "Efficient utilization of bandwidth in location aided routing," *IOSR Journal of Electronics and Communication Engineering*, vol. 7, no. 5, pp. 38–42, 2013.

[21] N. C. Wang, Y. F. Huang, J. S. Chen, S. M. Wang and C. L. Chen, "An improved location-aided routing protocol for mobile ad hoc networks with greedy approach," *WSEAS Transactions on Communications*, vol. 8, no. 8, pp. 970–979, 2009.

[22] T. F. Shih and H. C. Yen, "Location-aware routing protocol with dynamic adaptation of request zone for mobile ad hoc networks," *Wireless Networks*, vol. 14, no. 3, pp. 321–333, 2008.