

## Research Article

# IMOC: Optimization Technique for Drone-Assisted VANET (DAV) Based on Moth Flame Optimization

Rehan Tariq , Zeshan Iqbal, and Farhan Aadil

*Department of Computer Science, University of Engineering and Technology Taxila, Pakistan*

Correspondence should be addressed to Rehan Tariq; rehan.tariq@students.uettaxila.edu.pk

Received 1 June 2020; Revised 3 September 2020; Accepted 29 September 2020; Published 7 November 2020

Academic Editor: Sungchang Lee

Copyright © 2020 Rehan Tariq et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Technology advancement in the field of vehicular ad hoc networks (VANETs) improves smart transportation along with its many other applications. Routing in VANETs is difficult as compared to mobile ad hoc networks (MANETs); topological constraints such as high mobility, node density, and frequent path failure make the VANET routing more challenging. To scale complex routing problems, where static and dynamic routings do not work well, AI-based clustering techniques are introduced. Evolutionary algorithm-based clustering techniques are used to solve such routing problems; moth flame optimization is one of them. In this work, an intelligent moth flame optimization-based clustering (IMOC) for a drone-assisted vehicular network is proposed. This technique is used to provide maximum coverage for the vehicular node with minimum cluster heads (CHs) required for routing. Delivering optimal route by providing end-to-end connectivity with minimum overhead is the core issue addressed in this article. Node density, grid size, and transmission ranges are the performance metrics used for comparative analysis. These parameters were varied during simulations for each algorithm, and the results were recorded. A comparison was done with state-of-the-art clustering algorithms for routing such as Ant Colony Optimization (ACO), Comprehensive Learning Particle Swarm Optimization (CLPSO), and Gray Wolf Optimization (GWO). Experimental outcomes for IMOC consistently outperformed the state-of-the-art techniques for each scenario. A framework is also proposed with the support of a commercial Unmanned Aerial Vehicle (UAV) to improve routing by minimizing path creation overhead in VANETs. UAV support for clustering improved end-to-end connectivity by keeping the routing cost constant for intercluster communication in the same grid.

## 1. Introduction

Vehicular ad hoc networks (VANETs) are different from mobile ad hoc networks (MANETs); therefore, clustering algorithms designed for MANETs cannot be applied to VANETs. In traditional VANETs, infrastructure, like roadside units (RSUs), is used to provide network services to vehicular nodes, selecting the optimal paths and transmitting data. This infrastructure provides road safety information, road congestion, alternative routes, along with weather conditions to drivers. In urban areas where RSU support is available, VANETs work efficiently, but in those areas where infrastructure is not available, VANETs do not perform well [1]. On the other hand, scalability is one of the challenges in VANETs. Clustering is used to solve the scalability issue, but in the high-speed environment on highways where the vehicle speed is relatively much faster than in urban areas, the

clustering does not work well, resulting in degraded network performance due to the higher rate of reclustering [2]. Existing VANET routing and clustering algorithms are computationally expensive, so we need to build a heterogeneous routing algorithm (for flying ad hoc network- (FANET-) assisted VANET) with low routing overhead, efficient utilization of computational resources, and high overall network throughput [3]. The addition of UAVs in existing VANETs is a challenging task because they have very distinct features as compared with ground nodes/vehicle. Another challenge is the efficient utilization of flight time of UAVs because UAVs carry limited energy resources [4]. In VANET, partial infrastructure support is available through RSUs; replacing the RSUs with UAVs to form a fully ad hoc network is another challenge to be addressed.

The current traffic system has many problems like road congestion, accident risks, mobility, node energy, node

TABLE 1: VANET routing challenge.

Topology-based routing	Geography-based routing
Performance at stake in rural areas	Performance on stake in urban areas
Transmission can be delayed	Transmission of data for longer distances
Higher routing overhead	Incorrect GPS coordinates for a node
Higher packet drop ration	Inherent loops can occur
Routes are broken more frequently	Network partitioning more frequently

TABLE 2: UAV classification.

UAV type	Weight (kg)	Altitude (m)	Hovering time (hrs)	Range (km)
Micro	<5	250	1	<10
Mini	150	150-300	<2	<10
Close range	150	3000	2-4	10-30
Short range	200	3000	3-6	30-70
Medium range	1250	5000	6-10	70-200
Medium-range endurance	1250	8000	10-18	>500
Low-altitude deep penetration	350	50-9000	0.5-1	>250
Low-altitude-long-endurance	<30	3000	>24	>500
Medium-altitude-long-endurance	1500	14000	24-48	>700

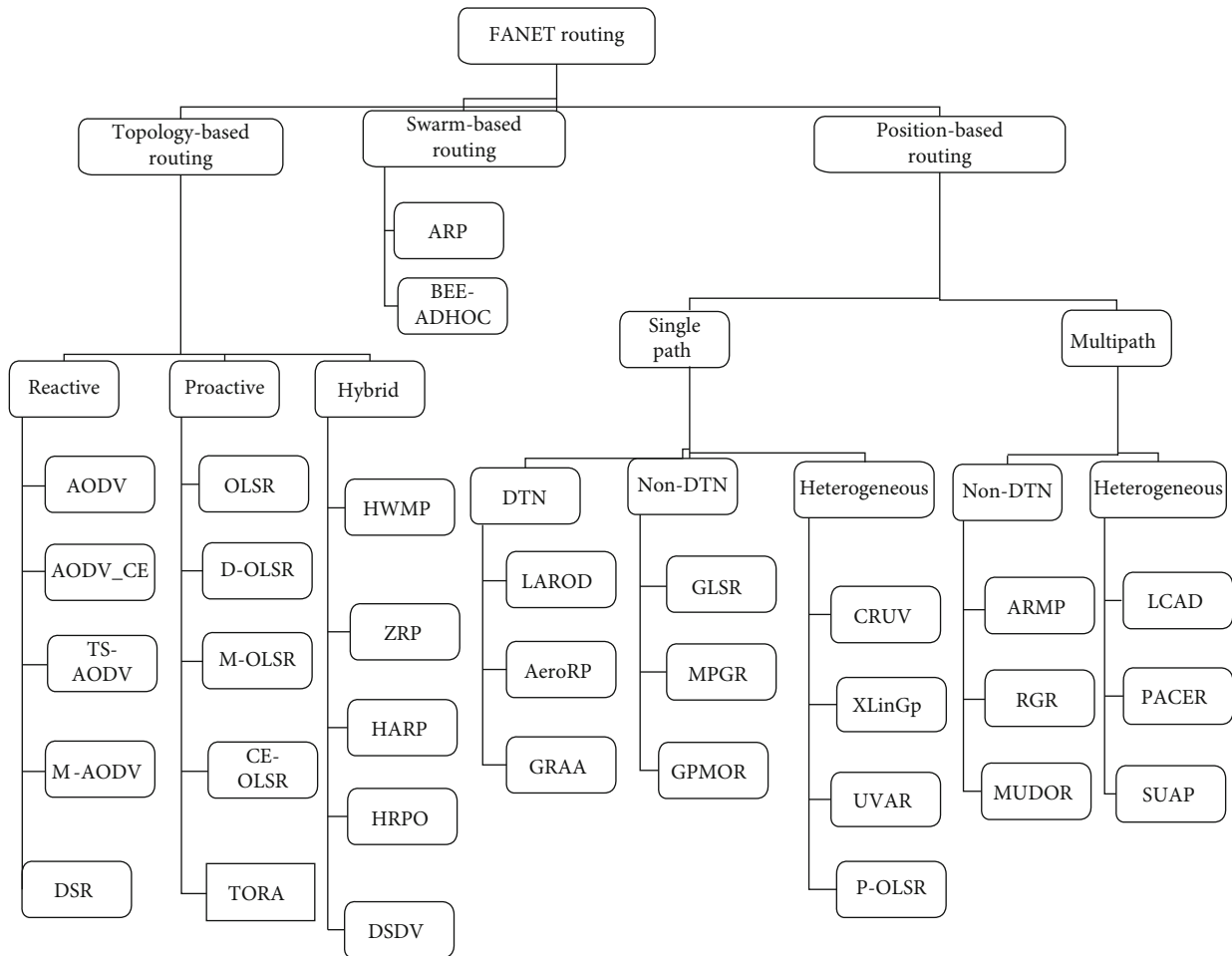


FIGURE 1: FANET routing classification.

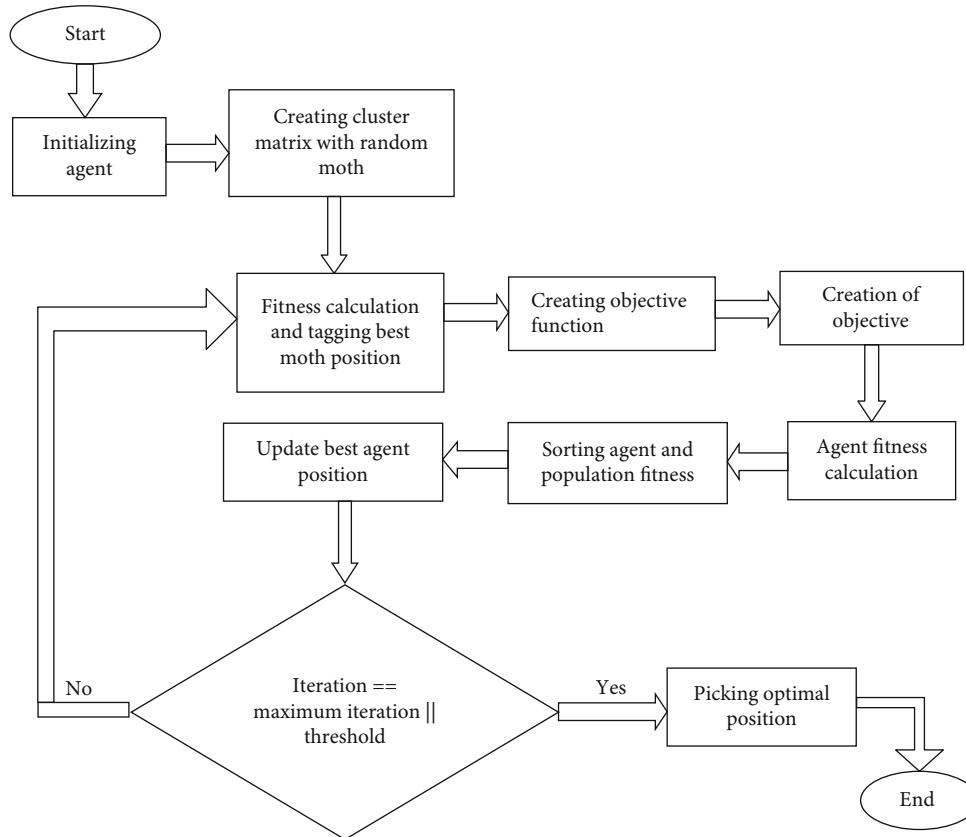


FIGURE 2: Flow chart of proposed IMOC algorithm.

physical condition, and received signal strength of node [5]. If nodes are participating in path construction and path maintenance phase, then the route will be considered as reliable [6]. The reliable route improves packet delivery ratio, reliability, and packet delays and achieves low overhead during transmission of data. Route reliability is essential and robust for application such as disaster management and audio and video conferencing. If the route is lost, then the packet takes a lot of time to reach a destination with higher travelling cost. So, to solve these issues, FANET assistance will provide a better solution to solve irregularities in traditional VANETs.

Genetic algorithms/programming, evolutionary strategies, and learning classifier systems are some types evolutionary algorithms [7, 8]. Evolutionary algorithms offer a decent solution for the problems that cannot be solved with other techniques. In situations where we must find a solution for unsolvable problems, evolutionary techniques are widely accepted. EA might be computationally expensive, but finding a near-optimal solution for unsolvable problems is acceptable. In FANETs and VANETs for a continuous node clustering problem, the choice of evolutionary algorithms is effective [9].

The natural evolution model of biological evolution is the base for evolutionary algorithms [10]. An environment will be generated in which possible solutions will be evolved to find a solution for the problem. For problem factors with

regard to constructed surroundings, it is possible to get the best possible solution through evolution. To solve the scalability issue, nodes are grouped and they share the same geographical coordinates [11] [12]. To provide solutions for network scalability, clustering is one of the methods [13]. The clustering solution ensures the effective utilization of resources with load balancing in each cluster. A moth flame optimizer is one of the finest clustering techniques to provide an optimal number of clusters. Moths are the insects like butterflies. About 16000 species of moths are identified to date. Like other insects, moth larvae convert into cocoons in adulthood. The moths navigate at night-time and follow moonlight. The traverse orientation method is used for traveling by moths. During traveling, moths follow moonlight by keeping a fixed angle toward the moon. Their going after moonlight with a fixed angle keeps them in a straight line. Humans adopted the same method for traveling in a straight line at night [14, 15]. For example, at night, if a man wants to walk toward the west, the moon position must be on the northern side of the sky. By keeping the moon on the right side, a man can easily travel in straight line. With regard to the efficacy of transverse alignment, often, moths are tricked by nonnatural light and are inclined to fly spirally towards nonnatural light. If the light source is far away, then the same behavior for transverse orientation performs well.

Once artificial light comes across a straight path that is being followed, moths try to keep the angle toward the

```

1. START
2. Define grid size
3. In the 2d grid random deployment of vehicular nodes
4. Broadcasted position of each node in the search space
5. Node IDs as vertex, mesh topology is formed
6. Assignment of values to edges in the mesh network by distance calculation of each node
7. Create search space of  $m \times n$  order, initialized moth position
8.   When  $i = 1$ , loop from  $i$  the total maximum number of search agents. FOR  $j$  to the total number of dimensions where  $j$  starts from 1
9.   Position of each moth updated by moth position  $(i, j) =$  upperbound minus lowerbound both starts from I divided by node position in the grid plus lower-bound
10.   End loop
11. If simulation stalled or ended (20 iterations)
12.   FOR moth  $i$  to flame size and  $i$  starts from 1
13.     Calculation of fitness of moth_position (MP) as moth fitness = fitness_function (); fitness moth_position (MP) calculated
14.     WHILE node list NOT empty for clustering nodes
15.       Allocation of the best solution to moth with cluster fitness of each moth less than the finest result
16.     END_WHILE
17.   END_FOR
18. Sorted_fitness (), sorting fitness values of all moths
19. Population_sorted () w.r.t sorted_fitness () population is sorted
20. Among the updated fame position, the best obtained till now
21. Best flame_score = sorted_fitness (1);
22. Best flame_position = population_sorted (1, :)
23. Update moth position-based on the corresponding flame
24.   FOR  $i$  from one to maximum search_agents
25.     FOR  $j$  from one to total dimensions
26.       Compute distance for  $i$ th-moth for  $j$ th-flame; equivalent to absolute (population_sorted ( $i, j$ ) - MP ( $i, j$ ));
27.       Moth location update
28.     END_FOR
29.   END_FOR
30. IF convergence_curve = convergence_curve iteration no 131.      stall iteration++;
31. ELSE
32.   stall_iteration =0;
33. END_IF
34. Iteration++;
35. END loop
36. Best solution from search is equal to total number of
37. END

```

ALGORITHM 1: Intelligent moth flame clustering optimization for VANETs.

artificial light source. Deadly paths for moths occur when the artificial light source is too close to moths, because moths must converge toward the light. This convergence property of moths can be exploited mathematically as a moth flame optimizer (MFO) algorithm [16]. In this research, we proposed an intelligent moth flame clustering optimization for VANET (IMOC) to optimize the clustering problem in VANETs with air assistance of FANETs.

*1.1. Vehicular Ad Hoc Network.* The moving vehicles are equipped with advanced communication capabilities to form a wireless network referred to as VANETs. VANETs offer intelligent transportation services including road conditions, traffic density, alternative routes, vehicle conditions, nearby rest areas, and weather updates to drivers. Intelligent transportation integrated information systems, communication sensors, advanced mathematical methods, and high technol-

ogies to traditional transportation infrastructure. Traffic-matics is the term used in intelligent transportation system (ITS) where moving vehicles act as network nodes for transceiving and routing packets in a network [17]. To ensure a safe and secure route for vehicles is the main application of ITS. Information including unseen traffic, road conditions, weather information, traffic density, and infotainment is broadcast to make the trip safer for drivers and passengers. To provide short wireless networks between vehicles, radio devices and onboard units (OBUs) are installed on the vehicle. These devices are used to provide communication between OBU and RSU to form VANETs [18].

To get an accurate geographical position of moving nodes, vehicles are equipped with a global positioning system (GPS) and a differential global positioning system (D-GPS). RSUs serve as a cellular base transceiver system and act as a backbone to provide communication between vehicles in

TABLE 3: Parameters for simulation.

Parameters	IMOC	CLPSO	GWO	ACO
Total population size	100	100	100	100
Maximum number iteration	150	150	150	150
Total runs	10	10	10	10
Weight for inertia	0.90	0.694	0.694	—
Rate of evaporation	—	—	—	0.5
C1	2	2	2	2
C2	2	2	2	2
Grid size for simulation	500 m <sup>2</sup> , 1000 m <sup>2</sup> , 1500 m <sup>2</sup> , 2000 m <sup>2</sup>	500 m <sup>2</sup> , 1000 m <sup>2</sup> , 1500 m <sup>2</sup> , 2000 m <sup>2</sup>	500 m <sup>2</sup> , 1000 m <sup>2</sup> , 1500 m <sup>2</sup> , 2000 m <sup>2</sup>	500 m <sup>2</sup> , 1000 m <sup>2</sup> , 1500 m <sup>2</sup> , 2000 m <sup>2</sup>
Number of vehicles	10 to 100	10 to 100	10 to 100	10 to 100
Interval between vehicles	+20	+20	+20	+20
Transmission ranges	25 m to 200 m	25 m to 200 m	25 m to 200 m	25 m to 200 m
Vehicle position	Fixed	Fixed	Fixed	Fixed
Minimum_distance between vehicles	1.5 m	1.5 m	1.5 m	1.5 m
Maximum_distance between vehicles	5 m	5 m	5 m	5 m
W1 (1st objective function's weight)	0.5	0.5	0.5	0.5
W2 (2nd objective function's weight)	0.5	0.5	0.5	0.5

VANETs [19]. The mobility model is not random as vehicles follow the road trajectory, but speed is relatively high as compared with MANETs. The energy is not a critical issue because transceivers utilize engine power to establish communication in VANETs. The number of RSUs in VANETs depends on the communication protocol. The communication in VANETs might be intervehicle, vehicle to RSU, and routing-based communication [20]. The information needs to be broadcast efficiently in VANETs for effective information interchange during communication between nodes. To provide such capability, there is a need to have efficient routing protocols. The proactive (table-driven) and reactive (on-demand) are two main classifications for routing protocols.

In proactive routing protocols, the routing information is available every time in its packet header. Optimized link-state routing (OLSR) and fishy state routing (FSR) are types of proactive protocols. FSR minimizes the overhead because it does not broadcast; it only exchanges topological change with its neighboring nodes [21]. OLSR uses multipoint relays to the optimized broadcasting process of the control message to keep the routing table updated. Hello and topology control message are used to discover and disseminate link-state information. Nodes share topological change to their neighboring subset as nodes have limited repetitions for broadcasting [22]. In reactive protocols, the route is not stored permanently which helps in minimizing communication overhead and routes are established on-demand. In the route construction process, the control message is broadcasted through flooding to look up participating nodes for communication. Ad hoc on-demand distance vector (AODV), dynamic source routing (DSR), and temporally ordered routing

algorithm (TORA) are examples of reactive routing. AODV apply route discovery by hop count and sequence number. Destination sequenced number is checked in the route construction process based on the route request/route reply messages [23]. DSR routing information is attached to the data packet header from the source. Route recovery or maintenance is the limitation of DSR [24]. The temporally ordered routing protocol (TORA) is based on a three-level route construction, route maintenance, and erasing route by using query (QRY), update (UPD), and clear (CLR) messages. Topological change does not have an effect on routing information until a complete path from source to destination has been lost [25]. To provide fast message data delivery along curved roads overlays the node selection based on optimal position and exponent partition range [26].

Geographic or position-based routing protocols used GPS to pick exact coordinates of nodes and used their current location for routing data [27]. GPSR, geographic source routing (GSR), and greedy perimeter coordinator routing (GPCR) are some of the examples of position-based routing protocols. A greedy perimeter stateless routing protocol works on greedy approval for transmission of data between the sender and receiver. The locations of the transmitting node and the destination are used to find other nodes to construct a route [28]. In GSR, the shortest path is calculated between the sender and receiver based on their locations [29]. GPCR coordinating nodes are given preference with noncoordinating nodes. Communication is established between the sender and the receiver on their geographical locations and road conditions [30]. Anchor-based street traffic-aware routing (A-STAR) is the best route established

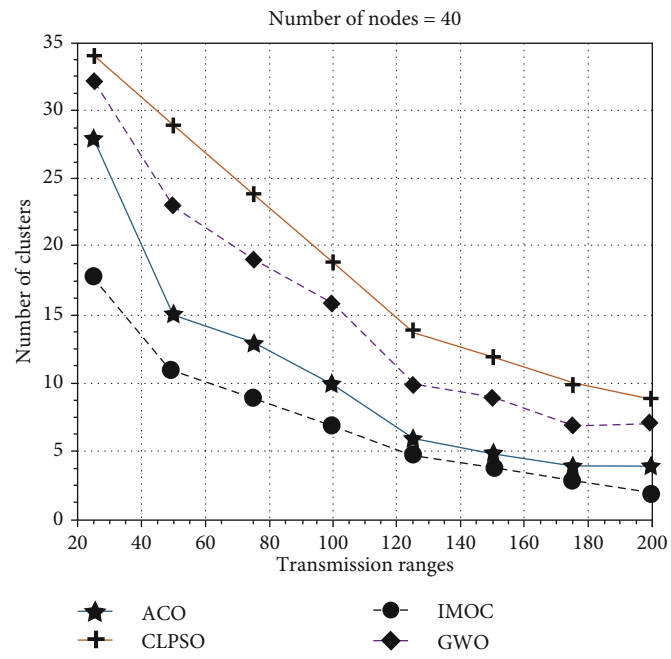
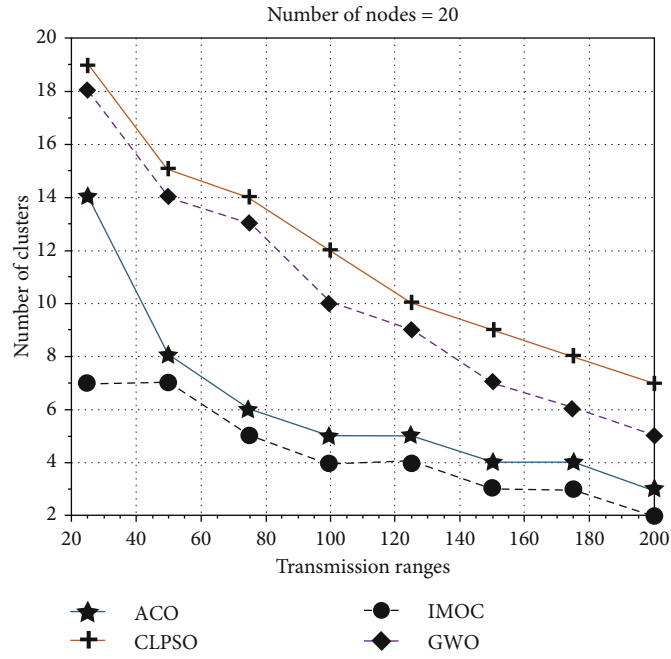


FIGURE 3: Continued.

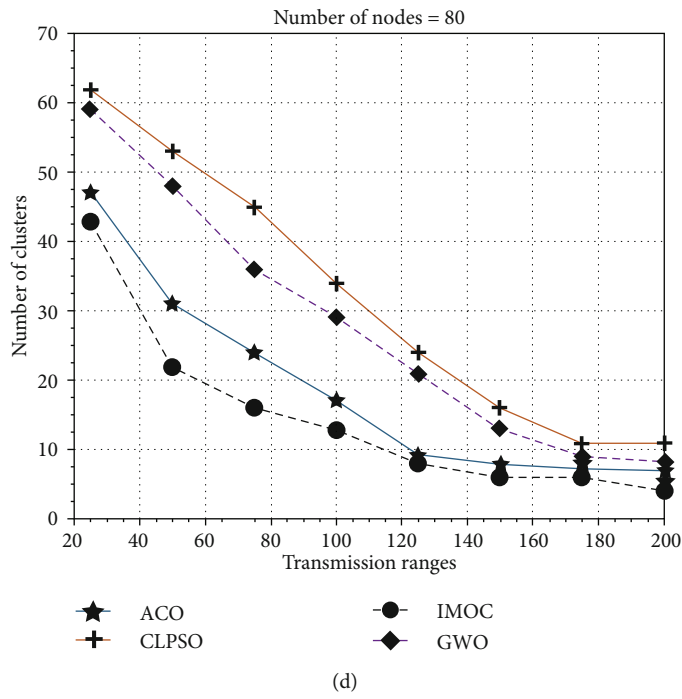
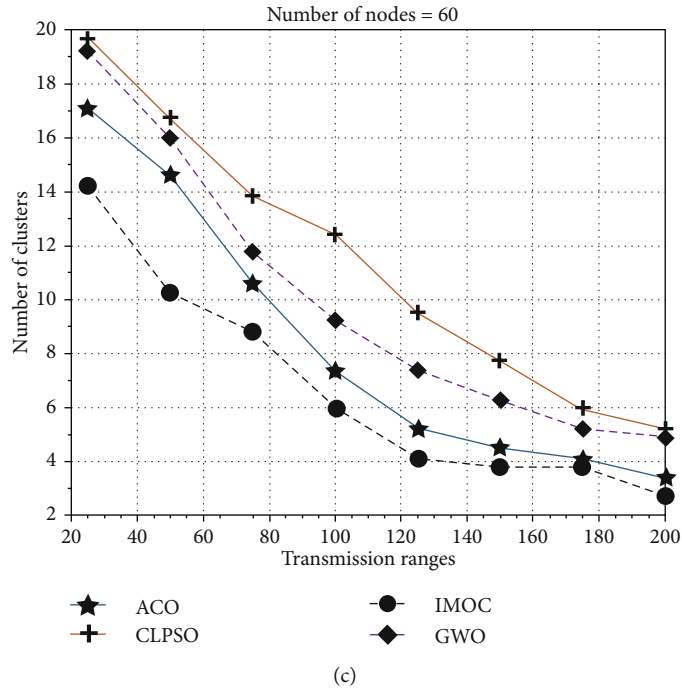


FIGURE 3: Continued.

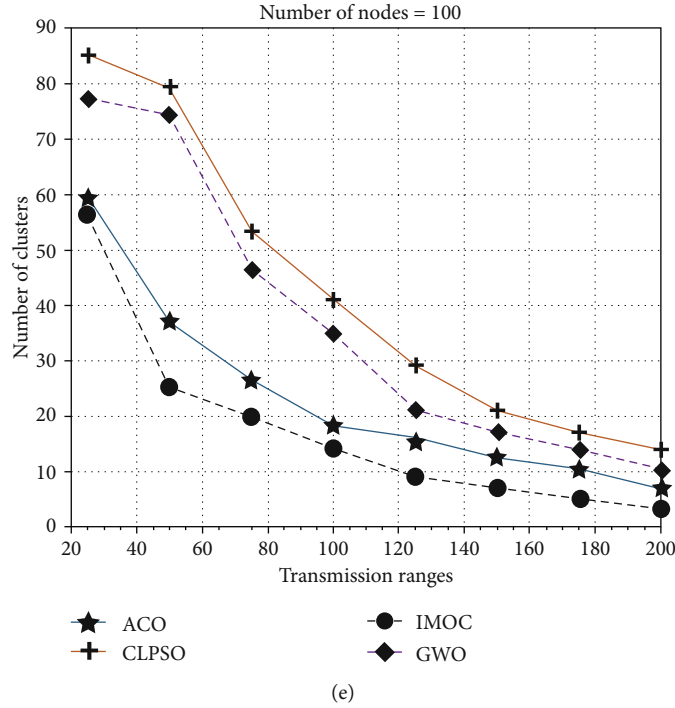


FIGURE 3: Grid\_size 500 m  $\times$  500 m, nodes 10 to 100.

on information gathered from nodes including their location and trust. The anchor path is computed with Dijkstra's least weight path [31]. Table 1 presents the routing challenges for topology-based and geographical-based routing protocols in VANET.

*1.2. Flying Ad Hoc Network.* Availability of low-cost Wi-Fi radio interfaces, GPS, micro-embedded systems, high-resolution cameras, and sensors raised a path for developing intelligent flying vehicles or UAVs [32]. These UAVs created a relatively new era of networks known as FANETs. To integrate drones with an existing vehicular network to improve overall network performance is known as a drone-assisted vehicular network (DAVN) [33]. The UAV-assisted applications have their unique features, competitive advantages, and characteristics [34]. The fundamental operation in Internet of Things (IoT) application is data aggregation. It can be seen in distributed internet-based industrial computing and control systems [35]. The FANET applications can be found everywhere from civilian to military use [36]. Such applications are traffic monitoring, disaster monitoring, providing coordination between rescue teams, crop monitoring, fire monitoring where human access is difficult, infotainment, autonomous tracking, and border surveillance [37, 38]. Two types of applications for FANET can be classified on the deployment of an aerial node in topology: one is a single aerial node application and the second is multi-aerial node applications. In the single aerial node application, only one aerial node (AN) is deployed in the middle of base stations localized on the ground; the AN serves as a router between multiple base stations, whereas in multi-UAV application, a team of ANs works together to provide services [39].

Table 2 shows the classification for UAV type, coverage range, weight, climb rate, and endurance time in the air. FANETs are considered as a subclass of MANETs; UAV routing becomes more complex as AN characteristics vary from other ad hoc networks. The characteristics, including mobility rate, number of ANs, transmission range, weather conditions, and residual energy, need to be critically addressed for designing a routing scheme. Under these limitations, higher communication failures can result in high dynamic movements of ANs. In FANETs, effective routing will support to keep services and applications stable and available all the time. The FANET is an additional support to enhance the effectiveness of existing technologies such as VANETs and MANETs [40]. The ANs can be placed and dispatched in multiple scenarios to improve VANET and MANET applications to provide end-to-end connectivity between ground nodes.

Figure 1 presents the classification for FANET routing protocols. The classification for topological-based protocols can be divided as proactive, reactive, and hybrid. The position-based routing protocols are classified as single path schemes and multipath schemes. The swarm-based protocols are listed for FANET routing. The parameters for FANET-routing protocols are node density, link information between ANs, residual energy, coverage area, and mobility pattern.

The concept of FANET-assisted VANETs is the focus of researchers these days. Various approaches were proposed where drones/UAVs were used to assist VANETs. One of the initial approaches is a multi-UAV-aided network [41]. This approach proposed two-layer networking, i.e., aerial networking and ground networking where the former is responsible for air-to-air communication and the latter is a



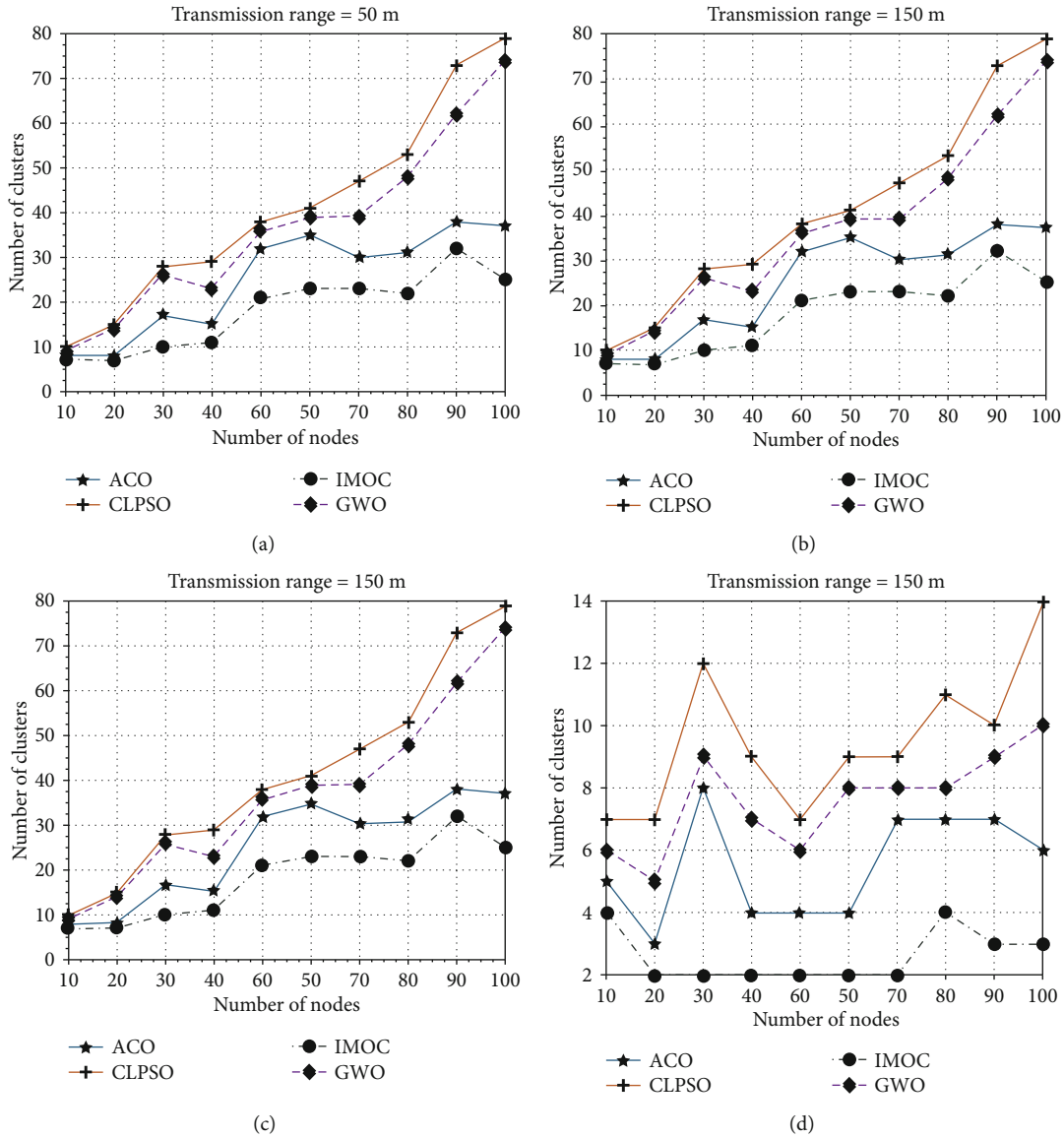


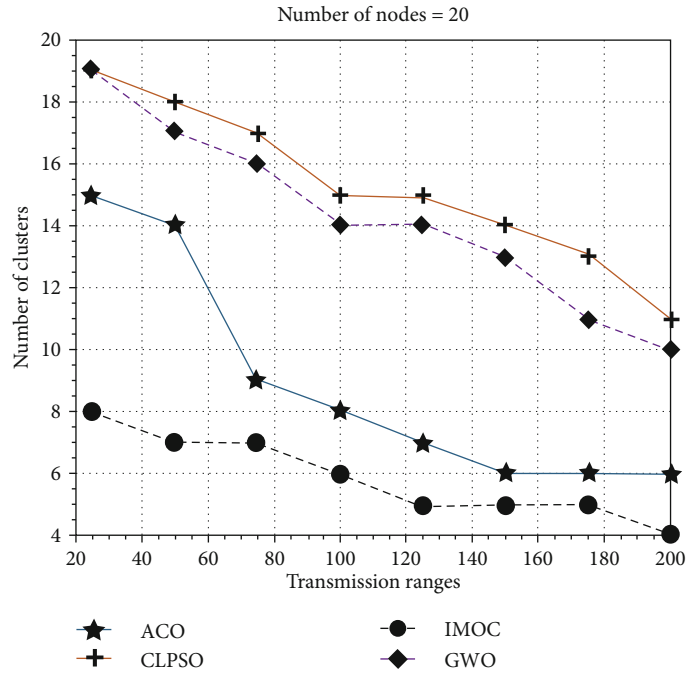
FIGURE 4: Grid\_size 500 m × 500 m, transmission\_ranges 50 to 200.

VANET which transfers the data among vehicles. A special channel is established between these two channels for transferring information such as road conditions. A UAV-assisted VANET routing protocol (UVAR) is a delay tolerant protocol [3] in which UAVs are used which have global knowledge of the network. UAR has two subcomponents: UVAR-G is responsible for transferring packets among connected vehicles by considering traffic density, whereas UVAR-S works by forwarding packets to the UAV. UVAR-S is an on-demand routing protocol that considers multipath toward UAVs and selects the most connected one as the preferred path.

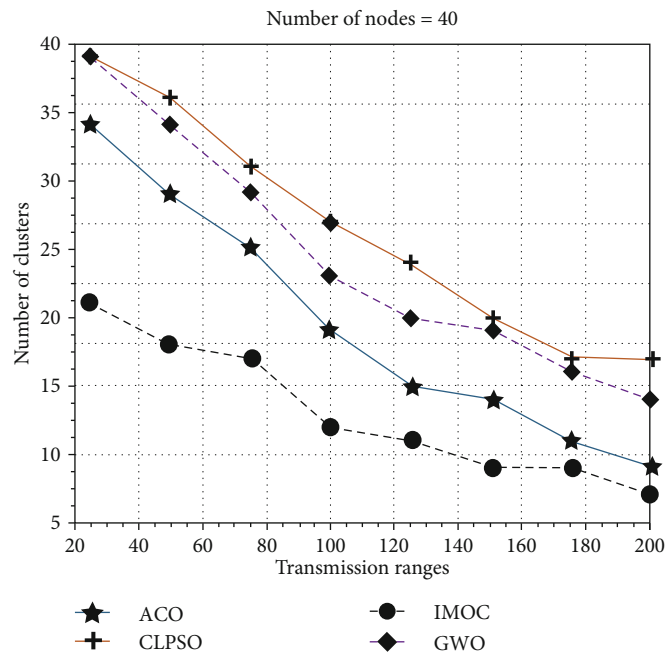
To get reliable data delivery and guarantee robust paths, the flooding-based techniques are used in providing efficient routing solutions. The existing UAVs cooperate in an ad hoc fashion with vehicles [42]. The U2RV routing protocol is proposed in [43]. U2RV is a four-phase process. In the first

phase, various paths are discovered; the paths can include any path established through the UAV; based on source and destination, a suitable path is selected from the set of paths discovered in the first phase. This is followed by the actual data delivery in the third phase. The final phase deals with the discovery of an alternative path which is necessary as the routes in VANETs are dynamic. UAVs are proposed to be used in VANETs [44] for help in finding the disconnected segments and work as relay nodes in VANET infrastructure. A central ground station is at the heart of the scheme which sends instructions to UAVs for storing and forwards the data and dispatches it towards disconnected segments. Disconnected segments are identified through the exchange of hello messages between vehicles and ground stations.

UADD, a protocol for smart transportation networks, is proposed in [45]. To provide communication between UAVs



(a)



(b)

FIGURE 5: Continued.



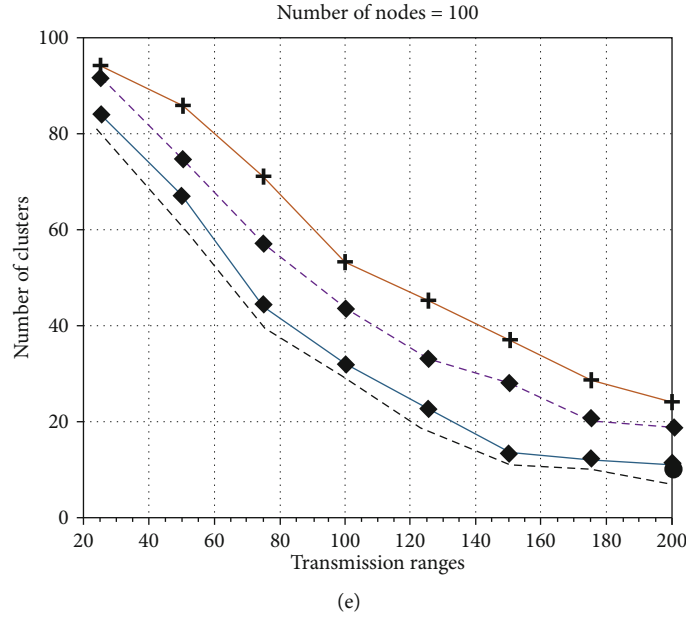


FIGURE 5: Grid\_size 1000 m  $\times$  1000 m, nodes 10 to 100.

and vehicles, an opportunistic virtual interaction scheme was introduced. The crux of this research work is forwarding the data on the optimal intersection to the UAV when the vehicle-to-vehicle communication is not possible. Like the previous scheme, the UAV is used to store and forward packets to the other vehicle. The use of UAV mitigates the effects of jamming in VANETs [46]. It has been observed that hackers observe the traffic pattern between OBUs and vehicles, so they launch a jamming attack. Authors have proposed to use UAVs to shield against this threat. A technique based on reinforcement learning was used to achieve an optimal relay policy adopted by UAVs to avoid the aforementioned attack.

One of the fundamental aspects of any ad hoc network is its mobility model. In [47], the authors proposed a mobility model for UAV- and VANET-based communication. In the proposed model, UAVs follow the movements of vehicles on the road. To maintain the connectivity, the received signal strength (RSSI) from the vehicle is used. The UAV selects the vehicle with the lowest RSSI value and tries to improve the RSSI, so that packets can be delivered successfully to the said vehicle.

To improve communication performance of VANETs between ongoing OBUs and UAVs against smart jammers, the UAVs are used to induce a specific strategy according to the jammer attack [48]. To enhance the network life time and mitigate the “hot spot” problem, a new algorithm is proposed, an asynchronous clustering and mobile data gathering based on timer mechanism (ACMDGTM) [49]. In the curved road scenario which overlays the node selection method, adaptive relay-node selection (ARNS) is used to redefine the optimal position of the node while considering obstacle distribution. The broadcasting characteristics of ARNS are used to classify the road structure [50].

## 2. IMOC-Proposed Methodology

The flow of the proposed IMOC algorithm is shown in Figure 2. During the initialization phase in solution space ( $m \times n$ ), the random position assigned to each moth and moth array is equipped with fitness values. For flames, a similar ordered matrix and array are generated. The best value for the moth found so far is stored in the flame matrix. It is an iterative process, so the optimal number of flames in search space with the best moth against its flame is attained during each iteration. After each best find, it updates the moth position. A moth travels in the solution space until they have found an optimal solution or the searching operation is terminated.

In order of the  $m \times n$  solution space, the random position assigned to each moth during the initialization phase and moth array is stored as fitness values. Similarly, the flame matrix and corresponding array are generated. The moth's best value found so far is stored in the flame matrix. To find an optimal solution or terminate the search operation, moths are moved in a solution space.

This operation used the dimension of lowerbound-upperbound of the search space. Further, it is used to evaluate the fitness value of each moth based on their location in the search space. The creation of a fitness matrix is an iterative process; updated values are stored in the matrix in ascending order. For each moth, the lower fitness value is provided by the fitness matrix. The optimal best score for the flame is calculated by combining the position of the moth and its fitness value and is used to update the moth position in the search space. For optimal solution, a linear decreasing factor “ $x$ ” was used for convergence. For effective communication, the minimal number of clusters required is also obtained by using the same convergence technique.

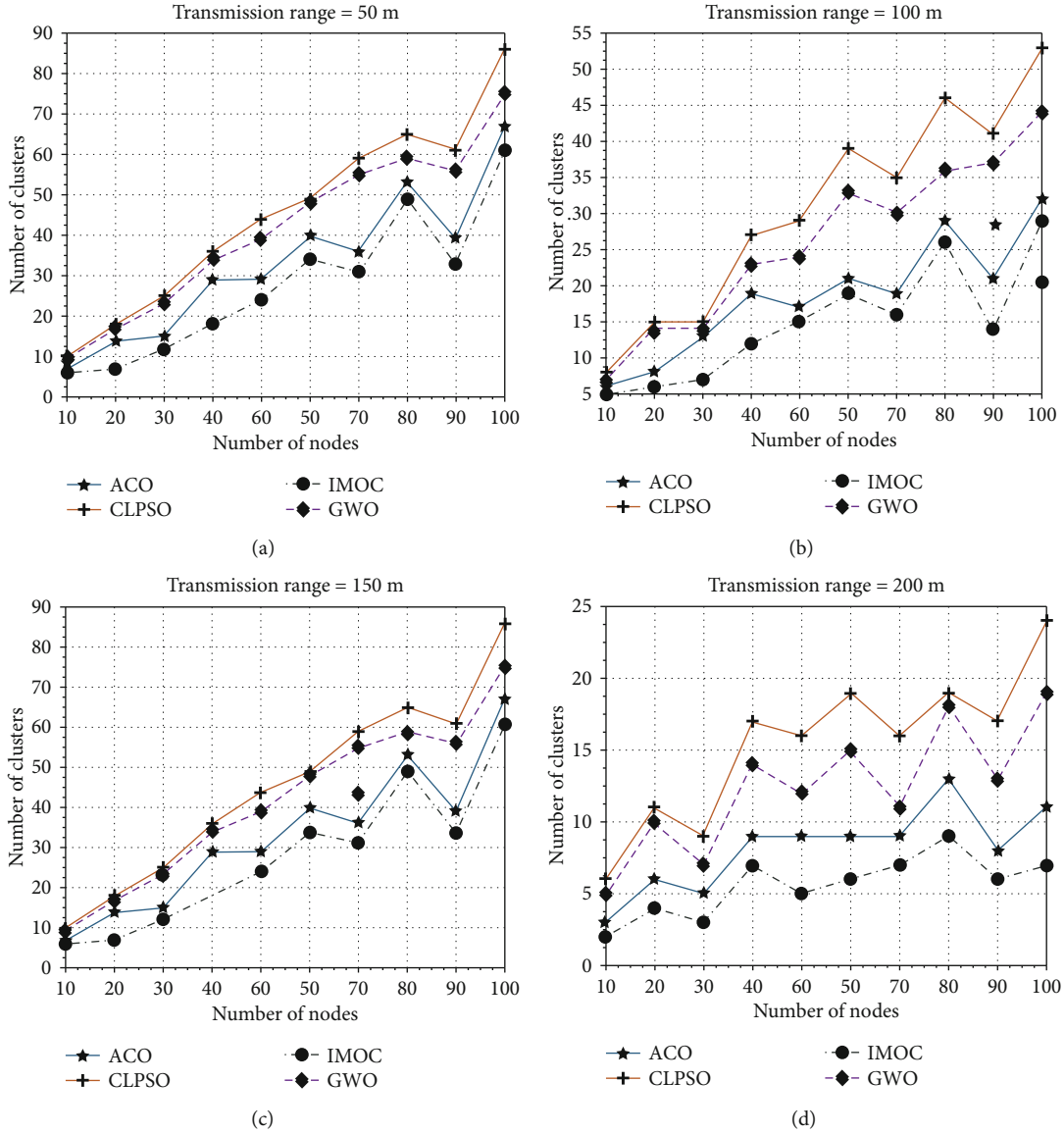


FIGURE 6: Grid\_size 1000 m × 1000 m, transmission\_ranges 50 to 200.

After creating clusters, selection of the cluster head (CH) is the next phase. Multiple parameters like grid size, node density, node connectivity, load balance factor, and transmission range are the parameters used in the CH selection process. These parameters are passed to fitness function with assigned weights. An important part of IMOC is to carry selection using a fitness function. Cluster lifetime increased by selecting the best CH resulting in minimizing network energy and limiting unnecessary broadcast overhead. The following equation (1) is used to calculate the fitness value for the IMOC algorithm:

$$\text{Fitness} = \frac{W1 \times \text{Energy\_Resi}}{(W2 \times \text{avg\_dis})(W3 \times \text{delta\_diff})}. \quad (1)$$

The residual energy of the vehicular node is denoted by  $\text{Energy\_Resi}$ , the average distance between neighboring

nodes is  $\text{avg\_dis}$ , and the load balancing factor (LBF) is considered by  $\text{delta\_diff}$ . Weight for energy is  $W1$ , the average distance is  $W2$ , and the delta difference is  $W3$ . Achieving an equal number of cluster members only results in an ideal scenario. In a real scenario, it is difficult to achieve as vehicular nodes change their positions and other parameters. The ideal node degree deviation of movement from its neighbors is computed by

$$\text{Delta\_Diff} = \text{ABS}(\text{Ideal\_Degree} - \text{Node Degree}). \quad (2)$$

Inappropriate selection of CH might result if selection criteria for CH are static and a single parameter might bias the fitness function [21–23, 38]. Depending on the scenario, weight is assigned to parameters dynamically by IMOC to counter the biasing problem and negatively impact the fitness values. In the first step, each value for the parameter normalized between the range of 0 and

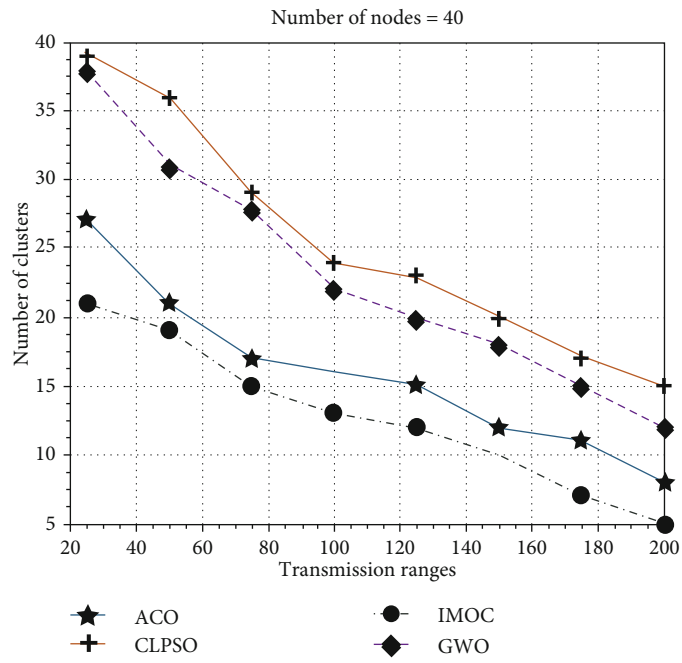
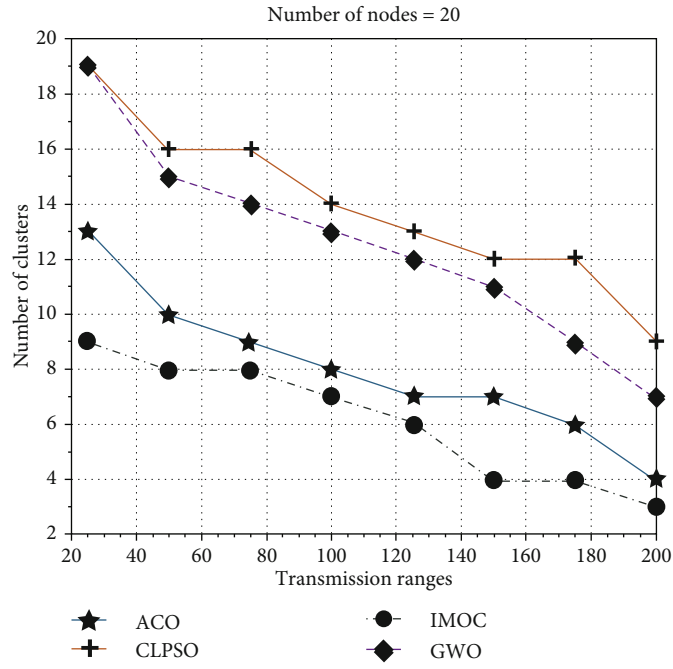
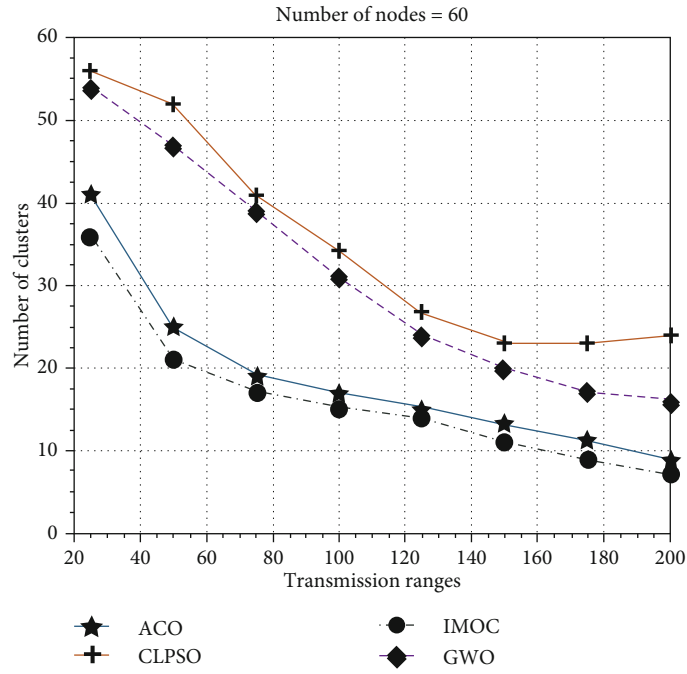
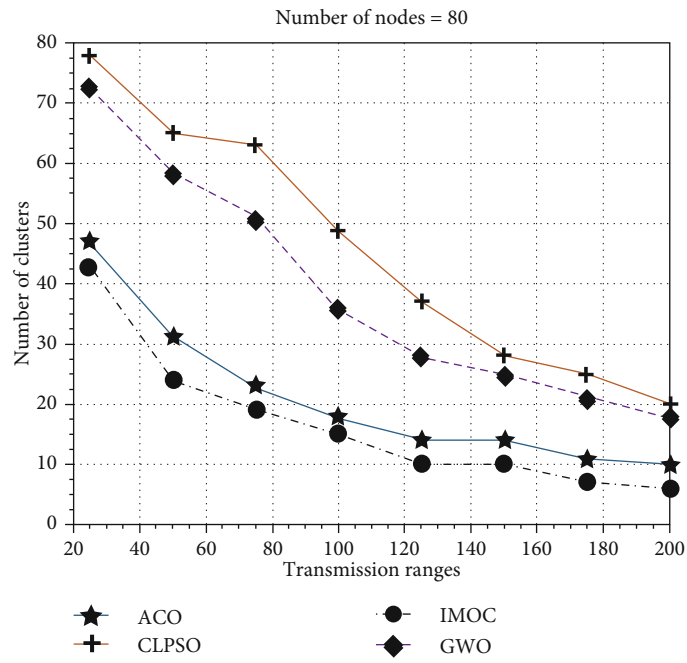


FIGURE 7: Continued.



(c)



(d)

FIGURE 7: Continued.

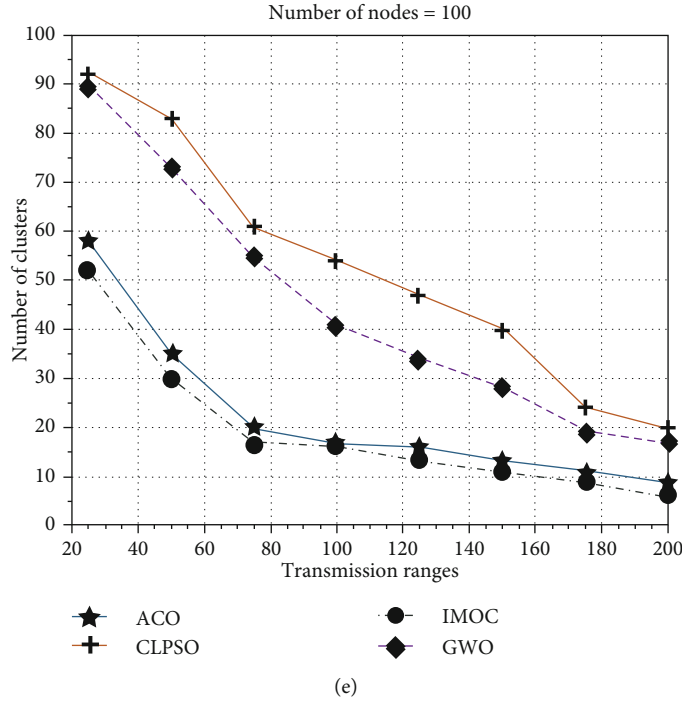


FIGURE 7: Grid\_size 1500 m × 1500 m, nodes 10 to 100.

10. In equation (3), each parameter deviation is calculated based on negative impact.

$$\text{Dev}_-(p) = \text{ABS}(\_mean - \_parameter(p)). \quad (3)$$

Penalized outlier parameters are used in equation (3), to add penalty on weirdness from their mean, and are used to compute updated values for parameters. To penalize the outlier penalty, another equation is used with

$$w(p) = \frac{1}{\text{dev}(p)}. \quad (4)$$

The aggregated total of all weight essentially is equivalent to “1.” Fitness for each node can be calculated for all parameters by equation (1).

### 3. Experimental Results and Analysis

Table 3 shows the parameter setting for simulation; the total population for each algorithm set is to be 100 and 150, and the maximum iteration for each solution is set to ten, depending on the nature of the algorithms, inertia, weight, and evaporation rate used. Inertia is set as CLPSO 0.694, GWO 0.694, and IMOC 0.90, and evaporation rate is set as 0.5 for ACO. Four grid sizes are used to perform simulations 500 m × 500 m, 1000 m × 1000 m, 1500 m × 1500 m, and 2000 m × 2000 m. The vehicle maintains a minimum distance of 1.5 m in the simulations; the interval between vehicles is set to be 2 m. Transmission ranges for all simulations are considered from 25 m to 200 m and node density from 10 to 100. An assumption is considered wherein vehicle

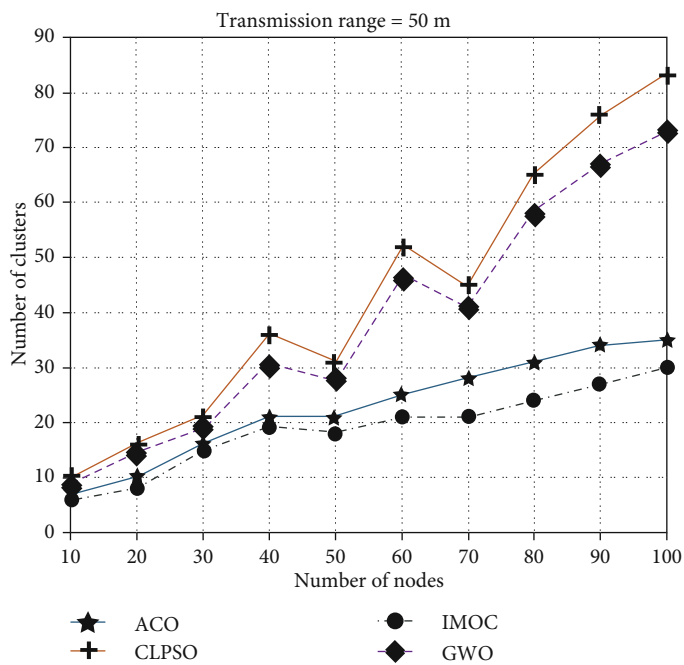
mobility remains fixed or is moved with constant velocity. IMOC experimentations were compared with ACO, GWO, and CLPSO which are some the state-of-the-art evolutionary clustering protocols.

### 4. Results and Discussion

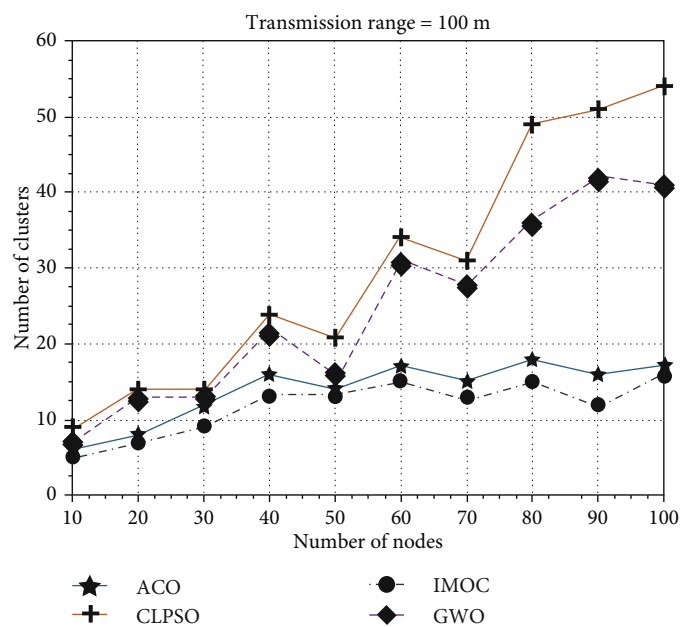
To measure the efficiency of the proposed IMOC algorithm with CLPSO, ACO, and GWO, numerous experimentations were performed. Their performance is presented in the following figures. To check the efficacy of IMOC, node density in the grid and transmission ranges for nodes were evaluated in multiple scenarios. IMOC maintained its supremacy and flexibility in the results. In Figure 3, transmission ranges for nodes were set from 25 m to 200 m keeping the grid size to 500 m × 500 m, and 10 to 100 vehicular nodes were deployed. In Figure 3(b) where the node density is 40 and the transmission size is 25 m, clusters created by CLPSO = 29, GWO = 27, and ACO = 23, but IMOC created only 19. When the transmission range was increased to 100 m, IMOC created only six clusters, in comparison to ACO 14, GWO 18, and CLPSO 21. The proposed technique showed consistent performance measure compared with the existing techniques. When the transmission range for the nodes was increased, few numbers of clusters were created. IMOC results outperformed other algorithms when nodes increased from 20, 40, 60, 80, to 100. IMOC created an optimal number of clusters for all sets of experimentation in the 500 m × 500 m grid size.

Figure 4 shows the optimal performance of IMOC in comparison with existing techniques. To strengthen this matter, an additional set of experimentation is performed,





(a)



(b)

FIGURE 8: Continued.

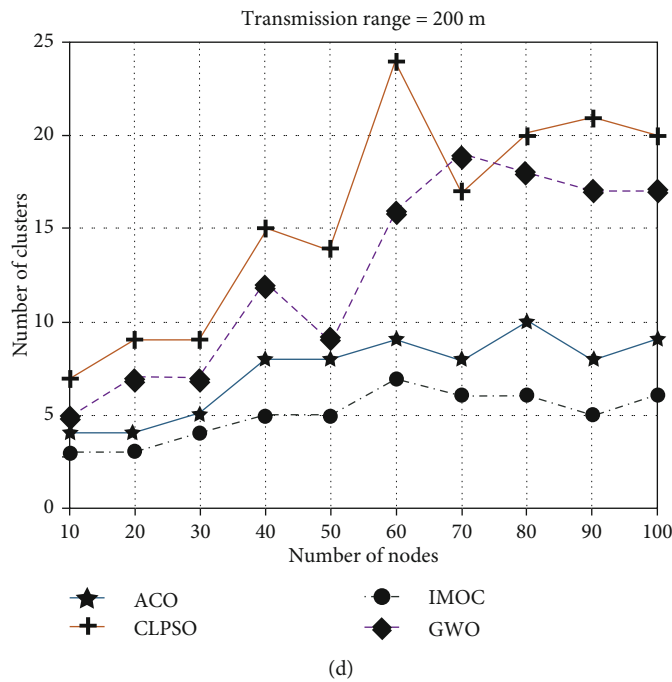
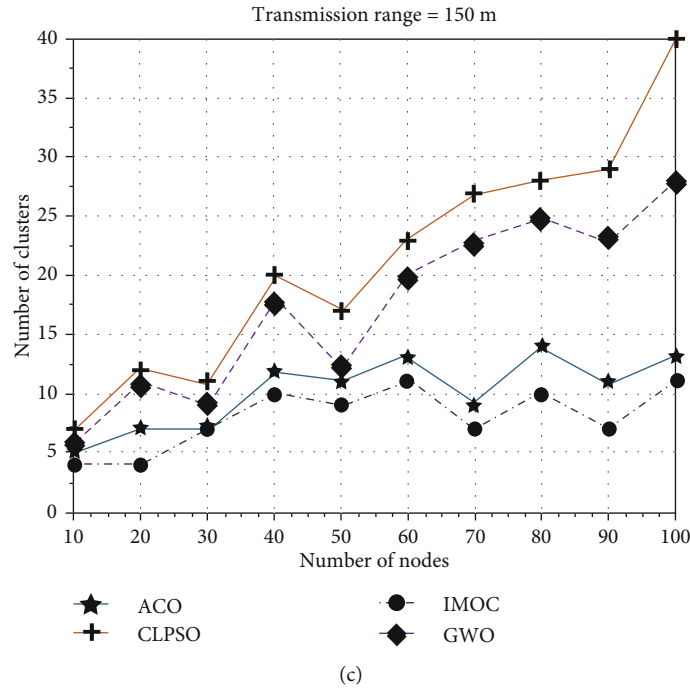
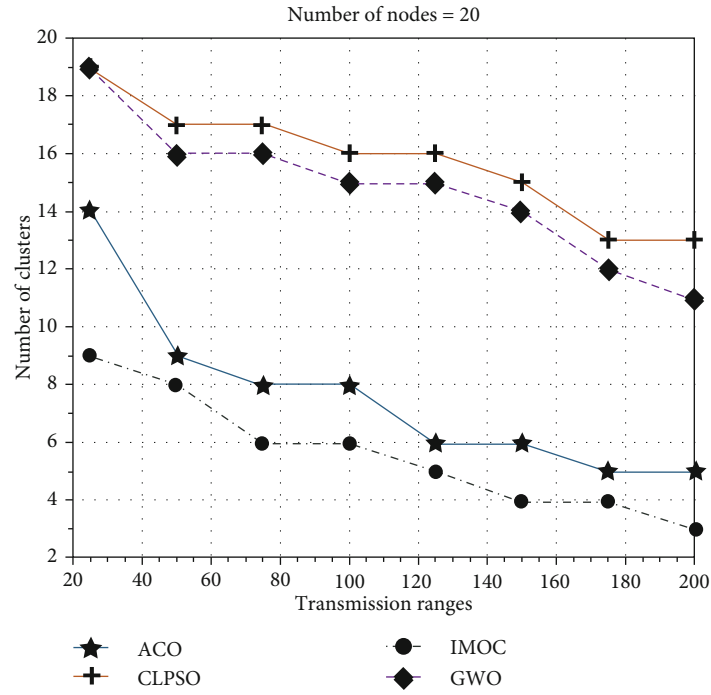


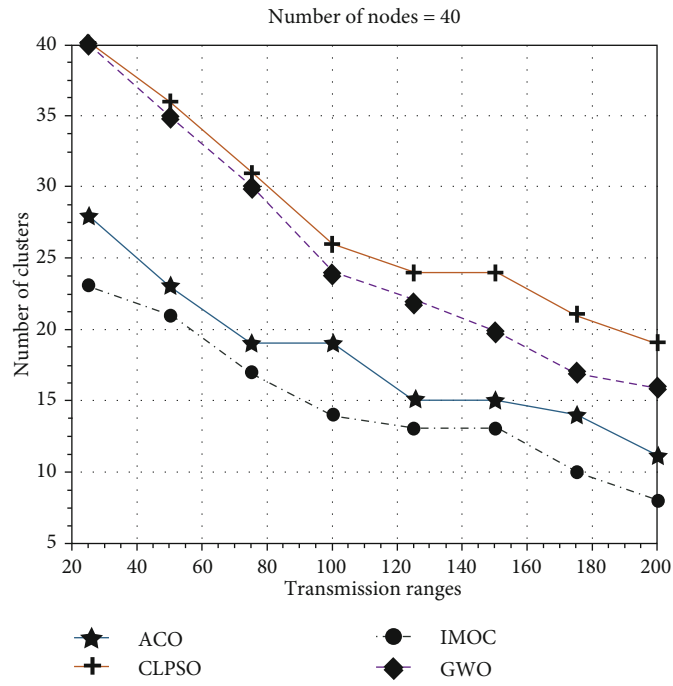
FIGURE 8: Grid\_size 1500 m × 1500 m, transmission\_ranges 50 to 200.

where nodes were fixed to 50 m and the cluster creation for grid size 500 m × 500 m was checked by varying node density from 10 to 100 nodes (Figure 4(a)). In these experiments, the tested transmission ranges were 50 m, 100 m, 150 m, and 200 m for node density from 10 to 100 nodes in Figure 4. With varying transmission ranges, it is noticed that IMOC performance was equally good and consistent; results for all instances produced were better, and ACO results were the closest to IMOC results in these settings. It was noticed that IMOC performance was also the best even if the grid size

increased to 1 km × 1 km. Figure 5 depicts results for this scenario. The experimental setting was updated, and results for node density 50 for the 75 m transmission range show that IMOC created 18 clusters in comparison with 31, 36, and 23, respectively, for GWO, CLPSO, and ACO in Figure 5(a). The minimum number of clusters was created by IMOC. In Figure 5(e), to check the results, the accuracy for node density was increased to 100 and the transmission range to 125 m; GWO created 33 clusters, CLPSO 45, and ACO 23. In comparison, IMOC created only 18 clusters.



(a)



(b)

FIGURE 9: Continued.

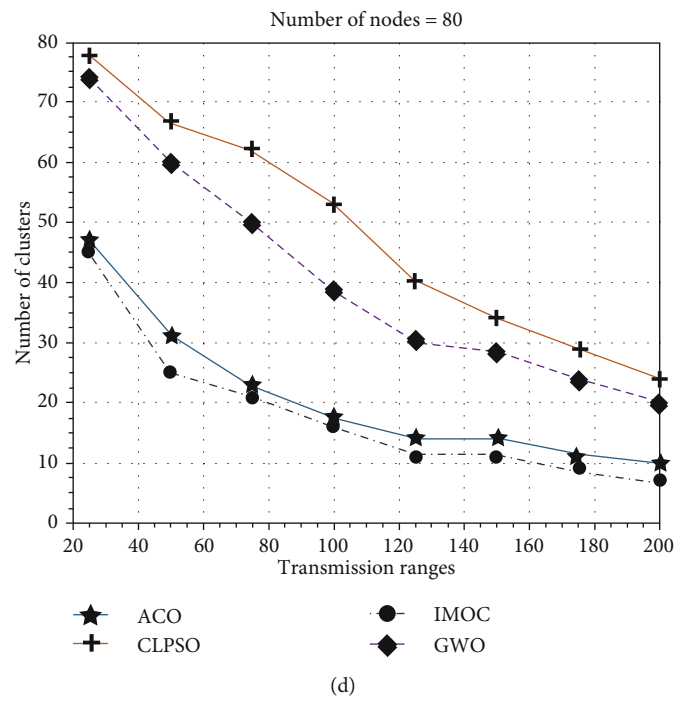
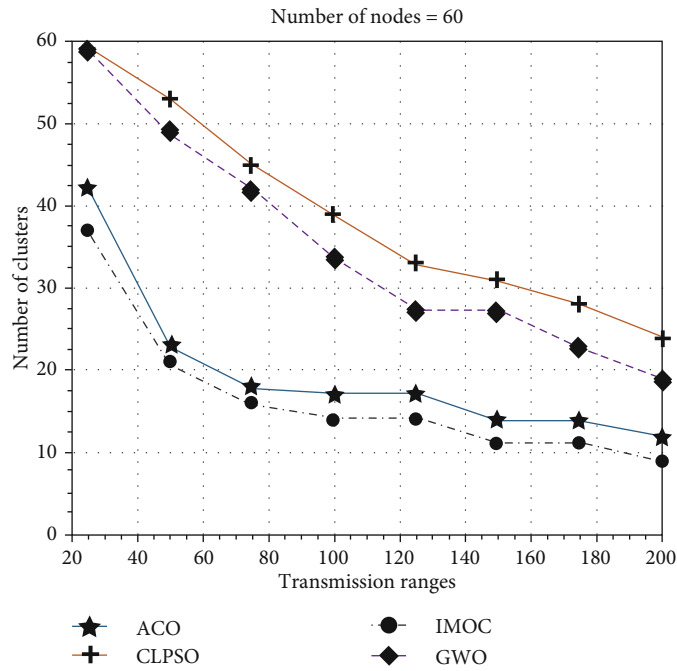


FIGURE 9: Continued.

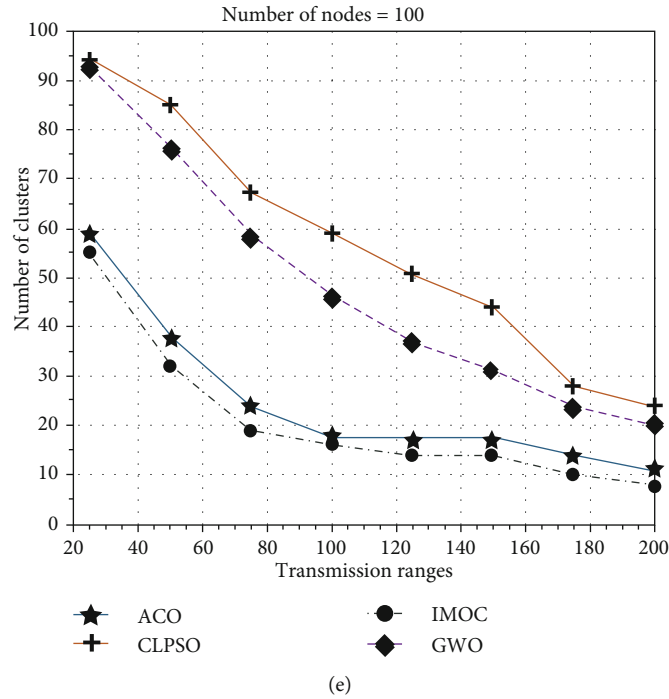


FIGURE 9: Grid\_size 2000 m  $\times$  2000 m, nodes 10 to 100.

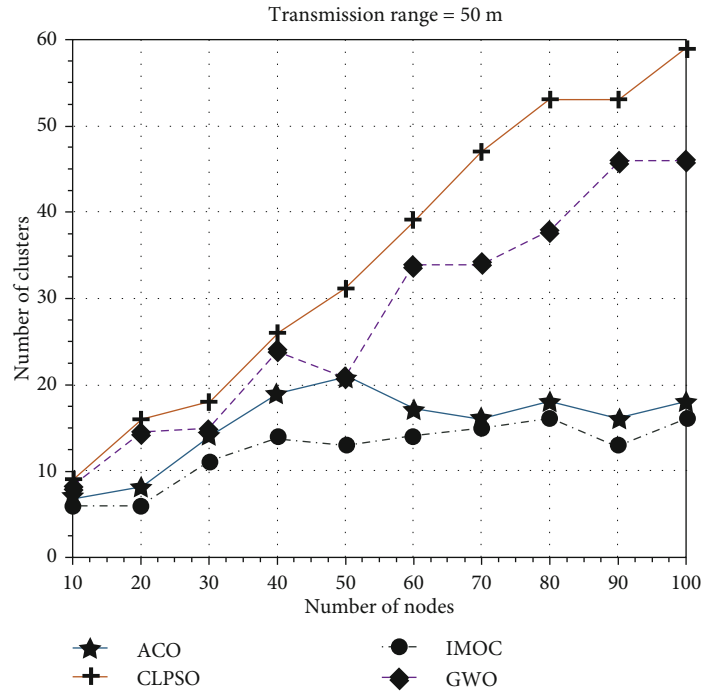
These results ensured that for any combination of nodes and transmission range for the same grid size, IMOC creates a minimum number of clusters in which its results optimized the said routing problem. The creation of a few clusters directly depends on the transmission size. The number of clusters increased with a short transmission range.

To analyze the IMOC performance with an existing algorithm, experimentations were performed with different transmission ranges. Experimentations were performed against transmission ranges of 50 m, 100 m, 150 m, and 200 m for node density from 10 to 100 nodes. Figure 6 displays the result that IMOC gives an optimal number of clusters within the 1000 m  $\times$  1000 m grid size. Simulation was performed to strengthen the results of the proposed algorithm by increasing the grid size to 1500 m  $\times$  1500 m. Figure 7 presents results for the subjected scenario for multiple nodes and transmission ranges. Figure 8(b) shows that when the transmission range is 100 and the node size 80, similar improved results for IMOC are presented.

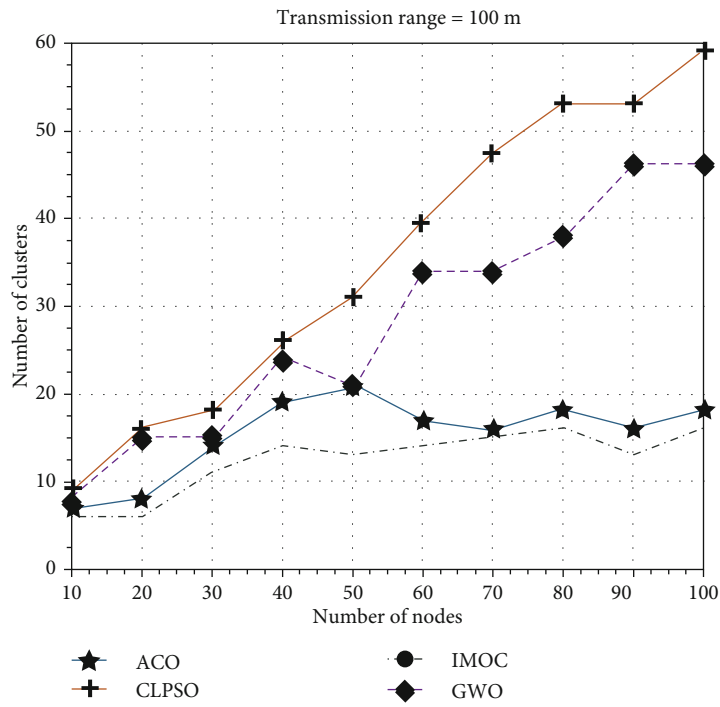
IMOC generated only 15 clusters in comparison with ACO 18, GWO 36, and CLPSO 49. In Figure 7, it can be visualized that for lower transmission range and higher transmission range, IMOC resulted in an optimal number of clusters in comparison to ACO, GWO, and CLPSO. Results have shown that IMOC performance improves for an increasing number of nodes in the grid. The efficiency and performance of IMOC remain optimal for any number of nodes and transmission range. IMOC produced 36 and 7 clusters at the transmission range of 25 m and 200 m respectively for 60 nodes, while ACO generates 41 and 9 clusters for the same transmission ranges and number of node. ACO remains the closest minimal cluster producer for the problem under observation.

In Figure 8, the performance of the proposed algorithm was tested against multiple transmission ranges by changing node densities. On each point, IMOC performance shows better results. We can conclude that the higher the transmission range, the lesser the number of clusters; Figure 8(d) seconds this statement. It shows that our proposed technique generates 4 clusters at the transmission range of 200 m for 10 nodes, while for 100 nodes, it produces only 6 clusters. The efficiency of our proposed technique IMOC increases with the node density as compared to ACO, GWO, and CLPSO. Figure 8(d) shows that when we set a transmission range of 200 m for 10 and 100 nodes, IMOC produced only four clusters for ten number of nodes and only six for 100 nodes in Figure 9, when the grid size was increased to 2000 m  $\times$  2000 m, similar behavior of IMOC was observed by generating an optimal number of clusters. It can be seen that for extreme parameter settings for lower and higher nodes and transmission ranges, IMOC performance seems optimal. For 100 nodes and transmission range of 200 in Figure 9(e), IMOC created only 8, ACO 11, GWO 21, and CLPSO 24. Simulation results prove that IMOC for any given parameter setting creates an ideal number of clusters.

Efficient results were produced by IMOC compared with ACO, GWO, and CLPSO. Figure 10 depicts the optimal number of clusters by increasing the node density and keeping transmission ranges at 50 m, 100 m, 150 m, and 200 m. For any given parameter setting, IMOC performed best in producing an optimal number of clusters. In simulations, it can be observed that IMOC provided the best solution for any given scenario by creating a minimal number of clusters. To justify the performance of IMOC clustering in a more realistic scenario, in Figures 11 and 12, results are presented in the 3D grid by keeping the simulation setting as

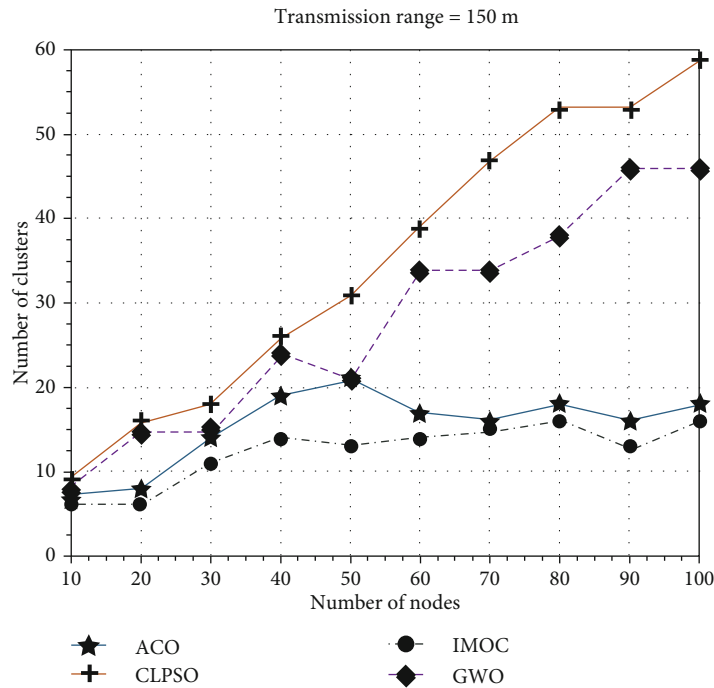


(a)

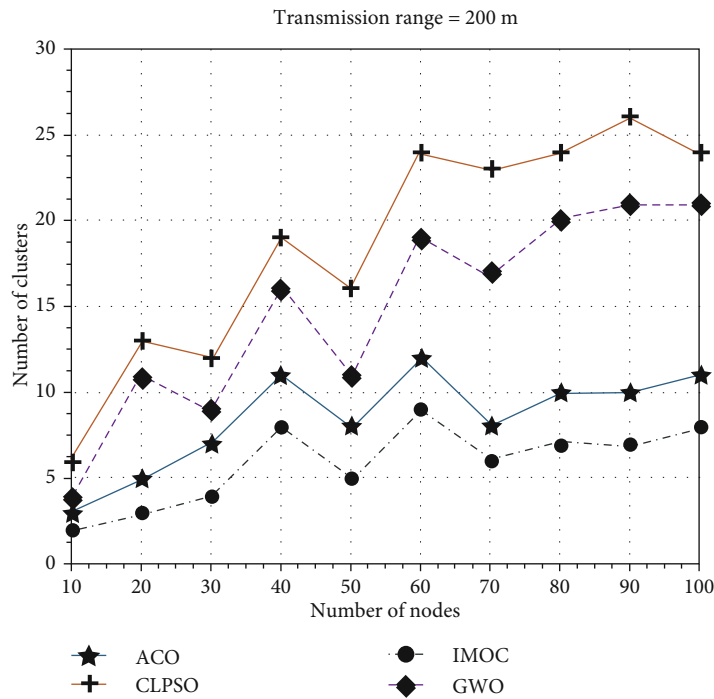


(b)

FIGURE 10: Continued.



(e)



(d)

FIGURE 10: Grid\_size 2000 m × 2000 m, transmission\_ranges 50 to 200.

transmission ranges of 50 m, 100 m, 150 m, and 200 m and node density at 40 and 80 nodes for all sets of grid sizes from 500 m × 500 m to 2000 m × 2000 m.

Figure 11 depicts the results for node 40, and Figure 12 presents the results for node 80. The result justifies the performance of IMOC as optimal at any point for any scenario. The optimal number of clusters produced by IMOC for all

grid sizes from 500 m × 500 m, 1000 m × 1000 m, 1500 m × 1500 m, and 2000 m × 2000 m.

**4.1. Load Balance Factor.** It is unrealistic to have an equal number of clusters for simulation. In some scenarios, one CH might be overloaded with a maximum number of cluster members in comparison to the cluster with a smaller number

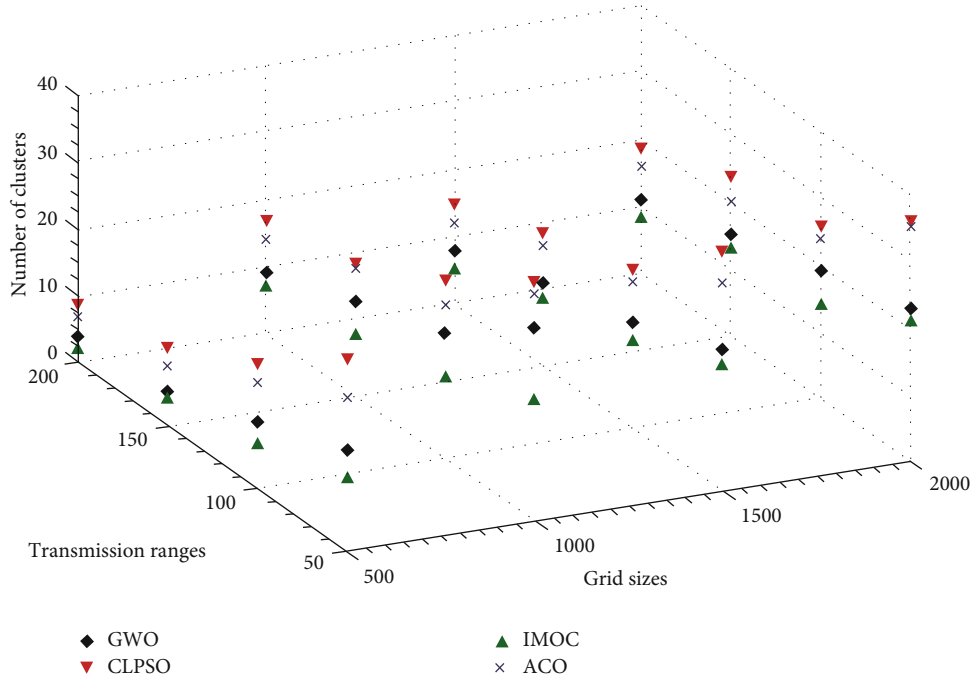


FIGURE 11: LBF for nodes 40.

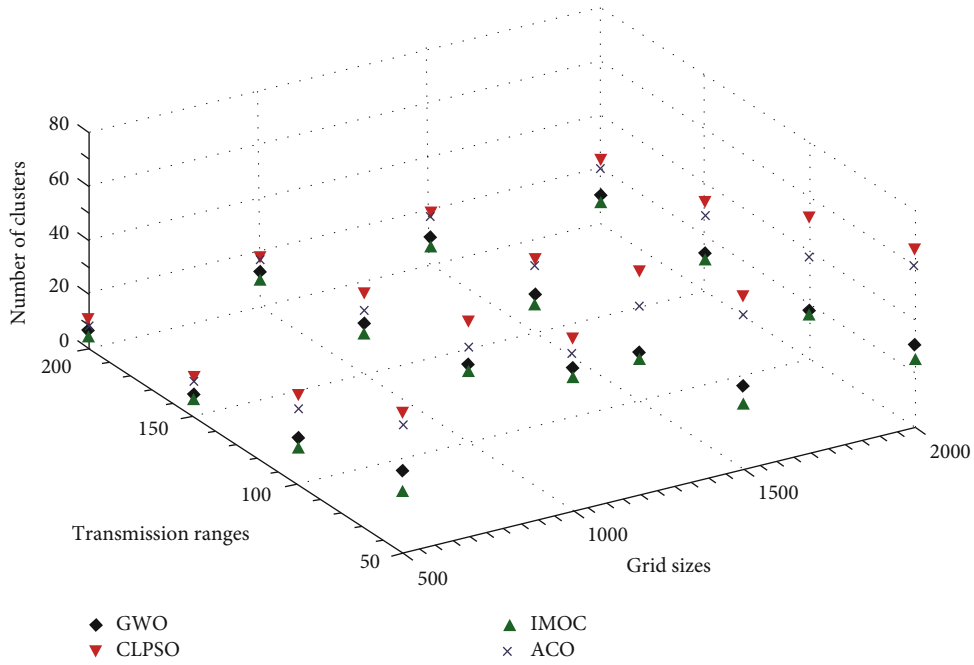


FIGURE 12: LBF for nodes 80.

of nodes. To mitigate the overloading of clusters, the load balance factor for each CH calculated by using

$$\text{Load balance factor} = \frac{1}{n_c \times \sum_i (x_i - u)^2}. \quad (5)$$

In equation (5),  $n_c$  represents the total number of  $X_i$  which is the cluster cardinality, and an average number of

CH neighbor is  $-u$ . Figure 13 and 14 are for nodes 20 and 80 with varying transmission ranges from 25 m to 200 m for all grid sizes. IMOC performance is good in the case when the CH neighbor's number reached a threshold in terms of LBF.

A framework proposed to improve the routing efficiency of IMOC with support of UAV. In Figure 15, where grid size is 500 m  $\times$  500 m for node 60, IMOC created 19 clusters. CH



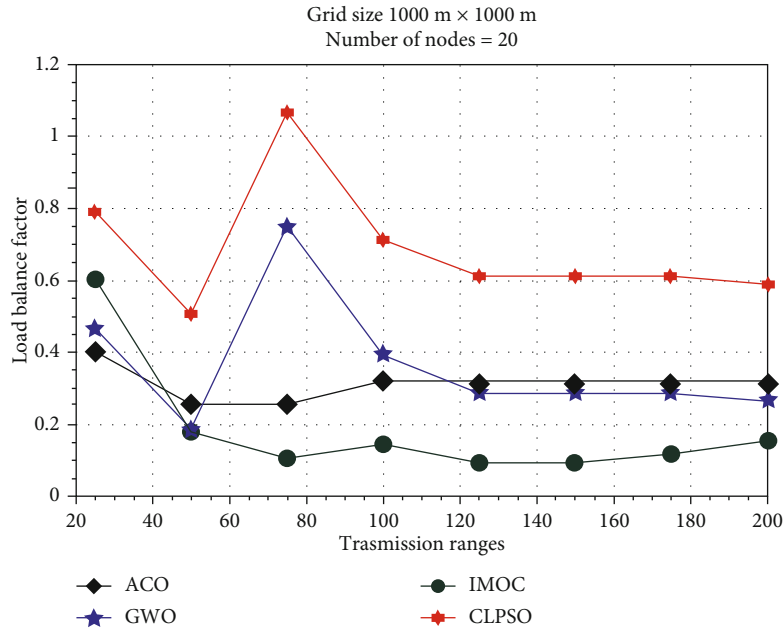


FIGURE 13: LBF for node 20.

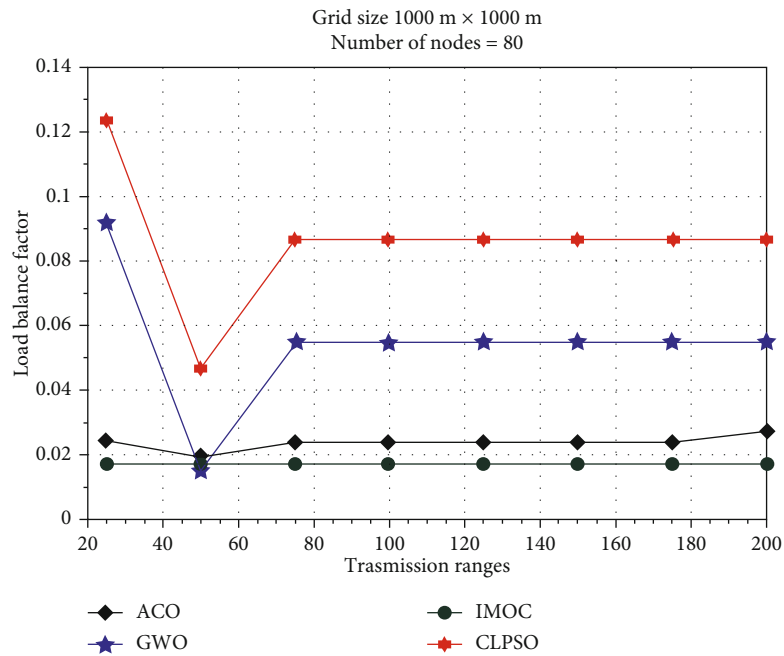


FIGURE 14: LBF For node 80.

and their associated members are depicted in Figure 15. Two kinds of communication existed between CH and cluster members (CM). Intercluster communication and intracluster communication can exist. To limit the overhead, CM cannot broadcast the message into the entire topology. Clustering ensured that CM can only send a request to its CH. In intra-cluster communication, the sender and receiver both are the members of the same CH. CM sends a request to CH to manage all services for cluster members. CH acts as

a link between the sender and receiver to provide communication services. During inter-cluster communication, the sender and receiver both are not members of the same cluster. Conventionally, the sender node sends a request to its CH, and CH broadcasts this message into the entire topology. CHs receive this message and reply against the request, for path creation between destinations by connecting CHs. The path must be live and keep track of CHs for communication between the sender and receiver. In this scenario, hop count

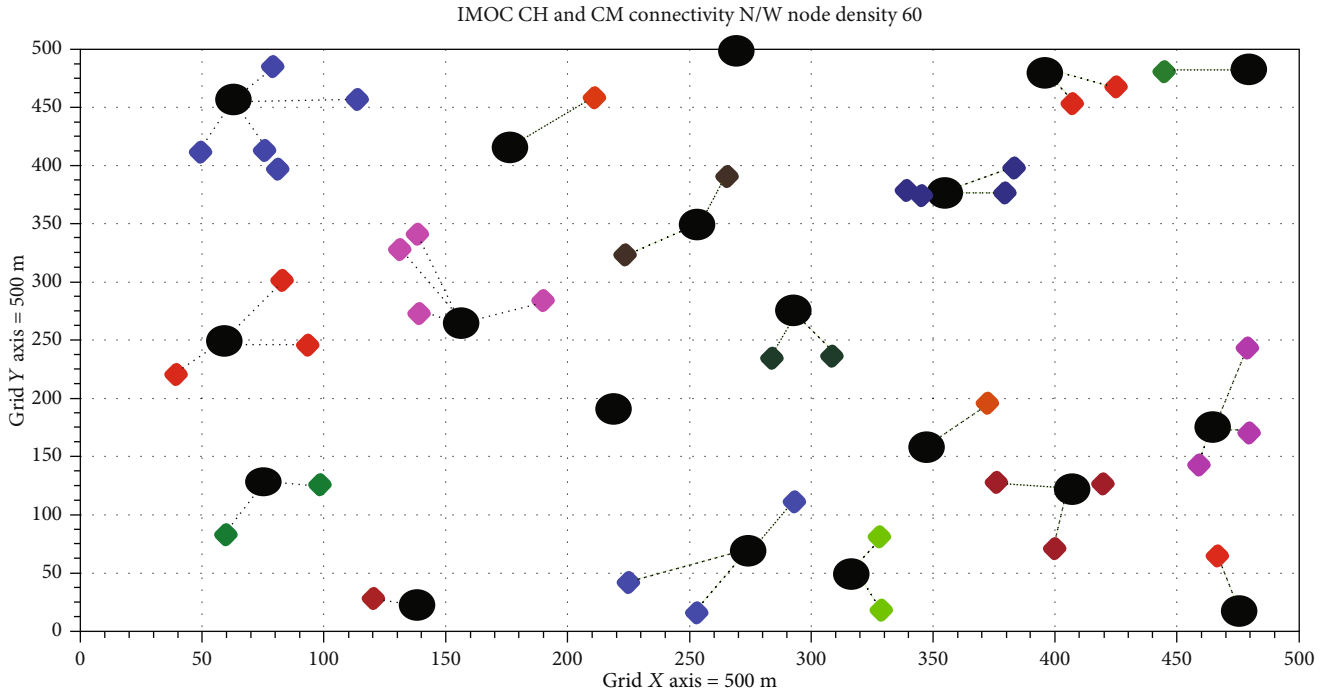


FIGURE 15: IMOC CH and cluster member association (500 m × 500 m).

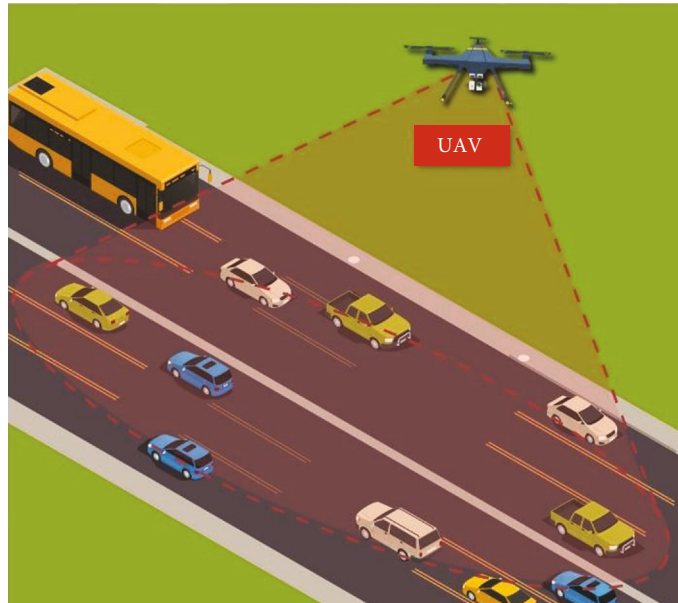


FIGURE 16: UAV and CH connectivity.

for multiple transmission varies from 1 to 18, depending upon the CH involved in communication. This will add additional computational complexity in routing. We proposed a solution to keep constant the hop count for all communications. Figures 16 and Figures 17 show that in the center of a topology-deployed AN (drone).

The complete grid is in the range of AN. All CHs are in the range of AN and listed. To reduce the intercluster communication problem and path construction and maintain it

during communication are overhead between the sender and receiver. Considering conventional path construction based on broadcast request messages from the sender to its neighboring nodes, node reply to this request is the path members. In this problem, if the sender wants to send data outside the cluster, CH sends a request to AN, and AN sends a message to all CHs. Destination CH responds against AN request. The path will be established between the sender's CH to AN to destination CH. For all nodes, the hop count

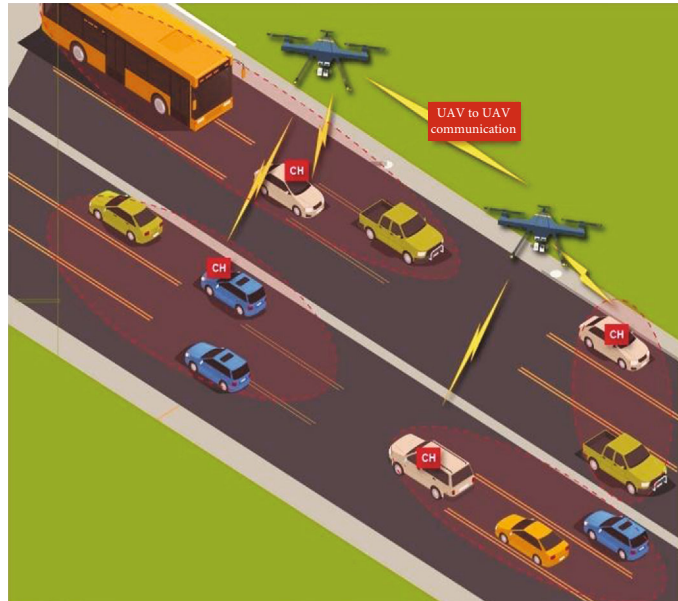


FIGURE 17: UAV assistance to VANET.

will be constant as two hop counts. As a result, unnecessary broadcast overhead is reduced for intercluster communication. This also improves the efficiency of the network. Clustering provides an optimal number of clusters to minimize broadcasting for intracluster communication. Additionally, AN with the IMOC clustering solution limits unnecessary broadcasting and improves network performance for intercluster communication.

## 5. Conclusion

To solve the VANET routing problem, IMOC solution presented an evolutionary algorithm based on cluster optimization. The MFO technique is used to find near-optimal solutions in search space. The IMOC algorithms work iteratively to find solutions from search spaces. The IMOC is an efficient algorithm for VANETs as a minimal number of clusters are achieved. It reduced unnecessary broadcasting and helped in minimizing the routing cost. The routing cost decreased as the near-optimal number of clusters was obtained from the search space. The VANET support in clustering also improved routing performance, by restricting unnecessary broadcasts and keeping the hop count constant for all communications. The proposed IMOC algorithm's efficiency was evaluated by performing a diverse set of simulations while varying topological parameters. To validate the optimization results, simulations were performed and monitored with multiple node densities in the search space with variable transmission ranges of vehicular nodes. In the said topological constraints, IMOC presented near-optimal solutions and creates a minimal number of clusters in the search space. Result comparison of renowned evolutionary algorithms, GWO, CLPSO, and ACO with IMOC shows that the proposed algorithm is the best solution for the problems under observation.

## Data Availability

No dataset is utilized during the experimentation process.

## Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

## References

- [1] C. Cooper, D. Franklin, M. Ros, F. Safaei, and M. Abolhasan, "A comparative survey of VANET clustering techniques," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 1, pp. 657–681, 2017.
- [2] J. Nzouonta, N. Rajgure, G. Wang, and C. Borcea, "VANET routing on city roads using real-time vehicular traffic information," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 7, pp. 3609–3626, 2009.
- [3] O. S. Oubbati, A. Lakas, F. Zhou, M. Güneş, N. Lagraa, and M. B. Yagoubi, "Intelligent UAV-assisted routing protocol for urban VANETs," *Computer Communications*, vol. 107, pp. 93–111, 2017.
- [4] M. A. Khan, A. Safi, I. M. Qureshi, and I. U. Khan, "Flying ad-hoc networks (FANETs): a review of communication architectures, and routing protocols," in *2017 First International Conference on Latest trends in Electrical Engineering and Computing Technologies (INTELLECT)*, pp. 1–9, Karachi, Pakistan, 2017.
- [5] R. C. Biradar and S. S. Manvi, "Reliable ring based multicast routing scheme in MANET: an agent based approach," in *2009 IEEE International Conference on Automation Science and Engineering*, pp. 507–512, Bangalore, India, 2009.
- [6] L. Junhai, Y. Danxia, X. Liu, and F. Mingyu, "A survey of multicast routing protocols for mobile ad-hoc networks," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 1, pp. 78–91, 2009.

- [7] A. Zhou, B.-Y. Qu, H. Li, S.-Z. Zhao, P. N. Suganthan, and Q. Zhang, "Multiobjective evolutionary algorithms: a survey of the state of the art," *Swarm and Evolutionary Computation*, vol. 1, no. 1, pp. 32–49, 2011.
- [8] A. Trivedi, D. Srinivasan, K. Sanyal, and A. Ghosh, "A survey of multiobjective evolutionary algorithms based on decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 21, pp. 440–462, 2017.
- [9] N. Maslekar, M. Boussedjra, J. Mouzna, and H. Labiod, "A stable clustering algorithm for efficiency applications in VANETs," in *2011 7th International Wireless Communications and Mobile Computing Conference*, pp. 1188–1193, Istanbul, Turkey, 2011.
- [10] S. J. Nanda and G. Panda, "A survey on nature inspired metaheuristic algorithms for partitional clustering," *Swarm and Evolutionary Computation*, vol. 16, pp. 1–18, 2014.
- [11] A. Daeinabi, A. G. P. Rahbar, and A. Khademzadeh, "VWCA: an efficient clustering algorithm in vehicular ad hoc networks," *Journal of Network and Computer Applications*, vol. 34, no. 1, pp. 207–222, 2011.
- [12] N. Kumar, N. Chilamkurti, and J. H. Park, "ALCA: agent learning-based clustering algorithm in vehicular ad hoc networks," *Personal and Ubiquitous Computing*, vol. 17, no. 8, pp. 1683–1692, 2013.
- [13] B. R. Senapati and P. M. Khilar, "Optimization of performance parameter for vehicular ad-hoc NETWORK (VANET) using swarm intelligence," in *Nature Inspired Computing for Data Science*, pp. 83–107, Springer, 2020.
- [14] K. D. Frank, C. Rich, and T. Longcore, "Effects of artificial night lighting on moths," in *Ecological Consequences of Artificial Night Lighting*, pp. 305–344, Island Press, 2006.
- [15] P. Garg and A. Gupta, "Optimized open shortest path first algorithm based on moth flame optimization," *Indian Journal of Science and Technology*, vol. 9, p. 48, 2017.
- [16] S. Mirjalili, "Moth-flame optimization algorithm: a novel nature-inspired heuristic paradigm," *Knowledge-Based Systems*, vol. 89, pp. 228–249, 2015.
- [17] S. Jinyuan, Z. Chi, and F. Yuguang, "An ID-based framework achieving privacy and non-repudiation in Vehicular Ad Hoc Networks," in *MILCOM 2007 - IEEE Military Communications Conference*, pp. 1–7, Orlando, FL, USA, 2007.
- [18] D. Zhu, G. Cui, and J. Huang, "The design of scheduling scheme ADCSA for mobile wireless sensor networks," *The Journal of Computer Information Systems*, vol. 8, pp. 7607–7618, 2012.
- [19] J. Wu, M. Fang, H. Li, and X. Li, "RSU-assisted traffic-aware routing based on reinforcement learning for urban Vanets," *IEEE Access*, vol. 8, pp. 5733–5748, 2020.
- [20] A. Dahiya and R. Chauhan, "A comparative study of MANET and VANET environment," *Journal of Computing*, vol. 2, pp. 87–92, 2010.
- [21] B. Paul, M. Ibrahim, M. Bikas, and A. Naser, "Vanet routing protocols: pros and cons," 2012, <https://arxiv.org/abs/1204.1201>.
- [22] T. Clausen and P. Jacquet, "Rfc3626: optimized link state routing protocol (olsr)," *Experimental*, vol. 51, 2003 <https://www.ietf.org/rfc/rfc3626.txt>.
- [23] C. Perkins, E. Belding-Royer, and S. Das, *Ad hoc on-demand distance vector (AODV) routing*, University of Cincinnati, 2003.
- [24] N. S. M. Usop, A. Abdullah, and A. F. A. Abidin, "Performance evaluation of AODV, DSDV & DSR routing protocol in grid environment," *IJCSNS International Journal of Computer Science and Network Security*, vol. 9, pp. 261–268, 2009.
- [25] A. K. Gupta, H. Sadawarti, and A. K. Verma, "Performance analysis of AODV, DSR & TORA routing protocols," *International Journal of Engineering and Technology*, vol. 2, p. 226, 2010.
- [26] D. Cao, Y. Liu, X. Ma et al., "A relay-node selection on curve road in vehicular networks," *IEEE Access*, vol. 7, pp. 12714–12728, 2019.
- [27] S. Giordano and I. Stojmenovic, "Position based routing algorithms for ad hoc networks: a taxonomy," in *Ad hoc wireless networking*, pp. 103–136, Springer, 2004.
- [28] S. A. Rao, M. Pai, M. Boussedjra, and J. Mouzna, "GPSR-L: greedy perimeter stateless routing with lifetime for VANETS," in *2008 8th International Conference on ITS Telecommunications*, pp. 299–304, Phuket, Thailand, 2008.
- [29] Y. Yu, R. Govindan, and D. Estrin, *Geographical and energy aware routing: a recursive data dissemination protocol for wireless sensor networks*, UCLA Computer Science Department Technical Report, 2001.
- [30] J. Bernsen and D. Manivannan, "Greedy routing protocols for vehicular ad hoc networks," in *2008 International Wireless Communications and Mobile Computing Conference*, pp. 632–637, Crete Island, Greece, 2008.
- [31] F. Li and Y. Wang, "Routing in vehicular ad hoc networks: a survey," *IEEE Vehicular Technology Magazine*, vol. 2, no. 2, pp. 12–22, 2007.
- [32] S. Rosati, K. Kruzelecki, G. Heitz, D. Floreano, and B. Rimoldi, "Dynamic routing for flying ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 65, pp. 1690–1700, 2015.
- [33] W. Shi, H. Zhou, J. Li, W. Xu, N. Zhang, and X. Shen, "Drone assisted vehicular networks: architecture, challenges and opportunities," *IEEE Network*, vol. 32, no. 3, pp. 130–137, 2018.
- [34] B. Alzahrani, O. S. Oubbati, A. Barnawi, M. Atiquzzaman, and D. Alghazzawi, "UAV assistance paradigm: state-of-the-art in applications and challenges," *Journal of Network and Computer Applications*, vol. 166, article 102706, 2020.
- [35] B. Yin and X. Wei, "Communication-efficient data aggregation tree construction for complex queries in IoT applications," *IEEE Internet of Things Journal*, vol. 6, pp. 3352–3363, 2018.
- [36] S. Kumar and A. Bansal, "Performance investigation of topology-based routing protocols in flying ad-hoc networks using NS-2," in *IoT and Cloud Computing Advancements in Vehicular Ad-Hoc Networks*, pp. 243–267, IGI Global, 2020.
- [37] M. Moorkamp, J.-L. Wybo, and E.-H. Kramer, "Pioneering with UAVs at the battlefield: the influence of organizational design on self-organization and the emergence of safety," *Safety Science*, vol. 88, pp. 251–260, 2016.
- [38] I. Bekmezci, O. K. Sahingoz, and Ş. Temel, "Flying ad-hoc networks (FANETs): a survey," *Ad Hoc Networks*, vol. 11, no. 3, pp. 1254–1270, 2013.
- [39] O. K. Sahingoz, "Networking models in flying ad-hoc networks (FANETs): concepts and challenges," *Journal of Intelligent & Robotic Systems*, vol. 74, no. 1-2, pp. 513–527, 2014.
- [40] O. S. Oubbati, A. Lakas, M. Güneş, F. Zhou, and M. B. Yagoubi, "UAV-assisted reactive routing for urban VANETs," in *Proceedings of the Symposium on Applied Computing*, pp. 651–653, New York, NY, USA, 2017.

- [41] Y. Zhou, N. Cheng, N. Lu, and X. S. Shen, "Multi-UAV-aided networks: aerial-ground cooperative vehicular networking architecture," *IEEE Vehicular Technology Magazine*, vol. 10, pp. 36–44, 2015.
- [42] O. S. Oubbati, N. Chaib, A. Lakas, P. Lorenz, and A. Rachedi, "UAV-assisted supporting services connectivity in urban VANETs," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 4, pp. 3944–3951, 2019.
- [43] O. S. Oubbati, N. Chaib, A. Lakas, S. Bitam, and P. Lorenz, "U2RV: UAV-assisted reactive routing protocol for VANETs," *International Journal of Communication Systems*, vol. 33, article e4104, 2019.
- [44] H. Sedjelmaci, M. A. Messous, S. M. Senouci, and I. H. Brahmi, "Toward a lightweight and efficient UAV-aided VANET," *Transactions on Emerging Telecommunications Technologies*, vol. 30, article e3520, 2019.
- [45] W. Li, X. Liu, X. Ma, X. Wang, and Y. Zhou, "UAV-aided data delivery scheme based on opportunistic virtual intersections for smart transportation networks," *Journal of Advanced Transportation*, vol. 2019, Article ID 1576908, 11 pages, 2019.
- [46] Q. Wu, H. Wang, X. Li, B. Zhang, and J. Peng, "Reinforcement learning-based anti-jamming in networked UAV radar systems," *Applied Sciences*, vol. 9, no. 23, p. 5173, 2019.
- [47] S. A. Hadiwardoyo, J.-M. Dricot, C. T. Calafate, J.-C. Cano, E. Hernandez-Orallo, and P. Manzoni, "UAV mobility model for dynamic UAV-to-car communications," in *Proceedings of the 16th ACM International Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, & Ubiquitous Networks*, pp. 1–6, New York, NY, USA, 2019.
- [48] L. Xiao, X. Lu, D. Xu, Y. Tang, L. Wang, and W. Zhuang, "UAV relay in VANETs against smart jamming with reinforcement learning," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 5, pp. 4087–4097, 2018.
- [49] J. Wang, Y. Gao, W. Liu, W. Wu, and S.-J. Lim, "An asynchronous clustering and mobile data gathering schema based on timer mechanism in wireless sensor networks," *Computers, Materials & Continua*, vol. 58, no. 3, pp. 711–725, 2019.
- [50] D. Cao, Y. Jiang, J. Wang et al., "ARNs: adaptive relay-node selection method for message broadcasting in the internet of vehicles," *Sensors*, vol. 20, no. 5, p. 1338, 2020.