

An Evolutionary Algorithm-Based Vehicular Clustering Technique for VANETs

YASER ALI SHAH¹, FARHAN AADIL¹, AMAAD KHALIL², MUHAMMAD ASSAM³,
IBRAHIM ABUNADI⁴, (Member, IEEE), ALA SALEH ALLUHAIDAN⁵, AND FAHD N. AL-WESABI⁶

¹Department of Computer Science, COMSATS University Islamabad, Attock 43600, Pakistan

²Department of Computer Systems Engineering, University of Engineering and Technology, Peshawar 25000, Pakistan

³College of Computer Science and Technology, Zhejiang University, Hangzhou 310027, China

⁴Department of Information Systems, Prince Sultan University, Riyadh 11586, Saudi Arabia

⁵Department of Information Systems, College of Computer and Information Sciences, Princess Nourah Bint Abdulrahman University, Riyadh 11671, Saudi Arabia

⁶Department of Computer Science, College of Science and Arts, King Khalid University, Abha 62529, Saudi Arabia

Corresponding authors: Yaser Ali Shah (yasir@cuiatk.edu.pk) and Fahd N. Al-Wesabi (falwesabi@kku.edu.sa)

This work was supported in part by the Deanship of Scientific Research, King Khalid University, under Grant RGP2/209/42; in part by the Princess Nourah Bint Abdulrahman University Researchers through the Princess Nourah Bint Abdulrahman University, Riyadh, Saudi Arabia; in part by the Princess Nourah Bint Abdulrahman University Researchers Supporting Project PNURSP2022R234; and in part by the Prince Sultan University (by paying the Article Processing Charges (APC) of this publication).

ABSTRACT Many precious lives are lost world-wide due to road accidents. To counter this issue, the ultimate solution is Vehicular ad hoc networks (VANETs). Due to the high mobility of vehicles and varying network topology in VANETs, efficient communication among the vehicular nodes is of extreme importance. To enhance communication proficiency in VANETs, clustering is a renowned procedure. Therefore, a clustering algorithm based on Moth-Flame Optimization (MFO), titled AMONET, is projected which can effectively work in high mobility nodes scenario of VANETs. AMONET is based on bio inspired procedure and creates optimized clusters for reliable and efficient communication. Our algorithm is assessed experimentally with well-known procedures such as Ant Colony Optimization (ACO), Comprehensive Learning Particle Swarm Optimization (CLPSO) and Multi Objective Particle Swarm Optimization (MOPSO). Several experiments are conducted to measure the comparative efficiency of these procedures. The average cumulative results for all the grid sizes are 27.1% for AMONET while 36.3% for ACO, 54.9% for CLPSO and 58.7% for MOPSO. The results signify that AMONET produces near ideal results, covers the entire network and generates least number of clusters. It is an efficient technique to accomplish vehicular clustering with the purpose of improving the network's overall performance and consequently reducing the routing cost of the vehicular network.

INDEX TERMS Clustering, evolutionary algorithms, intelligent transportation system (ITS), moth-flame optimization (MFO), swarm-based algorithm, vehicular ad hoc networks (VANETs).

I. INTRODUCTION

Accidents are one of the major reasons behind sudden deaths worldwide. The Global status report of 2018 on safety of roads [1], providing statistics of 180 countries, shows that the number of deaths due to road accidents globally, has soared to 1.35 million annually. The USA's Fatality Analysis Reporting System (FARS) [2], indicates 35,092 people lost their lives due to 32,166 fatal automobile accidents in the United States only in 2015. National economies are massively affected by road accidents and crashes. These comprises of

The associate editor coordinating the review of this manuscript and approving it for publication was Hisao Ishibuchi.

medical costs, damage to the property and cost on emergency/police services. The present traffic system has many problems. It has accident risks, problem of congestion. It is more time consuming to reach a specific destination and the travel cost is higher. So, to avoid all these issues, VANETs are the decisive solution. VANETs come in the scope of Intelligent Transportation Systems (ITS). Within the past few years VANETs are reasonably a vibrant area amongst researchers. Numerous research papers have been published having recommendations, plug-ins, theoretical evaluation of the concerns/problems involved, simulation outcomes and enhancements in order to improve current techniques and algorithms [3].

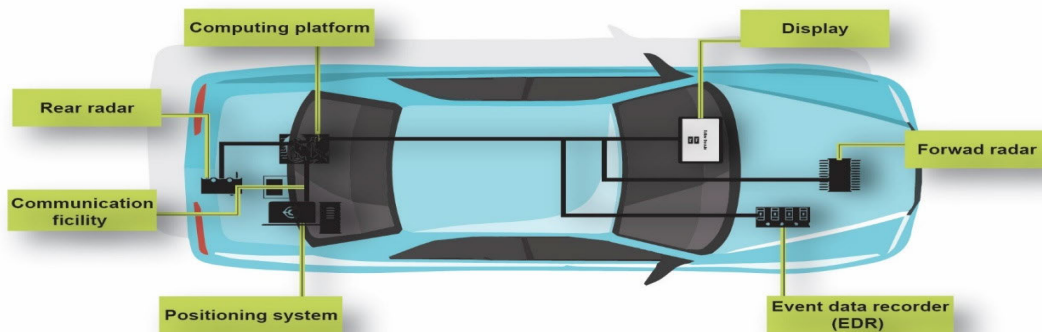


FIGURE 1. Smart vehicle architecture.

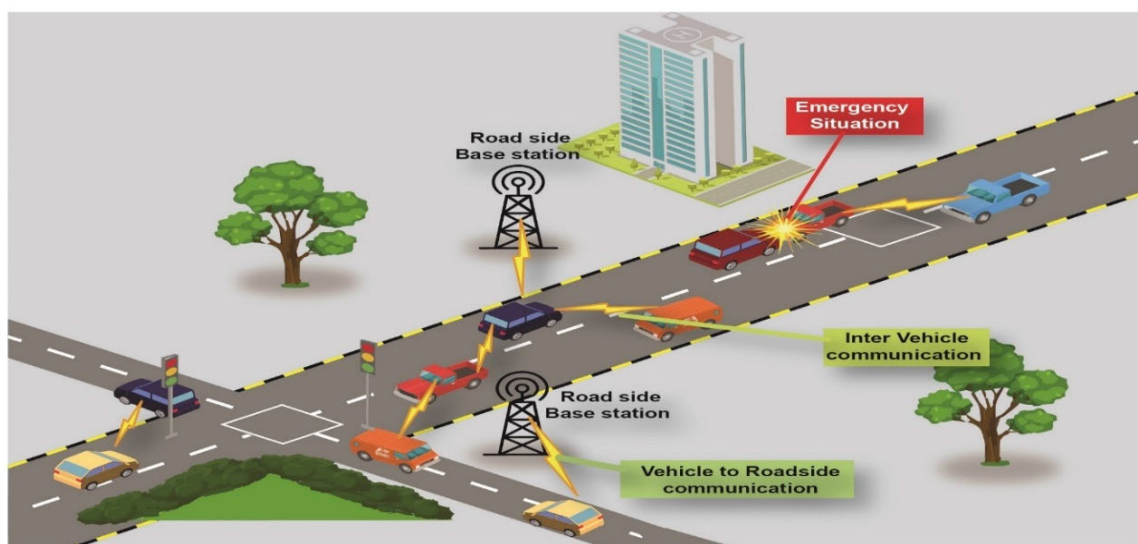


FIGURE 2. Infrastructure of VANETs.

Nowadays automobiles are no more the conventional mechanical devices that we once recognized. Vehicles have got smart having numerous sensors that can measure diverse attributes. A smart vehicle will carry the desired gadgets on smart roads. A smart vehicle is depicted in Figure 1. It comprises of numerous sensors like forward and rear radar, and a Global Positioning System (GPS) An event data recorder (EDR) device also exists whose working is analogous to the airplane’s black box. Figure 2 presents a simple infrastructure of VANETs. In the coming years numerous innovative applications will emerge in VANETs which will support V2V and V2I communications. Numerous applications linked with safety of roads, traffic management and infotainment are the focal areas of research.

Scalability is a research field of noteworthy importance for network designers. New algorithms and procedures must be formulated which considers vehicle’s movement patterns. Vehicles can communicate with other vehicles and infrastructure to transfer information within a VANETs scenario.

Vehicles collect diverse kinds of information for instance traffic/road conditions and the information is transmitted to the desired units. However, it’s not a simple job to direct the information to the anticipated destination as the vehicles move in high speed and may travel in opposite directions. Clustering/platooning of vehicles is an ultimate answer to tackle this problem.

Vehicular clustering is the core interest of this research work. Clusters are produced by grouping the nodes that are present in a similar geographical neighborhood. The network become more scalable [4], [5]. Numerous clustering techniques are present in literature [6], [7]. One of the fundamental objectives of a clustering algorithm is the stability of the clustering algorithm. The maintenance cost of cluster is very high, so forming stable clusters is exceptionally important. Cluster stability can be achieved if cluster head (CH) changes are rare or occasional [8]. Therefore, selecting the most appropriate node as the CH can ensure a stable cluster. Different procedures have different solutions as the

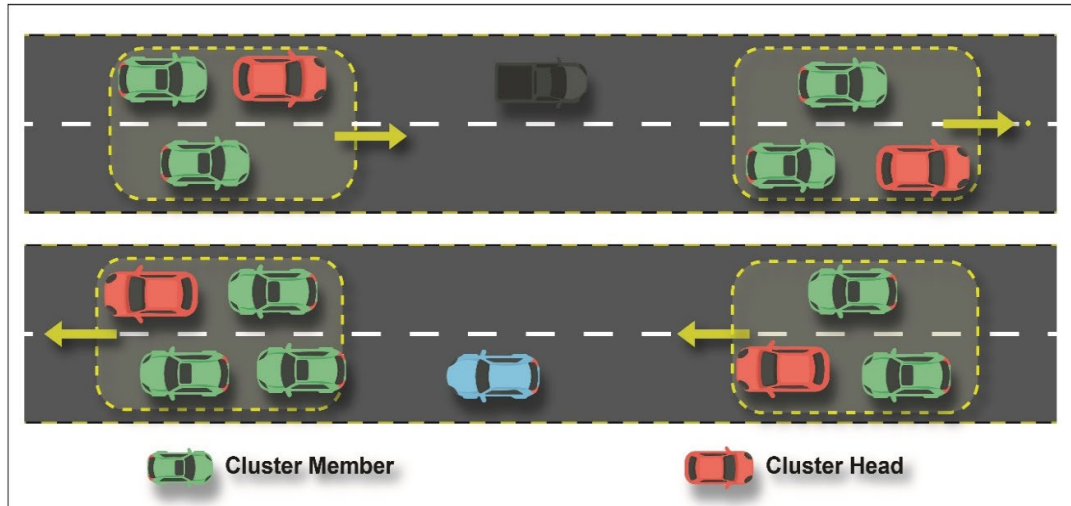


FIGURE 3. Cluster-based routing.

cluster's size depends on the transmission range of the nodes. Similarly, cluster nodes change their CH with the passage of time, as vehicles move along on the road. Minimizing this switching ensures formation of stable clusters [9].

The criteria for a node to be included in a specific cluster depends on typically three important factors namely the speed, the direction and the distance [10]. Routing in vehicles based on clusters is depicted in Figure 3. Clustering of vehicle is a continuous problem and must be solved accordingly. It has been experimentally proved that evolutionary algorithms are efficient in solving continuous problems [11], therefore a method is anticipated for improving the vehicular clustering issue in VANETs. Clustering is an NP hard problem [12], and these techniques have been applied to search near-optimal solutions. Our research work took inspiration from this. Here Moth Flame Optimization (MFO) algorithm, which is an evolutionary procedure, is utilized to accomplish clustering in VANETs.

Evolutionary algorithms offer suitable estimated clarifications to problems which are not easily determined by other approaches. Evolutionary algorithms exhibit very effective results for optimization problems [13]. To ascertain a precise answer exhaustive computation are typically made. In such problems it is not compulsory to find the ideal answer rather sometimes a near-optimal result is acceptable. Evolutionary techniques are fruitful in situations like these. MFO is a bio-inspired optimization procedure which is utilized to handle complex optimization problems. Clustering is regarded as an optimization problem so MFO algorithm is much suitable for it. Consequently, it becomes the inspiration of this work where a novel bio-inspired procedure for clustering of vehicle is projected. The algorithm is based on transverse orientation, the navigating procedure of moths in nature. The procedure has been applied with respect to a highway model. Numerous parameters are considered for instance the transmission range, the vehicle's direction and speed, the number of nodes

in a network and the grid size. For an improved analysis, the simulation results are exhibited graphically and evaluated with other renowned meta-heuristics procedures. The average cumulative results for all the grid sizes are 27.1% for AMONET while 36.3% for ACO, 54.9% for CLPSO and 58.7% for MOPSO. The results specify that our algorithm produces near optimum results and helps in generating clusters in VANETs efficiently. The rest of the paper is organized as following:

Section 2 clarifies the key concepts of VANETs and presents some of the related research work. It tries to justify the usage of an evolutionary algorithm to resolve a multi objective problem of clustering of vehicles in VANETs. Section 3 presents the research vision and proposes our algorithm. The experimental arrangement and results are described in detail in section 4. Eventually, section 5 concludes this paper and provides some guidelines for future analysis.

II. RELATED WORK

In ad hoc networks the group of nodes are connected for temporary communication. Vehicular ad hoc networks (VANETs) are a type of ad hoc network, where nodes (vehicles) are connected with each other without a fixed hardware infrastructure. The nodes are connected wirelessly and the topology changes instantly and arbitrarily. These kinds of networks might function in a standalone manner or they may be linked to a broader network. It is a distributed and a decentralized sort of wireless network. In wired networks, routers are used to set up the network. Similarly access points are employed in wireless networks. However, there is no pre-existing infrastructure in these networks, as every node communicate with each other and contributes in routing. Thus, communication is of extreme importance in ad hoc networks. The usage of numerous access technologies that assist V2V and V2I communication is recommended

in literature. WiFi and WiMAX (Worldwide Interoperability for Microwave Access) technologies are proposed in [14] and ZigBee and WiFi as access technologies in [15] while some have proposed Wireless Access for Vehicular Environments (WAVE) standard [16]. VANETs are foreseen by the automotive business as one of the most vital upcoming technologies. It can sustain a huge number of applications comprising of traffic management, safety and infotainment. VANETs growth will rise in the future and it is currently considered a critical research area [17].

One of the core concerns in VANETs is to transfer information from one vehicle to another efficiently, reliably and without any delay. As vehicles move at a great pace in VANETs, the network topology is ever changing, producing brief connection between the vehicles [18]. The task of transferring data becomes tremendously challenging [19]. To tackle all these issues and create a reliable and efficient network, vehicles are grouped into clusters. Clustering creates dispersed structure of categorized network arrangements by grouping nodes with each other centered on relative velocity and co-related spatial distribution. The reliability and scalability in VANETs are improved by clustering [20]. Besides enhancing routing of vehicles, the clusters assist in an accident situation or can also support in information propagation and entertainment applications.

There are numerous clustering techniques in literature. A file transfer method for VANETs based on clusters is presented in [21]. A vibrant multi clustering scheme is recommended in [22]. It adjusts to the subtleties of the cluster when traffic disruption occurs. All the vehicles are clustered as a central solitary cluster in a tiered scheme. However, it was found that the present vehicle disturbance adaptive method offers a low quality of service with a sole cluster of vehicles. Patel *et al.* [23] has presented certain routing protocols and their pros and cons. A clustering algorithm using agent technology is anticipated in [5]. The main objective is to define the properties of the agent. The vehicles having similar context information are clustered together. A dynamic on-demand algorithm for routing using ant colony optimization is projected in [24]. In literature several VANETs routing protocols are found [23], [25]. For efficient networking in VANETs there should be reduced number of CHs and lifetime of clusters should be longer.

Most of the algorithms were modelled mathematically before the emergence of heuristic optimization algorithms. These mathematical optimization approaches have a severe problem of local optima stagnation [26]. Therefore, these methods are not helpful in solving real world problems. Swarm intelligence procedures are founded on the collection of animals such as birds, ants, moths known as a swarm. The individual agents in a swarm have a self-organized behavior and follow the local rules, regardless to the global pattern, ultimately accomplish collective intelligence termed as swarm intelligence. This makes them very attractive in solving a problem. SI algorithms for instance bee-inspired algorithms, ACO, particle swarm, cuckoo search and several

others are recognized as efficient algorithms to tackle hard optimization problems in static scenarios [27].

In this research work, MFO is compared with PSO, MOPSO and ACO. PSO is a swarm-based procedure which is inspired by birds' movement. Each bird navigates in correspondence with the local and global best thereby converging to a near optimal position. The individuals apart from retaining their own best position also consider the other bird's position. If it finds that the other bird's position is better than its own, the adaptive change will occur. MOPSO is a modification in PSO. Here, the problem is dependent on more than two conflicting objectives. The number of objective functions is minimized or maximized concurrently. Those solutions are considered the best which deals with these conflicting objectives effectively. Ant colony optimization (ACO), is a swarm-based technique. It is based on the movement of ants and the pheromone (a chemical substance) trail. The ants which are following are being guided to follow the direction using pheromone. The concentration of pheromone decides the path for the later ants thus achieving indirect communication between the ants. These techniques have been widely used for solving complex optimization problems [28], [29].

MFO is a nature-inspired optimization procedure which is utilized to tackle complicated optimization problems [30], [31]. It is motivated by the navigation method of moths termed as transverse orientation, an efficient movement technique. A moth flies by keeping a fixed angle in relation with the moon thereby ensuring a straight-line movement in lengthy expanses. MFO has been employed to solve numerous problems in literature. It is a highly effective method in solving optimization problems [32], [33]. It proved to be an efficient technique to train the multi-layer perceptron [34]. It has achieved flourishing results in medical diagnosis [35], [36]. Similarly, MFO has been implemented in image segmentation yielding positive results [37]. Consequently, this becomes the inspiration of this research work where we employed MFO based algorithm named AMONET to solve the vehicular clustering problem in VANETs.

III. PROPOSED METHODOLOGY

The No Free Lunch (NFL) theorem [38] declares that not a single technique can handle all kinds of optimization problems simultaneously. Therefore, an optimizer may accomplish decent results in a scenario while may fail in another. Accordingly, new optimizers may accomplish better results than the existing optimizers in each set of problems. This becomes the motivation of our work. This research work presents AMONET, a novel approach for realizing efficient clustering in VANETs by utilizing Moth Flame Optimization (MFO) algorithm.

A. MOTH FLAME OPTIMIZATION (MFO) ALGORITHM

MFO algorithm is based on moth's movement. Moths are bugs quite like butterflies. In excess of 160,000 various types of moths exist. Moths can travel at night by flying in accordance with the position of the moon by using an

analogous angle, a movement termed as transverse orientation. The method helps in travelling in straight and long path [39]. Transverse orientation is effective, yet it is commonly observed that moths move around man-made lights in a spiral manner. They are misled by these artificial lights. Transverse orientation is an efficient movement approach if the light source is distant but is inefficient if the light source is nearby. Whenever a moth notices a man-made source of light, it attempts to maintain a similar angle with it to fly in a straight trail. However, this light source is extremely close comparing with the moon, maintaining a similar angle causes a moth to fly spirally, which eventually becomes lethal for it. This movement pattern can help in optimization. MFO is a population-based procedure and has been effectively applied to solve complex optimization problems [40]. Additionally, MFO avoids local optima stagnation and has very high diversification [41], [42].

In this procedure, the candidate solutions are the moths and the moth's position in space are considered the variables of the problem. The movement of a moth can be 1D, 2D, 3D or hyper dimensional. The matrix M shows the set of moths which is depicted as:

$$M = \begin{pmatrix} m_{1,1} & m_{1,2} & m_{1,3} & \cdots & \cdots & m_{1,d} \\ m_{2,1} & m_{2,2} & m_{2,3} & \cdots & \cdots & m_{2,d} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ m_{n,1} & m_{n,2} & m_{n,3} & \cdots & \cdots & m_{n,d} \end{pmatrix} \quad (1)$$

Here n implies the number of moths and the number of dimensions (variables) is represented by d.

An array OM is created where the resulting fitness values for all the moths are saved, shown as:

$$OM = \begin{bmatrix} OM_1 \\ OM_2 \\ OM_3 \\ \vdots \\ OM_n \end{bmatrix} \quad (2)$$

It is important to notice from the equations that the fitness function yields the fitness values for each moth. OM₁ for instance in the matrix OM represents the fitness value for the first moth and so on.

Flames are the other key component in this procedure. A flame matrix which is similar to the moth's matrix is shown below:

$$F = \begin{pmatrix} F_{1,1} & F_{1,2} & F_{1,3} & \cdots & \cdots & F_{1,d} \\ F_{2,1} & F_{2,2} & F_{2,3} & \cdots & \cdots & F_{2,d} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ F_{n,1} & F_{n,2} & F_{n,3} & \cdots & \cdots & F_{n,d} \end{pmatrix} \quad (3)$$

In this matrix, n represents the number of moths while d symbolizes the number of dimensions (variables).

Consequently, an array OF shows each flame's fitness value:

$$OF = \begin{bmatrix} OF_1 \\ OF_2 \\ OF_3 \\ \vdots \\ OF_n \end{bmatrix} \quad (4)$$

It is noteworthy to mention here that the moths and the flames both are regarded as solutions. Both are updated and treated differently in every iteration. Moths are regarded as the real search agents that fly in the search space, while the finest spot of moths attained thus far are represented by the flames. Put it in other way, the flames are regarded as pins or flags which are released by the moths while looking through the search space. Consequently, every moth explores nearby a flame (flag) and revises it whenever it finds a superior solution. A moth will not miss its best solution by applying this procedure.

B. METHODOLOGY

The general steps of our algorithm are shown in Figure 4. In the first step, the vehicles are initialized in the grid. The next step shows the mechanism of our proposed algorithm. In this step, all the basic tasks, for instance, creation of cluster matrix and objective matrix and fitness calculation of the candidate solution is performed. The calculation is extensively carried out to retrieve the optimal solution. The final step depicts the creation of cluster. The general block diagram of our algorithm is depicted in Figure 5. The procedure is started by initializing the agents, creating cluster matrix and creating objective matrix which defines the parameters. If the maximum iterations are not reached, the fitness of the search agent is calculated. Then the calculated fitness values are sorted and the whole population is sorted regarding the fitness values. In the next step, the best agent position is updated. This technique is executed till the maximum number of iterations are attained. Consequently, the best solution having the optimum number of clusters is achieved.

C. PSEUDO CODE

The stepwise function of our anticipated procedure is presented next. A pseudo code for AMONET is illustrated in Table 1.

The algorithm initializes by randomly positioning vehicles in a grid, with certain speed and directions. The vehicles are assigned ID's and Euclidian distance is computed among all of them to form a complete distance matrix of the entire network.

The position of moths assists in creating the search space. Formation of search space rely on some constraints for instance; dimensions, lower and upper bounds. Subsequently, the fitness of moths is calculated by using the moth's position in the search space. The fitness values are evaluated with the preceding iteration to form an organized matrix in ascending order of fitness values. This fitness matrix identified the lower fitness value of the moth. Hence, the fitness values and the

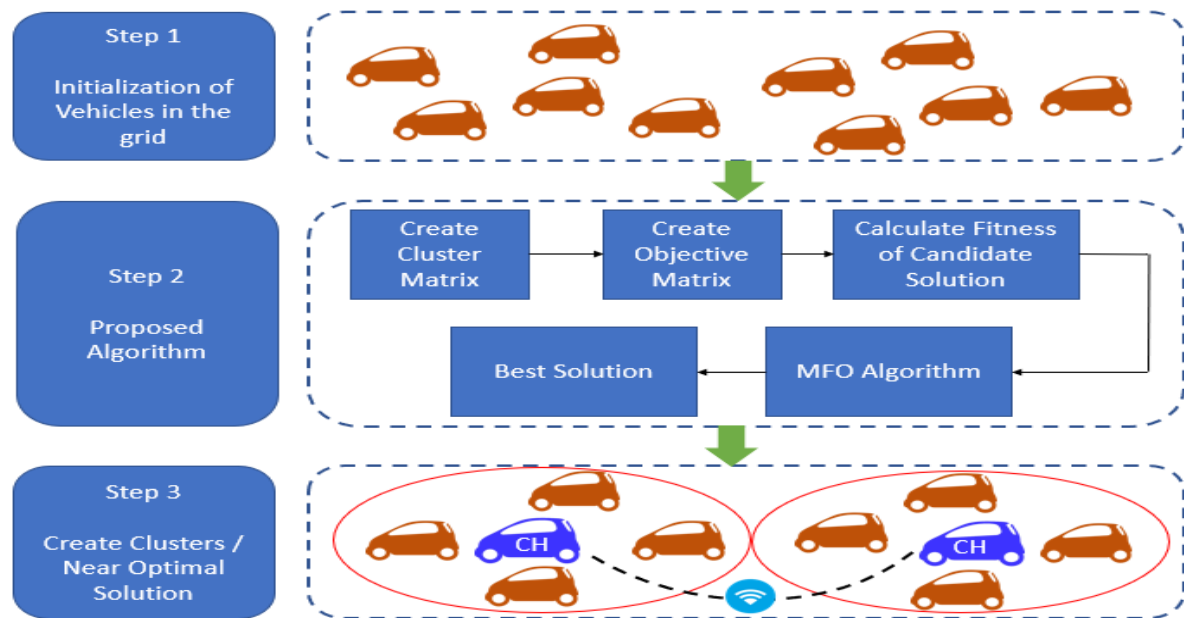


FIGURE 4. General steps of our algorithm.

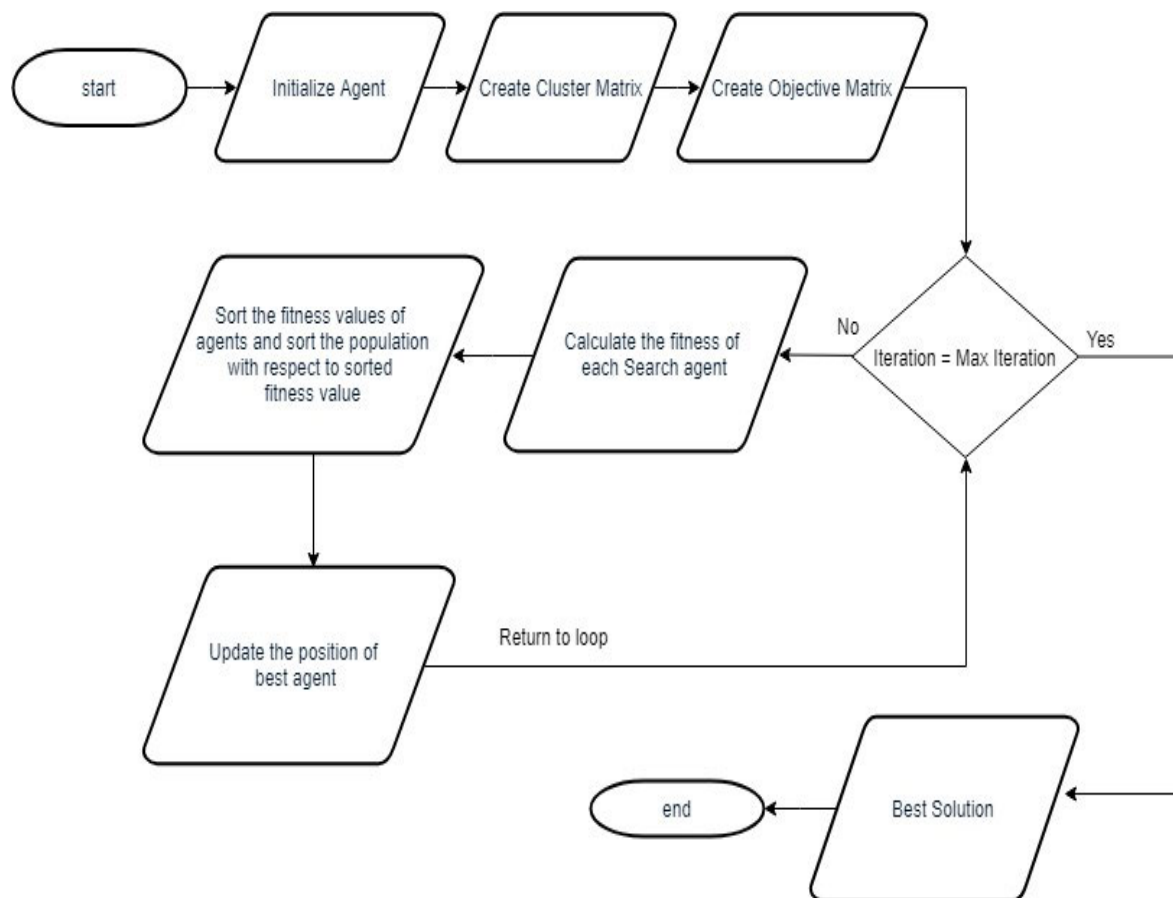


FIGURE 5. Methodology's block diagram.

moth position are used to obtain the best flame score. This helps us to update the value of moth position accordingly. A linearly decreasing factor 'a' is employed to converge

towards the solution. The range of 'a' is $[-2, -1]$, lowest the value of 'a' means better the chance to converge earlier. Lastly, the algorithm terminates if the maximum iterations

TABLE 1. Pseudo code.

```

1. Set the position of all the vehicles arbitrarily on the road.
2. Randomly initialize the direction of vehicles.
3. Initialize the vehicle's speed/velocity.
4. Generate a mesh topology between the vehicles while each vertex will depict a vehicle ID.
5. Calculate and normalize the distance between the vehicles. Then assign the distance values to the equivalent edges accordingly.
6. Initialize each moth's position to create a search space.
   | FOR each search agent
   | | FOR each dimension
   | | | Update the position of moths according to the upper and lower bounds of the moths to create a search space
   | | END FOR
   | END FOR

7. Initialize Number of Clusters to zero
   Initialize Cluster Fitness to infinity
   Initialize Number of nodes to zero
   WHILE Iteration is less than or equal to maximum iteration or StallIteration is less than 15
     | FOR each moth in a swarm
     | | Calculate the fitness of moths using the fitness function
     | | | WHILE Nodes for Clustering is not equal to zero
     | | | | Calculate the fitness of cluster using the fitness function
     | | | | | IF fitness of cluster is better than best solution thus far
     | | | | | | THEN assign fitness of cluster to the best solution
     | | | | | END IF
     | | | | END WHILE
     | | END FOR
   | END FOR

8. Sort the moth's fitness and the complete population with respect to fitness values.

9. Update the best flame position according to the sorted fitness values of flames obtained thus far.

10. Update the moth's position in accordance with its corresponding flame.
    | FOR each search agent
    | | FOR each dimension
    | | | Calculate the distance of Moth from Flame
    | | | Update the moth's position
    | | END FOR
    | END FOR

11. IF Convergence of current Iteration is identical to Convergence of previous Iteration
    | THEN INCREMENT StallIteration
    | ELSE
    | | Set StallIteration to zero
    | END IF

12. INCREMENT Iteration
13. END WHILE

Assign best solution to the number of clusters

```

have been reached or the result of previous 15 iterations is same. This helps in minimizing the randomness in the results and the algorithm converges effectively. Consequently, the convergence leads us towards the number of clusters required based on different parameters to form a robust communication network.

D. FITNESS FUNCTION

The anticipated algorithm AMONET, can be adopted for multi objective optimization, where weights can be assigned

to each objective according to the conditions preferred by user. The Fitness function can be derived as:

$$F_n = (F_1 * W1) + (F_2 * W2) \quad (5)$$

In this equation, W1 and W2 are the weights of the objective functions F_1 and F_2 respectively. Both the weights have been assigned similar value i.e., 0.5, however these values can be modified in accordance with the user's preference.

F_1 function represents the delta difference of the clusters in n (total number of clusters). The delta difference can be

calculated as:

$$F_1 = \sum_{i=1}^n \text{abs}(\text{DeltaDegree} - |CN_i|) \quad (6)$$

The *DeltaDegree* depicts the ideal degree of cluster density as specified by the user. For dense clusters, this value will be higher while it will be less for sparse clusters. The total number of vehicles in cluster i is represented by CN_i . The lower value of F_1 means the clusters have been generated almost optimally according to the user's specifications.

F_2 function represents the sum of distance of all CHs from their cluster nodes. The distance can be calculated as:

$$dCH_i = \sum_{j=1}^{|CN_i|} \text{ED}(\text{CH}_i, \text{CN}_{j,i}) \quad (7)$$

Here ED stands for Euclidean distance, the position of the i^{th} cluster head is shown as CH_i while the position of j^{th} cluster node in cluster i is indicated as $\text{CN}_{j,i}$. The sum of all these distances represents the function F_2 :

$$F_2 = \sum_{i=1}^n dCH_i \quad (8)$$

Here again n denotes the total number of clusters. The lower value of F_2 function is desirable. As lower value would mean the CH and the cluster node are nearer to each other, thus requiring low energy for transmission of data. More objective functions can be added to calculate the fitness function and also by varying the weights in accordance with the user's requirements.

E. COMPUTATIONAL COMPLEXITY

Computational complexity is one of the most vital features to evaluate an algorithm's efficiency. The MFO algorithm's computational complexity is defined by considering some vital factors for instance the total number of moths, variables, maximum iterations and flames ordering procedure in every iteration. As Quicksort procedure is applied here, the sort is of $O(n \log n)$ for the best case while for the worst case the sort is of $O(n^2)$.

The overall computational complexity of the P function (main function) of the algorithm, is formulated as following:

$$\begin{aligned} O(\text{MFO}) &= O(t(O(\text{Quick sort}) + O(\text{position update}))) \\ O(\text{MFO}) &= O(t(n^2 + n*d)) \\ O(\text{MFO}) &= O(tn^2 + tnd) \end{aligned} \quad (9)$$

In this equation, the number of moths is symbolized by n , t shows the maximum number of iterations and the symbol d represents the number of variables.

An optimal number of clusters in a VANET set-up indicates the stability and efficiency of a network as the network resources are effectively utilized. AMONET optimizes the number of clusters in the network owing to the evolutionary proficiency of MFO which empowers it to select the optimum number of solutions as it is competent and suitable for discrete and continuous variable problems.

IV. EXPERIMENTATIONS AND RESULTS

A. IMPLEMENTATION

The experimental arrangement along with the results are depicted in this section. The results of our anticipated algorithm are comparatively analyzed with three renowned algorithms, specifically MOPSO [43], CACONET [44] and CLPSO [45]. The experimentation proves that our algorithm generates smaller number of clusters in comparison to other techniques mentioned above. Consequently, it helps in curtailing the number of hops and message delivery rates. An efficient network is conceivable by reducing network routing cost.

B. EXPERIMENTAL SETUP

The experimentations have been achieved on four dimensions of road sections i.e., 1km^2 to 4km^2 . The transmission range of the vehicles is adjusted from 100m to 600m and the nodes are modified from 30 to 60. The movement of vehicles is along the X-axis in both directions like a highway scenario. The simulation factors have been set analogous for all the algorithms as shown in [Table 2](#).

C. CLUSTERS GENERATED WITH RESPECT TO GRID SIZE

The clusters are generated in this section, with respect to varying grid sizes keeping number of nodes and the transmission range static. The size of the network is altered from 1000m^2 to 4000m^2 . The nodes are kept static at 30, 40, 50 and 60 for [Figure 6](#) to [Figure 9](#) respectively.

In [Figure 6](#) (A), 30 nodes are considered, and the transmission range is also kept static at 300m. The clusters required to incorporate the complete network increases with the expansion of the network. It can be observed in [Figure 6](#) (A) to (D), the clusters required to cover the network minimizes because of the increase in the transmission range. The results show that our algorithm, AMONET is performing the best.

The clusters are created in [Figure 7](#) with respect to varying grid size. The nodes are taken as 40 whereas the transmission range is also kept static at 300m for (A), 400m for (B), 500m for (C) and 600m for (D). Even though MOPSO provides multiple answers to the problem, yet the results are not suitable in comparison with AMONET. There are some rare instances where CACONET, the algorithm based on ACO, performs better than our algorithm. This is due to the random nature of the algorithm. Nevertheless, our algorithm illustrates the best results in general. Now in [Figure 8](#) the experimentation has been carried out keeping the nodes to 50 and the size of network is modified from $1\text{km} \times 1\text{km}$ to $4\text{km} \times 4\text{km}$. One can observe from these experiments that as the number of nodes are increased, more clusters are generated. Conversely, the increase in transmission range lowers the clusters generated. Another noticeable point is that with the increase in network size, more clusters are generated to cover the entire network.

Lastly, in [Figure 9](#), the clusters are created with respect to varying grid size. The nodes are fixed to 60 whereas the

TABLE 2. Simulation parameters.

PARAMETERS	MOPSO	CLPSO	CACONET	AMONET
POPULATION SIZE	100	100	100	100
MAXIMUM ITERATIONS	150	150	150	150
SIMULATION RUNS	10	10	10	10
INERTIA WEIGHT $W /$	0.694	0.694	0.05	0.694
EVAPORATION RATE:				
$c1^1$	2	2	2	2
$c2^1$	2	2	2	2
VEHICLE VELOCITY RANGE	22 TO 30 (M/S)	22 TO 30 (M/S)	22 TO 30 (M/S)	22 TO 30 (M/S)
MAXIMUM ACCELERATION	1.5 M/S ²	1.5 M/S ²	1.5 M/S ²	1.5 M/S ²
SIMULATION AREA	1KM ² , 2KM ² , 3KM ² , 4KM ²	1KM ² , 2KM ² , 3KM ² , 4KM ²	1KM ² , 2KM ² , 3KM ² , 4KM ²	1KM ² , 2KM ² , 3KM ² , 4KM ²
TOTAL LANES	8	8	8	8
LANE WIDTH	50 M	50 M	50 M	50 M
TRANSMISSION RANGE	100M – 600M	100M – 600M	100M – 600M	100M – 600M
MINIMUM DISTANCE B/W	2 M	2 M	2 M	2 M
VEHICLES				
MAXIMUM DISTANCE B/W	5 M	5 M	5 M	5 M
VEHICLES				
MOBILITY MODEL	FREEWAY MOBILITY MODEL	FREEWAY MOBILITY MODEL	FREEWAY MOBILITY MODEL	FREEWAY MOBILITY MODEL
W1 (WEIGHT OF FIRST OBJECTIVE FUNCTION)	0.5	0.5	0.5	0.5
W2 (WEIGHT OF SECOND OBJECTIVE FUNCTION)	0.5	0.5	0.5	0.5

transmission range is also kept static at 300m for (A), 400m for (B), 500m for (C) and 600m for (D). It is apparent from these set of experiments that the performance of our algorithm is quite efficient comparatively.

D. PERCENTAGE OF CLUSTER HEADS SELECTION

The efficiency of a clustering algorithm is validated if it has lower percentage of cluster heads and higher ratio of cluster members. From Figure 10 to Figure 13, the percentage of nodes designated as CHs for MOPSO, CLPSO, AMONET (MFO) and CACONET are displayed for all the grid sizes from 1km² to 4km². These values are generated for all the transmission ranges from 100m to 600m and number of nodes from 30 to 60. For the grid size 1 km², our algorithm takes 13.1% of the nodes as cluster heads on average while this ratio is higher for other algorithms than our approach.

Conversely speaking, our algorithm considers 86.9% of the nodes as simple cluster members which shows its efficiency.

In Figure 11, the size of the network is increased to 2km². The percentage of nodes chosen as CHs for all the four algorithms is exhibited graphically. The average value is calculated for the nodes in the range from 30 to 60 and for all the transmission ranges from 100m to 600m. CAMONET takes 24.5% of the nodes as cluster heads. While the other three algorithms CLPSO, MOPSO and CACONET considers 50, 55 and 31.1 percent of the nodes as CHs respectively. The ratio of non-cluster heads produced is high for CAMONET which is around 75.5% while the other algorithms have 50%, 45% and 68.9% respectively.

Now in Figure 12, the corresponding average percentage values are displayed which validates the effectiveness of our approach. One can notice that these values are higher than the previous two instances where the size of network was

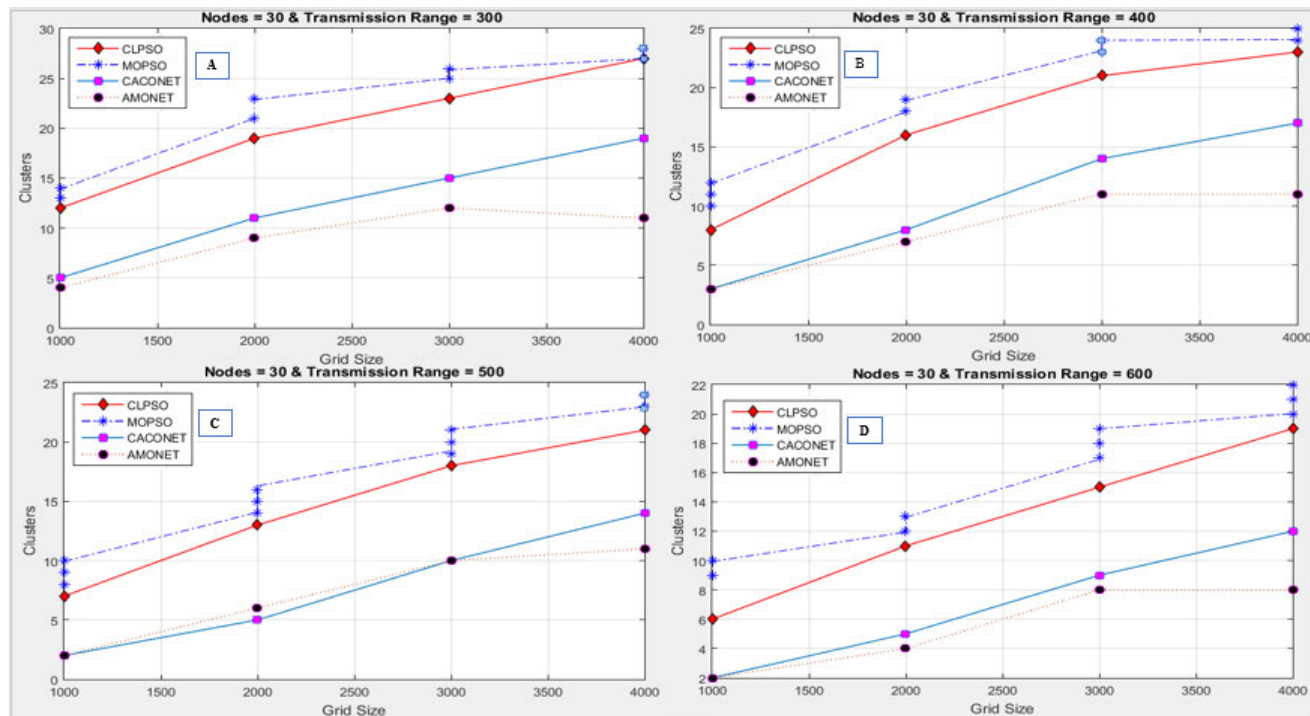


FIGURE 6. Clusters generated with respect to grid size. Transmission range: 300 to 600 m. Nodes = 30.

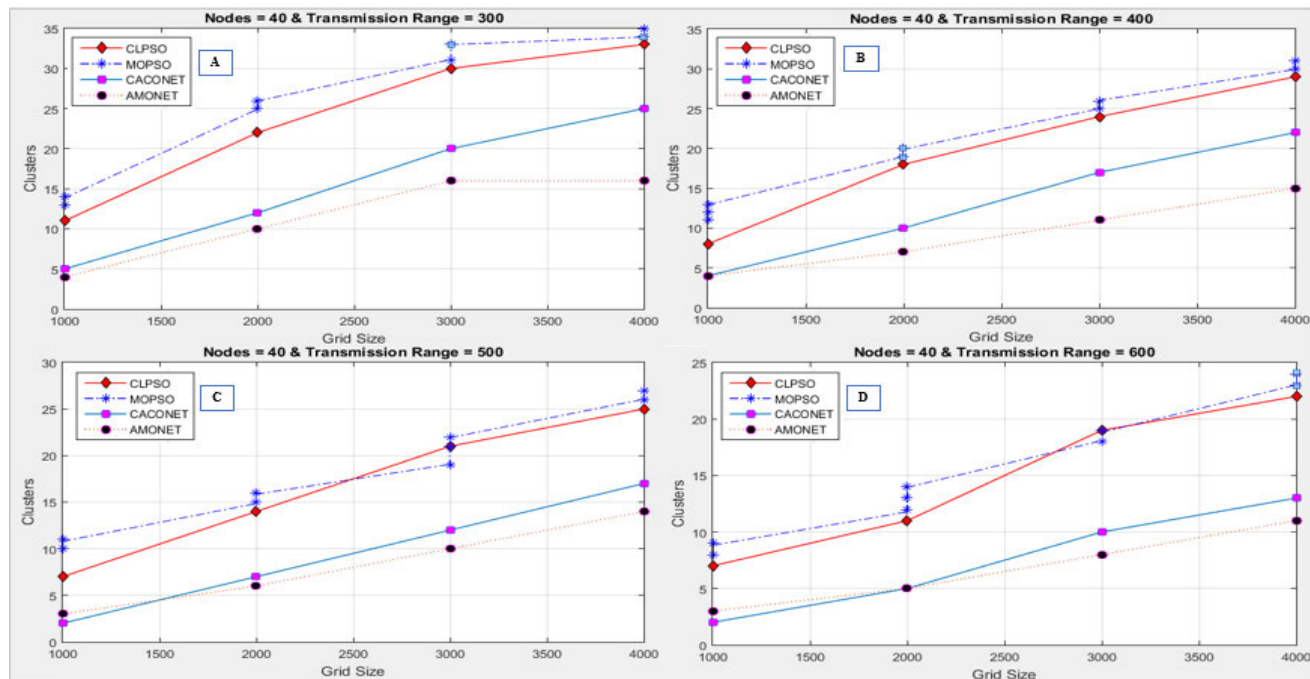


FIGURE 7. Clusters generated with respect to grid size. Transmission range: 300 to 600 m. Nodes = 40.

small compared to this grid size. The rationale behind this is relatively simple, more clusters will be created if the size of the network is expanded.

Finally, the network size is expanded to 4km². AMONET is considering 37.5% of nodes as cluster heads while

62.5% are taken as cluster members. These values are undoubtedly the best in comparison with other algorithms in discussion. This validates that our algorithm requires a smaller number of CHs to achieve the communication in the network.

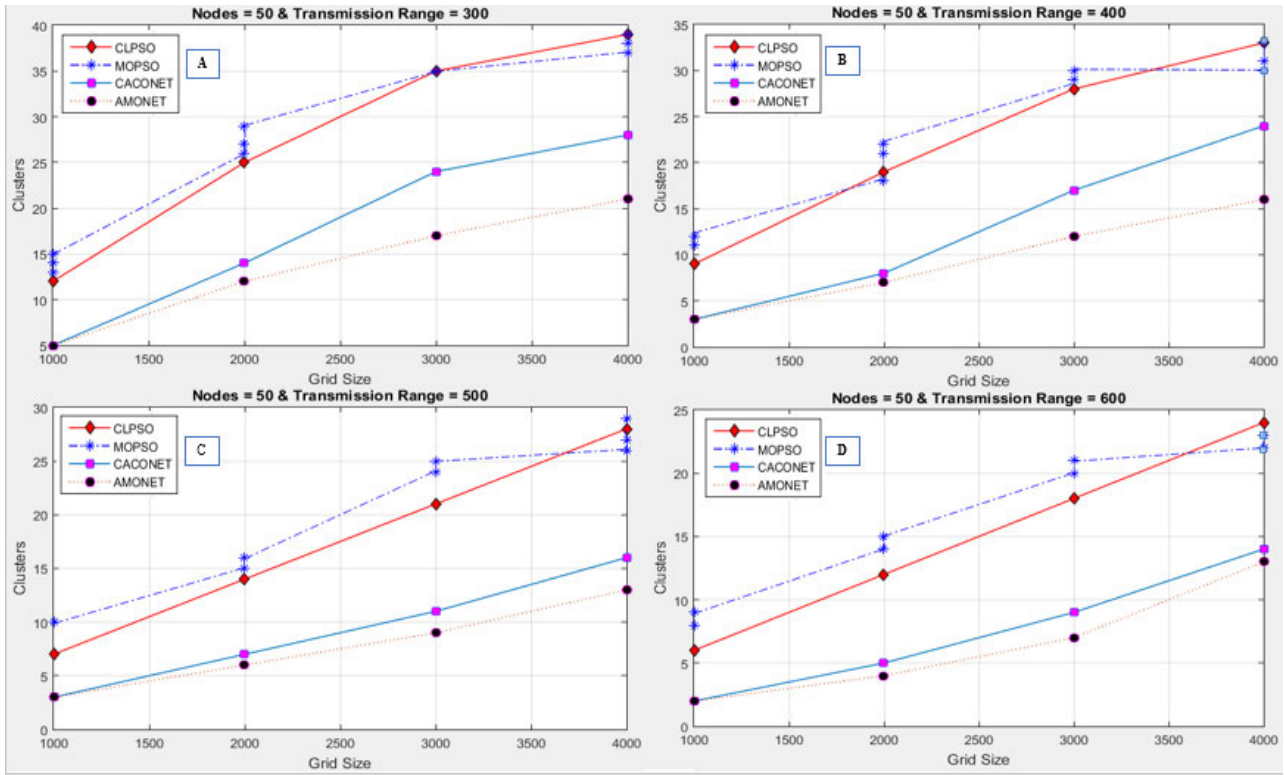


FIGURE 8. Clusters generated with respect to grid size. Transmission range: 300 to 600 m. Nodes = 50.

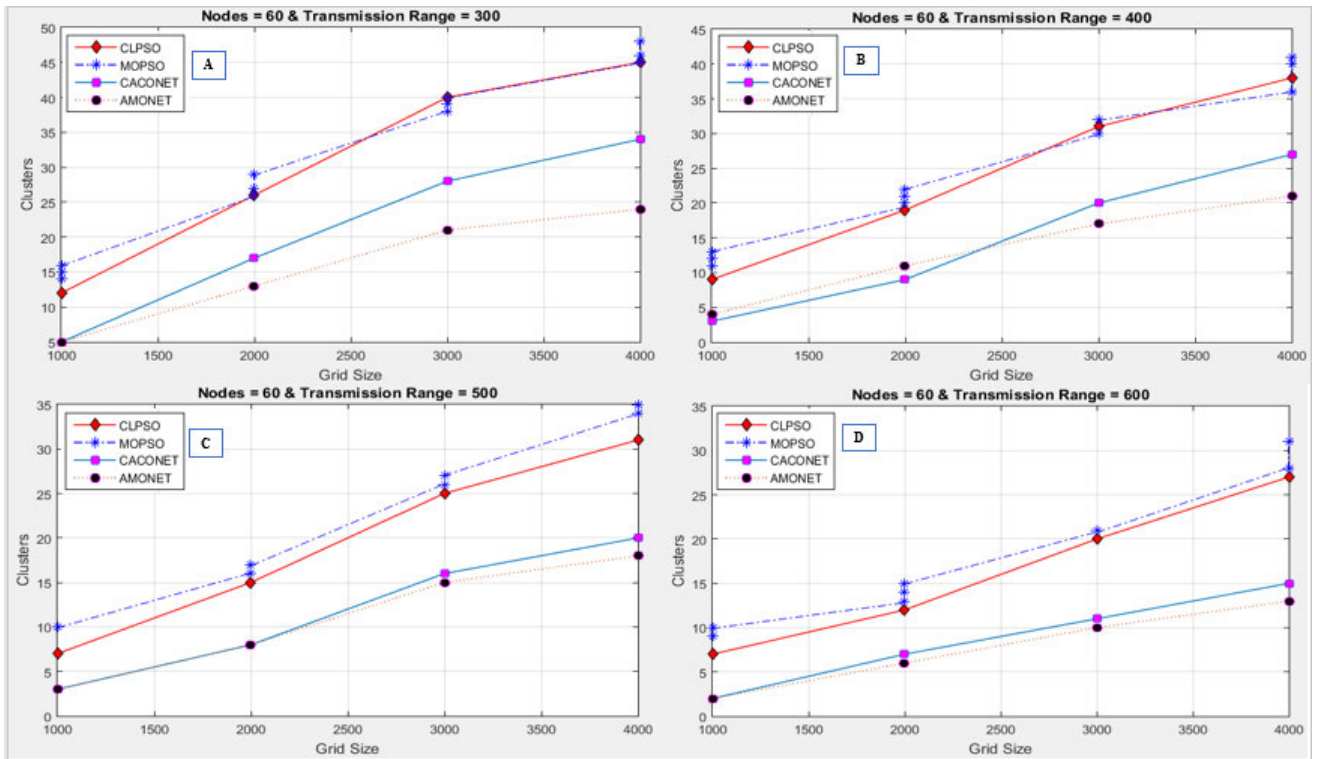


FIGURE 9. Clusters generated with respect to grid size. Transmission range: 300 to 600 m. Nodes = 60.

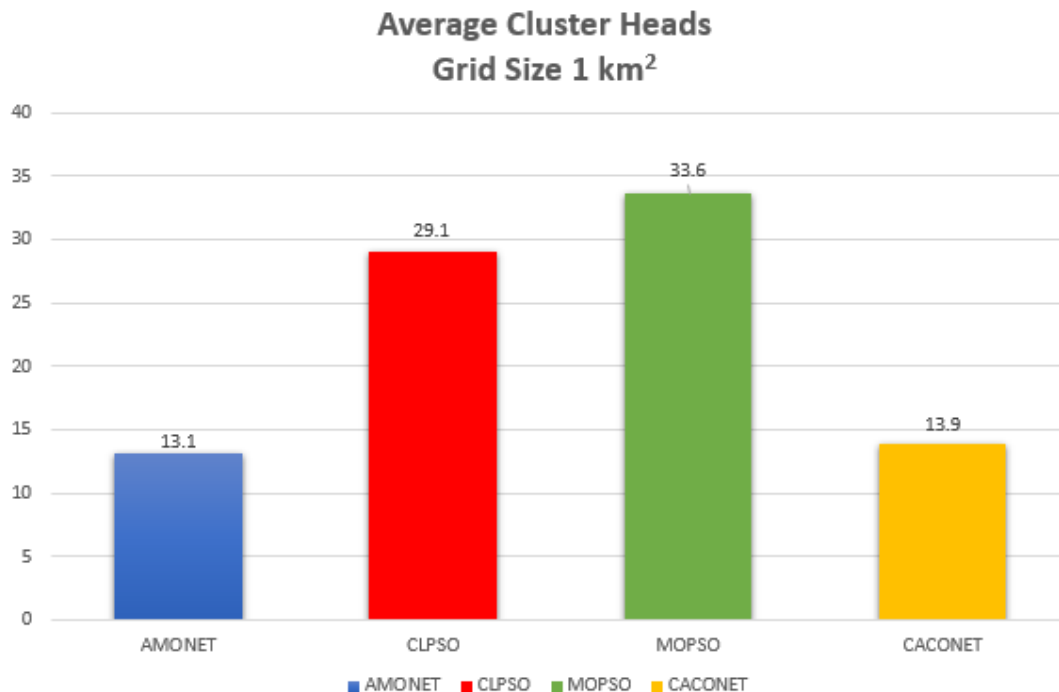


FIGURE 10. Average cluster heads in percentage. Grid size = 1 km².

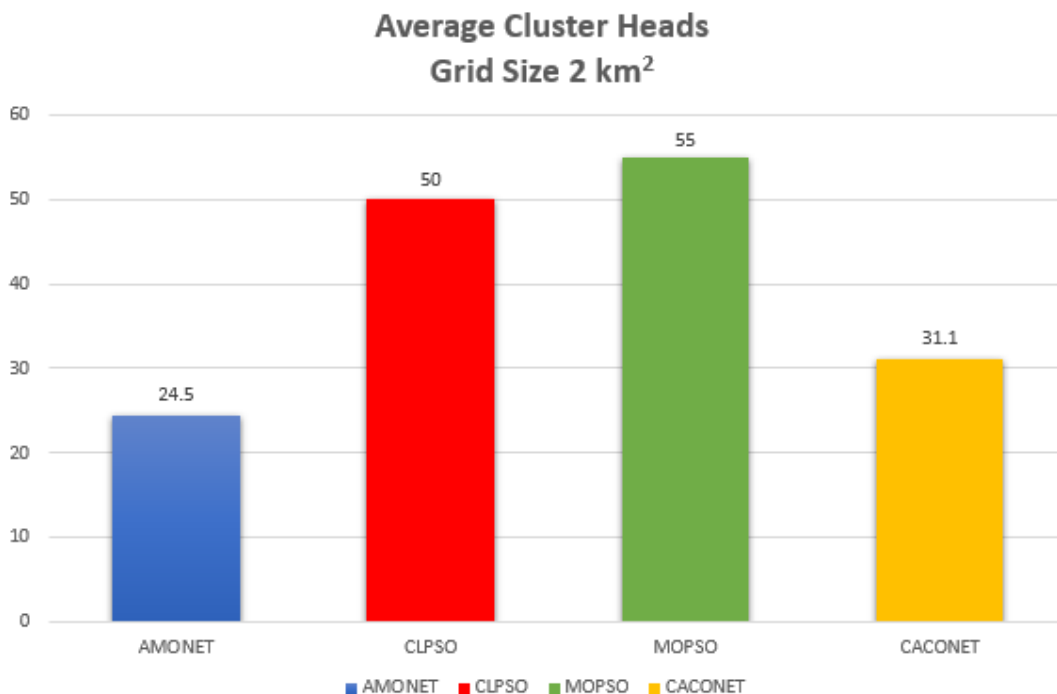


FIGURE 11. Average cluster heads in percentage. Grid Size = 2 km².

E. CLUSTERS GENERATED FOR ALL THE TRANSMISSION RANGES, NODES AND GRID SIZES

To clarify the results, [Table 3](#) to [Table 6](#) depicts the complete results of the experimentation. [Table 3](#) represents the

clusters generated by modifying the transmission range and the number of nodes whereas the grid size is kept fixed at 1km x 1km. The transmission range has been modified from 100m to 600m, while the nodes are increased from

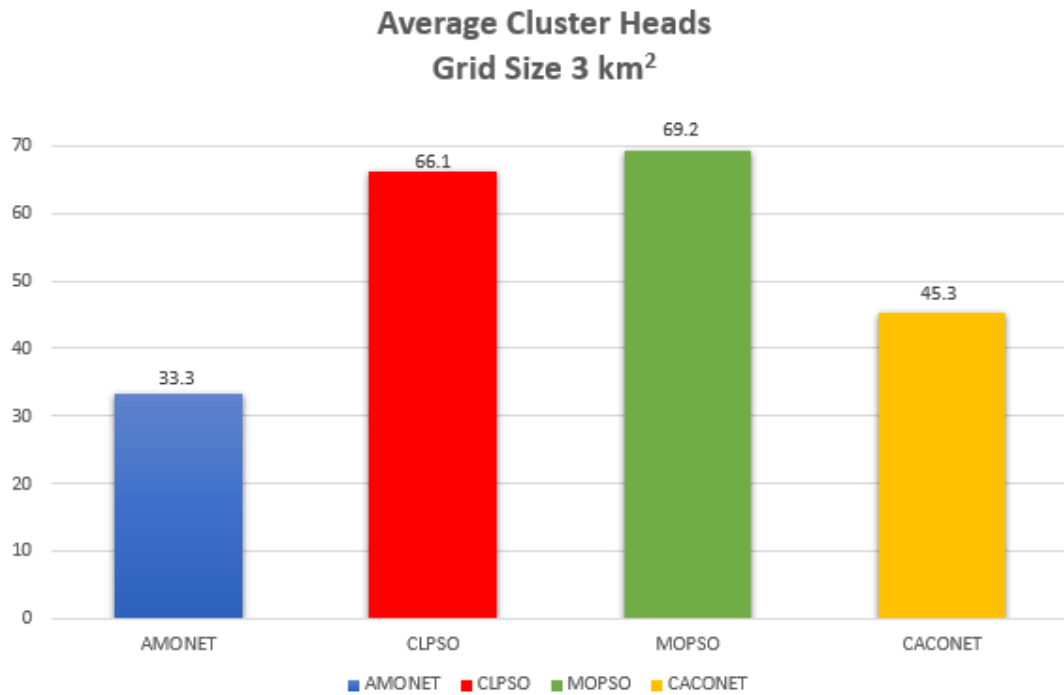


FIGURE 12. Average cluster heads in percentage. Grid size = 3 km².

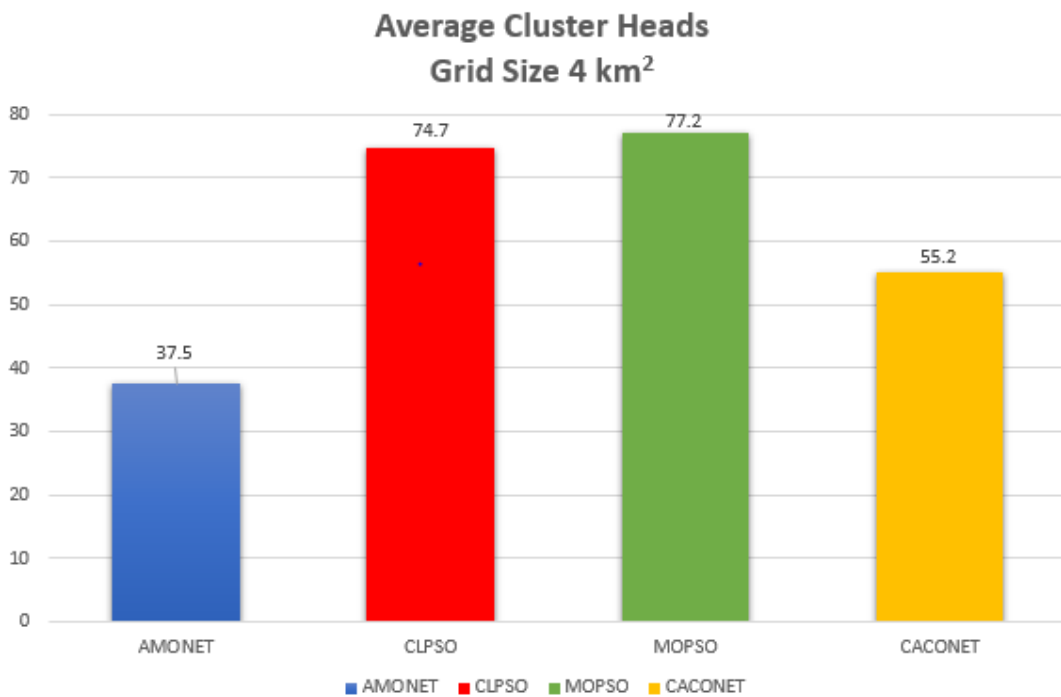


FIGURE 13. Average cluster heads in percentage. Grid size = 4 km².

30 to 60. MOPSO generates multiple solutions but to simplify the table, we have considered single output for it. Here it can be observed that AMONET is producing the best results. If we examine Table 3 thoroughly, it is apparent that

as the range of transmission increases, the clusters required to cover the entire network reduces. The rationale for this inverse relationship is that as the range of transmission increases, more vehicles are included in the transmission

TABLE 3. Clusters generated by modifying transmission range and nodes. Grid size: 1 km².

Grid Size 1km ²		AMONET				CLPSO				MOPSO				CACONET			
Transmission Range	Nodes	30	40	50	60	30	40	50	60	30	40	50	60	30	40	50	60
	100	10	12	21	23	23	30	32	38	26	30	33	38	13	19	23	24
	150	8	12	12	18	19	23	23	26	21	24	25	27	10	12	13	15
	200	7	6	9	10	16	19	18	19	18	19	18	21	7	8	9	10
	250	5	5	5	7	14	14	14	15	15	15	16	17	6	6	6	7
	300	4	4	5	5	12	11	12	12	13	13	14	15	5	5	5	5
	350	4	4	4	4	10	10	10	11	11	12	13	13	4	4	4	4
	400	3	4	3	4	8	8	9	9	11	11	12	12	3	4	3	3
	450	3	4	3	3	8	8	7	8	10	10	10	10	3	3	3	3
	500	2	3	2	3	7	7	7	7	9	10	10	10	2	2	3	3
	550	2	3	2	2	6	7	7	7	9	9	10	10	2	2	2	2
600	2	3	2	2	6	7	6	7	9	9	9	9	2	2	2	2	

TABLE 4. Clusters generated by modifying transmission range and nodes. Grid size: 2km².

Grid Size 2km ²		AMONET				CLPSO				MOPSO				CACONET			
Transmission Range	Nodes	30	40	50	60	30	40	50	60	30	40	50	60	30	40	50	60
	100	14	19	22	26	29	37	44	51	30	40	48	54	21	28	35	41
	150	13	18	21	25	26	33	38	44	29	37	42	46	18	25	27	33
	200	12	13	17	24	23	29	33	37	27	32	36	38	15	19	25	29
	250	9	11	16	16	21	26	28	31	24	29	31	33	14	15	20	20
	300	9	10	12	13	19	22	25	26	21	25	27	27	11	12	14	17
	350	9	9	9	11	17	20	21	21	19	22	23	23	9	10	11	12
	400	7	7	7	11	16	18	19	19	18	20	21	21	8	10	8	9
	450	7	6	7	8	14	16	16	17	17	17	18	18	7	7	8	8
	500	6	6	6	8	13	14	14	15	14	16	16	17	5	7	7	7
	550	5	5	6	7	12	13	13	14	13	15	15	15	5	6	6	6
600	4	5	4	6	11	11	12	12	13	14	14	14	5	5	5	5	

range of a specific vehicle, thus lower number of clusters are generated.

Table 4 provides the detailed results for all algorithms by varying the quantity of vehicles and the transmission capabilities of these vehicles. The network size is kept static at 2km x 2km. The results can be viewed in detail where it is apparent that AMONET is generating the ideal number of clusters. The power of our algorithm can further be validated that it produces the best results for all the values of transmission ranges or number of vehicles. There are rare instances, where the results of CACONET are better than our technique. This is due to the random nature of the algorithm.

The complete results for the grid size of 3km² is presented in Table 5. One can notice that the network size has increased so has the number of clusters. This confirms the direct relationship of network size with the number of clusters needed to encompass the complete network. Similarly, more clusters are generated if the nodes are increased. This is understandable as the vehicles are increased in number and placed at arbitrary positions, it will require more clusters to cover the complete network efficiently. The results provided in Table 5 validates the effectiveness of our anticipated procedure.

Table 6 is provided to further clarify the results of Figure 6 to Figure 9. The results are attained by keeping the grid size

TABLE 5. Clusters generated by modifying transmission range and nodes. Grid size: 3km².

Grid Size 3km ²		AMONET				CLPSO				MOPSO				CACONET			
Transmission Range	Nodes	30	40	50	60	30	40	50	60	30	40	50	60	30	40	50	60
	100	14	22	25	30	30	39	50	58	30	40	50	59	24	32	38	47
	150	13	19	25	28	29	36	47	53	30	39	48	56	22	30	35	42
	200	13	19	22	26	28	33	42	49	29	37	44	49	18	27	31	37
	250	12	16	18	22	26	33	37	44	28	33	39	45	17	25	29	31
	300	12	16	17	21	23	30	35	40	26	31	35	40	15	20	24	28
	350	12	12	15	19	22	27	33	35	25	28	32	35	15	19	21	24
	400	11	11	12	17	21	24	28	31	24	26	29	32	14	17	17	20
	450	10	10	10	15	20	22	23	27	22	23	26	28	11	14	14	16
	500	10	10	9	15	18	21	21	25	20	21	24	26	10	12	11	13
	550	10	9	8	12	16	19	19	22	19	20	22	23	10	10	9	11
	600	8	8	7	10	15	19	18	20	18	19	20	21	9	10	9	9

TABLE 6. Clusters generated by modifying transmission range and nodes. Grid size: 4km².

Grid Size 4km ²		AMONET				CLPSO				MOPSO				CACONET			
Transmission Range	Nodes	30	40	50	60	30	40	50	60	30	40	50	60	30	40	50	60
	100	15	21	24	30	30	40	50	59	30	40	50	60	24	33	40	49
	150	14	18	24	30	29	39	48	57	30	40	50	59	23	31	40	45
	200	14	17	24	29	29	38	45	54	30	38	48	56	21	31	37	42
	250	12	17	21	26	27	36	42	49	29	38	45	53	21	28	34	39
	300	11	16	21	24	27	33	39	45	28	35	39	48	19	25	28	34
	350	11	15	16	21	25	31	36	41	26	33	33	43	17	23	28	31
	400	11	15	16	21	23	29	33	38	24	31	30	40	17	22	24	27
	450	11	15	14	19	22	27	31	34	24	29	28	37	15	18	21	24
	500	11	14	13	18	21	25	28	31	23	26	26	35	14	17	16	20
	550	9	11	13	15	20	24	25	28	22	25	24	30	13	15	15	17
	600	8	11	13	13	19	22	24	27	21	22	23	28	12	13	14	15

fixed at 4km x 4km but modifying the number of nodes and the transmission range for all the algorithms. The network size has become very huge, yet the results of our technique are superior than the other procedures. One can notice more clusters are created here in the grid size of 4km² owing to the huge network size.

It is apparent from these results that clusters are reduced drastically by boosting the transmission range, while number of clusters increases if the grid size is increased. Consequently, the number of clusters essential in a network are directly linked to the total grid size. In contrast, it is inversely proportional to the transmission range. Another notable fact to consider here is that as the nodes are increased, more

clusters are required to cover the network showing a direct connection between the number of nodes and the clusters being generated. From the above discussion, the following equations can be derived.

$$X \propto G_s \tag{10}$$

$$X \propto \frac{1}{T_r} \tag{11}$$

$$X \propto N \tag{12}$$

In these equations, number of clusters is depicted by X; the grid size is represented by G_s, T_r represents the transmission range and N signifies the number of nodes.

Equation (10) implies that increasing the grid size (from 1km^2 to 4km^2 in our experiments) will gradually yield more clusters as the overall area has increased enormously. So, to incorporate the complete area, more clusters will be required to cover all the nodes present in the scenario.

Equation (11) in contrast, suggests that if the transmission range is increased (100m to 600m in our experiments), fewer clusters are formed. A logical reason for this is that a vehicle can communicate with more vehicles if it has higher transmission range. It will require smaller number of clusters to encompass the complete network. Conversely, if the transmission range is low, it will not be able to cover those vehicles which are outside its range, hence more clusters will be required to efficiently connect the communication network.

Equation (12) proves that more clusters are generated if the total number of nodes are increased showing the direct relationship between them. This is due to the fact that more clusters will be required to incorporate the increasing number of vehicles. So, it can be concluded that in urban areas, more clusters would be needed while in rural area, where the traffic density is low, a smaller number of clusters will accomplish the job effectively. All the above results demonstrate that the projected clustering algorithm is efficient and adaptable in comparison with different approaches in a VANET scenario. The performance of the proposed method is better than other algorithms in comparison since the diversification of MFO algorithm is very high and it avoids local optima quite efficiently.

V. CONCLUSION AND FUTURE WORK

This paper proposed AMONET, a clustering algorithm founded on Moth-Flame Optimization (MFO) for VANETS. The experimental results are compared with other eminent procedures i.e., CLPSO, MOPSO and CACONET. Due to the evolutionary capability of our procedure, better search spaces are explored, and it creates optimal number of clusters. The results illustrate the efficiency of the procedure which makes it the best amongst the procedures discussed. Smaller number of clusters ensures reduced packet delays and the number of hops is also diminished. As a result, a proficient vehicular network based on clusters can be envisioned. This technique will improve the complete performance of the network by optimizing the clusters and increasing the lifetime of clusters. To extend this work in the future some guidelines are described here. Owing to the high mobility feature, the network shape or topology is ever changing in VANETS. Therefore, re-affiliation frequency of node can be calculated. If several nodes move along and leave a specific cluster, there must be a mechanism to recall the algorithm and re-generate the clusters so that optimal performance can be accomplished. Recalling the clustering algorithm can be one of the vital future lines of action which ensures an efficient vehicular network. To present a vigorous environment, the range of transmission and the total number of nodes can be

changed dynamically. The experiments can be carried out using other new evolutionary algorithms such as Grasshopper Optimization Algorithm (GOA), Whale Optimization Algorithm (WOA) or Dragonfly Algorithm (DA) to achieve optimal results.

REFERENCES

- [1] *Global Status Report on Road Safety 2018: Summary*, World Health Organization, Geneva, Switzerland, 2018.
- [2] *U.S. Department of Transportation, Fatality Anal. Reporting Syst.*, Washington, DC, USA, 2016.
- [3] A. Ullah, X. Yao, S. Shaheen, and H. Ning, "Advances in position based routing towards ITS enabled FoG-oriented VANET—A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 2, pp. 828–840, Feb. 2020.
- [4] A. Daeinabi, A. G. P. Rahbar, and A. Khademzadeh, "VWCA: An efficient clustering algorithm in vehicular ad hoc networks," *J. Netw. Comput. Appl.*, vol. 34, no. 1, pp. 207–222, 2011.
- [5] S. Harrabi, I. B. Jaafar, and K. Ghedira, "A novel clustering algorithm based on agent technology for VANET," *Netw. Protocols Algorithms*, vol. 8, no. 2, pp. 1–19, 2016.
- [6] I. Tal and G.-M. Muntean, "Clustering and 5G-enabled smart cities: A survey of clustering schemes in VANETS," in *Paving the Way for 5G Through the Convergence of Wireless Systems*. Hershey, PA, USA: IGI Global, 2019, pp. 18–55.
- [7] C. Cooper, D. Franklin, M. Ros, F. Safaei, and M. Abolhasan, "A comparative survey of VANET clustering techniques," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 1, pp. 657–681, 1st Quart., 2016.
- [8] L. A. Maglaras and D. Katsaros, "Distributed clustering in vehicular networks," in *Proc. IEEE 8th Int. Conf. Wireless Mobile Comput., Netw. Commun. (WiMob)*, Oct. 2012, pp. 593–599.
- [9] X. Cheng and C. Huang, "A center-based secure and stable clustering algorithm for VANETS on highways," *Wireless Commun. Mobile Comput.*, vol. 2019, Jan. 2019, Art. no. 8415234.
- [10] A. N. Hassan, A. H. Abdullah, O. Kaiwartya, Y. Cao, and D. K. Sheet, "Multi-metric geographic routing for vehicular ad hoc networks," *Wireless Netw.*, vol. 24, no. 7, pp. 2763–2779, Oct. 2018.
- [11] H. R. Maier, S. Razavi, Z. Kapelan, L. S. Matott, J. Kasprzyk, and B. A. Tolson, "Introductory overview: Optimization using evolutionary algorithms and other metaheuristics," *Environ. Model. Softw.*, vol. 114, pp. 195–213, Apr. 2019.
- [12] E. Elhamifar and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2765–2781, Nov. 2013.
- [13] T. Chugh, K. Sindhya, J. Hakanen, and K. Miettinen, "A survey on handling computationally expensive multiobjective optimization problems with evolutionary algorithms," *Soft Comput.*, vol. 23, no. 9, pp. 3137–3166, May 2019.
- [14] R. N. Fries, M. R. Gahrooei, M. Chowdhury, and A. J. Conway, "Meeting privacy challenges while advancing intelligent transportation systems," *Transp. Res. C, Emerg. Technol.*, vol. 25, pp. 34–45, Dec. 2012.
- [15] S. Vodopivec, J. Beöter, and A. Kos, "A survey on clustering algorithms for vehicular ad-hoc networks," in *Proc. 35th Int. Conf. Telecommun. Signal Process. (TSP)*, 2012, pp. 52–56.
- [16] S. Bu, F. R. Yu, X. P. Liu, P. Mason, and H. Tang, "Distributed combined authentication and intrusion detection with data fusion in high-security mobile ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 60, no. 3, pp. 1025–1036, Mar. 2011.
- [17] A. Gardezi, T. Umer, F. Butt, R. Young, and C. Chatwin, "Vehicle monitoring under vehicular ad-hoc networks (VANET) parameters employing illumination invariant correlation filters for the Pakistan motorway police," *Proc. SPIE Opt. Pattern Recognit.*, vol. 9845, Oct. 2016, Art. no. 984508.
- [18] T. Zhang, T. Zhang, and X. Liu, "Novel self-adaptive routing service algorithm for application in VANET," *Appl. Intell.*, vol. 49, no. 5, pp. 1866–1879, 2019.
- [19] S. Benkerdagh and C. Duvallat, "Cluster-based emergency message dissemination strategy for VANET using V2 V communication," *Int. J. Commun. Syst.*, vol. 32, no. 5, p. e3897, Mar. 2019.
- [20] Y. Chen, M. Fang, S. Shi, W. Guo, and X. Zheng, "Distributed multi-hop clustering algorithm for VANETS based on neighborhood follow," *J. Wirel. Commun. Netw.*, vol. 2015, no. 98, pp. 1–12, 2015.
- [21] Q. Luo, C. Li, Q. Ye, T. H. Luan, L. Zhu, and X. Han, "CFT: A cluster-based file transfer scheme for highway," 2017, *arXiv:1701.01931*.

- [22] K. Padmanabhan, I. Jeyasubramanian, J. S. Pandian, and J. V. Rajendran, "Improving QoS in VANET using dynamic clustering technique," *Int. J. Netw. Commun.*, vol. 6, no. 4, pp. 72–79, 2016.
- [23] D. Patel, M. Faisal, P. Batavia, S. Makhija, and M. M. Roja, "Overview of routing protocols in VANET," *Int. J. Comput. Appl.*, vol. 136, no. 9, pp. 4–7, 2016.
- [24] A. M. Oranj, R. M. Alguliev, F. Yusifov, and S. Jamali, "Routing algorithm for vehicular ad hoc network based on dynamic ant colony optimization," *Int. J. Electron. Elect. Eng.*, vol. 4, no. 1, pp. 79–83, Feb. 2016.
- [25] R. Brendha and V. S. J. Prakash, "A survey on routing protocols for vehicular Ad Hoc networks," in *Proc. 4th Int. Conf. Adv. Comput. Commun. Syst. (ICACCS)*, 2017, pp. 1–7.
- [26] G. G. Tejani, V. J. Savsani, V. K. Patel, and S. Mirjalili, "An improved heat transfer search algorithm for unconstrained optimization problems," *J. Comput. Design Eng.*, vol. 6, no. 1, pp. 13–32, Jan. 2019.
- [27] M. Mavrovouniotis, C. Li, and S. Yang, "A survey of swarm intelligence for dynamic optimization: Algorithms and applications," *Swarm Evol. Comput.*, vol. 33, pp. 1–7, Apr. 2017.
- [28] J. Wang, J. Cao, B. Li, S. Lee, and R. S. Sherratt, "Bio-inspired ant colony optimization based clustering algorithm with mobile sinks for applications in consumer home automation networks," *IEEE Trans. Consum. Electron.*, vol. 61, no. 4, pp. 438–444, Nov. 2015.
- [29] D.-G. Zhang, J.-Y. Du, T. Zhang, and H.-R. Fan, "New method of traffic flow forecasting based on QPSO strategy for Internet of Vehicles," in *Proc. IEEE Int. Conf. Smart Internet Things (SmartIoT)*, Oct. 2020, pp. 102–108.
- [30] N. Jangir, M. H. Pandya, I. N. Trivedi, R. H. Bhesdadiya, P. Jangir, and A. Kumar, "Moth-flame optimization algorithm for solving real challenging constrained engineering optimization problems," in *Proc. IEEE Students' Conf. Electr., Electron. Comput. Sci. (SCEECS)*, Mar. 2016, pp. 1–5.
- [31] M. Shehab, L. Abualigah, H. Al Hamad, H. Alabool, M. Alshinwan, and A. M. Khasawneh, "Moth-flame optimization algorithm: Variants and applications," *Neural Comput. Appl.*, vol. 32, no. 14, 2020, pp. 9859–9884.
- [32] B. S. Yildiz and A. R. Yildiz, "Moth-flame optimization algorithm to determine optimal machining parameters in manufacturing processes," *Mater. Test.*, vol. 59, no. 5, pp. 425–429, 2017, doi: [10.3139/120.1110242017](https://doi.org/10.3139/120.1110242017).
- [33] S. Khalilpourazari and S. Khalilpourazary, "An efficient hybrid algorithm based on water cycle and moth-flame optimization algorithms for solving numerical and constrained engineering optimization problems," *Soft Comput.*, vol. 23, no. 5, pp. 1699–1722, 2019.
- [34] W. Yamany, M. Fawzy, A. Tharwat, and A. E. Hassanien, "Moth-flame optimization for training multi-layer perceptrons," in *Proc. 11th Int. Comput. Eng. Conf. (ICENCO)*, Dec. 2015, pp. 267–272.
- [35] M. Wang, H. Chen, B. Yang, X. Zhao, L. Hu, and Z. Cai, "Toward an optimal kernel extreme learning machine using a chaotic moth-flame optimization strategy with applications in medical diagnoses," *Neurocomputing*, vol. 267, pp. 69–84, Dec. 2017.
- [36] G. I. Sayed and A. E. Hassanien, "Moth-flame swarm optimization with neurosophic sets for automatic mitosis detection in breast cancer histology images," *Appl. Intell.*, vol. 47, no. 2, pp. 397–408, Sep. 2017.
- [37] M. A. El Aziz, A. A. Ewees, and A. E. Hassanien, "Whale optimization algorithm and moth-flame optimization for multilevel thresholding image segmentation," *Expert Syst. Appl.*, vol. 83, pp. 242–256, Oct. 2017.
- [38] D. H. Wolper and W. G. Macready, "No free lunch theorems for optimization," *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 67–82, Apr. 1997.
- [39] M. Ebeed, S. Kamel, and H. Youssef, "Optimal setting of STATCOM based on voltage stability improvement and power loss minimization using moth-flame algorithm," in *Proc. 18th Int. Middle East Power Syst. Conf. (MEPCON)*, Dec. 2016, pp. 815–820.
- [40] S. Mirjalili, "Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm," *Knowl.-Based Syst.*, vol. 89, pp. 228–249, Nov. 2015.
- [41] P. Tripathi and U. Tomar, "Solving economic load dispatch problems through moth flame optimization algorithm," *Singhal, Circuits*, vol. 3, no. 3, pp. 25–33, 2019.
- [42] P. Singh and S. Prakash, "Optical network unit placement in fiber-wireless (FiWi) access network by moth-flame optimization algorithm," *Opt. Fiber Technol.*, vol. 36, pp. 403–411, Jul. 2017.
- [43] J. E. Alvarez-Benitez, R. M. Everson, and J. E. Fieldsend, "A MOPSO algorithm based exclusively on Pareto dominance concepts," in *Proc. Int. Conf. Evol. Multi-Criterion Optim.* Springer, 2005, pp. 459–473.
- [44] F. Aadil, K. B. Bajwa, S. Khan, N. M. Chaudary, and A. Akram, "CACONET: Ant colony optimization (ACO) based clustering algorithm for VANET," *PLoS ONE*, vol. 11, no. 5, May 2016, Art. no. e0154080.
- [45] W. Shahzad, F. A. Khan, and A. B. Siddiqui, "Clustering in mobile ad hoc networks using comprehensive learning particle swarm optimization (CLPSO)," in *Communication Networking* Springer, 2009, pp. 342–349.



YASER ALI SHAH received the B.S. and M.S. degrees in computer systems engineering from the University of Engineering and Technology, Peshawar, in 2010 and 2013, respectively, and the Ph.D. degree in computer engineering from the University of Engineering and Technology, Taxila, Pakistan, in 2021. He is currently working as a Lecturer at COMSATS University Islamabad, Attock Campus, Pakistan. His research interests include vehicular *ad-hoc* networks, image and video processing, and bio-inspired algorithms.



FARHAN AADIL received the B.S. degree in computer science from Allama Iqbal Open University, Pakistan, in 2005, and the M.S. degree in software engineering and the Ph.D. degree in computer engineering from the University of Engineering and Technology, Taxila, Pakistan, in 2011 and 2016, respectively. He pursued a career in computer science for four years (2005 to 2009). He is currently working as an Assistant Professor at COMSATS University Islamabad, Attock Campus, Pakistan. His research interests include vehicular *ad-hoc* networks, intelligent transportation systems, and bio-inspired algorithms.



AMAAD KHALIL received the B.S. and M.S. degrees in computer systems engineering and the Ph.D. degree in computer engineering from the University of Engineering and Technology, Peshawar, Pakistan, in 2010, 2013, and 2021, respectively. He is currently working as a Lecturer with the University of Engineering and Technology, Peshawar. His research interests include wireless communications, intelligent transportation systems, and bio-inspired algorithms.



MUHAMMAD ASSAM received the B.Sc. degree in computer software engineering from the University of Engineering and Technology, Peshawar, Pakistan, in 2011, and the M.Sc. (18Y) degree in software engineering from the University of Engineering and Technology, Taxila, Pakistan, in 2018. He is currently pursuing the Ph.D. degree in computer science and technology with Zhejiang University, China. He has been working as a Lecturer (on study leave) with the Department of Software Engineering, University of Science and Technology Bannu, Khyber Pakhtunkhwa, Pakistan, since November 2011. His research interests include brain-machine interface, medical image processing, machine/deep learning, the Internet of Things (IoT), and computer vision.



IBRAHIM ABUNADI (Member, IEEE) received the Ph.D. degree in information systems from the School of Information Communication Technology, Griffith University, Australia. He is an Associate Professor; the Chair of the IS Department, College of Computer and Information Sciences, Prince Sultan University; and a fellow of the British Higher Education Academy. He taught many courses, including human-computer interaction, business process management, enterprise

architecture, technology innovations, business analysis, computer databases, and computer applications for business. He previously worked as an IT Analyst for Computer Associates and as a Strategic Consultant for the Saudi Computer Association. His research interests include data science, blockchain, health informatics, and organizational information systems. He has numerous publications in the field of information systems and continuously conducts reviews for many conferences and journals in the same fields. He is a member of the following associations: the Saudi and Australian Computer Societies, ACM, and the Association of Information Systems.



FAHD N. AL-WESABI received the B.S. degree in computer science from the University of Science and Technology, Sana'a, Yemen, in 2006, the M.S. degree in computer information systems from The Arab Academy for Banking and Financial Sciences, Sana'a, in 2009, and the Ph.D. degree in computer science from SRTM University, India, in 2015. From 2006 to 2009, he was a Research Assistant, and from 2010 to 2015, he was a Lecturer with the Faculty of Engineering, University

of Science and Technology. From 2015 to 2018, he was an Assistant Professor with the Faculty of Computer and Information Technology, Sana'a University, Yemen, and since October 2018, he has been an Assistant Professor with the Computer Science Department, King Khalid University, Saudi Arabia. He is the author of ten books, more than 80 articles, and many funded research projects. His research interests include AI, the IoT, smart cities, machine learning, biomedical, software engineering, applied soft computing, information security, and enterprise systems.

...

ALA SALEH ALLUHAIIDAN received the B.Sc. degree in computer science from Princess Nourah Bint Abdulrahman University, Riyadh, Saudi Arabia, the M.Sc. degree in computer information systems from Grand Valley State University, MI, USA, and the Ph.D. degree in information systems and technology from Claremont Graduate University, CA, USA. She is currently an Assistant Professor with the Department of Information Systems, Princess Nourah Bint Abdulrahman University. Her research interests include health informatics, big data analytics, and machine learning.