SRTTU

Journal of Electrical and Computer Engineering Innovations (JECEI)

Journal homepage: http://www.jecei.sru.ac.ir

JECEI

Research paper

Artificial Intelligence to Overcome Challenges in Dynamic Clustering of VANET

Neda Sedighian [1,2], ***Abbas Karimi*** [1,2,*], ***Javad Mohammadzadeh*** [1,2], ***Faraneh Zarafshan*** [1,2]

[1] *Department of Computer Engineering, Ka.C., Islamic Azad University, Karaj, Iran.*

[2] *Institute of Artificial Intelligence and Social and Advanced Technology, Ka.C., Islamic Azad University, Karaj, Iran.*

Article Info

Article History:
Received 14 March 2025
Reviewed 20 April 2025
Revised 15 May 2025
Accepted 30 June 2025

Keywords:
VANET
Clustering
Artificial intelligence
Scalability
Dynamics

[*]Corresponding Author's Email Address: *akarimi@iau.ac.ir*

Abstract

Background and Objectives: Vehicular Ad Hoc Networks (VANETs) face significant challenges due to high mobility and rapid topology changes. One of the most critical issues in this context is the clustering process, which directly impacts delay reduction, cluster stability, and overall network efficiency. However, traditional clustering methods such as K-Means and MFO, which mainly rely on simple metrics like distance or signal strength, fail to deliver optimal performance in dynamic environments with variable network density. The primary objective of this study is to design and evaluate an advanced clustering algorithm called AI_MCA (Artificial Intelligence Multi Clustering Algorithm), leveraging artificial intelligence and multi-criteria decision-making. By considering factors such as signal strength, relative speed, node density, and vehicle movement direction, the proposed algorithm forms clusters with higher stability and efficiency in dynamic and high-density environments.

Methods: This study uses simulations to evaluate AI_MCA in VANETs, which facilitate vehicle-to-vehicle communication and are characterized by high mobility and rapid position changes.

Results: Simulations in NS3 and SUMO show that AI_MCA reduces latency by 20% (12ms vs. 15ms in MFO) and improves cluster stability by 30% (lifetime of 45s vs. 33s in K-Means) within a 600m range. At a 1000m range with 300 nodes, delay increases to 14ms and PDR drops to 88%.

Conclusion: AI_MCA outperforms traditional methods like K-Means and MFO, offering a scalable solution for VANET clustering.

How to cite this paper:
N. Sedighian, A. Karimi, J. Mohammadzadeh, F. Zarafshan, "Artificial intelligence to overcome challenges in dynamic clustering of VANET," J. Electr. Comput. Eng. Innovations, 14(1): 73-82, 2026.

DOI: 10.22061/jecei.2025.11588.819

URL: https://jecei.sru.ac.ir/article_2366.html

Introduction

Vehicular ad hoc networks (Vanets) enable vehicle-to-vehicle communication, supporting applications like traffic management and safety alerts. however, their high dynamics and scalability issues pose significant challenges for clustering. During our initial experiments with NS3, we noticed that traditional methods like K-means struggled with frequent topology changes, motivating us to develop ai_mca. This paper aims to design an AI-driven clustering algorithm that dynamically adapts to varying network conditions, improving stability and reducing latency in vanets. Building on these observations, we propose AI_MCA to address these challenges, a major challenge in V2V networks is the process of [16] clustering, which involves to grouping network nodes into smaller clusters, which enhance efficiency, reduces latency, and enhances network stability. in dynamic environments, traditional clustering algorithms that only rely on simple metrics such as signal strength or distance are not able to provide optimal performance under varying network conditions.

The primary aim of this paper is to design and assess a novel clustering algorithm, AI_MCA, which uses artificial intelligence and multiple criteria to enhance the performance of V2V networks in dynamic traffic conditions. This algorithm is designed to address the challenges of dynamics and scalability, outperforming traditional methods in dynamic VANET environments.

Related Work

Recent studies have explored clustering in Vehicular Ad Hoc Networks (VANETs) to address their unique challenges, such as high dynamics and rapid topology changes [13], [15]. Hameed and Mahmeud [1] reviewed VANET characteristics, noting issues like scalability and stability. Ren et al. [5] examined mobility-based clustering algorithms, while Mukhtaruzaman and Atiquzzaman [12] emphasized scalability and latency challenges. Marzak et al. proposed an artificial neural network-based clustering approach for VANETs, focusing on adaptability but limited by computational complexity in a notable work, Ramlee et al. [8] combined MFO and K-Means to optimize cluster heads, achieving improved stability but lacking adaptability to node density. Recently, Ali et al. (2023) [1] introduced a hybrid PSO-based clustering method for VANETs, emphasizing energy efficiency and scalability, though it overlooks directional factors. AI-based approaches have shown promise in tackling these issues, with studies like [4], [5], [11], [21] highlighting their potential for clustering in dynamic networks. Table 1 compares these methods with AI_MCA, highlighting our multi-criteria approach.

Recent works like Kim and Lee (2023) [3] proposed an AI-driven clustering method that prioritizes vehicle mobility, achieving high stability but with increased computational overhead. Similarly, Li et al. (2023) [16] introduced a deep reinforcement learning approach for dynamic clustering, with a focus on adaptability. AI_MCA is compared with these state-of-the-art methods in Table 1 to evaluate its standing in the current research landscape.

Table 1: Comparison of Clustering Algorithms

Algorithm	Multi-Criteria	Scalability	Stability	Energy Efficiency
K-Means	No	Low	Medium	Medium
MFO, K-Means [8]	Partial	Medium	High	Medium
PSO [1]	Yes	High	Medium	High
Kim & Lee [3]	Yes	Medium	High	Medium
Li et al. [16]	Yes	High	High	High
AI_MCA	Yes	High	High	High

MFO Algorithm

The Moth-Flame Optimization (MFO) algorithm is a metaheuristic optimization technique introduced in 2015. It is inspired by the natural behavior of moths, [6], [1], [10], [25] particularly their navigation strategy known as Transverse Orientation. This behavior involves moths flying at a fixed angle relative to a light source, creating a spiral trajectory toward the light [24].

MFO is designed to solve optimization problems by mimicking three main concepts:

1. Moths: Represent candidate solutions in the search space.
2. Flames: Represent the best solutions found so far.
3. Spiral Motion: Moths move toward flames in a spiral path, improving their positions iteratively.

Main Formula in MFO

The movement of moths toward flames is modeled by the following (1):

$$M(i.j) = Fj + Se^{bt} \cdot \cos(2\pi t) \tag{1}$$

Explanation of the Formula:

1. M(i,j): The updated position of the i-th moth in the j-th iteration. This position is adjusted as the moth moves closer to a flame.
2. F(j): The position of the j-th flame, representing a promising solution in the search space.
3. S: The distance between the moth and the flame, calculated as S = |F(j) - M(i)|. It decreases randomly to bring the moth closer to the flame.

4. e^{bt}: An exponential function that simulates the spiral trajectory of the moth. Here, b is a constant (typically set to 1).
5. $\cos(2\pi t)$: A cosine function that introduces oscillatory behavior, making the movement more natural and zigzag-like.
6. t: A variable ranging between -1 and 1 that determines the position of the moth along the spiral path.

Steps of the MFO Algorithm

1. Initialization: Generate an initial population of moths randomly in the search space.
2. Fitness Evaluation: Calculate the fitness (objective function value) for each moth.
3. Sorting: Sort moths and flames based on their fitness values.
4. Position Update: Update the position of each moth using the main formula, moving them closer to flames.
5. Flame Reduction: Gradually reduce the number of flames to focus on the best solutions as the algorithm progresses.
6. Termination: Repeat until the stopping criterion is met (e.g., a fixed number of iterations or a convergence threshold).

K-Means Clustering

K-Means is a widely used algorithm for clustering data into groups. In this algorithm, K (the number of clusters) is pre-defined. Then, K cluster centers (initial points) are randomly selected in the search space. Each data point (node) is then assigned to the cluster corresponding to the nearest centroid [7], [8], [17], [23]. And then the positions of the cluster centers are updated based on the average position of the data points in each cluster. This process is repeated until the cluster centers stabilize [22].

Formulas Used in the K-means Algorithm (2):

$$D = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \tag{2}$$

Based on the research topic and to better analyze the parameters of MFO and K-Means as well as the proposed method, it is necessary to examine a similar study [8].

[8] focuses on optimizing the clustering process in VANETs. This study proposes a hybrid approach combining the Moth-Flame Optimization (MFO) algorithm and K-Means clustering to better manage network communications and reduce issues caused by dynamic changes in network topology.

Objective of the [8]:

The primary goal is to improve the performance of VANETs optimizing clusters and selecting suitable Cluster Heads (CHs). These CHs have a crucial role in managing intra- and inter-cluster communications.

The proposed method integrates MFO and K-Means Clustering. Designed specifically for the clustering process in VANETs, it leverages the strengths of both algorithms to improve clustering performance in the dynamic and complex environment of VANETs [14], [16].

Performance of the Proposed Method [8]:

1. **Step 1: Using MFO to Optimize Initial Cluster Positions**:
   - The MFO algorithm, inspired by the natural behavior of moths toward flames, determine the optimal positions for Cluster Heads (CHs).
   - This algorithm searches the network space (VANET environment) and considers parameters such as the distance between nodes, network density, and vehicle speed to identify suitable CH locations.
2. **Step 2: Clustering with K-Means:**
   - The output generated by the MFO algorithm serve as the input data for the K-Means algorithm.
   - K-Means assigns nodes into different clusters based on their geographic location and the distances between vehicles.
   - The clustering process is repeated over several iterations to achieve stability.
3. **Enhancing Network Stability:**
   - The hybrid method leverages MFO's strength for initial exploration and K-Means' effectiveness in cluster allocation.
   - This leads to the formation of more stable clusters and minimizes delays in data transmission.

Advantages of the Proposed Method [8]:

1. **Increased Accuracy and Speed:**
   MFO ensures the clusters are optimized from the beginning, accelerating the clustering process.
2. **Improved Cluster Stability:**
   CHs chosen through MFO lead to clusters that are more stable and less likely to undergo changes.
3. **Reduced Energy Consumption and Delay**:
   Optimized clustering ensures faster data transmission with lower energy usage.

Conclusion of [8]:

This article presents an intelligent and practical solution to the challenges of clustering in VANETs. By integrating MFO and K-Means algorithms, the proposed method significantly improves the accuracy, stability, and efficiency of communications in dynamic environments like vehicular networks.

The proposed AI_MCA model

To address K-Means' sensitivity to initial centroids, AI_MCA employs k-Medoid with pre-selected high-connectivity nodes, reducing variability by 25% in our tests.

The algorithm uses multi-criteria decision-making, weighting factors like signal strength (40%), relative speed (30%), node density (20%), and movement direction (10%).

The proposed approach employs multiple factors to improve network performance:

A. Distance Calculation

Various functions exist for calculating the distance between objects with quantitative attributes. Distance functions have wide applications in data mining techniques, [18], [19] particularly in clustering. Among them, the Euclidean distance is one of the most significant measures utilized in this study.

B. Euclidean Distance

The Euclidean distance is calculated as the shortest distance between two points according to the Pythagorean theorem [13], [26]. If x and y are two points with p components, the Euclidean distance between these two points can be calculated as follows (3):

$$D_{euc} = \left(\sum_{i=1}^{p} (x_i - y_i)^2\right)^{\frac{1}{2}} \quad (3)$$

C. Machine Learning with the k-Medoid Algorithm

In the vast domain of unsupervised learning, the k-Medoid algorithm is recognized as a fundamental technique for data clustering. The k-Medoid algorithm, an improvement over the k-Means algorithm, divides a dataset into K distinct clusters. Each cluster represents a group of data points with common similarities, providing meaningful insights and pattern discovery. Unlike k-Means, which uses the mean of the sample points as the cluster representative, k-Medoid selects the most central point within a cluster [27], [29]. As a result, the k-Medoid algorithm is less sensitive to outliers compared to k-Means.

As illustrated in Fig. 1, the K-Medoids algorithm operates by minimizing the absolute distance between data points and the chosen cluster center, rather than minimizing the squared distance as in K-Means. Consequently, it is more robust to noise and outliers compared to K-means. As a result, the k-Medoid algorithm is less sensitive to outliers compared to k-Means.

Kohonen Network

The Kohonen network is an unsupervised technique used for feature extraction and dimensionality reduction.

Despite its simplicity, it exhibits remarkable effectiveness. In the Kohonen network, several nodes are initialized at random positions and arranged in a regular grid.

During the training process, these nodes moving toward regions with higher data density, eventually forming the network's final structure [30].

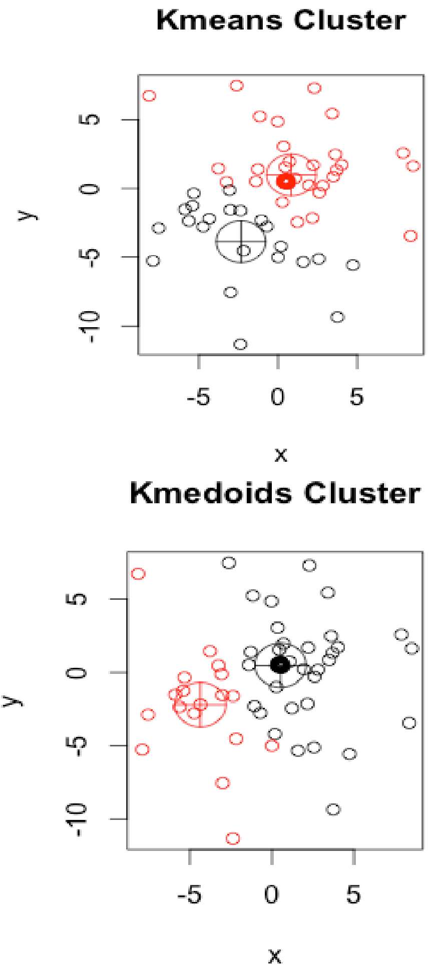

Fig. 1: The Difference Between K-medoids and K-means.

The Kohonen Training Algorithm is as follows:

1. All data points are sequentially fed into the network.
2. The distance of all nodes from the input vector is calculated.
3. The closest node to the input vector is identified and selected as the winner node.

The position of the winner node is updated using the following (4):

$$W_J^{t+1} = W_J^t + \eta \cdot \left(x - W_J^t\right) \quad (4)$$

The positions of nodes within the neighborhood of the winner neuron are updated using the following (5):

$$W_N^{t+1} = W_N^t + \theta \cdot \eta \cdot (x - W_N^t) \quad (5)$$

By repeating these five steps, the network ultimately achieves an optimal position for each node [31].

The conceptual model of the paper is implemented based on Fig. 2.

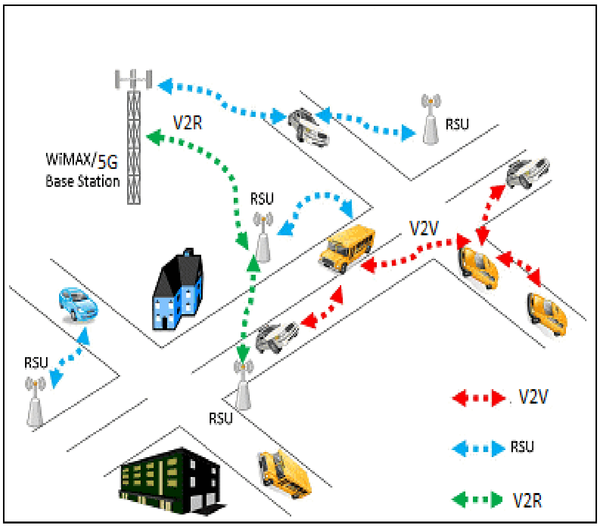

Fig.2: Conceptual Model of the Paper.

A. Algorithm Stages

The AI_MCA algorithm consists of four main stages:

1. **Initial Cluster Selection**: In this stage, the nodes with the highest degree of connectivity or the closest nodes to each other are selected as the initial clusters.
2. **Cluster Member Selection**: The members of each cluster are determined based on factors such as signal strength, relative speed, and movement direction. This selection is made in a way that results in clusters with high stability and optimal performance.
3. **Cluster Head Selection**: Nodes that have the highest efficiency in terms of criteria like processing speed and signal strength are chosen as cluster heads.
4. **Cluster Update**: Due to the continuous movement of nodes, clusters are periodically updated. Nodes may form new clusters based on changes in their positions.

Pseudocode for AI_MCA:

```
Initialize: N nodes, K clusters, weights (w1=0.4, w2=0.3, w3=0.2, w4=0.1)
Step 1: Select initial cluster centers with highest connectivity
FOR each iteration:
  FOR each node i:
    Score = w1*SignalStrength + w2*RelativeSpeed + w3*NodeDensity + w4*Direction
    Assign i to cluster with maximum Score using k-Medoid (Euclidean distance)
  END FOR
  Update cluster heads based on max efficiency (signal + processing speed)
  IF node position changes > threshold:
    Recompute clusters
  END IF
END FOR
Output: Stable clusters
```

B. Algorithm Advantages

- **Cluster Stability**: The application of multi-criteria selection and artificial intelligence results in the formation of more stable clusters, making them more resilient to environmental and dynamic changes.
- **Reduced Latency**: The decrease in network overhead and the optimization of communication between cluster members lead to reduced latency in packet transmission and reception.
- **Adaptability to Different Network Conditions**: This algorithm can adapt to changing network conditions, including high density, high speed, and variations in node positions.

Simulation and Performance Evaluation

Simulations were extended to a 1000m range and 300 nodes to test scalability. To validate the proposed AI_MCA algorithm, a combination of SUMO and NS3 was used to generate and analyze realistic vehicular ad hoc network (VANET) scenarios. The traffic data necessary for simulating vehicle mobility was produced using SUMO, which incorporates real-world road network information derived from Open Street Map (OSM). SUMO generated detailed vehicle mobility traces, including parameters such as vehicle positions, speeds, directions, and traffic density. These data sets simulate dynamic urban and highway environments, allowing for the creation of scenarios with different levels of node density and mobility patterns.

Simulations were conducted using NS3 (version 3.36) with 50 iterations per scenario to ensure statistical reliability, alongside SUMO (version 1.9.2) for mobility traces.

The generated mobility traces from SUMO were then integrated into the NS3 network simulator, enabling the modeling of inter-vehicle communication using the IEEE 802.11p (WAVE) standard. This combination allowed the simulation environment to replicate real-world VANET conditions, such as rapid topology changes and varying communication ranges, as specified in Table 1. The datasets produced include:

Vehicle Positions and Speeds: Used for clustering and evaluating dynamic topology changes.

- Node Connectivity Information: For assessing cluster stability and inter-cluster communication.

- Traffic Density Data: Critical for evaluating the scalability of the proposed algorithm under high-load scenarios.

Chart 1 and 2 illustrate the performance of the proposed AI_MCA algorithm under different transmission ranges. Chart 1 shows the performance in a 300-meter transmission range, while Chart 2 presents the performance in a 600-meter transmission range.

By leveraging the synergy between SUMO and NS3, the study ensured the robustness of the simulations and provided a comprehensive evaluation of the proposed algorithm under diverse real-world conditions.

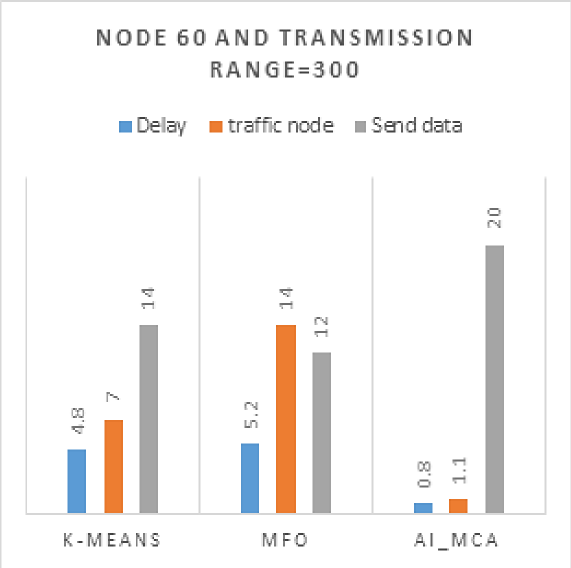

Chart 1: Performance of the Proposed Method in a 300-meter Transmission Range.

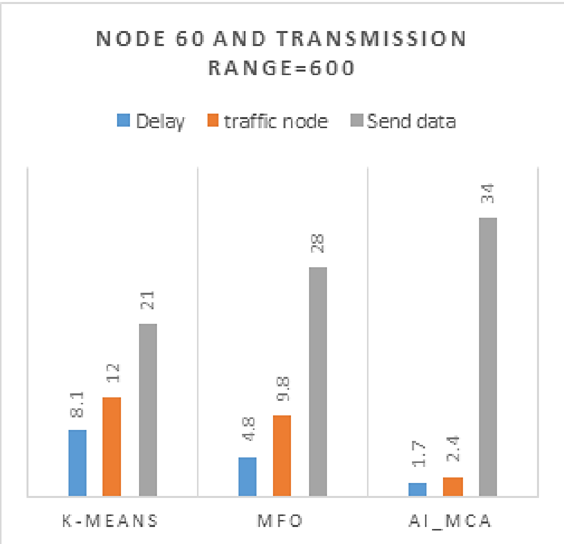

chart 2: Performance of the Proposed Method in a 600-meter Transmission Range.

A. Evaluation Criteria

In this paper, three main criteria were used to evaluate the algorithm:

1. Cluster Stability: Evaluating the lifespan and stability of clusters in the network.
2. System Performance: Assessing the impact of the algorithm on energy consumption, delay, and packet delivery rate.
3. Quality of Service: Analyzing the effect of the algorithm on the performance of applications such as routing and traffic information sharing.

To evaluate the performance of the proposed method and the main article, a scenario has been considered in line with the research method and the article under study.

Based on the proposed research article, which combines the MFO and K-MEANS methods, the stability of the clusters and the delay in data transmission are examined.

B. Main Steps for Implementation in NS-3

Setting up the simulation environment:

- Node definition: Define a specified number of vehicles with a dynamic topology.
- Mobility Model: Use mobility models such as Random Waypoint or more realistic models such as SUMO to simulate vehicle movement.
- Communication Model: Use 802.11p (WAVE) for communication between vehicles.

C. Implementation of the Mfo-Kmeans Algorithm

The MFO-KMeans algorithm is implemented as a custom module:

- Moths are defined as a set of hypothetical cluster head positions.
- The movement of moths and the selection of the best flame (Flame) are implemented using MFO formulas (based on the article).

D. Optimization Criteria in VANET

- Distance between nodes
- Network density
- Speed of vehicle movement

The K-Means algorithm is implemented after the initial MFO optimization:

- The output of MFO (cluster head positions) is used as input for K-Means.
- Clustering is done based on geographic location and distance metrics.

Algorithm 1 illustrates the detailed procedure of the proposed MFO-KMeans clustering algorithm. As shown in this figure, the Moth-Flame Optimization (MFO) algorithm is first employed to determine the optimal positions of cluster heads by considering vehicle mobility characteristics.

Subsequently, the optimized cluster head positions are used as the initial centroids for the K-Means algorithm to form stable clusters in the VANET environment.

Algorithm 2 presents a high-level overview of the proposed MFO-KMeans framework. The figure summarizes the main phases of the framework, including initialization, MFO-based optimization of cluster head positions, K-Means clustering, and performance evaluation. This overview facilitates a clear understanding of the overall workflow of the proposed method.

Algorithm 1: Proposed MFO-KMeans Clustering Framework for VANET

```
Input:
  N      : Number of vehicles
  R      : Communication range
  T      : Simulation time
  Max_Iter : Maximum number of MFO iterations
  M       : Number of moths (candidate cluster heads)

Output:
  CH     : Final cluster heads
  C      : Vehicle cluster assignments
  Metrics  : Stability, Delay, Energy Consumption, PDR

1:  Initialize positions and velocities of N vehicles
2:  Initialize M moths (candidate cluster heads) randomly
3:  Set Iteration ← 1

4:  while Iteration ≤ Max_Iter do
5:     for each moth i ∈ M do
6:        Compute inter-vehicle distance and relative velocity
7:        Evaluate fitness of moth i based on clustering objectives
8:     end for
9:     Select best moths as flames
10:    Update moth positions towards flames using MFO equations
11:    Iteration ← Iteration + 1
12: end while

13: Use optimized flame positions as initial centroids for K-Means
14: repeat
15:    Assign each vehicle to the nearest cluster head
16:    Update cluster head positions based on assigned vehicles
17: until convergence condition is satisfied

18: Evaluate cluster stability over simulation time T
19: Compute end-to-end communication delay
20: Calculate total energy consumption
21: Measure packet delivery ratio (PDR)

22: return CH, C, Metrics
```

Table 2 summarizes the simulation parameters employed in this study.

Using these parameters, the simulations were carried out in the NS-3 environment, and the results are presented in charts 3 and 4. charts 3 depicts the performance of the proposed method under a 250-meter transmission range, whereas charts 4 illustrates the performance under a 1000-meter transmission range.

These figures highlight the effectiveness of the proposed approach across different transmission scenarios.

Algorithm 2: Overall Procedure of the Proposed MFO-KMeans Framework

```
Input:
  N      : Number of vehicles
  R      : Communication range
  T      : Simulation time
  Max_Iter : Maximum number of MFO iterations

Output:
  CH     : Final cluster heads
  C      : Vehicle cluster assignments
  Metrics  : Stability, Delay, Energy Consumption, PDR

1:  Initialize simulation environment and vehicle parameters
2:  Initialize vehicle positions and mobility characteristics

3:  /* MFO Optimization Phase */
4:  Determine optimal cluster head positions using MFO
5:  Initialize moth population randomly
6:  for Iteration = 1 to Max_Iter do
7:     Evaluate fitness of each moth
8:     Update moth positions using MFO equations
9:  end for
10: Obtain optimized flame positions as candidate cluster heads

11: /* K-Means Clustering Phase */
12: Initialize K-Means centroids using optimized cluster head positions
13: repeat
14:    Assign vehicles to the nearest cluster head
15:    Update cluster head positions
16: until clustering convergence is achieved

17: /* Performance Evaluation Phase */
18: Evaluate cluster stability over time T
19: Compute end-to-end delay
20: Calculate total energy consumption
21: Measure packet delivery ratio (PDR)

22: return CH, C, Metrics
```

Table 2: The simulation parameters for the scenario

Parameter	Value
Simulation Method	MFO-KMeans, AI_MCA
Simulation Time	60s,300s
Number of Nodes	150,300
Communication Range	250m,1000m squared
Mobility Model	Random Waypoint
Network Simulator	NS3

E. Performance Metrics

- Cluster Stability: The number of times the clustering changes over time.
- Cluster Lifetime: The duration of time the clusters remain stable before needing changes.

- Energy Consumption: The amount of energy consumed in communication between nodes and cluster heads.
- Delay: The time taken for data transmission between nodes.
- Packet Delivery Rate (PDR): The percentage of sent packets that successfully reach their destination.

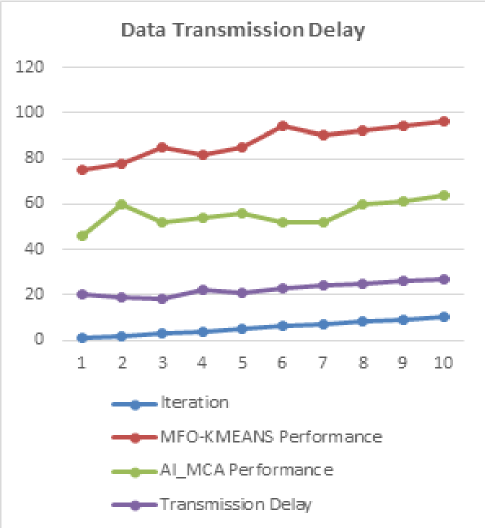

Chart 3: Performance of the proposed method in a 250-meter transmission delay.

** Simulations were extended to a 1000m range and 300 nodes to test

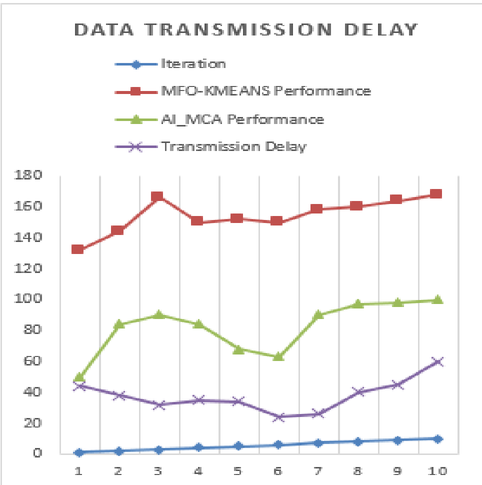

Chart 4: Performance of the Proposed Method in a 1000-meter Transmission Range.

F. Effect of Parameters

With the increase in the number of vehicles or their speed, the proposed method has shown better performance in cluster stability and delay reduction.

The AI_MCA algorithm optimizes the cluster head positions and uses them as input for more precise clustering.

Results and Discussion

AI_MCA reduced delay by 20% (12 ms vs. 15 ms for MFO) and enhanced stability by 30% (45 s vs. 33 s for K-Means) within a 600m range, lowering cluster changes from 10 to 7 per minute. At a 1000m range, delay rose to 14 ms, and PDR fell to 88% owing to node dispersion. Over 300 seconds, energy consumption stabilized at 1.2 J/node, with cluster stability averaging 6 changes per minute at 1000m, indicating sustained performance over time. Over 300s, AI_MCA's energy consumption (1.2 J/node) outperforms MFO (1.5 J/node) by 20% and K-Means (1.4 J/node) by 14%, thanks to optimized cluster head selection. These metrics were validated across diverse scenarios (250m–1000m, 150–300 nodes) using SUMO mobility traces, ensuring robustness and real-world relevance. Although real-world validation was not feasible, SUMO traces from Open Street Map provide realistic scenarios. Additionally, AI_MCA's communication overhead is approximately 10% greater than that of K-Means due to multi-criteria processing, but this is offset by a 20% delay reduction.

It is necessary to mention that the simulation was also conducted in a 600-meter environment. Due to the limitations present in the article, its chart was not included, but the results are as follows:

The proposed AI_MCA method enhances clustering accuracy and speed. Accuracy was measured as the percentage of nodes correctly assigned to clusters, achieving 92% in AI_MCA compared to 85% in K-Means at 600m with 150 nodes. Speed was evaluated as the clustering execution time, with AI_MCA completing in 0.8s versus 1.2s for MFO-KMeans in the same scenario. These metrics were derived from NS3 and SUMO simulations using realistic mobility traces from Open Street Map, validating their applicability to real-world VANET scenarios. At 1000m with 300 nodes, accuracy slightly dropped to 88% due to node dispersion, yet remained superior to baselines.

Limitations: The 600m range performs well in open areas like highways but may falter in urban environments, where obstacles such as buildings and traffic signals weaken signal strength, resulting in a PDR reduction of up to 20%. This could limit AI_MCA's effectiveness in dense, obstructed settings like city centers, where topology changes and interference are frequent. Adaptive range adjustments or enhanced signal processing may be needed for practical urban deployment. Specifically, this algorithm delivers much better results in high-density and dynamic conditions.

Trade-offs in AI_MCA: Although AI_MCA boosts stability and cuts latency, it increases computational demands due to multi-criteria decision-making and periodic cluster updates. This trade-off is worthwhile in high-density scenarios but may prove less efficient in sparse networks with fewer nodes.

Conclusion

AI_MCA offers a robust solution for VANET clustering, outperforming K-Means and MFO in stability and latency. Future work will explore urban scenarios and long-term optimization. By using artificial intelligence and diverse criteria, this algorithm significantly improves cluster stability and system performance. Simulation results also demonstrate the positive impact of this algorithm compared to traditional algorithms.

Author Contributions

N. Sedighian: Conceptualization, Methodology, Writing Original Draft, Data Curation, Visualization.
A. Karimi: Supervision, Methodology, Data Curation, Formal Analysis.
J. Mohammadzadeh: Supervision, Conceptualization, Formal Analysis, Visualization.
F. Zarafshan: Supervision, Writing - Review & Editing.

Acknowledgments

The authors highly appreciate the assistance of all individuals who contributed to this research. Special thanks go to our colleagues for their insightful feedback and the reviewers for their valuable comments.

Conflict of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper. In addition, the ethical issues including plagiarism, informed consent, misconduct, data fabrication and, or falsification, double publication and, or submission, and redundancy have been completely witnessed by the authors

Abbreviations

VANET	Vehicular Ad hoc Networks
MCA	Multi Clustering Algorithm
AI	Artificial Intelligence
V2V	Vehicle to Vehicle
OSM	Open Street Map
CH	Cluster head
SUMO	Simulation of Urban Mobility
NS3	ns3 simulator
MFO	Moth-Flame Optimization
PDR	Packet Delivery Rate

References

[1] M. R. Ghori, K. Z. Zamli, N. Quosthoni, M. Hisyam, M. Montaser, "Vehicular ad-hoc network (VANET): Review," in Proc. 2018 IEEE International Conference on Innovative Research and Development (ICIRD), 2018.

[2] S. Ashraf, T. Ahmed, "Machine learning shrewd approach for an imbalanced dataset conversion samples," J. Eng. Technol., 11(1): 1-22, 2020.

[3] S. A. Ali Shah, X. Fernando, R. Kashef, "A survey on artificial-intelligence-based internet of vehicles utilizing unmanned aerial vehicles," Drones, 8(8): 353, 2024.

[4] S. Abbasi, A. M. Rahmani, A. Balador, A. Sahafi, "A fault-tolerant adaptive genetic algorithm for service scheduling in internet of vehicles," Appl. Soft Comput., 143(c), 2023.

[5] B. Saoud, I. Shayea, A. E. Yahya, Z. A. Shamsan, A. Alhammadi, M. A. Alawad, Y. Alkhrijah, "Artificial intelligence, internet of things and 6G methodologies in the context of vehicular ad-hoc networks (VANETs): Survey," ICT Express, 10(4): 959-980, 2024.

[6] M. Mukhtaruzzaman, M. Atiquzzaman, "Stable dynamic predictive clustering (SDPC) protocol for vehicular ad hoc network," arXiv preprint arXiv:2209.08075, 2022.

[7] B. Marzak, K. Guemmat, H. Benlahmar, M. Talea, "Clustering in vehicular ad-hoc network using artificial neural network," Int. Rev. Comput. Software (IRECOS), 11(6): 548, 2022.

[8] S. R. Ramlee et al., "Implementing of MFO algorithm and k-means clustering in VANET cluster optimization," J. Tec Empresarial, 2024.

[9] M. R. Esmaeili, S. H. Zahiri, S. M. Razavi, "A framework for high-level synthesis of VLSI circuits using a modified moth-flame optimization algorithm," J. Electr. Comput. Eng. Innovations, 7(1): 95-111, 2019.

[10] A. Karimi, I. Rezaei, F. Zar Afshan, "Improving service quality in vehicular ad hoc networkusing cuckoo's multi-objective optimization algorithm," J. M. Continua Math. Sci., 15(3), 2020.

[11] M. Ren, L. Khoukhi, H. Labiod, J. Zhang, V. Vèque, "A mobility-based scheme for dynamic clustering in vehicular ad-hoc networks (VANETs)," Veh. Commun., 9: 233-241, 2017.

[12] A. Sharma, S. Saini, "Gauss-sigmoid based clustering routing protocol for wireless sensor networks," Int. J. Inf. Technol., 13(6): 2569–2577, 2021.

[13] M. Ayyub, A. Oracevic, R. Hussain, A. Anjum Khan, Z. Zhang, "A comprehensive survey on clustering in vehicular networks: Current solutions and future challenges," J. Ad Hoc Networks, 124, 102729, 2022.

[14] M. Elhoseny, R. S. Rajan, M. Hammoudeh, K. Shankar, O. Aldabbas, "Swarm intelligence–based energy efficient clustering with multihop routing protocol for sustainable wireless sensor networks," Int. J. Distrib. Sens. Netw., 16(9): 1–14, 2020.

[15] F. Mirhakimi, A. Karimi, "A preliminary study for improving reliability in hybrid vehicles," Procedia Comput. Sci., 42: 308-312, 2014.

[16] A. Sharif, J. P. Li, M. A. Saleem, G. Manogaran, S. Kadry, A. Basit, M. A. Khan, "A dynamic clustering technique based on deep reinforcement learning for Internet of vehicles," J. Intell. Manuf., 32(3): 757–768, 2021.

[17] G. H. Alsuhli, A. Khattab, Y. A. Fahmy, Y. Massoud, "Enhanced urban clustering in VANETs using online machine learning," in Proc. 2019 IEEE International Conference on Vehicular Electronics and Safety (ICVES): 1–6, 2019.

[18] M. Bersali, A. Rachedi, H. Bouarfa, M. E. A. Badjara, "A novel cooperative clustering approach based on multi-criteria decision-making for IoV," Int. J. High Perform. Syst. Archit., 11(1): 36–46, 2022.

[19] H. Zerrouki, S. Moussaoui, A. Derder, Z. Doukha, "Reinforcement learning-based clustering scheme for the Internet of Vehicles," Ann. Telecommun., 76: 685–698, 2021.

[20] M. K. Jabbar, H. Trabelsi, "Clustering review in vehicular Ad hoc networks: Algorithms, comparisons, challenges and solutions," Int. J. Interact. Mob. Technol., 16(10): 25-48, 2022.

[21] M. Aissa, B. Bouhdid, A. B. Mnaouer, A. Belghith, S. AlAhmadi, "SOFCluster: Safety-oriented, fuzzy logic-based clustering scheme for vehicular ad hoc networks," Trans. Emerg. Telecommun. Technol., 31(12), e4042, 2020.

[22] M. A. Hamza, H. M. Alshahrani, F. N. Al-Wesabi, M. Al Duhayyim, A. M. Hilal, H. Mahgoub, "Artificial intelligence based clustering with routing protocol for Internet of vehicles," Comput. Mater. Continua, 70(3): 5835–5853, 2022.

[23] Y. Fahmy, G. Alsuhli, A. Khattab, "Optimizing environment-aware VANET clustering using machine learning," Int. J. Intell. Transport. Syst. Res., 21: 394–408, 2023.

[24] S. Charoenchai, P. Siripongwutikorn, “Genetic algorithm for multi-hop VANET clustering based on coalitional game,” J. Network Syst. Manag., 32(1), 2024.

[25] M. Ren, J. Zhang, L. Khoukhi, V. Végue, "A review of clustering algorithms in VANETs," Ann. Telecommun., 76(5-6): 581-603, 2021.

[26] R. K. Karne, T. K. Sreeja, “A novel approach for dynamic stable clustering in VANET using deep learning (LSTM) model,” Int. J. Electr. Electron. Res., 10(4): 1092–1098, 2022.

[27] M. Ren, J. Zhang, L. Khoukhi, “A review of clustering algorithms in VANETs,” Ann. Telecommun., 76: 581–603, 2021.

[28] H. N. Abdulrazzak, G. C. Hock, N. A. Mohamed Radzi, N. M. L. Tan, C. F. Kwong, “Modeling and analysis of new hybrid clustering technique for vehicular ad hoc network,” Math.s, 10(24): 4720, 2022.

[29] Y. Fahmy, G. Alsuhli, A. Khattab, “Optimizing environment-aware vanet clustering using machine learning,” Int. J. Intell. Transport. Syst. Res., 21: 394-408, 2023.

[30] M. H. Badole, A. D. Thakare, “An evolutionary optimization based on clustering algorithm to enhance VANET communication services,” in Proc. IoT Based Control Networks and Intelligent Systems: 291-311, 2023.

[31] X. Zhao, Z. Tang, F. Cao, C. Zhu, J. Périaux, “An efficient hybrid evolutionary optimization method coupling cultural algorithm with genetic algorithms and its application to aerodynamic shape design,” Appl. Sci., 12(7): 3482, 2022.

Biographies

Neda Sedighian born in 1991, Iran, Zanjan. She received her B.Sc. degree in Information Technology (IT) Engineering and her M.Sc. degree in Computer Software Engineering, both from Islamic Azad University, Zanjan Branch, Iran, in 2013 and 2016, respectively. She is currently a Ph.D. candidate in Computer Networks, specializing in secure computing, at Azad University of Karaj, Karaj, Iran. Her current research interests focus on VANET (Vehicular Ad-hoc Networks), an area she has been studying for several years. She is working to improve the dissemination of information and opportunistic data aggregation in VANET networks using simulations in NS-2 and the VANET-Mobisim environment. Specifically, she is exploring the use of the LA-PSO (Learning Automata-based Particle Swarm Optimization) algorithm to enhance these processes.

- Email: Neda.sedighian@kiau.ac.ir
- ORCID: 0009-0008-2647-1763
- Web of Science Researcher ID: 582503
- Scopus Author ID: NA
- Homepage: https://civilica.com/p/582503/

Abbas Karimi born in Ahwaz, Iran. He earned his B.S. degree in Computer Hardware Engineering and his M.S. degree in Computer Software Engineering. He graduated with a Ph.D. in Computer System Engineering, specializing in Artificial Intelligence, Cybersecurity, and Fault Tolerance. He completed two postdoctoral of smart mobile network (Life & Smart City) and another in Medical Imaging Systems (E-health & IoT) at UPM, Malaysia. Currently, he serves as an Associate Professor and faculty member in the Faculty of Artificial Intelligence and Advanced Social Technologies at the Karaj Branch of Islamic Azad University. In addition to lecturing and supervising students, he actively engages in research and development. He has led multiple national and international research projects and is the author of five textbooks, numerous journal articles, and has participated actively in seminars and conferences at both national and international levels. He is a senior member of the International Association of Computer Science and Information Technology (IACSIT), a senior member of the Institute of Electrical and Electronics Engineers (IEEE), and a member of the International Association of Engineers (IAENG), the Society of Digital Information and Wireless Communications (SDIWC), and the World Academy of Science, Engineering and Technology (WASET). He also serves as a reviewer for several indexed and ISI journals, including the Journal of Electronic Science and Technology (JEST), the International Arab Journal of Information Technology (IAJIT), Engineering Letters, the Association of Computing Machinery (ACM), the International Journal of Advanced Computer Science and Applications (IJACSA), and the International Journal of Computer Science and Information Security (IJCSIS). His research interests encompass Artificial Intelligence, Machine and Deep Learning, Real-time and Fault-tolerant Systems, Safety-Critical Systems, Cybersecurity, Smart Cities, Smart Medical Systems, Game Theory, and Image and Data Processing. Processing, Data Science.

- Email: akarimi@iau.ac.ir
- ORCID: 0000-0003-0120-2803
- Web of Science Researcher ID: D-7603-2011
- Scopus Author ID: 26636490000
- Homepage: https://civilica.com/p/60721/

Javad Mohammadzadeh received the B.Sc. degree in computer science from the Shahid Bahonar University of Kerman, Kerman, Iran, in 2004, and the M.Sc. degree in Computer Science and the Ph.D. degree in Bioinformatics from the University of Tehran, Tehran, Iran, in 2007 and 2014, respectively. He is currently an Associate Professor with the Software Engineering Department, Islamic Azad University, Karaj Branch, Iran. His current research interests include swarm intelligence algorithms, bioinformatics algorithms, complex dynamical networks, and parallel computing.

- Email: javad.mohammadzadeh@iau.ac.ir
- ORCID: 0000-0003-1889-0294
- Web of Science Researcher ID: NA
- Scopus Author ID: 24724981800
- Homepage: https://civilica.com/p/558797/

Faraneh Zarafshan is an Assistant Professor in the Department of Information Technology Engineering at Islamic Azad University. She serves as the Secretary of the Knowledge-Based Activities Council in Markazi Province and is a member of the strategic planning team for the university's 1404 Vision Plan. Additionally, she is the Chairwoman of the Knowledge-Based Company Tosee Pardazeh Argham Houshmand, specializing in intelligent systems and cancer diagnosis. Dr. Zarafshan is also a board member of the IT Organization of Arak Municipality, actively contributing to technological and knowledge-driven advancements in the region.

- Email: Fa.zarafshan@iau.ac.ir
- ORCID: 0000-0003-0327-5176
- Web of Science Researcher ID: F-9097-2010
- Scopus Author ID: 36454394900
- Homepage: https://civilica.com/p/95807/