Hindawi
Journal of Electrical and Computer Engineering
Volume 2023, Article ID 6327247, 15 pages
https://doi.org/10.1155/2023/6327247

Research Article

EtHgSC: Eigen Trick-Based Hypergraph Stable Clustering Algorithm in VANET

Mays Kareem Jabbar [1,2] **and Hafedh Trabelsi** [1]

[1] *CES_Lab, Ecole National d'Ingénieurs de Sfax (ENIS), Sfax University, Sfax 3029, Tunisia*
[2] *Faculty of Engineering, University of Misan, Al Amarah City, Misan Province 62001, Iraq*

Correspondence should be addressed to Mays Kareem Jabbar; m_mays85@uomisan.edu.iq

Received 15 October 2022; Revised 4 March 2023; Accepted 6 March 2023; Published 15 March 2023

Academic Editor: Bhargav Appasani

A smart city's vehicular communication strategy is important. A significant problem with vehicular communication is scalability. Clustering can help with vehicular ad hoc network (VANET) problems; however, clustering in VANET faces stability problems because of the rapid mobility of the vehicles. To achieve high stability for the VANET, this paper presents a new efficient Eigen-trick-based hypergraph stable clustering algorithm (EtHgSC). This algorithm has a twofold scheme for stable CH selection. In the first part of the proposed scheme, the cluster generation is handled using an improved hypergraph-based spectral clustering algorithm using the Eigen-trick method. The "Eigen-trick" method is used to partition both vertices and hyperedges, which provides an approach for reducing the computational complexity of the clustering. The cluster head (CH) is chosen in the second part, taking into account the requirements for keeping a stable connection with most neighbors. In addition to relative speed, neighboring degree, and eccentricity that are used to select the CH, the vehicle time to leave metric is introduced to increase the CH stability. The grey relational analysis model is used to find each vehicle's score, and the CH is selected based on the maximum vehicle's score. The results show the supremacy of our proposed scheme in terms of CH lifetime, cluster member (CM) lifetime, and the change rate of CH. Also, the proposed scheme achieves a considerable reduction in terms of packet delay.

1. Introduction

An intelligent transportation system (ITS) has emerged as a popular research area in recent years because of the ongoing development of wireless communication technologies and embedded systems. An essential component of the ITS is the vehicular ad hoc network (VANET). VANET is a subset of mobile ad hoc networks (MANET) [1]. Additionally, the majority of the VANET clustering algorithms were evolved from the earlier MANET clustering algorithms, such as the mobility-based metric for clustering (MOBIC) in [2] and the weighted clustering algorithm (WCA) in [3]. In a VANET, vehicles communicate with each other and fixed infrastructure via vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I), respectively. Clustering the vehicles can address problems with scalability, broadcast storms, hidden terminal problems, and quality of service that VANETs experience because of the high mobility of the vehicles [4]. Vehicles group together to form clusters according to similar factors, including the vehicles' speed, direction, number of cars within the transmission range, and position. Each cluster has a cluster head (CH) that is responsible for the cluster and the cluster members (CMs) [5]. Many clustering algorithms have been introduced, but their effectiveness declines as the number of vehicles in a city environment increases. Because of the high mobility of vehicles, cluster stability is reduced, so VANET necessitates a special clustering technique that considers mobility-related characteristics to prevent frequent cluster breaking. The communication link weakness is also one of the major issues in the urban environment.

The method used for cluster formation and CH selection in any clustering algorithm is very important to achieve stability. In a graph representation of a network of vehicles,

a vehicle node is connected to two other vehicle nodes [6–8]. This graphical representation might work well in areas with low population density, such as highways or sparsely populated cities. In contrast, graph theory does not apply in dense urban environments where a vehicle is always connected to more than two other vehicles. A good way to represent dense vehicle networks is with hypergraphs.

A hypergraph is frequently used to describe multiple interactions between items in a cohesive way, and spectral clustering is one of the best methods for dividing those objects (vertices) into various communities [9]. Hypergraph spectral clustering has been used to cluster the vehicles in the cluster formation phase. The authors in [10] proposed hypergraph partitioning using the tensor trace maximization (TTM) method.

This paper presents the Eigen-trick-based hypergraph stable clustering algorithm (EtHgSC), which has a twofold scheme for stable clustering. In the first part of the proposed scheme, the cluster generation is handled using an improved hypergraph-based spectral clustering algorithm using the Eigen-trick method. "Eigen-trick" method is used to partition both vertices and hyperedges, which provides an approach for reducing the computational complexity of the clustering. To improve clustering, the Eigen trick is used to calculate the modified Laplacian value in the TTM [10]. The CH is chosen in the second part, considering the requirements for keeping a stable connection with most neighbors. In addition to relative speed, neighboring degree, and eccentricity that are used to select the CH, the vehicle time to leave metric is introduced to increase the clustering stability. The grey relational analysis model is used to find each vehicle's score. The vehicle with a high score is selected as a CH for each cluster at each instant. The contributions of this paper are listed as follows:

(i) The hypergraph-based spectral clustering algorithm is introduced for cluster formation, and the Laplacian value of the TTM is modified using the Eigen-trick method. It considers the transformable connection between the vertex Laplacian and the hyperedge Laplacian, which can speed up the solution of eigenproblems without losing information.

(ii) The fast-moving vehicle may leave the cluster at any time and go out of the CH's transmission range. In that case, the packet drop would be higher. It is better to know the leaving time of the vehicle in the cluster. Time to leave (T_{leave}) is an important parameter to consider when selecting a stable CH.

(iii) A vehicle may change its direction, so estimating the next position of the vehicle beforehand is necessary to know. Thus, the predicted vehicle direction at the next instant is calculated with the help of the predicted vehicle position by the predictive directional greedy routing protocol (PGRP). It helps to select the lane and its length to get the necessary estimated time to leave T_{leave}'. This parameter reduces the frequent cluster breakage at junctions.

(iv) A relational analysis is developed using grey relational analysis (GRA) between four CH selection parameters instead of knowledge-based weightage. GRA is a decision-making model that uses for a few data (when we don't have too much data). The scheme for selecting a CH is used by the GRA to find each vehicle's score for selecting the CH. By using this scheme, strong connectivity and a stable link lifetime are obtained.

The rest of this article is organized as follows. The related works are introduced in Section 2; Section 3 presents our proposed EtHgSC model with its parts (cluster generation and CH selection phases). Simulation results and discussion are presented in Section 4. Finally, the conclusion and future work are introduced in Section 5.

2. Related Works

Different clustering algorithms have been introduced for VANET, which their primary goal is to increase cluster stability; we present some of them in this section.

Affinity propagation algorithm (APROVE) was presented in [11]. APROVE algorithm was proposed using the affinity propagation algorithm in a distributed manner. The mobility metrics are used in CH selection. The authors asserted the existence of clusters with excellent stability. With the same concept of mobility metric, another algorithm was proposed. A mobility-based scheme for dynamic clustering in VANETs (MoDyC) was presented in [12] to improve clustering stability.

Vehicular multihop algorithm for stable clustering (VMaSC) was proposed in [13], and it is used in VANET to build stable multihop clusters with a small number of CHs. The CH is selected based on the node with the least mobility via multihops, which is determined as a function of the speed differential between neighboring nodes. Also, the VMaSC improved in [14] by integrating IEEE 802.11p-based multihop clustering and the long-term evolution (LTE). This algorithm has high citations; it was used as a reference in many articles.

Based on the concept of vehicular mobility in a highway environment, Arkian et al. [15] presented a new clustering scheme based on multi-hop.

Some algorithms used other metrics for the selection of the CH. Passive multihop clustering (PMC) was presented in [16], and it uses the priority neighbor following strategy for selecting the CH. This strategy improves clustering stability and reduces the cost of clustering effectively.

The authors in [17, 18] used the fuzzy logic scheme to improve the network's reliability and stability.

Other algorithms are considered a VANET as a graph. Clustering techniques in graph theory are divided into two categories: hierarchical clustering and partitioned clustering. On the basis of the existing cluster structure, hierarchical algorithms construct clusters. These algorithms build partitions sequentially, whereas partitioning methods only use one partition to separate nodes into clusters.

Some methods employed a K-mean approach and combined it with other algorithms to build stable clusters, such as K-mean and Floyd-Warshall algorithms (KMFW) in [19], the K-means clustering method, a new routing protocol (KMRP) in [20], and a betweenness centrality based clustering (BCBC) in [21]. The K-mean method is one of the simplest partitioned clustering methods. The K-mean method is not capable of disconnecting the nonlinearly distinguishable clusters. To resolve this, spectral clustering is used based on eigenvectors of the matrix [10].

The authors in [22] introduced position-based prioritized clustering (PPC), which generates clusters based on geographic location information and vehicle priorities. The CH election process is similar to the computation of minimum dominating sets used in graph theory. Other algorithms used heuristic algorithms for cluster formation [23]. Hybrid clustering algorithm based on roadside (HCAR) was presented, and this algorithm is a centralized approach with road side units (RSUs) distribution.

However, a cluster-based VANET-oriented evolving graph (CVoEG) [6] is an evolving graph that uses spectral clustering approach for cluster formation. Another algorithm proposed by Khan et al. [7] is also based on graph theory and uses a heuristic algorithm for cluster formation. The authors selected CHs based on connectivity and determined its eccentricity. The CH was chosen as the highest eccentric vehicle in a cluster. This algorithm forms stable clusters and increases the probability of connectivity of the elected route. Table 1 presents clustering algorithms in the literature in terms of the CH selection metrics, and simulator tools used.

As we said, the graphical depiction might work well in sparsely populated areas like highways or small cities, but not in crowded urban scenarios. Graph theory, on the other hand, does not work in urban dense scenarios where a vehicle is always connected to more than two other vehicles. A good representation of dense vehicle networks is a hypergraph.

Also, we see that many clustering algorithms are based on mobility and speed measures to select the CH, but these measures are lost in the urban scenario, especially when the vehicle density is high because the vehicle speed will be low. So, it is better to search for new metrics for selecting the CH.

So, our primary goal is to propose a new clustering algorithm that aims to increase the clusters' stability. Our proposed model has two main parts: the cluster formation is presented using a hypergraph spectral clustering algorithm. The hypergraph spectral clustering algorithm is used with a modified Laplacian value in the TTM by using the Eigen-trick method to improve the clustering. The Eigen-trick method improves clustering efficiency by using higher-order information in eigenvalues. It considers the transformable connection between the vertex Laplacian and the hyperedge Laplacian, which can speed up the solution of eigenproblems without losing information. The second part is the CH selection phase; in addition to relative speed, neighbor degree, and eccentricity, the vehicle's time to leave parameter is used in the selection of the CH to increase the CH stability. The vehicle's direction is unpredictable at the road junction, which reduces the CH's stability [24]. So, our proposed scheme has solved the problem of CH stability at the junction using the estimated next time to leave parameter. Strong connection and robust link lifetime are ensured by selecting a suitable CH using the GRA model.

3. The Proposed EtHgSC Model

A new clustering algorithm for a VANET structure in an urban scenario is presented in this article. The proposed approach (EtHgSC) includes two main parts: the cluster generation part, which uses the improved hypergraph spectral clustering method using the Eigen-trick method for generating the clusters. The second part is the CH selection. CH stability is governed by using four factors: inclusive of relative speed, neighboring degree, eccentricity, and the vehicle's time to leave with the estimated next vehicle's position. These two parts are explained in detail in this section. The flow diagram of our proposed model is shown in Figure 1.

3.1. Clusters Generation of EtHgSC. In this proposed model, a multilane road structure is taken into account in an urban scenario. The real map incorporates the total number of cars N together with their speeds and positions. The transmission range Tr_v is the same for all on-board unit (OBU) equipment. The RSUs communicate with the same communication module, with the transmission range $\mathrm{Tr}_{\mathrm{RSU}}$. The OBU contains both IEEE 802.11p radio technology and a global positioning system (GPS) unit. Every car in the network functions as a node (V) acting as a source, destination, or router.

In the process of spectral clustering, information loss is evident [25]. To prevent it, the TTM clustering has been proposed for spectral clustering of vehicles. It is called TTM because the hypergraph partition is equivalent to tensor decomposition into a lower rank. Despite the fact that the TTM was designed to further partition the hypergraph into lower nondecomposable graphs, the information loss can be further minimized with the help of the Eigen trick in TTM hypergraph partitioning. In this section, a weighted hypergraph TTM is proposed for partitioning, with its improvement using the Eigen-trick method for the large-scale VANET for clustering. Figure 2 presents an example of our proposed model.

3.1.1. Cluster Formation as a Hypergraph. In spectral clustering, the most common criterion method is the radio cut method. For a set of clusters $V_1, V_2, \ldots, V_k$ satisfying $V_1 \cap \ldots \cap V_k = \varnothing$ and $V_1 \cup \ldots \cup V_k = V$. And as we mentioned before, L is a spectral clustering major tool.

In weighted hypergraph $\mathscr{H} = (V, \mathscr{E}, W)$, V is a vertices, $V \in \{v_1, v_2, v_3, \ldots\ldots, v_n\}$ is the vehicles (nodes), $\mathscr{E}$ is the edges $\mathscr{E} \in \{e_1, e_2, e_3, \ldots\ldots, e_m\}$, it is the links among vehicles (c_{ij}), it depends on the distance between vehicles (d), and W is the weight of the edges, $W \longrightarrow [0, 1]$.

Table 1: State-of-the-art comparison.

Reference	Algorithm	CH selection metric	Simulator tools
[11]	APROVE	Distance and speed	NS2
[12]	MoDyC	Relative position, moving direction, and link lifetime	NS2, SUMO
[19]	KMFW	Average distance	NS2
[13]	VMaSC	Average speed	NS3, SUMO
[14]	VMaSC-LTE	Average speed	NS3, SUMO
[15]	Arkian et al. [15]	Speed, neighbor	OMNET++, SUMO
[16]	PMC	Speed, neighbors, link lifetime, and position	NS2, vanetMobiSim
[22]	PPC	Travel time, ID, and relative velocity	NS2
[23]	HCAR	Lowest ID	NS2, VANET MobiSim
[6]	CVoEG	Link lifetime, position, and relative speed	MATLAB, SUMO, MOVE
[18]	FCMS	Relative speed, degree, security, and trustworthiness	—
[24]	JCV	Movement at the junction, relative position, time, and degree of a node	SUMO, CVANETSIM, JAVA
[7]	Khan et al. [7]	Link connectivity	MATLAB, SUMO, MOVE
[20]	KMRP	Velocity, free buffer size, and node degree	NS2
[10]	HGCM	Relative speed, neighboring, trust score	MATLAB, SUMO
[21]	BCBC	Betweenness centrality method	MATLAB, SUMO

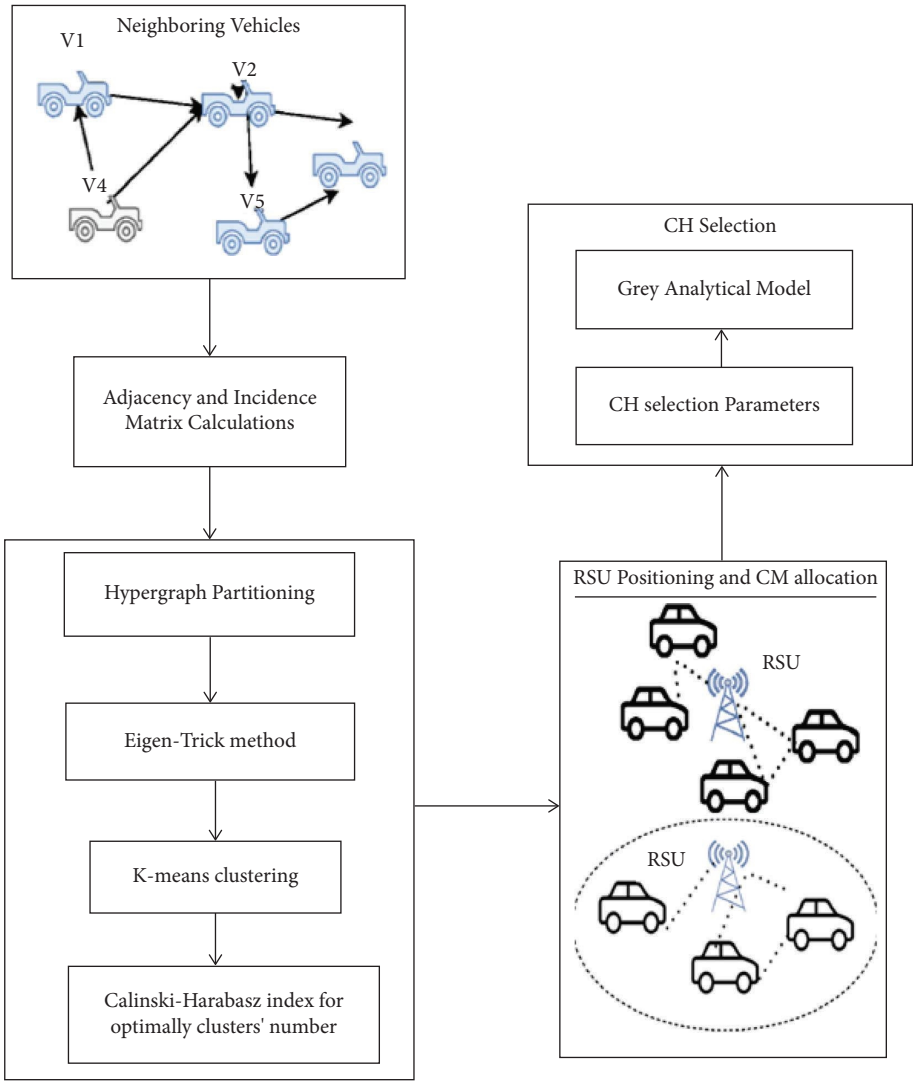

Figure 1: The EtHgSC flow diagram.

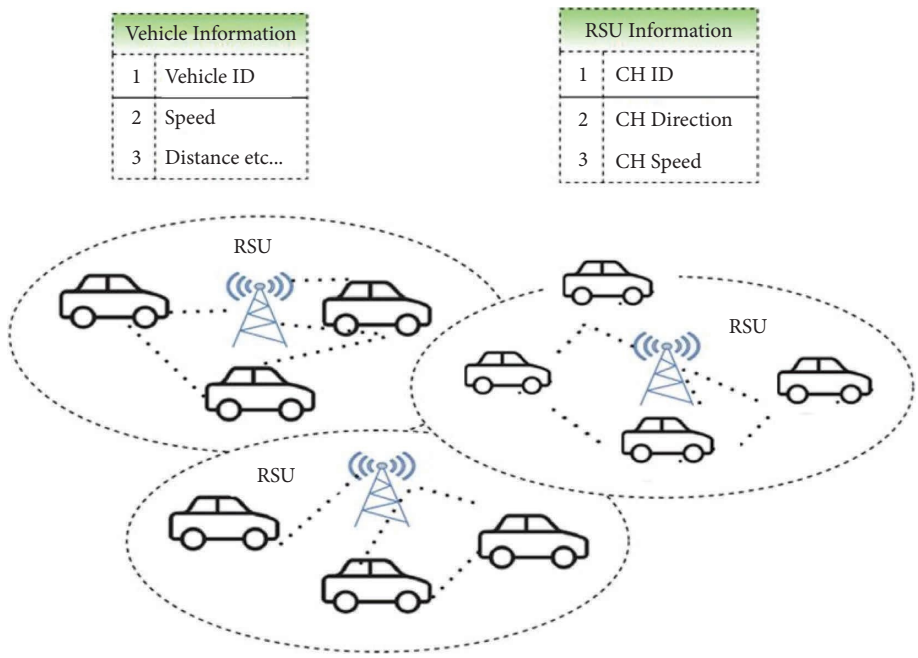

Figure 2: Proposed model.

$$c_{ij} = \begin{cases} 1, & d_{ij} \le Tr_v, \\ 0, & d_{ij} > Tr_v. \end{cases} \tag{1}$$

The problem is partitioning the weighted hypergraph V into k disjoint sets, $V_1, \ldots\ldots V_k$. The followings are the steps for partitioning the hypergraph with the modified TTM using the Eigen-trick method:

(1) The degree of any node is the number of vehicular nodes connected to other nodes $v \in \mathscr{V}$, deg $(v) = \sum_{e \in E} w_e$. The vertex diagonal matrix (degree matrix) is $\mathrm{Dig}_v \in \mathbb{R}^{N \times N}$. The other important term associated is a volume, which is coined as $\mathrm{vol}(\mathscr{V}_1) = \sum_{v \epsilon \mathscr{V}_1} \deg(v)$, which is the number of nodes incident on node $\mathscr{V}_1$ such that $\mathscr{V}_1 \subseteq \mathscr{V}$. The associativity of $\mathscr{V}_1$ is defined as the sum of incident edges' weights as $\mathrm{assoc}(\mathscr{V}_1) = \sum W_e$. The normalized associativity of these separate partitions is given as follows:

$$N - \mathrm{assoc}(\mathscr{V}_1, \ldots\ldots \mathscr{V}_k) = \sum_{i=1}^{k} \frac{\mathrm{assoc}(\mathscr{V}_i)}{\mathrm{vol}(\mathscr{V}_i)}. \tag{2}$$

This problem can be presented as the tensor problem as discussed in [25], and this is the reason it is termed TTM. The clustering of the tensors is formulated as the problem of maximizing the diagonal terms, and edge weights are the diagonal elements in the VANET hypergraph. So, the problem is coined the tensor maximization problem. The solution to this problem is searched by spectral relaxation by computing the k eigenvectors of the normalized adjacency matrix [25].

(2) The defined adjacency matrix is used for spectral clustering. It is computed using the following formula, which is specified here for the tensor (order m):

$$A_{i_1, i_2, \ldots\ldots i_m} = \begin{cases} w_{\{i_1, i_2 \ldots\ldots i_m\}}, & \text{if } i_1, i_2, \ldots\ldots\ldots, i_m \text{ are distinct}, \\ 0, & \text{otherwise}. \end{cases} \tag{3}$$

Also, the incidence matrix (I) is defined, $I \in \mathbb{R}^{N \times m}$. It represents the connection between vertices and edges. $i(v, e) = 1$, if there is a connection between a vertex v and an edge e, and otherwise, $i(v, e) = 0$. The edges' diagonal matrix $\mathrm{Dig}_e \in \mathbb{R}^{m \times m}$ is obtained as follows:

$$\deg(e) = \sum_{v \in V} i(v, e). \tag{4}$$

In our work, the weighted incidence matrix is used, and it is calculated as follows:

$$i(v, e) = \begin{cases} w_e, & \text{if connected}, \\ 0, & \text{otherwise}. \end{cases} \tag{5}$$

(3) The Fiedler vector-based unnormalized Laplacian matrix was used in the TTM approach in [25]:

$$L_v = \mathrm{Dig}_v^{-1/2} A \mathrm{Dig}_v^{-1/2}. \tag{6}$$

In our work, by utilizing higher-order information in eigenvalues, the Eigen trick improves clustering efficiency. It considers the transformable connection between the vertex Laplacian L_v and the hyperedge Laplacian L_e, which can speed up the solution of

eigenproblems without losing information. During this time, the partitions for vertices and hyperedges are both obtained simultaneously [26].

The hyperedge Laplacian L_e, $L_e \in \mathbb{R}^{m\times m}$, is calculated as follows [26]:

$$L_e = \mathrm{Dig}_e^{-1/2} W^{(1/2)} I^T \mathrm{Dig}_v^{-1} I W^{(1/2)} Dig_e^{-1/2}. \quad (7)$$

In actuality, there is a link between the symmetry of L_v and L_e. In "Eigen trick," an interesting attribute of two Laplacians is that the k trailing eigenvectors of the vertex Laplacian $U_v \in \mathbb{R}^{N\times k}$ can be computed from the corresponding eigenvectors of the hyperedge Laplacian $U_e \in \mathbb{R}^{m\times k}$ and vice versa. $U_e = \mathrm{eig}(L_e)$.

(4) Computed the $U_v \in \mathbb{R}^{N\times k}$ from the hyperedge eigenvectors U_e.

$$U_v = \mathrm{Dig}_v^{-1/2} I \, \mathrm{Dig}_e^{-1/2} W^{(1/2)} U_e. \quad (8)$$

(5) View each row of V as a vector for a node, and cluster the N nodes into k clusters $(V_1, V_2, \ldots, V_k)$ through k-means clustering.

(6) The optimal set of clusters and the clustering efficiency are evaluated using the Calinski–Harabasz index (s). Using this index, we can see how closely vehicles are clustered together in a cluster, as well as how far apart all clusters are. The s index is calculated as follows [27]:

$$s = \frac{\mathrm{tr}(B_k)}{\mathrm{tr}(Z_k)} \times \frac{\mathrm{Vehi}_{\mathrm{num}} - k}{k-1}. \quad (9)$$

This index evaluates how closely each cluster's vehicles are spaced out. In this case, k stands for the number of clusters, while $\mathrm{Vehi}_{\mathrm{num}}$ is the size of each cluster. The dispersion between clusters is denoted by $\mathrm{tr}(B_k)$, and the dispersion among cars inside a cluster is denoted by $\mathrm{tr}(z_k)$. Equations (10) and (11) are used to calculate these two factors.

$$Z_k = \sum_{q=1}^{k} \sum_{x\in C_q} (x - c_q)(x - c_q)^T, \quad (10)$$

$$B_k = \sum_{q=1}^{k} n_q (c_q - c_E)(c_q - c_E)^T. \quad (11)$$

A set of points in cluster q are shown here as c_q, and the cluster center is x. The clusters with n_q points in them have a center called c_E.

The maximum value of s is used to select an ideal set of clusters from the pool of generated clusters using this index. $C_{\mathrm{optimal}} = [C_{\mathrm{num}} : \forall \max(s)]$.

The complete cluster formation algorithm is as in Algorithm 1.

3.1.2. RSUs Deployment. VANET's weaknesses as well as network performance are enhanced by RSUs. The RSUs can successfully meet increased reliability, vehicular density, and decreased overhead delay, particularly in urban environments. When connecting to one or more vehicle clusters, the RSU functions as a gateway inside a router. It can also connect with RSUs using wireless or optical networks through an inclusive deployment [28]. Few studies in the literature addressed RSU deployment, but the RSU is a component of VANET and is critical, as we have developed in our work. The actual VANET network needs RSU communication too [29].

So, after Algorithm 1 has produced the ideal set of clusters, the fetching location for RSU starts. Using betweenness centrality, a traffic graph's dynamic structure is created. A network is considered as a graph $G = (V, \mathscr{E})$, where V vehicles' connections are connected by a certain set of edges $\mathscr{E}$. A graph's centrality matrix serves as a measure of its compactness [30]. The graph's most frequently visited vertex is determined by centrality. For a vehicle v, it can be calculated as follows:

$$C_B(v) = \sum_{s\neq v\neq u\in V} \frac{\sigma_{st}(v)}{\sigma_{st}}. \quad (12)$$

Here, the number of shortest paths from node s to node u is shown here as σ_{st}, and the number of pathways that pass through v is $\sigma_{st}(v)$. The center of the cluster is determined by the vehicle with the highest centrality value $\mathrm{Vehi} = \max(C_B)$. This is where the RSU will be installed.

Now, the cluster generation phase for our scheme is complete.

3.2. CH Selection Measures. The next step is the selection of the CH, which is the vehicular node in a cluster that coordinates or heads the cluster. Continuous communication between the cluster and RSU, security, and upkeep of the routing path all fall under the purview of a CH. The stability of the CH in VANET is the main motive for designing any clustering algorithm. The proposed methodology for the CH selection is based on four parameters; relative speed, neighboring degree, eccentricity, and the vehicle's time to leave with the estimated next vehicle's position. These parameters are calculated for each vehicle at each instant t. The four parameters are discussed in the following.

3.2.1. Relative Speed Score (ψvehi). It is characterized as a shift in a node's relative average vehicle speed with respect to its neighbors' speeds. The long-term velocity of vehicles is taken into consideration by a reward function that is developed. The speed of each vehicle (V_{vehi}) is calculated. As a result, their speed receives an absolute value (δ) that can be used to reward or penalize them. As a result, the relative average speed is increased or decreased, and ψvehi is calculated as follows [15]:

$$\begin{aligned} \psi_{\mathrm{vehi}}(t+1) &= \psi_{\mathrm{vehi}}(t) + \delta; \left|V_{\mathrm{vehi}} - V_{\mathrm{avg}}\right| \le S_{\mathrm{thr}}, \\ \psi_{\mathrm{vehi}}(t+1) &= \psi_{\mathrm{vehi}}(t) - \delta; \left|V_{\mathrm{vehi}} - V_{\mathrm{avg}}\right| > S_{\mathrm{thr}}, \end{aligned} \quad (13)$$

Input: maximum vehicles: $\text{Vehi}_{\max}$, vehicle's location: Vehi_{Loc}
(1) Choose the time period t during which there is the largest number of vehicles $\text{Vehi}_{\max}$
(2) A hypergraph formation $\mathbb{H} = (V, \mathscr{E}, W)$
(3) The similarity matrix A is calculated based on the distance proximity between the vehicles $A \in \mathbb{R}^{N\times N}$, and also, an incidence matrix I is generated, its size of $I \in \mathbb{R}^{N\times m}$
(4) A diagonal matrix for vehicles $\text{Dig}_v \in \mathbb{R}^{N\times N}$, and for edges $\text{Dig}_e \in \mathbb{R}^{m\times m}$
(5) Find the hyperedge Laplacian $L_e = \text{Dig}_e^{-1/2} W^{(1/2)} I^T \text{Dig}_v^{-1} I W^{(1/2)} \text{Dig}_e^{-1/2}$
(6) The k dominant eigenvector of L_e is calculated $U_e \in \mathbb{R}^{m\times k}$
(7) Computed the $U_v \in \mathbb{R}^{N\times k}$ from the hyperedge eigenvectors U_e
(8) Normalize each row of $\overline{U_v} = U_v$
(9) Run k-means on the rows of $\overline{U_v}$
(10) C_{num} is obtained through k-means partition $Par = \{V_1, \ldots\ldots V_k\}$
(11) Compute the Calinski–Harabasz (s) index for each cluster C_{num}
(12) $C_{\text{optimal}} = [C_{\text{num}}: \forall \max(s)]$ is the optimal number of clusters
Output: C_{optimal}, partitioning vehicles in clusters

Algorithm 1: Cluster formation using Eigen-trick method.

where S_{thr} is the variable that makes sure the vehicle going with velocity is nearly moving at the same speed as that of the neighbors. Using the traffic control interface (TraCI) parameters, the initial value of ψ_{vehi} is calculated, and δ is set to 0.01.

3.2.2. Neighboring Degree (n_d). The two vehicular nodes vi and vj's connection state at time t in the cluster created C_{optimal} with vehicle density Vehi_{num} are defined as follows [31]:

$$n_d = \sum_{j=1}^{\text{Vehi}_{\text{num}}} c_{ij}; \forall 1. \tag{14}$$

Since n_d is high, the CH won't be dynamic for a very long period. The total number of nearby vehicles is the neighborhood degree. A vehicle's neighbors are those that are within its transmission range. If the distance between two cars at time t is less than or equal to Tr_v, then c_{ij} is 1.

3.2.3. Eccentricity (Ecc). Due to the vehicles' high speed in real time, communication links break more frequently. An evolving cluster model is necessary for maintaining the link. Reclustering will typically be unavoidable after the CH resigns or no longer meets the criteria to remain a CH. Eccentricity is a concept that is introduced to ensure stability. Here, spectral clustering is used to create a dynamic graph-based model.

It is intended for a vehicular graph topology to be hypergraph $\mathscr{H} = (V, \mathscr{E}, W)$, and the concept of eccentricity (Ecc) is introduced. Ecc is the mean/average eigenscore (λ_i) of each vehicle in a cluster; it is calculated as follows [6]:

$$\text{Ecc} = \frac{1}{|\text{Vehi}_{\text{num}}|} \sum_{\lambda_i \epsilon \text{Vehi}_{\text{num}}} \lambda_i. \tag{15}$$

3.2.4. Vehicle's Time to Leave (T_{leave}). It is the amount of time needed for a vehicle to reach the last section of the lane. This parameter guarantees to choose a CH with enough time remaining to finish the lane that causes it to head for a longer period. It is determined using the length of the lane L, the distance covered by a vehicle on the road segment D, and the vehicle speed at that time v:

$$T_{\text{leave}} = \frac{L - D}{v}. \tag{16}$$

The possibility of the vehicle changing direction changes the lane length. Figure 3 can give a picture of the problem. T_{leave} fails to estimate the road junctions. The unpredictable change in the direction of the vehicle at the junction can abruptly change the prediction. The lane lengths may differ, and so, the vehicle's estimated next position must be known ahead of time. It helps to select the lane and its length to get the necessary T_{leave}.

3.2.5. Estimated Next Position. To predict the next vehicle position at time $t+1$, firstly the new vehicle coordinates should be calculated from the current coordinates (X_2, Y_2), distance $d = v_2 t_2$, and vehicle current direction θ_2, and it is calculated as follows [32]:

$$\begin{cases} X_2' = X_2 + v_2 t_2 \cos\theta_2, \\ Y_2' = Y_2 + v_2 t_2 \sin\theta_2. \end{cases} \tag{17}$$

Then, a vehicle new direction using the predicted position (X_2', Y_2') is calculated by using

$$\theta_2' = \tan^{-1} \frac{(Y_2 - Y_2')}{(X_2 - X_2')}. \tag{18}$$

From the vehicle's new direction θ_2', the next lane length L and D are known now, and then, we can calculate the T_{leave}' with the predicted next vehicle's position using equation (16).

After calculating the four measures for each vehicle at each instant of time, the GRA model is used to find the vehicles' score. The GRA is explained in the next subsection.

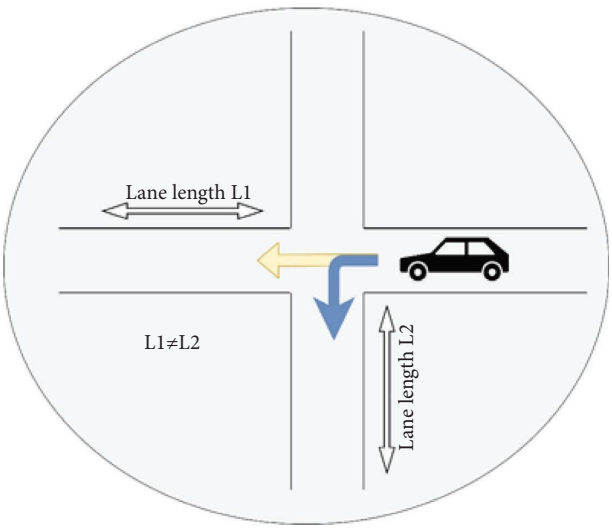

Figure 3: Problem in T_{leave} without the next position estimation.

3.3. Grey Relational Analysis Model. GRA is a component of grey system theory, which is useful for resolving problems involving complex interrelationships between multiple factors and variables. The basic step of GRA is to convert the performance of each option into a comparability sequence, which is denoted by the letters X_{ij}. The term for this step is "grey relational generating" [33]. A reference sequence is defined in accordance with these sequences. Following that, the grey relational coefficient between the reference sequence and all comparability sequences is determined $\gamma(X_{0j}, X_{ij})$. The grey relational grade between each comparability sequence and the reference sequence X_0 is then determined using these grey relational coefficients. If a comparability sequence translated from an alternative has the highest grey relational grade between the reference sequence and itself, that alternative will be the best choice. The grade $g(v)$ is calculated as follows [34]:

$$g(v) = \sum_{j=1}^{n} w_j \gamma\left(X_{0j}, X_{ij}\right), \text{for } i = 1, 2, \ldots\ldots, \text{Vehi}_{\text{num}}, \tag{19}$$

where $n = 4$ (the number of CH selection parameters in our work), i is a number of alternatives. $\gamma(X_{0j}, X_{ij})$ is the grey relational coefficient, how near X_{ij} is to X_{0j} is determined using it. w_j is the weight of the four parameters, where $\sum_{j=1}^{n} w_j = 1$.

The grade for each vehicle in a cluster is calculated using the GRA. The vehicle with the maximum grade is selected as a CH for each cluster at each instant of time t.

The pseudocode for the CH selection phase is introduced in Algorithm 2.

3.4. Time Complexity of the EtHgSC Scheme. The total time complexity of the EtHgSC scheme is expressed as follows:

$$O_{\text{TOT}} = O_{\text{CF}} + O_{\text{RSU}} + O_{\text{CH}}, \tag{20}$$

where O_{CF} stands for the time complexity of cluster formations, O_{RSU} for RSU deployment, and O_{CH} for CH selection. The formation of a hypergraph, the application of Laplacian, and the Eigen-trick technique to solve eigenproblems, followed by K-means to get the ideal set of clusters, are the key steps in the cluster formation process.

In hypergraph, the quotient of similarity is taken between each of the vehicles, and this is calculated by constructing a d nearest neighbour graph with complexity $O(N^2)d$. By using the Eigen-trick method, the hyperedge Laplacian L_e is processed instead of vertex Laplacian L_v, and the time complexity for L_e is $O(\text{NNZ}(I^2)/N)$.

"Eigen trick" provides us a way to calculate U_v with lower time and space complexity. The time complexity for solving the eigenproblem of the vertex is $O(N^3)$, while the corresponding cost to hyperedge is $O(m^3)$. Instead of solving the eigenproblem of L_v directly, the Eigen-trick method is proposed to solve it using the eigenproblem of L_e. It will reduce the computational complexity from $O(N^3)$, to $O(m^3)$ significantly $(m < N)$. Another benefit is that the conversion between U_e and U_v is accurate without any information loss [26]. The last is the K-means complexity which is dependent on the $O_{C_{\text{optimal}}} = O(\tau \text{NC}_{\text{num}})$. So, O_{CF} will be

$$O_{\text{CF}} = O\left(N^2\right)d + O\left(\frac{\text{NNZ}\left(I^2\right)}{N}\right) + O\left(m^3\right) + O\left(\tau \text{NC}_{\text{num}}\right). \tag{21}$$

The complexity is reduced by removing the terms of less computational power

$$O_{\text{CF}} = O\left(N^2\right)d + O\left(m^3\right). \tag{22}$$

Due to the use of a graph in the RSU deployment, the computational complexity is as follows:

$$O_{\text{RSU}} = O\left(N^2\right)d. \tag{23}$$

The CH selection parameters are relative speed score (ψ_{vehi}), neighbourhood degree (η), eccentricity $(\mathfrak{E})$, and time to leave estimation (T'_{leave}).

$$O_{\text{CH}} = O_{\psi_{\text{vehi}}} + O_{\eta} + O_{\mathfrak{E}} + O_{T'_{\text{leave}}}. \tag{24}$$

The time complexity for ψ_{vehi}, η, and $\mathfrak{E}$ is calculated as follows:

$$\begin{aligned} O_{\psi_{\text{vehi}}} &= O(N), \\ O_{\eta} &= O\left(\log N^2\right). \end{aligned} \tag{25}$$

The relative speed is calculated using a straightforward threshold function based on the vehicle's speed. The neighborhood is the next, and it is determined by the c_{ij} affinity matrix for neighboring vehicles.

The affinity matrix and eigenvalue decomposition of spectral clustering methods are used to determine the eccentricity:

Input: number of clusters, number of vehicles in each cluster at each instant of time, vehicles coordinators (X, Y), current lane length (L), current direction θ
For $t = 1$: instant of time
  For $i = 1$: clusters
    For $j = 1$: vehicles in cluster
      Calculate the CH parameters: n_d, ψ_{vehi}, and Ecc
      Predicted the next vehicle's coordinators (X′, Y′)
      Find the new vehicle's direction from predicted coordinators θ'
      Calculate T'_{leave} using predicted vehicle's position
    End for
    Input the four matrices to GRA model with their weights, where $\sum_{j=1}^{n} w_j = 1$
    Generate a matrix X of $\text{Vehi}_{\text{num}} \times 4$
    Normalize the matrix X
      For i = 1: Vehi_{num}
(a)   For j = 1: n
        Calculate the grey relational coefficient $\gamma(X_{0j}, X_{ij})$
(b)   end for
      end for
    Generate a graph object from grey relational coefficient $\gamma(X_{0j}, X_{ij})$
    Calculate the Vehicle's grade $g(v)$ with the help of $\gamma(X_{0j}, X_{ij})$
      Highest-grade vehicle is selected as a CH CH = max $g(v)$
  End for
End for
Output: CH vehicle

Algorithm 2: The pseudocode for CH selection using the GRA model.

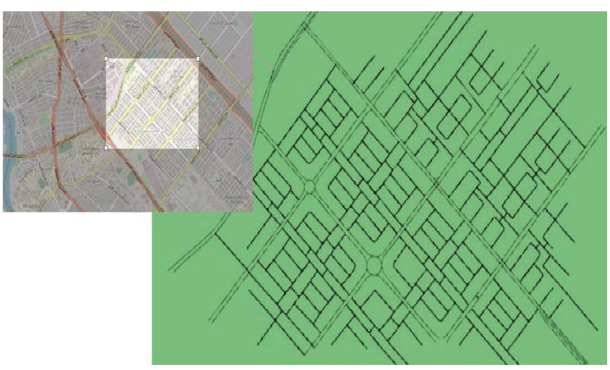

Figure 4: Map of Baghdad in SUMO.

$$O_{\mathfrak{C}} = O\left(N^2\right)d + O\left(N^3\right). \tag{26}$$

The complexity of time to leave is calculated for all the vehicles in the network

$$O_{T'_{\text{leave}}} = \mathrm{O}(N). \tag{27}$$

Thus, the complete time complexity is reduced to moving all the terms with less complexity than cubic and quadratic terms

$$O_{\text{CH}} = O\left(\log\left(N^2\right)\right) + O\left(N^3\right). \tag{28}$$

The overall complexity is primarily dependent on the hypergraph

$$O_{\text{TOT}} = 2O\left(N^2\right)d + O\left(m^3\right) + O\left(\log\left(N^2\right)\right) + O\left(N^3\right). \tag{29}$$

4. Performance Evaluation and Simulation Results

The simulation is done using MATLAB as a network simulator, SUMO as a traffic simulator [35], and with the help of TraCI [36]. The area considered for the study is the real map of Iraq's capital, Baghdad, with latitude = 33.3730°N and longitude = 44.3960°W; it is extracted from OSM, Figure 4. The goal of this study is to evaluate cluster stability at low and high vehicle densities (100 and 1000, respectively), as well as in a dynamic scenario. Also, our proposed model is tested at different vehicle speeds (10, 15, 20, and 25 m/s). Table 2 presents the simulation parameters.

Thirteen clusters are generated using Algorithm 1. Each cluster is depicted in a different color. RSUs serve as a cluster center; they are portrayed in a triangle shape in Figure 5. The discussion of the results is divided into two stages: the effect of the predicted next vehicle's position on the cluster's stability at different vehicle speeds and for the worst-case scenario (1000 vehicles) and a state-of-the-art comparison in terms of stability using different cluster performance parameters.

The following four cluster performance parameters are used to evaluate the performance of our EtHgSC scheme [1]:

(1) CH lifetime: It is the time between a vehicle becoming a CH and subsequently changing to another state. The stability of a cluster depends highly on this metric. If the CH has a long lifetime, the necessity of creating a new cluster will be low.

Table 2: Simulation parameters.

Parameters	Values
Network parameters	IEEE 802.11p
Scenario	Urban
Lanes	458 lanes
Vehicle density	100, 1000
Simulation time (t)	1500 s
T_v	200 m
T_{RSU}	350 m
Maximum speed of vehicle	10–25 m/s

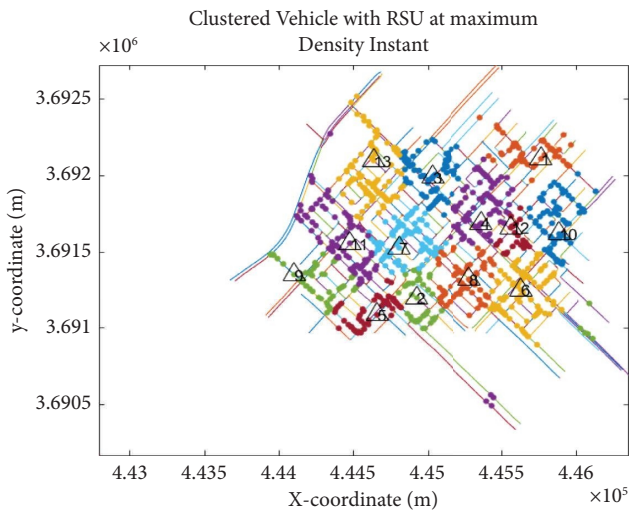

Figure 5: Clusters formation along with RSU deployment with maximum vehicle density.

(2) CM lifetime: It refers to the period beginning with the vehicle's connection to the cluster and becoming one of its members and ending with its departure.

(3) CH change rate: It is the number of state transitions from CH to another state per unit of time. A lower CH change rate is preferable for the stability of VANET clustering.

(4) Clusters number: The number of clusters that form while the network is in use is intended. When there are fewer cluster numbers, the clustering method performs better.

4.1. Effect of Predicted Next Vehicle's Position on the Stability at Different Vehicle Speeds. Figures 6–8 demonstrate the effect of using T'_{leave} with predicted next vehicle's position and without predicted next vehicle's position (based on current position) on the CH lifetime, CM lifetime, and the change rate of CH, respectively. The proposed scheme is tested at different vehicle speeds for the worst-case scenario of 1000 vehicles. In a comparison with our proposed model using T_{leave} without predicated, we see that our proposed model using T'_{leave} with the predicted next position achieves the highest CH lifetime, CM lifetime, and lowest CH change rate, especially at high speed (25 m/s). The T'_{leave} with predicated next vehicle's position increases the CH lifetime by 22% and the CM lifetime by 12% and decreases the CH change rate by 31% at all vehicle speeds in a comparison with our EtHgSC using the current vehicle's position T_{leave}. The predicted next vehicle position increases the cluster stability because it solves the problem of changing the CH at junctions. So, it helps to select the vehicle as a CH, which has a high ability to stay in the cluster as long as possible.

4.2. State-of-the-Art Comparison. Table 3 compares the number of clusters and average CH lifetime for EtHgSC with other techniques from the literature. We compare with the CVoEG [6], Arkian et al. [15], JCV [24], VMaSC [14], and PMC [16]. The VMaSC and PMC are the most-cited clustering algorithms in the literature. This comparison occurs at low traffic density (100 vehicles), and the maximum vehicles speed is set to 25 m/s, as well as T_v is set to 200 m for all algorithms.

The number of clusters that formed over time was another indicator of the algorithm's success. The quality of the created clusters can be assessed using these numbers. Few clusters of limited mobility vehicles are able to connect effectively and maintain stable clustering. On the other hand, increased clustering eventually results in increased overhead and mergers. At low and high traffic levels, the EtHgSC constructs 5 and 13 clusters, respectively.

Khan et al. [6] presented the CVoEG algorithm. They evaluated a graph spectral clustering technique on a network of highways. According to the I-5 highway analysis of the California environment, CVoEG [6] produces 20 clusters with low traffic density along a road length of 12 km. It is anticipated to reach a stability level of 65.5%. Vehicle speed is employed in this study to simulate graph edges. As a result, with low variance, low cluster formation results from the eigenvalues being nearly identical due to the almost identical speeds of the cars.

Our proposed model has raised the concern of the CH stability at the junction, as the vehicle's direction is unpredictable at the road junction. A similar problem is highlighted by Mukhtaruzzaman and Atiquzzaman [24] in their work. In the junction-based clustering in VANET (JCV) [24], the vehicle direction and transmission range are considered for the clustering, and the relative position, direction at the junction, time spent on the road, and node's degree are considered as the CH selection parameters. With 16 clusters constructed, the JCV achieves 76% stability.

In the method proposed by Arkian et al. [15], for a highway length of 3000 m, a large number of dynamic clusters are projected with a low variance of only 90 vehicles using two-lane analysis. This method uses neighborhood analysis, so when there are just 90 vehicles, there must be a lot of clusters to cover all the vehicles in a sparse region. With low traffic flow and a large number of clusters, the CH stability is 58%.

The average CH lifetime of the VMaSC algorithm is 66%, with 17 clusters generated for 100 vehicles. This algorithm considers the relevant mobility metric as the CH selection factor. Because of the rapid movement of the vehicle nodes in VANET, the duration of each CH is extremely short.

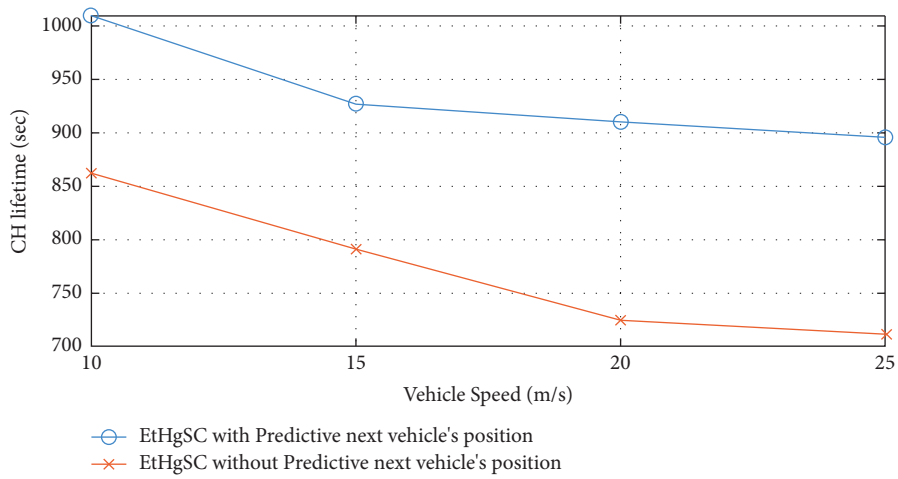

Figure 6: CH lifetimes for EtHgSC with and without predicted next vehicle's position.

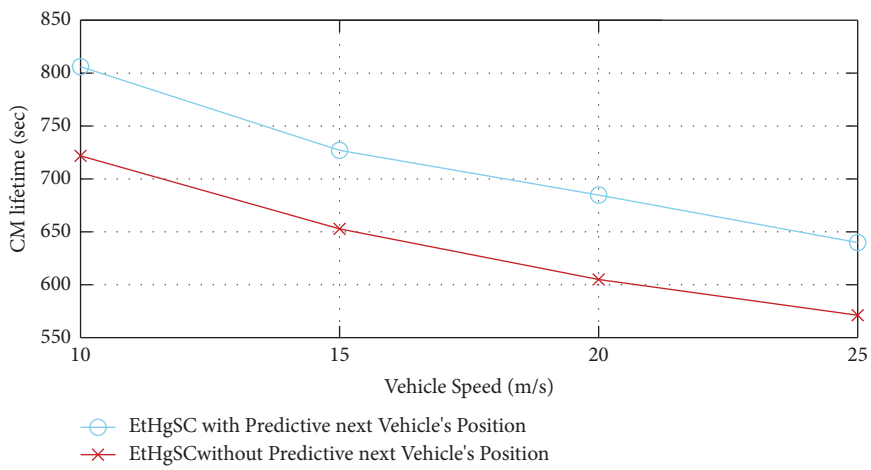

Figure 7: CM lifetimes for EtHgSC with and without predicted next vehicle's position.

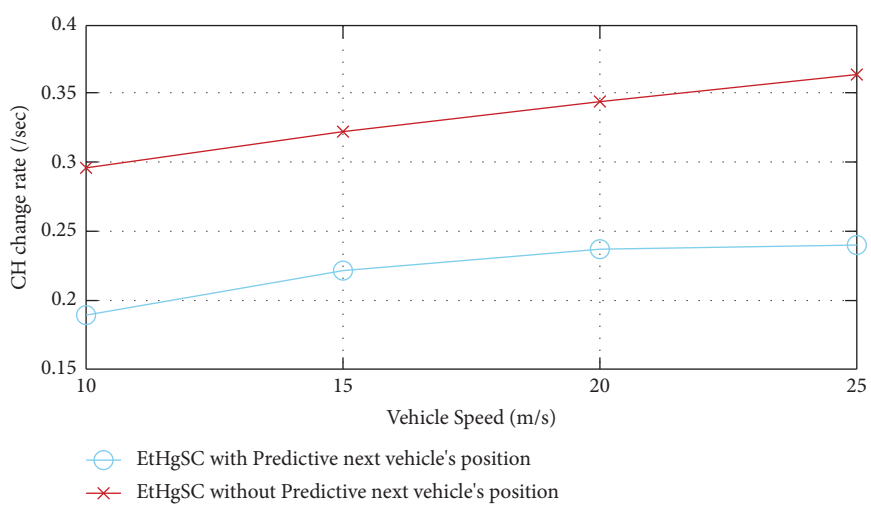

Figure 8: CH change rate for EtHgSC with and without predicted next vehicle's position.

Table 3: Comparative analysis of different algorithms at low traffic density.

Algorithm	Av. CH lifetime (%)	Cluster number
EtHgSC	81	5
CVoEG	65.5	20
JCV	76	16
Arkian et al. [15]	58	55
VMaSC	66	17
PMC	41	—

The PMC algorithm achieves 41% stability. This ratio is the lowest compared with all other algorithms although the CH selection is based on the neighbor following strategy. The authors did not mention the number of constructed clusters, such that Table 3 lacks that.

We can see that there is a significant effect of the number of constructed clusters on stability; a few clusters' numbers improve the clustering stability. The highest stability is achieved using our EtHgSC. So, we can conclude that the formation of VANET as a hypergraph improves the clustering efficiency compared with other algorithms. Also, the use of the Eigen-trick method to improve the hypergraph algorithm as well as the CH selection scheme of our EtHgSC algorithm helps to achieve high stability.

The results in Figures 9–12 evaluate the clustering stability using average CH lifetime, average CM lifetime, CH change rate, and delay. Our EtHgSC is evaluated at different speeds (10, 15, 20, 25, 30 m/s) and under low traffic level (100 vehicles) and compared with the JCV, VMaSC, and PMC algorithms to show its supremacy.

4.2.1. Average CH Lifetime. We can see that the average CH lifetime indicates a decreasing trend as vehicle speed increases. The network's architecture significantly changes as a result of the speed increase, breaking the connection. From Figure 9, the average CH lifetime for JCV is higher than that of VMaSC and PMC because the JCV algorithm takes into consideration the change of vehicle route at the junctions when selecting the CH, so it achieves a high CH duration. We can see that our EtHgSC has the highest duration time of CH in a comparison with JCV, VMaSC, and PMC. This achievement is due to our intelligent cluster formation algorithm and the CH selection scheme, which helps to select the vehicle that stays in the cluster as long as possible based on different measures. Also, our proposed model solved the problem of unpredicted changes for CH at the junctions, which led to an increase in the stability of the CH.

4.2.2. Average CM Lifetime. Figure 10 shows the lifetime of CM for different speeds. The PMC algorithm has a higher CM lifetime than VMaSC, and this is due to the use of the following vehicle with the highest priority. However, the average CH movement speed is an issue for VMaSC, so when a cluster's speed increases, joining to other clusters is simple, and therefore, the lifetime of CM decreases.

Although the duration of CMs decreases when the speed is increased, our proposed EtHgSC still maintains consistency. The EtHgSC achieves the highest lifetime for CMs of about 10%, 35%, and 86% compared with the JCV, PMC, and VMaSC, respectively, all vehicle speeds.

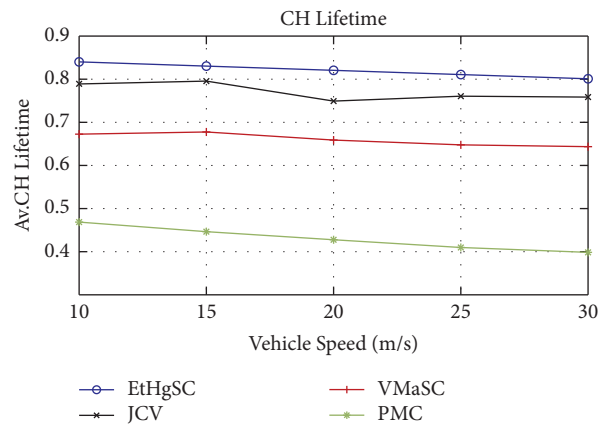

Figure 9: Average CH lifetime at different speeds.

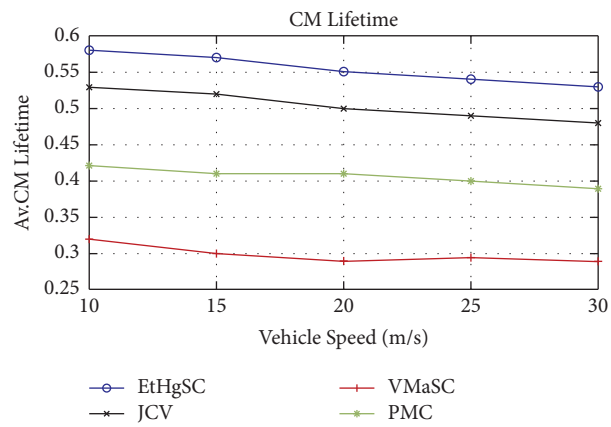

Figure 10: Average CM lifetime at different speeds.

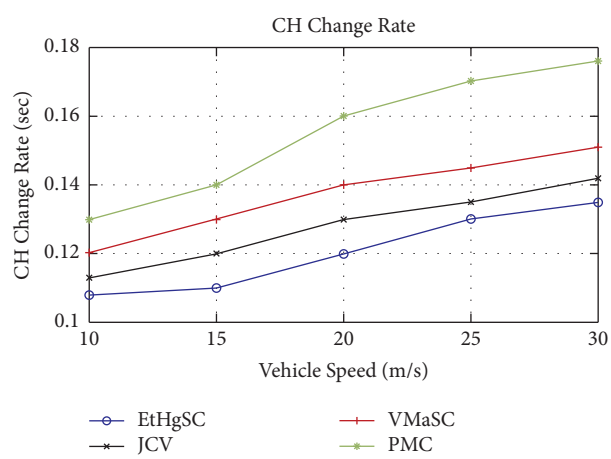

Figure 11: CH change rate at different speeds.

4.2.3. CH Change Rate. Figure 11 shows the relationship between the change rate of CH and the vehicle speed. The results show that the CH change rate increases with an

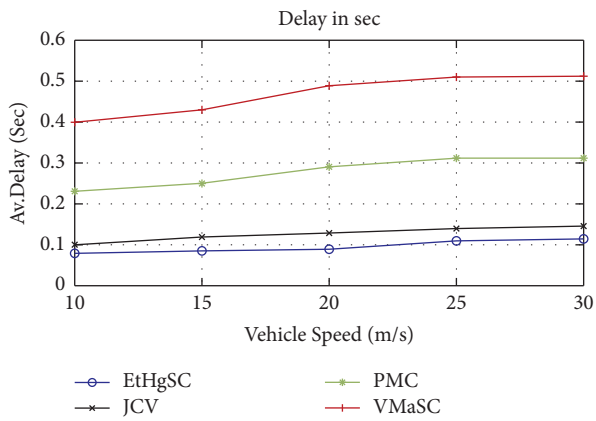

Figure 12: Average delay at different speeds.

increase in the maximum permitted speed. The CH change rate for our EtHgSC is lower than JCV, VMaSC, and PMC under all conditions. The change rate of CH in our EtHgSC is reduced by 6%, 14%, and 23% compared with JCV, VMaSC, and PMC, respectively, all vehicle speeds.

4.2.4. Average Delay. The average delay is also calculated for our proposed model; it is the time taken for a packet to transfer from source to destination. The delay is a distance-dependent parameter, and the optimal CH's location helps to reduce the delay, so the EtHgSC achieves less delay, approximately 24%, 66%, and 80% in a comparison with the JCV, PMC, and VMaSC algorithms, respectively, as seen in Figure 12.

5. Discussion

This work has three novel contributions: the first one is the improved hypergraph algorithm using the Eigen-trick method to optimally cluster vehicles. The time to leave estimation T'_{leave} is another novel addition to our work. Also, based on the grey relational analysis model (GRA), a novel metric for selecting the CH is proposed that meets the requirements of maximum relative speed (ψvehi), neighborhood degree (n_d), eccentricity (Ecc), and maximum time to leave estimation (T'_{leave}).

The change in the vehicle's direction at the road junction hammers the stability. Finding the estimated next vehicle position with the aid of PGRP prediction aids in choosing the CH that has a high ability to stay in the cluster as long as possible, thus improving the overall stability. It is important to find the estimated next vehicle position when calculating the time to leave parameter. So, from the results, we can see that the CH is more stable when using the estimated time to leave T'_{leave} instead of the time to leave without estimation T_{leave}. As a result, the proposed approach achieves more stability and avoids frequent cluster breakage at the junctions.

Also, compared to the most common clustering algorithms in the literature, the JCV method follows our proposed EtHgSC method in terms of stability, because the two methods solve the problem of CH stability at junctions by preventing the frequent cluster breakage. The two methods take into consideration the change of vehicle route at the junctions when selecting the CH.

Finally, although the JCV method achieves good stability, our proposed EtHgSC method is the best, achieving 81% stability at low traffic levels while the JCV method achieves 76% stability. This is because of the intelligent cluster formation method using the Eigen-trick method, which is used to improve the hypergraph algorithm and make clustering more efficient. The scheme of CH selection using the four measures and the GRA model is also a reason why stability is getting better. In addition, we mentioned earlier that a few clusters number is able to connect effectively and maintain stable clustering. So, from the results section, we see that our efficient hypergraph algorithm constructs the fewest number of clusters (5 clusters) compared with other algorithms.

6. Conclusion and Future Work

In this article, an Eigen-trick-based stable clustering algorithm (EtHgSC) in VANET has been introduced with the goal of increasing clustering stability by employing efficient cluster formation and cluster head selection methods. The proposed scheme includes two main parts: the cluster formation phase and the cluster head selection phase.

The cluster formation has been handled using an improved hypergraph-based spectral clustering algorithm. The hypergraph algorithm has been improved using the Eigen-trick method; this method is used to partition both vertices and hyperedges, which provides an approach for reducing the computational complexity of the clustering. By utilizing higher-order information in eigenvalues, the Eigen-trick improves clustering efficiency. Four parameters have been used for selecting the cluster head: in addition to relative speed, neighboring degree, and eccentricity, the vehicle's time to leave parameter has been presented. In this parameter, due to the possibility of a vehicle changing its direction, the estimated next vehicle's position is used to get good and more stable results. So, the proposed approach achieves more stability and avoids frequent cluster breakage at the junctions. Our proposed approach has been applied to an urban scenario at low and high traffic levels and at different vehicle speeds. In comparison to other state-of-the-art algorithms, the results show that our proposed scheme is superior in terms of cluster stability and packet delay.

In future work, to achieve more effective clustering, we will try to employ a modularity matrix rather than an adjacency matrix to generate more efficient clustering along with a vehicle's lane index corresponding to other vehicles to improve the CH stability.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] M. K. Jabbar and H. Trabelsi, "Clustering review in vehicular ad hoc networks: algorithms, comparisons, challenges and solutions," *Int. J. Interact. Mob. Technol.*vol. 16, no. 10, pp. 25–48, 2022.

[2] P. Basu, N. Khan, and T. D. C. Little, "A mobility based metric for clustering in mobile ad hoc networks," in *Proceedings of the 21st international conference on distributed computing systems workshops*, pp. 413–418, Mesa, Arizona, April 2001.

[3] M. Chatterjee, S. K. Das, and D. Turgut, "WCA: a weighted clustering algorithm for mobile ad hoc networks," *Cluster Computing*, vol. 5, no. 2, pp. 193–204, 2002.

[4] M. K. J. Alsabah, H. Trabelsi, and W. Jerbi, "Survey on clustering in VANET networks," in *Proceedings of the 2021 18th International Multi-Conference on Systems, Signals and Devices (SSD)*, pp. 493–502, Monastir, Tunisia, March 2021.

[5] M. K. Jabbar and H. Trabelsi, "A review on clustering in VANET: algorithms, phases, and comparisons," in *Proceedings of the 2022 19th International Multi-Conference on Systems, Signals and Devices (SSD)*, pp. 444–451, Sétif, Algeria, May 2022.

[6] Z. Khan, P. Fan, S. Fang, and F. Abbas, "An unsupervised cluster-based VANET-oriented evolving graph (CVoEG) model and associated reliable routing scheme," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 10, pp. 3844–3859, 2019.

[7] Z. Khan, A. Koubaa, S. Fang, M. Y. Lee, and K. Muhammad, "A connectivity-based clustering scheme for intelligent vehicles," *Applied Sciences*, vol. 11, no. 5, p. 2413, 2021.

[8] S. Elaraby and S. M. Abuelenin, "Connectivity analysis of directed highway vehicular ad hoc networks using graph theory," *International Journal of Communication Systems*, vol. 34, no. 5, Article ID e4745, 2021.

[9] P. S. Chodrow, N. Veldt, and A. R. Benson, "Generative hypergraph clustering: from blockmodels to modularity," *Science Advances*, vol. 7, no. 28, Article ID eabh1303, 2021.

[10] M. K. Jabbar and H. Trabelsi, "A novelty of hypergraph clustering model (HGCM) for urban scenario in VANET," *IEEE Access*, 2022.

[11] C. Shea, B. Hassanabadi, and S. Valaee, "Mobility-based clustering in VANETs using affinity propagation," in *Proceedings of the GLOBECOM 2009-2009 IEEE Global Telecommunications Conference*, pp. 1–6, Honolulu, HI, USA, December 2009.

[12] M. Ren, L. Khoukhi, H. Labiod, J. Zhang, and V. Veque, "A mobility-based scheme for dynamic clustering in vehicular ad-hoc networks (VANETs)," *Vehicular Communications*, vol. 9, pp. 233–241, 2017.

[13] S. Ucar, S. C. Ergen, and O. Ozkasap, "VMaSC: vehicular multi-hop algorithm for stable clustering in vehicular ad hoc networks," in *Proceedings of the 2013 IEEE wireless communications and networking conference (WCNC)*, pp. 2381–2386, Shanghai, China, April 2013.

[14] S. Ucar, S. C. Ergen, and O. Ozkasap, "Multihop-cluster-based IEEE 802.11 p and LTE hybrid architecture for VANET safety message dissemination," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 4, pp. 2621–2636, 2016.

[15] H. R. Arkian, R. Ebrahimi Atani, A. Pourkhalili, and S. Kamali, "Cluster-based traffic information generalization in vehicular ad-hoc networks," *Vehicular Communications*, vol. 1, no. 4, pp. 197–207, 2014.

[16] D. Zhang, H. Ge, T. Zhang, Y.-Y. Cui, X. Liu, and G. Mao, "New multi-hop clustering algorithm for vehicular ad hoc networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 4, pp. 1517–1530, 2019.

[17] M. A. Saleem, S. Zhou, A. Sharif et al., "Expansion of cluster head stability using fuzzy in cognitive radio CR-VANET," *IEEE Access*, vol. 7, pp. 173185–173195, 2019.

[18] K. Bylykbashi, D. Elmazi, K. Matsuo, M. Ikeda, and L. Barolli, "Effect of security and trustworthiness for a fuzzy cluster management system in VANETs," *Cognitive Systems Research*, vol. 55, pp. 153–163, 2019.

[19] I. Hussain and C. Bingcai, "Cluster formation and cluster head selection approach for vehicle ad-hoc network (VANETs) using K-means and floyd-Warshall technique," *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 12, p. 1, 2017.

[20] K. Kandali, L. Bennis, and H. Bennis, "A new hybrid routing protocol using a modified K-means clustering algorithm and continuous hopfield network for VANET," *IEEE Access*, vol. 9, pp. 47169–47183, 2021.

[21] M. K. Jabbar and H. Trabelsi, "A betweenness centrality based clustering in VANETs," in *Proceedings of the 2022 15th International Conference on Security of Information and Networks (SIN)*, pp. 1–4, Sousse, Tunisia, November 2022.

[22] Z. Wang, L. Liu, M. Zhou, and N. Ansari, "A position-based clustering technique for ad hoc intervehicle communication," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 38, no. 2, pp. 201–208, 2008.

[23] O. Senouci, S. Harous, and Z. Aliouat, "A new heuristic clustering algorithm based on RSU for internet of vehicles," *Arabian Journal for Science and Engineering*, vol. 44, no. 11, pp. 9735–9753, 2019.

[24] M. Mukhtaruzzaman and M. Atiquzzaman, "Junction-based stable clustering algorithm for vehicular ad hoc network," *Annales des Telecommunications*, vol. 76, no. 9-10, pp. 777–786, 2021.

[25] D. Ghoshdastidar and A. Dukkipati, "Uniform hypergraph partitioning: provable tensor methods and sampling techniques," *Journal of Machine Learning Research*, vol. 18, no. 1, pp. 1638–1678, 2017.

[26] Y. Yang, S. Deng, J. Lu et al., "GraphLSHC: towards large scale spectral hypergraph clustering," *Information Sciences*, vol. 544, pp. 117–134, 2021.

[27] X. Wang and Y. Xu, "An improved index for clustering validation based on Silhouette index and Calinski-Harabasz index," *IOP Conference Series: Materials Science and Engineering*, vol. 569, no. 5, Article ID 052024, 2019.

[28] D. Kim, Y. Velasco, W. Wang, R. N. Uma, R. Hussain, and S. Lee, "A new comprehensive RSU installation strategy for cost-efficient VANET deployment," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 5, pp. 1–4211, 2016.

[29] R. Adrian, S. Sulistyo, I. W. Mustika, and S. Alam, "A controllable rsu and vampire moth to support the cluster stability in vanet," *International Journal of Computer Networks and Communications*, vol. 13, no. 3, pp. 79–95, 2021.

[30] D. R. White and S. P. Borgatti, "Betweenness centrality measures for directed graphs," *Social Networks*, vol. 16, no. 4, pp. 335–346, 1994.

[31] M. S. Kakkasageri and S. S. Manvi, "Multiagent driven dynamic clustering of vehicles in VANETs," *Journal of Network*

and Computer Applications, vol. 35, no. 6, pp. 1771–1780, 2012.

[32] R. Karimi and S. Shokrollahi, "PGRP: Predictive geographic routing protocol for VANETs," *Computer Networks*, vol. 141, pp. 67–81, 2018.

[33] A. Jayant, "Use of grey relational analysis in solving multiple attribute decision-making problem: a case study of warehouse location selection," *Advances in Industrial Engineering and Management*, vol. 4, no. 2, pp. 157–164, 2015.

[34] Y. Kuo, T. Yang, and G.-W. Huang, "The use of grey relational analysis in solving multiple attribute decision-making problems," *Computers and Industrial Engineering*, vol. 55, no. 1, pp. 80–93, 2008.

[35] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, "SUMO--simulation of urban mobility: an overview," in *Proceedings of the SIMUL 2011, Third Int. Conf. Adv. Syst. Simul*, Barcelona, Spain, October 2011.

[36] R. Adrian, S. Sulistyo, and I. W. Mustika, "A study on communication system in VANET," in *Proceedings of the 2018 4th International Conference on Science and Technology (ICST)*, pp. 1–6, Limerick, Ireland, October 2018.