Hindawi
Wireless Communications and Mobile Computing
Volume 2019, Article ID 8415234, 10 pages
https://doi.org/10.1155/2019/8415234

Research Article

A Center-Based Secure and Stable Clustering Algorithm for VANETs on Highways

Xiaolu Cheng[1] and Baohua Huang[2]

[1] *Virginia Commonwealth University, Richmond, Virginia 23220, USA*
[2] *Guangxi University, Nanning, Guangxi 530004, China*

Correspondence should be addressed to Baohua Huang; bhhuang66@gxu.edu.cn

Received 15 October 2018; Revised 5 December 2018; Accepted 20 December 2018; Published 2 January 2019

Guest Editor: Zaobo He

Currently, communications in the vehicular ad hoc network (VANET) can be established via both Dedicated Short Range Communication (DSRC) and mobile cellular networks. To make use of existing Long Term Evolution (LTE) network in data transmissions, many methods are proposed to manage VANETs. Grouping the vehicles into clusters and organizing the network by clusters are one of the most universal and most efficacious ways. Since the high mobility of vehicles makes VANETs different from other mobile ad hoc networks (MANETs), the previous cluster-based methods for MANETs may have trouble for VANETs. In this paper, we introduce a center-based clustering algorithm to help self-organized VANETs forming stable clusters and decrease the status change frequency of vehicles on highways and two metrics. A novel Cluster Head (CH) selection algorithm is also proposed to reduce the impact of vehicle motion differences. We also introduce two metrics to improve the security of VANETs. A simulation is conducted to compare our mechanism to some other mechanisms. The results show that our mechanism obtains high stability and lower packet loss rate.

1. Introduction

As a key component of Intelligent Transportation Systems (ITS), vehicular ad hoc network (VANET) has attracted plenty of researchers from different fields, and massive research efforts have been made.

In VANETs, there are two types of communications [1]. VANETs enable both vehicle-to-vehicle (V2V) communications and vehicle-to-infrastructure (V2I) communications. In VANETs, vehicles and the infrastructures, such as Roadside Units (RSU) and application servers, exchange information for navigation, safe driving, entertainment, and so on.

Generally, communications in VANETs are roughly categorized into two classes according to the adopted radio interfaces. One class of approaches is based on Dedicated Short Range Communication (DSRC). The other class is based on existing cellular technology [2].

DSRC began to be used for V2V communication from the 90s. It has a shortage in medium range, which is about 300 meters. It is inadequate for large-scale deployment [3] because its coverage radius is not large enough. With the rapid improvement of mobile cellular networks, some researchers supposed to utilize the existing mobile cellular infrastructures and technologies for communications in VANETs. Mobile cellular networks provide wider and larger coverage, while their delay is longer than DSRC for real-time information exchanges in local areas [4]. Therefore, both DSRC and mobile cellular networks cannot fully meet the needs of ITS. As a result, VANETs support communication not only via LTE but also via DSRC.

To make use of existing mobile cellular networks for data transmissions, many methods are proposed to manage VANETs. However, if VANETs are fully managed by infrastructures, low efficiency will be a big issue, while fully decentralized VANETs must create a lot of overhead. Therefore, VANETs usually combine some centralized parts and decentralized parts. To decrease the overhead via DSRC channels and the probability of LTE channel congestion, VANETs are centralized by cellular-based connections and scheduling. Meanwhile, vehicles may also exchange messages with their neighbors via DSRC. Dividing vehicles into clusters is a common and reasonable approach for VANETs management. In a cluster-based framework, vehicles are signed into clusters. The range of a cluster is smaller or equal to the range of 802.11p, so that vehicles in the same cluster can exchange messages via DSRC. A single eNodeB manages many clusters around it. Within a cluster, at least one vehicle performs as a Cluster Head (CH) to collect information of all Cluster Members (CM) via DSRC and exchanges data with the eNodeB via TLE. This architecture decreases the management overhead while utilizing both DSRC and LTE.

Compared to other MANETs, nodes in VANETs have higher mobility and higher speed. Cluster reforming and CH changing must be much more frequent than other typical MANETs. To decrease the management overhead and increase communication quality, the clustering algorithm for VANETs should be able to form stable clusters. To achieve this goal, in this paper we propose a stable clustering algorithm for VANETs. We propose a novel approach to form and maintain stable clusters for VANETs on highways to avoid continual cluster reforming. A center-based clustering algorithm is used to locate the initial clusters' centers. In every cluster, a suitable CH is chosen by vehicles' position, speed, and maximal acceleration. A cluster maintenance algorithm is proposed to keep CMs in its CH's transmission range.

The rest of the paper is organized as follows. The Related Work briefly reviews the current literature on clustering algorithms in VANETs. The proposed scheme is detailed in the Proposed Scheme. The simulation parameters, simulation results, and analysis are shown in the Performance Evaluation. In the Conclusion, we state the conclusion.

2. Related Work

In the literature, clustering is the process to group vehicles in VANETS.

Ref. [5] proposes a method, named LTE4V2X, to organize vehicular networks. In the centralized vehicular networks, eNodeB manages vehicles in its coverage and divides them into clusters. LTE4V2X protocol defines how the self-organized network works. In LTE4V2X, eNodeB creates clusters which contain the largest number of nodes circulating in the same direction.

Ref. [6] extends LTE4V2X to increase information dissemination efficiency. It selects CH by the distance from vehicles to eNodeB. Although, comparing to the original approach, the complexity is lower and the LTE channel quality is higher, the power consumption of message exchanging is not optimized. Nevertheless, [6] states that the system can calculate the transmit power of DSRC channels by the distance between vehicles so that the transmit power could be dynamically adjusted.

Road condition affects the speed and direction of vehicles. For example, vehicle's speed is lower on the bumpy road than a smooth road. Vehicle mobility is determined by human behavior. Take a street connected megapolis and a village as an example. In the morning, most vehicles move from the village (home) to the megapolis (office). In the evening, most vehicles run following the reverse path. Ref. [7] quantifies temporal locality similarity to measure the relation of two vehicles' mobility. Then, they utilize the relation of vehicles' movements to form stable clusters. The locality can also be used for reducing energy consumption [8].

Ref. [9] proposes a clustering approach to minimize the total power consumed by DSRC communications. They use a weighted distance matrix to indicate power consumed between each pair of vehicles. In this way, the CH selection problem is formulated as a variant of the p-median problem in graph theory [10]. In this approach, the number of clusters p is determined first based on LTE coverage radius and DSRC coverage radius. The p cluster zones are determined by vehicle number and 802.11P coverage radius. p Cluster Heads that are closet to the eNodeB are selected. Then, the system dynamically selects new CH to minimize the transmission power between CMs and CH based on weighted distance and the p-median issue in graph theory. Although this approach minimizes the power consumption within a single cluster, the power consumption of V2I communications has not been considered. The method to decide the zones is vague and complicated. Moreover, this approach is not suitable for the scenario that CMs not only send their information to CH but also communicate among themselves.

Ref. [11] proposes a high-integrity file transfer scheme for VANETs on highways named Cluster-based File Transfer (CFT) scheme. In this scheme, CMs help their CH to download file fragments and then transmit fragments to the CH which requests the file. Since the very high speed of vehicles on highways, CFT is a good approach to help the vehicles download files which they have not enough connection time to download. However, CFT just considers the bidirection environment. In addition, with CFT CH broadcasts its request to its neighbors; then, neighbors that receive the invitation join the cluster and broadcast the request to invite more vehicles to join the cluster until there are enough vehicles. Therefore, CFT may not able to apply in complicated environment, and it may cause network congestions.

Ref. [12] proposes an evolutionary game theoretic (EGT) framework for clustering and CH selecting. Their protocol is based on game theory. They defined the net utility of a CH to select the CH which may achieve high throughput. A cluster size is added in the utility function for CH to optimize the size of a cluster. Ref. [13] proposes an intelligent naive Bayesian probabilistic estimation practice (ANTSC) method. This method is based on the traffic flow. To increase the stability of cluster, a CH must be in the lane having the heaviest traffic flow. Naive Bayes algorithm is used to select the CH which may make the cluster most stable. However, [12] just compared the EGT clustering with one clustering algorithm proposed in 2010. Both [12, 13] did not consider the security of the network.

3. Proposed Scheme

3.1. Overview and Assumption. Clustering algorithm groups a set of unlabeled nodes into clusters. In cluster-based VANETs, all vehicles send their information to eNodeB.

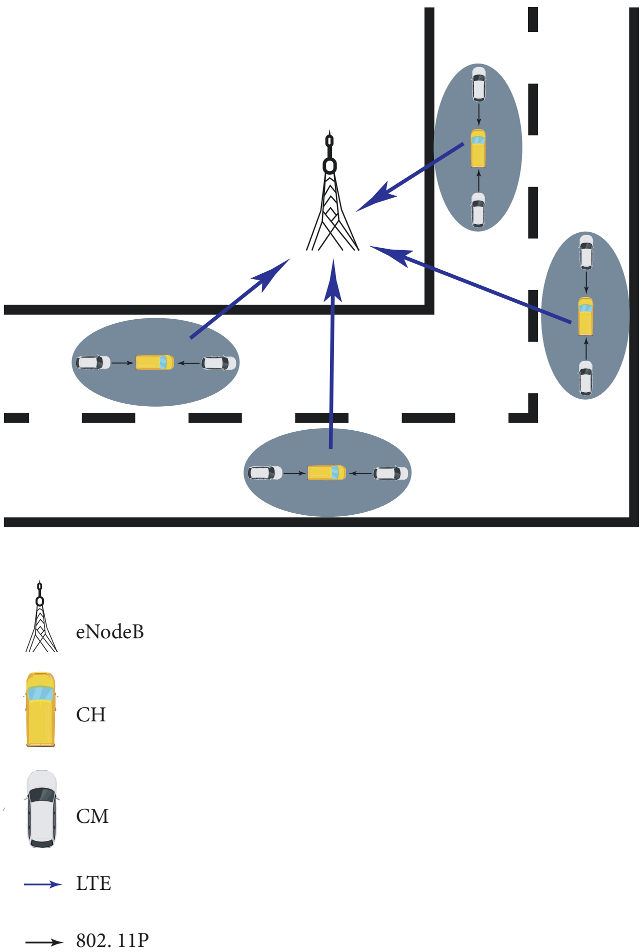

Figure 1: Communications within one cluster.

Then, eNodeB manages the vehicles by clusters. A CH acts as a messenger to help eNodeB and CMs exchange information.

We assume all vehicles are able to communicate via both LTE and DSRC. The size of cluster is smaller or equal to the range of 802.11p, so that vehicles in the same cluster can exchange messages via DSRC. DSRC coverage radius is about 300 meters. LTE coverage radius is about 1 kilometer. Therefore, a single eNodeB manages many clusters around it. Within a cluster, a vehicle acts as a CH to collect information of all CMs via 802.11p and exchanges data with the eNodeB via TLE. Figure 1 is a simplified view of a cluster-based vehicular network.

For cluster in this paper, we have some assumptions:

(1) All vehicles have both LTE and 802.11p interfaces

(2) All vehicles are equipped with Global Positioning System (GPS) devises. So, they have accurate geolocations

(3) All vehicles know their destination, speed, and maximal acceleration

Based on the assumptions, we propose a center detection based clustering algorithm. We group the vehicles in the region where the density of vehicles is higher than other areas into clusters with the help of blob detection method or an improved high-degree algorithm. Some parameters, such as speed and acceleration, are added to the CH selection metric to make the cluster stabler and decrease the CH reselection frequency.

3.2. Cluster Formation. In our proposed algorithm, in the initialization stage of cluster formation, vehicles send beacon messages to the eNodeB. The beacon message of one vehicle contains the vehicle's ID k, current position (x_k, y_k), current speed v_k, maximal acceleration a_k, and direction type t_k.

Direction type is decided by the angle from the current position to the destination. For vehicle k, whose destination position is (x'_k, y'_k), the direction angle θ_k is

$$\theta_k = \tan^{-1}\frac{y'_k - y_k}{x'_k - x_k} \qquad (1)$$

When $\theta_k \in [0^\circ, 90^\circ)$, $t_k = 1$. When $\theta_k \in [90^\circ, 180^\circ)$, $t_k = 2$. When $\theta_k \in [180^\circ, 270^\circ)$, $t_k = 3$. When $\theta_k \in [270^\circ, 360^\circ)$, $t_k = 4$. Vehicles that have different t are managed, respectively.

The clustering algorithm is described in Algorithm 1.

After receiving the beacon messages, the system analyzes vehicles' position information and detects the centers of the ranges where the vehicle density is higher than in other areas. If the vehicle quantity or the vehicle density is not very large, an improved Highest-Degree Algorithm is applied. Several vehicles which have more neighbors in their transmit range are detected. We improve the original Highest-Degree Algorithm to make sure the distance between any two vehicles we detected is larger than the DRSC range. The positions of detected vehicles will be the centers we use in the clustering algorithm. Otherwise, when the vehicle quantity and the vehicle density are very large, to decrease the computing complexity and analyze time, the system draws dots on the map to indicate vehicles. Then, we can carry out the blob detection. The blob detection is able to detect the regions where the gray pixel value is greater. Thus, we can use the blob detection algorithm, e.g., [14], to detect the centers of regions on the map where dot density is higher.

All vehicles whose distances to the center are not larger than the range of DSRC are labeled as one cluster. Then, the system selects one nearest intersection for every center among all intersections that meet the following conditions:

(1) The distance from it to the points in P is not smaller than the range of DSRC

(2) The intersection is not in any cluster's region

Vehicles near those selected intersections are grouped into clusters. Then, eNodeB uses the same way to select intersections near the selected intersections and groups vehicles. After iterations, ungrouped vehicles are grouped into clusters. The distance between two vehicles in the same cluster is not larger than the range of DRSC. To further decrease computing complexity, in line 8 of clustering algorithm, a vehicle or infrastructure located in the center or intersection can broadcast a request to invite neighbors to join the cluster. In line 37, the chosen vehicle e can broadcast an invitation instead of calculating distance by the system.

```
Input: Vehicle set V
Output: Initial clusters
Initialize center set $C = \phi$;
Locate the centers and add them into $C$;
Initialize point set $P = C$;
while $P \neq \phi$ do
    foreach point $p$ in $P$ do
        Initialize node set $cluster_p = \phi$;
        foreach vehicle $e$ in $V$ do
            if $d_{ep} \leq R$ then
                Add $e$ into $cluster_p$;
                Remove $e$ from $V$;
            end
        end
        if $cluster_p \neq \phi$ then
            Call Algorithm 2;
            Return set $cluster_p$;
            Estimate $S_{tc}$ of the CH $c$ of $cluster_p$;
            if $S_{tc}$ is remarkable high then
                Check all nodes in $cluster_p$ to detect attacker;
            end
            else
                $c$ check $S_a$ value of each CM;
                if $S_{am}$ is remarkable high then
                    Report to the server;
                end
            end
            Add the intersection nearest to $p$ which meets the conditions into $P$;
        end
        Remove $p$ from $P$;
    end
end
while $V \neq \phi$ do
    foreach point $c$ in $C$ do
        Select an element $e$ in $V$ nearest to $c$;
        Initialize set $cluster_e = \{e\}$;
        Remove $e$ from $V$;
        Set $e$ as CH;
        foreach vehicle $v$ in $V$ do
            if $d_{ev} \leq R$ then
                Add $v$ into $cluster_e$;
                Remove $v$ from $V$;
            end
        end
        Return $cluster_e$;
        Estimate $S_{tc}$ of the CH $e$ of $cluster_e$;
        if $S_{tc}$ is remarkable high then
            Check all nodes in $cluster_e$ to detect attacker;
        end
        else
            $c$ check $S_a$ value of each CM;
            if $S_{am}$ is remarkable high then
                Report to the server;
            end
        end
    end
end
```

Algorithm 1: Clustering algorithm.

3.3. Cluster Head Selection. Compared to other MANETs, VANETs have lower stability, because of the high mobility of vehicles. Although we divide the vehicles with the help of direction vector $\overrightarrow{v}_k$, the stability of clusters cannot be guaranteed. To select an appropriate CH which can increase the cluster lifetime and decrease the CH reselecting frequency, a relative mobility metric $\boldsymbol{M}$ is introduced for CH election.

For a vehicle k, which is in the cluster $cluster_i$, the position difference between it and all other N vehicles in the same cluster $cluster_i$ is

$$D_k = \sum_{n=1}^{N} \sqrt{(x_k - x_n)^2 + (y_k - y_n)^2} \tag{2}$$

The speed difference between k and all other N vehicles in the same cluster is

$$V_k = \sum_{n=1}^{N} |v_k - v_n| \tag{3}$$

The maximal acceleration difference between k and all other N vehicles in the same cluster is

$$A_k = \sum_{n=1}^{N} |a_k - a_n| \tag{4}$$

The relative mobility metric $\boldsymbol{M}$ is

$$\begin{aligned} \mathrm{M}_k = {} & \alpha \frac{D_k}{\max\{D_n \mid \forall n \in C_i\}} + \beta \frac{V_k}{\max\{V_n \mid \forall n \in C_i\}} \\ & + \gamma \frac{A_k}{\max\{A_n \mid \forall n \in C_i\}}, \end{aligned} \tag{5}$$

where α, β, and γ are weighted coefficients. $\alpha+\beta+\gamma = 1$. They can be adjusted to fit the different traffic conditions. When the traffic condition is good and all vehicles are driving at a similar speed, the distance between vehicles has a greater effect. Thus, the value of α should be higher than the other two. When vehicles are driving at high speed, the value of β is higher than the other two. When the vehicles enter an area which speed limit changes continually, a higher γ should be considered.

The relative mobility metric $\boldsymbol{M}$ evaluates the relative position, speed, and maximal acceleration differences between one vehicle and all other vehicles in the same cluster. A smaller $\boldsymbol{M}$ indicates the vehicle has lower relative mobility than other vehicles in this cluster. Algorithm 2 explains the process of Cluster Head selection. All clusters formed with the help of centers and intersections use Algorithm 2 to select CH. As a CH, the vehicle's relative mobility metric is smaller than any CMs. That means the motion mode of CH is similar to the whole cluster.

Input: Vehicles in one cluster
Output: Cluster head o of the corresponding cluster
Set $\mathbf{M}_{min} = +\infty$;
foreach *vehicle k* **do**
    Calculate the relative mobility metric $\mathbf{M}_k$;
    if $\boldsymbol{M}_k < \boldsymbol{M}_{min}$ **then**
        $\mathbf{M}_{min} = \mathbf{M}_k$;
        $o = k$;
    end
end
return o;

Algorithm 2: Cluster head selection algorithm.

3.4. Cluster Maintenance and Reforming. The unpredictability and mobility of traffic make the cluster lifetime temporary. It is infeasible to reform clusters in real time or very frequently. To minimize the frequency and overhead of cluster reforming, we propose a cluster maintenance algorithm. Algorithm 3 explains the cluster maintenance process.

(1) No Connections between CH and CM. When a CH cannot connect to a CM, the CH will delete the CM from its record and notice eNodeB. When a CM cannot reach its CH, the CM will check the signal it received via DSRC and join the cluster whose signal of CH is strongest. If the CM cannot receive a message strong enough, it will notice eNodeB via LTE and become a CH.

(2) No Connections between eNodeB and CH. When eNodeB notices it has lost connection to a CH, it recalls Cluster Head Selection Algorithm and a new vehicle will be CH of that cluster instead of the leaving vehicle.

(3) A Vehicle Joins the Network. When a vehicle comes into the network, it first tries to join the nearest cluster by broadcasting a CH request via DSRC. If it fails, it will send a message to eNodeB. eNodeB will help the vehicle to join a cluster or to be a CH and form a new cluster by itself.

(4) Two Clusters Are Too Close. With the movement of the vehicles, two clusters may be very close. When the distance between two CHs is shorter than R for a period Δt, the two clusters are merged into one cluster. The Cluster Head Selected Algorithm is recalled. A new CH for the new cluster is selected. Then, all vehicles, which are out of the transmission range of the new CH, leave this cluster and check the invitation signal they have received via DSRC and join the cluster whose signal of CH is the strongest. If a vehicle does not find a cluster to join in, it notices eNodeB via LTE and becomes a CH.

3.5. Security Mechanism. To further improve the VANETs security and availability, a novel security mechanism is proposed to detect malicious nodes.

In clustered networks, the availability and security of CHs are incredibly crucial. CHs help the servers to collect and transmit messages to CMs. If an attacker wants the access to other vehicles' private information, it should acts as a CH. The most common and most executable method for an attacker to be selected as a CH is launching a

Input: Initial clusters and vehicle set V
Output: New clusters
if *the eNodeB can not reach a CH* **then**
    Call Cluster Head Selection Algorithm;
end
if *the CH can not reach a CM* **then**
    Reomve the CM;
    Notice eNodeB;
end
if *the distance between two CHs* $\leq$ *R for a period* Δt **then**
    Merge the two clusters into one cluster;
    Call the Cluster Head Selected Algorithm;
    Check S_t values of new CHs;
    New CHs check S_a values of their CMs;
end
if *a CM can not reach the CH* **then**
    if *it can receive a signal from CHs* **then**
        Join the cluster whose signal of CH is strongest;
        CH check its S_a value;
    end
    else
        Notice eNodeB;
        The node performs as a CH;
    end
end

Algorithm 3: Cluster maintenance algorithm.

Sybil attack. In a Sybil attack, the vehicle controlled by a malicious attacker presents multiple identities (vehicles), and all of the vehicles have similar directions, positions, speeds, and maximal acceleration. Hence, these vehicles must have higher relative mobility metrics and higher probabilities to be selected as CH.

To protect the CMs' privacy, we introduce a trajectory similarity metric S_t to defend Sybil attacks. For a cluster contains N nodes, the trajectory similarity metric of its CH c is

$$S_{tc} = \frac{1}{N-1} \sum_{i=1}^{N-1} \left(\frac{\Delta t_i}{lifetime_i} \right), \quad (6)$$

where Δt_i is the duration of both c and node i that belong to the same cluster, and $lifetime_i$ is the lifetime of i in this network. Every time a CH is selected, the server estimates its trajectory similarity metric. If one CH has a remarkable higher trajectory similarity metric, the server will check all nodes in the cluster to detect the malicious attacker.

A denial-of-service attack (DoS attack) is another common attack in VANETs. In DoS attack, the attacker floods the CH or server to make the network services unavailable. The connections of authenticated vehicles to the network are temporarily broken. Therefore, the legitimate requests of server and authenticated vehicles cannot be actioned.

To protect the network availability, we introduce an activity similarity metric S_a to detect DoS attacks. For a vehicle m that belongs to a cluster containing N nodes, the activity similarity metric is

$$S_{am} = \frac{\sum_{i=1}^{N-1} p_i - p_m}{\sum_{i=1}^{N-1} p_i} \cdot \frac{N}{N-1}, \quad (7)$$

where p_i is the number of requests between vehicle i and CH c during a period of time Δt; p_m is the number of requests between vehicle m and CH c in the same time period. The higher the activity similarity metric, the greater the proportion of requests of vehicle m in all the requests of this cluster. If one CM has a remarkable higher similarity metric, the CH will regard it as a DoS attacker and report to the server. Meanwhile, the sever can also use activity similarity metrics to detect DoS attack from CHs.

4. Performance Evaluation

4.1. Simulation Parameter. We perform the simulation with the help of Veins LTE. Veins LTE is a simulator developed on Veins [15], which is an open source framework for simulation of vehicular networks based on both IEEE 802.11p and LTE. It integrates a network simulator named OMNeT++ and a traffic simulator named Simulation of Urban MObility (SUMO) [16].

In our experiment, vehicles run on a real map of Washington, DC, USA, obtained from OpenStreetMap [17]. We extract the data of highways in the center of Washington, DC. The total length of road is 30.38 km. The total lane length is

90.09 km. Every vehicle has random source and destination edge. The route from the starting point to the destination is the shortest path found by Dijkstra's algorithm [18]. The maximal acceleration ability of vehicles we have used is 2.6 m/s^2. The maximal deceleration ability of vehicles is 4.5 m/s^2. The vehicle's maximum velocity is 55.55 *m/s*.

We compare our proposed clustering algorithm, Center-Based Stable Clustering Algorithm (CBSC), with a K-Means-Based method (KMB) and SCalE algorithm [19]. K-means algorithm [20] is commonly used in VANETs for clustering, e.g., [21–23]. In KMB method, we divide the vehicles into two parts by the angle of the vehicles and perform KMB on them, respectively. The cluster maintenance algorithm KMB is proposed in [24]. The predefined threshold Δv_{th} is 5 *m/s*. In the simulations, all vehicles' movement information is resent to eNodeB for cluster status update in every 10 seconds. eNodeB needs exchange data with vesicles every 3 seconds. The simulation time is 503 seconds.

4.2. Results and Analysis. The goal of this paper is to propose a stable clustering algorithm for VANETs. To check whether a clustering algorithm can solve the high mobility of vehicles on the highways, the cluster stability should be evaluated. The metrics we use to show the performance of clustering algorithm are as follows:

(1) *Average CH Lifetime*. The CH lifetime is the period from the state when the vehicle is a CH to the state when it is not a CH (i.e., being a CM or leaving the system). When a CH ends its lifetime, a new CH is elected, or the cluster is dissolved.

(2) *Average CM Lifetime*. CM lifetime represents the duration at which a CM stays in the same cluster. The average CM lifetime is the average length of all vehicles' CM lifetime. It is another important metric to evaluate the stability of clusters.

(3) *Average Number of Reaffiliation Times per Vehicle*. The average number of reaffiliation times per vehicle represents the average number of times a vehicle changes the cluster it belongs to during the simulation time.

(4) *Packet Loss Rate*. Packet loss rate is the percentage of packets lost with respect to packets sent.

In the experimentation, we compare the four metrics of the three methods with different vehicle numbers, transmission ranges, or highway speed limits. Figures 2, 4, 6, and 8 show the results obtained with the variety of total vehicle number (N) and the variety of transmission range (R), when the highway speed limit (v) is 100 *km/h*. Figures 3, 5, 7, and 9 show the results obtained with the variety of transmission range (R) and the variety of highway speed limit (v), when the total vehicle number (N) is 300.

Figures 2 and 3 represent the average CH lifetime for the three methods. Those figures show that the CHs under KMB have a marked shorter lifetime. Although our CBSC has a higher value than SCalE a few times, in general, SCalE performs slightly better than CBSC on the average CH lifetime.

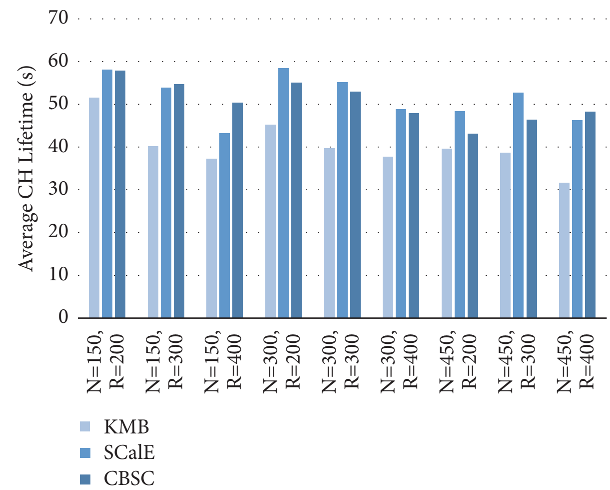

FIGURE 2: Average CH lifetime versus N and R.

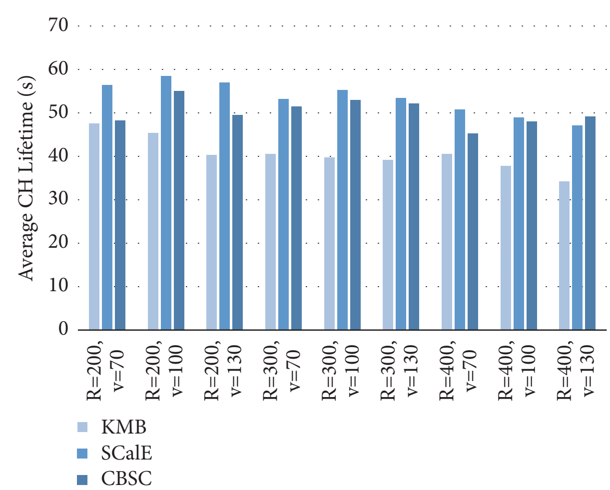

FIGURE 3: Average CH lifetime versus R and v.

The average CM lifetime values produced by KMB, SCalE, and the CBSC methods are shown in Figures 4 and 5. From those two figures, we can see that the average CM lifetime produced by CBSC is much longer than the other two methods. SCalE has the worst performance on the average CM lifetime.

Figures 6 and 7 show the average number of reaffiliation times per vehicle obtained in 503 seconds. Obviously, comparing to other two algorithms, vehicles with SCalE change status much more frequently. The data on the two figures shows CBSC not only produces a lower cluster status change frequency than KMB produces, its superiority, but also is bigger with the increase in highway speed limit.

The results of simulation illustrate that clusters under CBSC are the stablest in the three algorithms. They have the longest average CM lifetime and lowest average number of reaffiliation times per vehicle. Although SCalE performs slightly better than CBSC on the CH lifetime experiment, it produces a much shorter average CM lifetime. Besides, the number of CMs is much larger than the CHs in one system.

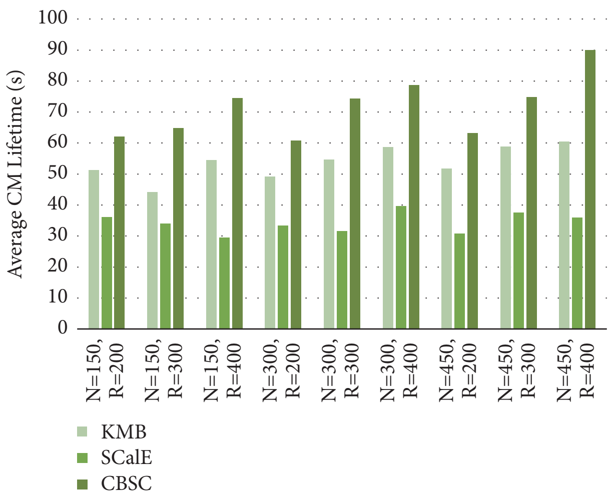

Figure 4: Average CM lifetime versus N and R.

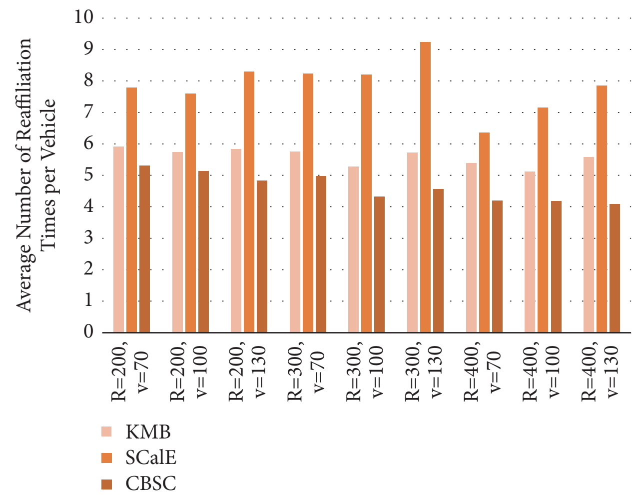

Figure 7: Average number of reaffiliation times per vehicle versus R and v.

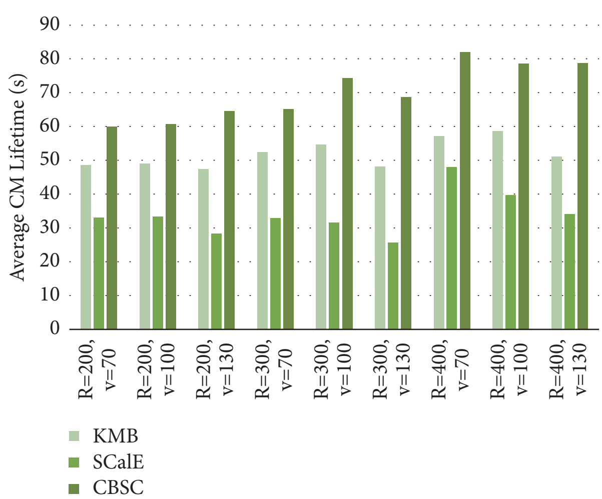

Figure 5: Average CH lifetime versus R and v.

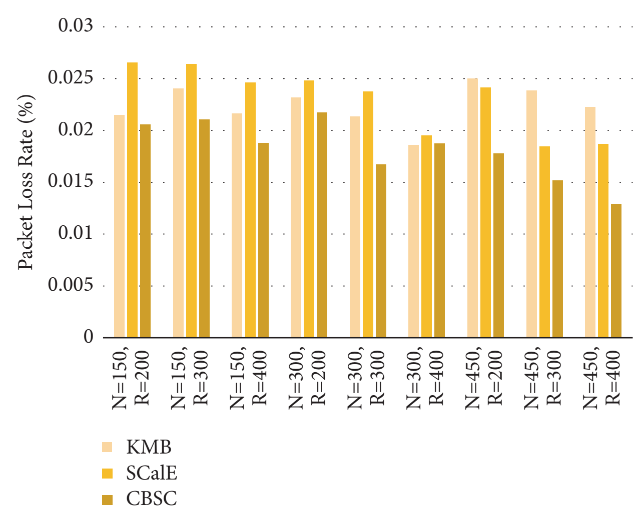

Figure 8: Packet loss rate versus N and R.

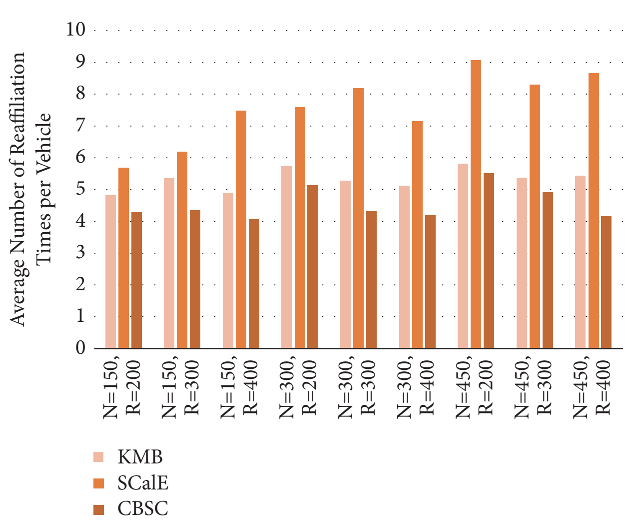

Figure 6: Average number of reaffiliation times per vehicle versus N and R.

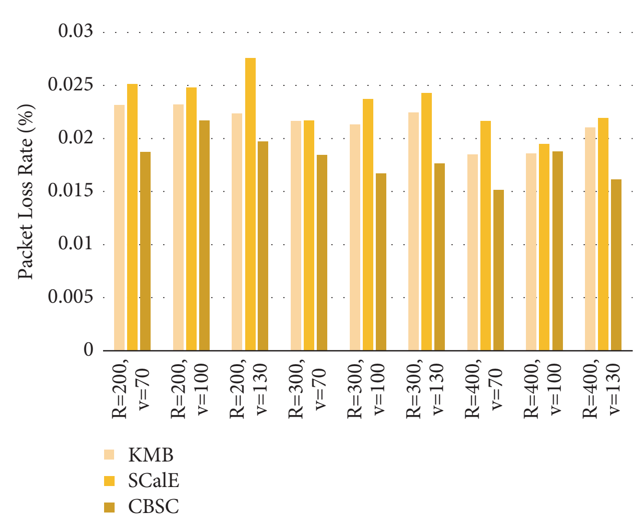

Figure 9: Packet loss rate versus R and v.

Therefore, we consider that CBSC has higher stability than SCalE.

The basic function of VANETs is supporting communication between separated vehicles and infrastructures. To test the performance of data dissemination in VANETs, we do experiment on packet loss rate with different methods. Packet loss means a packet fails to arrive at its destination. A high packet loss rate decreases the data dissemination efficiency and may cause network congestion. Therefore, an efficient data dissemination mechanism should have a low packet loss rate. In our experiment, all vehicles exchange data with eNodeB every three seconds. That means, in every three seconds, eNodeB sends data to all vehicles once, and each vehicle sends data to eNodeB once. Like the scene we described in the previous section, eNodeB communicates with the nodes in its record via CHs, and vehicles which are CMs send data to their CHs first. Figures 8 and 9 show the results of packet loss rate. With the increase in vehicle velocity or the transmission range, the packet loss rates obtained by all the three mechanisms decrease. But CBSC gets lower packet loss rate, while KMB performs the worst, when the amount of vehicle is larger. That insinuates CBSC has a good ability to handle a considerable amount of data. In the experiment, CBSC always obtains lowest packet loss rate. Since the interval between cluster status updates is 10 seconds, we can know that the probability of CM leaving its CH between two cluster status updates in CBSC is lower than the other two algorithms. We can consider that the proposed relative mobility metric $\boldsymbol{M}$ and CH selection algorithm of CBSC do reduce the impact of vehicle mobility on cluster stability.

5. Conclusion

To decrease the management overhead and increase the quality of communications, we try to make the clusters in VANETs as stable as possible while keeping the network performance acceptable. In this paper, we propose a stable clustering algorithm for VANETs on highways, which utilizes direction vector, the centers of vehicle denser areas, and intersections to group less quantity of more stable clusters. To reduce the impact of vehicle type and drivers' driving habits, we propose a novel CH selection algorithm and cluster maintenance algorithm, which use the relative mobility metric to reduce the influence of vehicle's distance, velocity, and maximal acceleration. To protect the vehicles' privacy and the network availability, we introduce two mechanisms to detect malicious attacker. In the simulation experiment, our algorithm's performance ranks up against the other two algorithms (KMB and SCalE) on both stability and package delivery rate. In the future, we would like to further improve the algorithm for the complex urban environment.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] S. Al-Sultan, M. M. Al-Doori, A. H. Al-Bayatti, and H. Zedan, "A comprehensive survey on vehicular Ad Hoc network," *Journal of Network and Computer Applications*, vol. 37, no. 1, pp. 380–392, 2014.

[2] J. Luo and J. P. Hubaux, "A survey of inter-vehicle communication," Tech. Rep., 2004.

[3] Y. J. Li, "An overview of the dsrc/wave technology," in *Proceedings of the International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness*, pp. 544–558, Springer, 2010.

[4] K. Zheng, Q. Zheng, P. Chatzimisios, W. Xiang, and Y. Zhou, "Heterogeneous vehicular networking: a survey on architecture, challenges, and solutions," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2377–2396, 2015.

[5] G. Remy, S.-M. Senouci, F. Jan, and Y. Gourhant, "LTE4V2X: LTE for a centralized VANET organization," in *Proceedings of the 54th Annual IEEE Global Telecommunications Conference: "Energizing Global Communications", GLOBECOM 2011*, pp. 1–6, Houston, TX, USA, December 2011.

[6] A. Memedi, F. Hagenauer, F. Dressler, and C. Sommer, "Cluster-based transmit power control in heterogeneous vehicular networks," in *Proceedings of the IEEE Vehicular Networking Conference, VNC 2015*, pp. 60–63, December 2015.

[7] Y. Li, M. Zhao, and W. Wang, "Intermittently connected vehicle-to-vehicle networks: detection and analysis," in *Proceedings of the 54th Annual IEEE Global Telecommunications Conference (GLOBECOM '11)*, pp. 1–6, December 2011.

[8] L. Zhang, Y. Deng, W. Zhu, J. Zhou, and F. Wang, "Skewly replicating hot data to construct a power-efficient storage cluster," *Journal of Network and Computer Applications*, vol. 50, pp. 168–179, 2015.

[9] P. Dong, X. Du, J. Sun, and H. Zhang, "Energy-efficient cluster management in heterogeneous vehicular networks," in *Proceedings of the IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 644–649, San Francisco, Calif, USA, April 2016.

[10] S. L. Hakimi, "Optimum distribution of switching centers in a communication network and some related graph theoretic problems," *Operations Research*, vol. 13, pp. 462–475, 1965.

[11] Q. Luo, C. Li, Q. Ye, T. H. Luan, L. Zhu, and X. Han, "CFT: A Cluster-based File Transfer Scheme for highway VANETs," in *Proceedings of the ICC 2017 - IEEE International Conference on Communications*, pp. 1–6, Paris, France, May 2017.

[12] A. A. Khan, M. Abolhasan, and W. Ni, "An Evolutionary Game Theoretic Approach for Stable and Optimized Clustering in VANETs," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 5, pp. 4501–4513, 2018.

[13] A. Mehmood, A. Khanan, A. H. H. M. Mohamed, S. Mahfooz, H. Song, and S. Abdullah, "ANTSC: An Intelligent Naïve Bayesian Probabilistic Estimation Practice for Traffic Flow to Form Stable Clustering in VANET," *IEEE Access*, vol. 6, pp. 4452–4461, 2017.

[14] T. Lindeberg, "Feature detection with automatic scale selection," *International Journal of Computer Vision*, vol. 30, no. 2, pp. 79–116, 1998.

[15] C. Sommer, D. Eckhoff, R. German, and F. Dressler, "A computationally inexpensive empirical model of IEEE 802.11p radio shadowing in urban environments," in *Proceedings of the 8th International Conference on Wireless On-Demand Network Systems and Services (WONS '11)*, pp. 84–90, January 2011.

[16] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, “Sumo–simulation of urban mobility: an overview,” in *Proceedings of the SIMUL 2011, The Third International Conference on Advances in System Simulation*, ThinkMind, 2011.

[17] (OSMF), O. F., “Openstreetmap.” http://www.openstreetmap.org/#map=14/38.9004/-77.0282.

[18] E. W. Dijkstra, “A note on two problems in connexion with graphs,” *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, 1959.

[19] G. V. Rossi, Z. Fan, W. H. Chin, and K. K. Leung, “Stable clustering for Ad-Hoc vehicle networking,” in *Proceedings of the 2017 IEEE Wireless Communications and Networking Conference, WCNC 2017*, pp. 1–6, March 2017.

[20] J. MacQueen, “Some methods for classification and analysis of multivariate observations,” in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, vol. 1, no 14, pp. 281–297, Oakland, CA, USA, 1967.

[21] E. Ben Hamida and M. A. Javed, “Channel-aware ECDSA signature verification of basic safety messages with K-means clustering in VANETs,” in *Proceedings of the 30th IEEE International Conference on Advanced Information Networking and Applications, AINA 2016*, pp. 603–610, March 2016.

[22] Q. Zhang, M. Almulla, Y. Ren, and A. Boukerche, “An efficient certificate revocation validation scheme with k-means clustering for vehicular ad hoc networks,” in *Proceedings of the 2012 IEEE Symposium on Computers and Communications (ISCC)*, pp. 000862–000867, Cappadocia, Turkey, July 2012.

[23] R. Chai, X. Ge, and Q. Chen, “Adaptive K-Harmonic Means clustering algorithm for VANETs,” in *Proceedings of the 14th International Symposium on Communications and Information Technologies, ISCIT 2014*, pp. 233–237, IEEE, September 2014.

[24] Z. Y. Rawashdeh and S. M. Mahmud, “A novel algorithm to form stable clusters in vehicular ad hoc networks on highways,” *EURASIP Journal on Wireless Communications and Networking*, vol. 2012, no. 1, p. 15, 2012.