Hindawi
Mobile Information Systems
Volume 2018, Article ID 9826782, 15 pages
https://doi.org/10.1155/2018/9826782

Research Article

Efficient and Reliable Cluster-Based Data Transmission for Vehicular Ad Hoc Networks

Xiang Ji [1,2] **Huiqun Yu** [1] **Guisheng Fan** [1] **Huaiying Sun,** [1] **and Liqiong Chen** [3]

[1] *Department of Computer Science and Engineering, East China University of Science and Technology, Shanghai 200237, China*
[2] *Shanghai Key Laboratory of Computer Software Evaluating and Testing, Shanghai 201112, China*
[3] *Department of Computer Science and Information Engineering, Shanghai Institute of Technology, Shanghai 201418, China*

Correspondence should be addressed to Huiqun Yu; yhq@ecust.edu.cn

Received 29 January 2018; Revised 7 April 2018; Accepted 29 April 2018; Published 30 July 2018

Academic Editor: Alessandro Bazzi

Vehicular ad hoc network (VANET) is an emerging technology for the future intelligent transportation systems (ITSs). The current researches are intensely focusing on the problems of routing protocol reliability and scalability across the urban VANETs. Vehicle clustering is testified to be a promising approach to improve routing reliability and scalability by grouping vehicles together to serve as the foundation for ITS applications. However, some prominent characteristics, like high mobility and uneven spatial distribution of vehicles, may affect the clustering performance. Therefore, how to establish and maintain stable clusters has become a challenging problem in VANETs. This paper proposes a link reliability-based clustering algorithm (LRCA) to provide efficient and reliable data transmission in VANETs. Before clustering, a novel link lifetime-based (LLT-based) neighbor sampling strategy is put forward to filter out the redundant unstable neighbors. The proposed clustering scheme mainly composes of three parts: cluster head selection, cluster formation, and cluster maintenance. Furthermore, we propose a routing protocol of LRCA to serve the infotainment applications in VANET. To make routing decisions appropriate, we nominate special nodes at intersections to evaluate the network condition by assigning weights to the road segments. Routes with the lowest weights are then selected as the optimal data forwarding paths. We evaluate clustering stability and routing performance of the proposed approach by comparing with some existing schemes. The extensive simulation results show that our approach outperforms in both cluster stability and data transmission.

1. Introduction

Vehicular ad hoc network (VANET) is the foundation for the intelligent transportation systems (ITSs), which aims at achieving seamless Internet connectivity between vehicles on the road [1]. With the developments of intelligent vehicle and new generation wireless communication techniques, vehicles equipped with wireless interfaces are able to provide ITS services [2] such as traffic monitoring, vehicle navigation [3], nearby information services, and mobile vehicular cloud computing.

Therefore, the creation of a stable network and communication management is the most challenging task due to the high mobility and uneven spatial distribution of vehicles in VANETs. The clustering technology is testified to be a promising solution to improve routing reliability and scalability by organizing similar vehicles into several virtual groups, called clusters [4]. Each cluster has a capital vehicle, named cluster head, which is responsible for managing communication in the cluster. Vehicles of a cluster can communicate directly via an intracluster communication, while vehicles in different clusters may achieve intercluster communication through cluster heads.

The originally notable clustering algorithms were designed for mobile ad hoc networks (MANETs) [4], such as the popular lowest identifier (LID) [5] and Mobility Clustering (MOBIC) [6]. Later, several other algorithms were designed for clustering in MANET. Recently, some of those algorithms were implemented in VANETs. However, due to the characteristic mobility and channel conditions of VANET, these approaches should be adapted according to the unique properties.

Clustering algorithms proposed for VANETs are used in communication networks to partition similar vehicles into clusters [7]. Therefore, clustering techniques can effectively limit the channel contention among cluster members to ensure fair channel access. Moreover, under the management of the cluster head, clustering algorithms can provide spatial reuse of resources such as bandwidth [8]. Given the high mobility of VANETs, how to select the cluster head and how to improve cluster stability become tough challenges.

Based on VANETs technology, numerous applications have been developed for the ITS. A typical kind of application is to disseminate safety messages among vehicles, including accident warning and congestion information [9, 10]. Another kind of application, infotainment, is also important for successful VANET deployment [11]. Infotainment services provide more pleasurable experience for both drivers and passengers with various applications, such as nearby information access and multimedia application [12].

To serve the infotainment services in urban VANET, this paper proposes a new LREL-based clustering scheme with the purpose of establishing a stable virtual network for data transmission. In order to form stable clusters, we propose an LLT-based neighbor sampling scheme to filter out unstable neighbor vehicles. Different from previous clustering schemes which focus on vehicular mobility, we propose new metric link reliability (LREL) for cluster head selection. The cluster heads are selected in a distributed way. In addition, we propose a routing protocol by using the proposed clustering architecture. We select bridge nodes at intersections to connect clusters in street scenarios. The bridge node acts as the routing path decision maker by monitoring the delay to incur for data transmission over road segments.

The contributions of this paper are mainly listed as follows:

(i) We propose an LLT-based neighbor sampling scheme to filter out the redundant unstable neighbors. A stable neighbor set is selected as a basis of clustering procedures.

(ii) We propose a new LREL-based clustering approach with the purpose of establishing a stable virtual network for efficient and reliable data transmission in urban VANET.

(iii) We propose a routing protocol for urban VANET by utilizing the structure constructed by LRCA. The protocol constructs routing path via the bridge node at intersection.

(iv) We analyze the performance of the proposed scheme by considering clustering performance metrics and routing metrics, including packet delivery ratio, end-to-end delay, and control overhead and clustering stability, by comparing with some existing cluster-based schemes.

The rest of the paper is organized as follows: Section 2 gives a brief description of related works. Section 3 describes the system model. Section 4 presents the proposed link reliability-based clustering scheme. Section 5 shows the data transmission utilizing the proposed clustering architecture. Next, Section 6 shows simulation results of the proposed scheme. Finally, conclusions are presented in Section 7.

2. Related Work

The original clustering algorithms are proposed in the late 1980s. Since then, a large amount of cluster-oriented researches have been introduced to MANETs in general and VANETs in particular [4]. Vehicle clustering is a potential approach to improve the scalability of networking protocols for VANET scenarios. For cluster-based routing protocols, cluster heads take responsibilities for the discovery and maintenance of routing paths, which reduce the control overhead to a great extent [13]. Due to the high-speed mobility of vehicles, network topology changes frequently [14]. Under this circumstance, the cluster maintenance cost increases significantly. Therefore, how to form the stable clusters and maintain their stability during communication are a vital issue in clustering techniques for VANETs.

Many clustering algorithms designed for VANETs have been proposed based on mobility metrics for cluster formation mechanisms. The mobility features, including speed, direction, and location of vehicles, are very important for VANET clustering procedures. Kayis and Acarman [15] proposed a passive clustering algorithm based on predefined speed intervals. They organize the vehicles within the same speed interval into groups. However, the speed interval is not a good metric for assessment because two vehicles with very similar speed around the interval gap might be divided into different clusters. Chen et al. [16] used the distance-based criteria in the cluster construction algorithms. Furthermore, they employ a central server to manage the cluster merging and splitting events. Shea et al. [17] proposed a distributed mobility-based clustering algorithm based on a data clustering technique called affinity propagation. They use the metric of vehicular position and mobility in cluster creation procedure by combining the current and future positions.

Some other clustering mechanisms have been proposed for VANET based on a sum of weighted values. Wang et al. [18] proposed a priority-based clustering approach. The priority is calculated according to the estimated travel duration and speed deviation. Almalag Mohammad et al. [19] presented a lane-based clustering algorithm which selects the vehicle as the cluster head with the highest cluster head level (CHL). CHL is a hybrid metric combining the condition of traffic flow, relative speed, and relative position of the vehicle. Morales et al. [20] proposed the clustering algorithm based on the destination of vehicles. According to their mechanism, vehicles with similar destinations are more likely to form a cluster. The weighted metric is computed as the combination of the current position, relative speed, relative destination, and final destination of vehicles.

Clustering algorithms for VANET can be categorized into two classes of one-hop clustering and multihop clustering. The aforementioned algorithms [16, 17, 19, 20] are based on single-hop clusters in which the cluster members are one-hop away from the CH. Vehicles only join the clusters where the CH is in its local vicinities. One-hop cluster topology can reduce the cluster re-affiliation and decrease cluster maintenance overhead because fewer information exchanges are required [21]. However, the transmission range and density of vehicles affect the size of cluster. In a high vehicular density, data collision may happen in the clusters. On the contrary, a vehicle may fail to detect neighbors in very low density. Recently, plenty of works have been proposed for multihop clustering algorithms. Wolny [22] proposed MDMAC, which is a modification of DMAC. MDMAC is able to form k-hop clusters by introducing the TTL (time-to-live) parameter in message delivery. Zhang et al. [23] proposed a multihop clustering scheme for VANETs. The multihop clusters are constructed based on the relative mobility between vehicles in multihop distance. Ucar et al. [9] proposed a novel multihop clustering scheme, known as VMaSC, which selects CHs based on relative mobility with respect to its neighbors. VMaSC reduces overhead during cluster formation by introducing a direct connection to the neighbor which is already a cluster head or a cluster member, instead of connecting to the CH multihops away. Further, the VMaSC claims to be the first multihop clustering algorithm which is analyzed under a realistic scenario. Ziagham and Noorimehr [24] proposed a single-hop clustering approach named MOSIC based on the changes of relative vehicular mobility. It uses the Gauss–Markov mobility (GMM) model for mobility predication and makes vehicle be able to prognosticate its mobility relative to its neighbors.

In recent years, some researchers build up semiclusters for VANET scenarios. Zhang et al. [25] proposed a novel variant of cluster, which called the microtopology (MT). The MT acts as a basic component of routing paths which consists of vehicles and wireless links among vehicles along the street. Togou et al. [26] proposed SCRP, which is an approximate cluster-based routing protocol based on connected dominating set (CDS). SCRP selects a small number of vehicles as dominating vehicles to form a virtual backbone in the network. Lin et al. [27] designed a moving-zone-based architecture for data delivery in VANETs. Similar to cluster formation, the moving zone is self-organized by vehicles which have similar movement patterns. Rivoirard et al. [28] proposed the chain-branch-leaf (CBL) clustering scheme which combines the information on road configuration, vehicle mobility, and link quality. CBL builds a stable vehicular network infrastructure by selecting vehicles with lower speed in the same traffic direction to form a stable backbone of branch nodes named Chain.

Data transmission over vehicular networks poses a number of challenges and has been widely studied. He et al. [29] proposed a SDN-based wireless communication solution to manage the network resources, which can schedule different network resources and minimize communication cost. Zeng et al. [30] proposed a channel prediction-based scheduling strategy for cooperative data dissemination in VANET, which reduces communication overhead and the data dissemination delay. Zhu et al. [31] proposed a distributed data replication algorithm with the idea of letting the data carrier distribute the data dissemination tasks to multiple nodes to speed up the dissemination process.

In the literature, clustering algorithms have been proposed for the purpose of load balancing, quality-of-service support, and data transmission in VANET scenario [2]. The built up clusters can serve as a hierarchical infrastructure-like overlay on top of an underlying ad hoc network, which can be used to route packets [4]. There have been some routing-oriented clustering algorithms which include both clustering and routing algorithms. For example, Song et al. [32] proposed a cluster-based directional routing protocol for VANETs which considers moving directions for cluster head selection. Ohta et al. [33] formed the clusters using position and direction information of vehicles. Unlike these researches, we consider the reliability of links between vehicles. We put up a new metric, named LREL, for cluster head selection. In addition, we propose a LLT-based neighbor sampling scheme to filter out unstable neighbors, which can reduce unnecessary message exchanges.

3. System Model

The prominent characteristics of VANET, including the high mobility and the uneven spatial distribution of vehicles, lead to frequent changes in the topologies and disconnections of the network. To solve these problems, we propose a clustered VANET structure to provide reliable connectivity for a group of vehicles. Figure 1 shows the general system architecture for vehicular service scenarios where vehicles are grouped into multiple moving clusters. Each cluster contains a capital vehicle (cluster head) which is responsible for managing information about the cluster members as well as data transmission.

In this paper, we propose a comprehensive cluster-based data transmission approach with pure vehicle-to-vehicle (V2V) communication type being considered. We assume that each vehicle has a unique identity and is equipped with an onboard unit (OBU). The GPS service is available for obtaining basic information, including vehicle's current location, velocity, and moving direction. Vehicles exchange their information with one another through beacon messages. The beacon message is broadcasted and collected at every beacon interval, which includes vehicle's identifier, current position, current velocity, moving direction, vehicle's current state, and cluster head's identifier if it is a cluster member.

In the proposed cluster algorithm, each vehicle may be in one of the following four states:

(i) *Initial node (IN)*: Initial state of the vehicles which do not belong to any cluster.

(ii) *Cluster head (CH)*: The state in which the vehicle acts as the leader of a cluster.

(iii) *Cluster member (CM)*: The state in which the vehicle is attached to an existing CH.

(iv) *Candidate cluster member (CCM)*: The state in which the vehicle intends to be a CM of an existing cluster before receiving a confirmation message.

The transitions among these states are triggered by different events. The details of the state transition process are presented in the following section. The notation used in this paper is presented in Table 1.

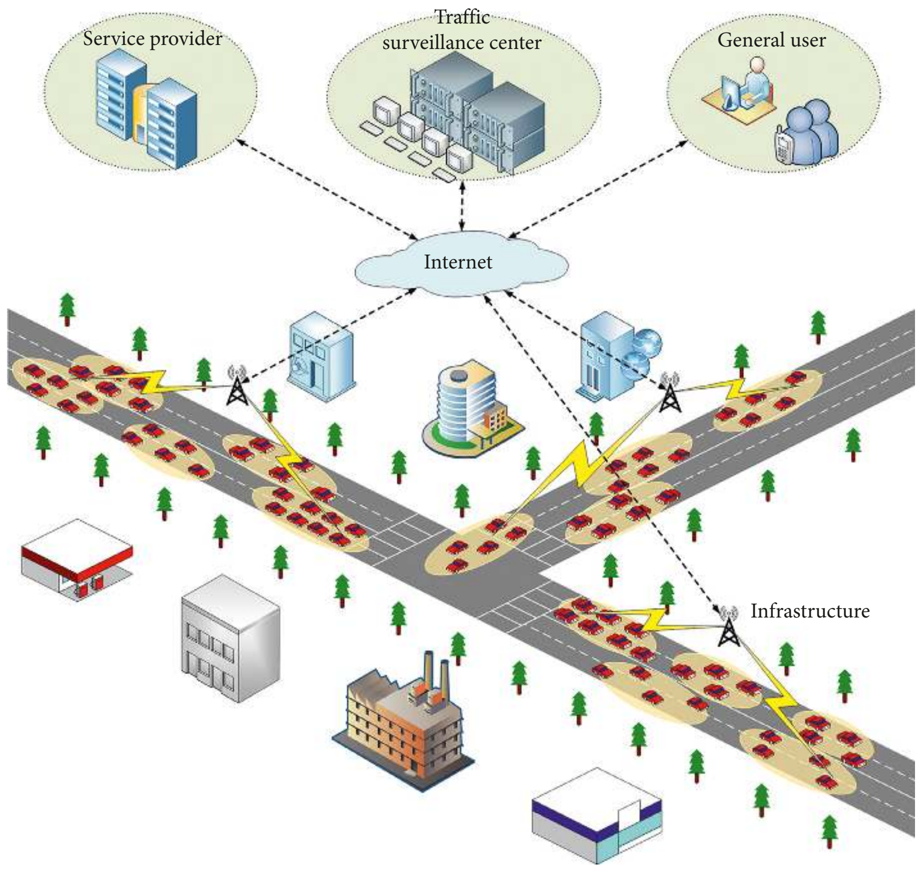

Figure 1: Cluster-based VANET architecture for vehicular service scenarios.

4. Vehicle Clustering

This paper proposes a link reliability-based clustering algorithm (LRCA) for urban VANETs. The LRCA is designed to provide efficient and reliable data transmission across the urban VANETs. Before clustering, we propose a novel LLT-based neighbor sampling strategy to filter out the redundant unstable neighbors. The proposed clustering scheme mainly composes of three parts: cluster head selection, cluster formation, and cluster maintenance. The general procedural flow of our proposed clustering algorithm is presented in Figure 2.

4.1. LLT-Based Neighbor Sampling. In VANET, vehicles exchange and collect information about their one-hop vicinities through periodic beacon messages. After receiving the beacons, each vehicle constructs the potential neighbor (PN) set [34]. However, not all the vehicles in PN are ideal for clustering. In order to improve the stability of clusters, this paper introduces an LLT-based neighbor sampling process to select a stable neighbor (SN) set from PN.

Link lifetime (LLT) [35], also called link expiration time (LET), represents the predicted duration time that two adjacent vehicles remain connected. Equation (1) defines LLT calculation. Δv_{ij} and Δd_{ij} represent the difference of velocity and distance between V_i and V_j, respectively. R denotes the transmission range of a vehicle. The LLT is introduced to evaluate the link sustainability. The link is more sustainable with a larger LLT value.

$$\mathrm{LLT}_{ij} = \frac{\left|\Delta v_{ij}\right| \cdot R - \Delta v_{ij} \cdot \Delta d_{ij}}{\left(\Delta v_{ij}\right)^2}. \tag{1}$$

The detail of the sampling process is presented in Algorithm 1. At the beginning, the system starts a timer (IN_TIMER) for gathering information. The vehicle in SN is defined as the neighboring vehicle which is going to have a constant connection for a predetermined time threshold δ_s. For each vehicle V_i, it maintains a set of stable neighbors SN_i, which contains several entries $\mathrm{SN}_i(j)$ of vehicle V_j.

4.2. Link Reliability Metric. The link reliability model for communication links between two vehicles in urban VANET is given in [36]. The network connectivity status is mainly determined by the velocity distribution over the vehicular traffic flow. The basic link reliability model is defined as the conditional probability in the following equation which describes the probability of the continuous link connectivity between two vehicles over a specified time duration:

$$r(l) = P\{l \text{ continues to } t + \mathrm{LLT} \mid l \text{ is available at } t\}, \tag{2}$$

where $r(l)$ represents the reliability of the link and l is the particular link between two vehicles. LLT indicates the

Table 1: Notations.

Notation	Description
IN	Initial node
CH	Cluster head
CM	Cluster member
CCM	Candidate cluster member
R	Communication range
V_i	Vehicle i; i is the ID of vehicle
LLT_{ij}	Link lifetime between V_i and V_j
LREL_i	Link reliability metric of V_i
PN_i	Potential neighbor set of V_i
SN_i	Stable neighbor set of V_i
$\mathrm{SN}_i(j)$	Entry of V_j in SN_i
$\mathrm{State}(V_i)$	State of V_i
IN_TIMER	Initial timer for gathering information
JOIN_TIMER	Join response timer
CH_TIMER	Cluster head selection timer
CH_ACK	Cluster head acknowledgement message
JOIN_REQ	Join request message
JOIN_RESP	Join response message
MERGE_REQ	Cluster merging request message
MERGE_RESP	Cluster merging response message
MERGE_ACK	Cluster merging acknowledgment message
$\mathrm{CM\_LIST}_i$	Cluster member list in V_i
$\mathrm{NUM\_CM}(V_i)$	Number of vehicles in $\mathrm{CM\_LIST}_i$
MAX_CM	Maximum number of vehicles a CH can serve
N_{pm}	Potential merged cluster size
δ_{s}	Time threshold for neighbor sampling
δ_{m}	Time threshold for cluster merging
δ_{i}	Time threshold for isolated CH

```
Input: PN_i;
Output: SN_i;
while IN_TIMER > 0 do
    for each vehicle V_j ∈ PN_i do
        if V_i receives a Beacon message from V_j then
            V_i calculates LLT_ij;
        end if
        if LLT_ij > δ_s then
            if V_j ∈ SN_i then
                V_i updates SN_i(j)
            else
                V_i adds a new SN_i(j) to SN_i
            end if
        end if
    end for
end while
```

Algorithm 1: LLT-based neighbor sampling.

prediction interval of link duration from time t. Equation (2) shows that if the link l is available at time t, the link will be available until $t + \mathrm{LLT}$.

The velocity of vehicles is the main parameter to calculate the link reliability. In this paper, we assume that the velocity satisfies a *normal distribution* [37]. Then the probability density function of the velocity $g(v)$ can be calculated as

$$g(v) = \frac{1}{\sigma\sqrt{2\pi}} e^{-((v-\mu)^2/2\sigma^2)}, \quad (3)$$

where μ and σ denote the mean and standard variation, respectively. Let Δv_{ij} be the relative velocity between vehicles V_i and V_j, that is, $\Delta v_{ij} = |v_i - v_j|$. Since v_i and v_j satisfy the normal distribution, Δv_{ij} should also obey the law of normal distribution. Let $f(T)$ be the probability density function of the communication duration T. $f(T)$ can be calculated as

$$f(T) = \frac{4R}{\sigma_{\Delta v_{ij}}\sqrt{2\pi}} \frac{1}{T^2} e^{\left(\left(2R/T - \mu_{\Delta v_{ij}}\right)^2 / \left(2\sigma^2_{\Delta v_{ij}}\right)\right)}, \quad \text{for } T \geq 0, \quad (4)$$

where $\mu_{\Delta v_{ij}}$ and $\sigma_{\Delta v_{ij}}$ denote the mean and the standard variation of relative velocity between v_i and v_j, respectively. Then, we can obtain the link reliability value $r(l_{ij})$ by integrating $f(T)$ in (4) from time t to $t + T$, as shown in the following equation:

$$r_t(l_{ij}) = \begin{cases} \int_t^{t+\mathrm{LLT}} f(T)\,dT, & \text{if LLT} > 0, \\ 0, & \text{otherwise.} \end{cases} \quad (5)$$

By using the Gauss error function Erf [38], the integral in (5) can be obtained as follows:

$$r_t(l_{ij}) = \mathrm{Erf}\left(\frac{((2R)/t) - \mu_{\Delta v_{ij}}}{\sigma_{\Delta v_{ij}}\sqrt{2}}\right) - \mathrm{Erf}\left(\frac{((2R)/(t+\mathrm{LLT})) - \mu_{\Delta v_{ij}}}{\sigma_{\Delta v_{ij}}\sqrt{2}}\right), \quad \text{when LLT} > 0. \quad (6)$$

The Erf function is calculated as follows:

$$\mathrm{Erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-\eta^2} d\eta, \quad -\infty < x < +\infty. \quad (7)$$

For a particular vehicle, it may have connections with multiple vehicles in the surrounding zone. The network connectivity could be diverse due to the complex traffic conditions. Therefore, we only consider those vehicles in the SN while defining the metric LREL:

$$\mathrm{LREL}_i(t) = \sum_{v_j \in \mathrm{SN}_i} r_t(l_{ij}). \quad (8)$$

4.3. Cluster Head Selection. This section describes the method whereby a CH is selected. We assess the fitness of a vehicle to act as a cluster head based on link reliability. The calculation of LREL has been presented in the previous section.

As shown in Algorithm 2, a vehicle V_i in the IN state firstly tries to join an existing cluster by listening to the CH_ACK message or beacon message from a CH during the time period CH_TIMER. If V_i fails to join an existing cluster when CH_TIMER expires, V_i calculates a weighted metric,

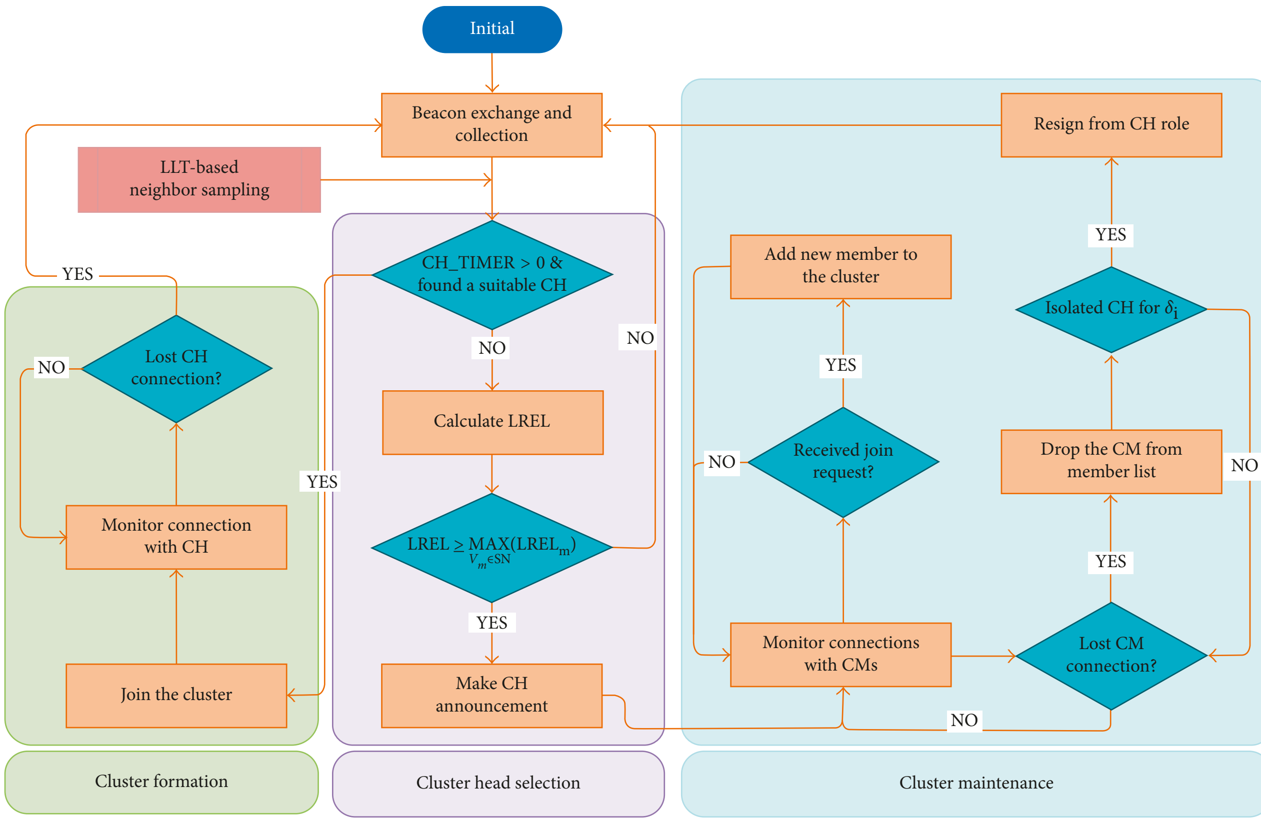

FIGURE 2: The general procedural flow of LRCA.

```
Input: Set of IN
Output: Set of CCM, CH
for each vehicle V_i where State(V_i) = IN do
    V_i starts CH_TIMER
    while CH_TIMER > 0 do
        if V_i receives CH_ACK or Beacon from CH_j then
            State(V_i) ← CCM
            goto Cluster Formation
        end if
    end while
    if V_i does not receive CH_ACK then
        V_i calculates LREL_i
        if LREL_i ≥ MAX_{V_m ∈ SN_i}(LREL_m) then
            State(V_i) ← CH
            V_i broadcasts CH_ACK
        end if
    end if
end for
```

ALGORITHM 2: Cluster head selection.

LREL_i, according to (8). Then V_i compares LREL_i with the neighbors in SN_i. If it turns out that V_i has the highest value of LREL, V_i broadcasts CH_ACK information to claim itself to be a CH.

4.4. Cluster Formation. As shown in Algorithm 3, the vehicle in the IN state will join a cluster if it receives the CH_ACK message or beacon message from a CH. The vehicle first transfers its state to CCM, then sends a JOIN_REQ message to the corresponding CH, and starts a join response timer (JOIN_TIMER). On reception of the JOIN_RESP message, the vehicle changes the state from CCM to CM. If the vehicle does not receive any response message, the vehicle resets its state to IN. A special case is that a vehicle V_i in the IN state receives messages from multiple CHs. In this case, the vehicle selects CH_j which has the highest LLT_{ij}.

The vehicle which is in the CH state maintains a CM_LIST to store the information of CMs. When the CH receives a JOIN_REQ message from the surroundings, the CH first checks the total number of members in the CM_LIST. If the number of CMs is less than a maximum number of members allowed (MAX_CM), the CH generates a JOIN_RESP message and unicasts it to the vehicle from which the JOIN_REQ message is received. At the meantime, the CH builds an entry of the vehicle and adds it to the CM_LIST.

4.5. Cluster Maintenance. Because of the high mobility of vehicles in VANET, the role of vehicles may keep changing frequently, which brings extra maintenance overhead. In our proposed scheme, the CH resigns from CH role and transfers to the IN state when losing all of its CMs. Otherwise, it remains as CH until the cluster merging process happens. Therefore, we only consider cluster merging (Algorithm 4) and vehicle leaving events in the cluster maintenance procedure.

```
Input: Set of IN, CCM
Output: Set of CM
for each vehicle V_i where State(V_i) = IN do
    if V_i receives CH_ACK or Beacon from CH_j then
        State(V_i) = CCM
        V_i unicasts JOIN_REQ to V_j
        V_i starts JOIN_TIMER
        while JOIN_TIMER > 0 do
            if V_i receives JOIN_RESP then
                State(V_i) ← CM
            end if
        end while
        if V_i does not receive JOIN_RESP then
            State(V_i) ← IN
        end if
    end if
end for
for each vehicle V_j where State(V_j) = CH do
    if V_j receives JOIN_REQ from V_i then
        if NUM_CM(V_j) < MAX_CM then
            V_j adds V_i into CM_LIST_j
            V_j unicasts JOIN_RESP to V_i
        end if
    end if
end for
```

Algorithm 3: Cluster formation.

```
Input: Two subclusters
Output: The merged cluster
if LLT_ij ≥ δ_m then
    if LREL_i ≥ LREL_j then
        CH_j sends MERGE_REQ to CH_i
        if CH_i receives MERGE_REQ then
            CH_i estimates N_pm
            if N_pm ≤ MAX_CM then
                CH_i sends MERGE_RESP to CH_j
            end if
        end if
        if CH_j receives MERGE_RESP then
            CH_j broadcasts MERGE_ACK
            State(V_j)←CM
        end if
    end if
end if
```

Algorithm 4: Cluster merging.

4.5.1. Cluster Merging. As time passes, clusters moving on the road may overlap with one another. When two moving clusters get closer to one another, the overlapping area of these two clusters becomes larger. Heavily overlapped moving clusters introduce redundant in intracluster management and communication overhead. Under the circumstances, the two clusters have potential to be merged. Instead of starting the cluster merging procedure immediately, the merging procedure begins if the two CHs (CH_i and CH_j) detect that they will stay neighbors for a certain time threshold $\delta_m(LLT_{ij} \geq \delta_m)$. This is because if two clusters just passing by one another quickly, the overlap of the two clusters is temporary and would not affect the overall performance in the long run. Once the cluster merging process begins, the two CHs share their cluster information and the CH with lower LREL (CH_j) sends MERGE_REQ to the higher CH (CH_i).

Upon reception of MERGE_REQ, CH_i estimates the potential merged cluster size (N_{pm}). If $N_{pm} \leq$ MAX_CM, cluster merging is permitted and CH_i then sends MERGE_RESP to CH_j. If CH_j receives MERGE_RESP, CH_j gives up the leadership and broadcasts MERGE_ACK to inform its CMs about the merge operation. Otherwise, the CHs remain the role as CHs. On reception of MERGE_ACK, CMs which also in the SN_i of CH_i automatically become cluster members of CH_i. The remaining vehicles then search new clusters and join in.

4.5.2. Leaving a Cluster. During every beacon period, each CH monitors the connections with its CMs dynamically. If a CH does not receive the beacon message from its CM for at least two beacon interval, the CH is considered to loss the connection with the CM. Every time when a CH receives a beacon message from its CM, it updates the CMs related information in its CM_LIST. When the disconnection occurs, CH deletes the entry of the CM from the CM_LIST. The vehicle which lost the connection with CH then transfers its state to IN and tries to find a new cluster. When a CH losses all of its CMs to become an isolated CH for a certain time δ_i, the CH resigns from CH role and turns to IN.

5. Routing Protocol of LRCA

The goal of our proposed LRCA architecture is to delivery data packets to a specified destination in the vehicular networks. For example, if a vehicle is heading for a particular shopping district, this vehicle can obtain the parking information nearby or promotion information from merchants in advance by sending inquires to the vehicles around the shopping district.

In LRCA, clusters span over every road segment. In order to connect them in street scenarios, we propose a bridge node selection scheme to nominate special nodes at intersections. Moreover, these selected bridge nodes are responsible for assessing the network condition over each road segment connected to the intersection.

5.1. Bridge Node Selection at Intersections. For each intersection, the vehicle which will stay longer at the intersection zone is preferred to be selected as the bridge node. In this case, those vehicles stopped by the red light will be the ideal candidates. If there is only one vehicle, we nominate the vehicle as the bridge node directly; if there are more vehicles, we randomly select a vehicle as the bridge node for the sake of simplicity; if there is no vehicle stopping at the intersection, we will select the vehicle which is approaching the intersection center with the lowest velocity.

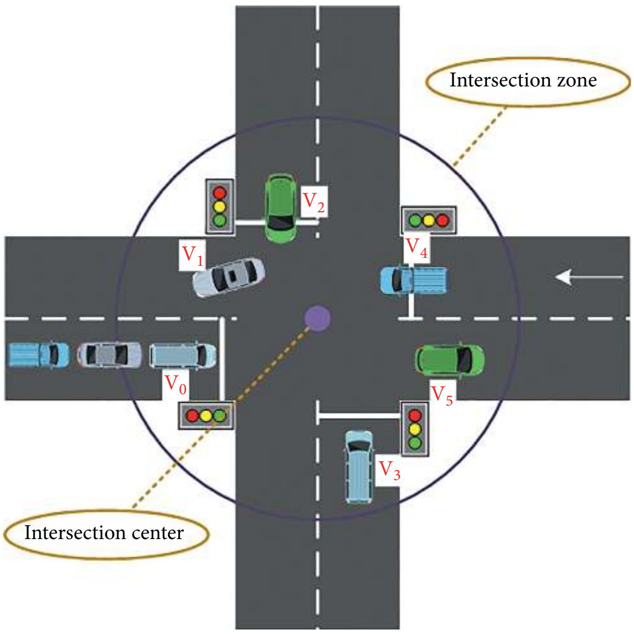

Figure 3: Bridge node selection at intersections.

For example, as shown in Figure 3, V_0 will be selected as the bridge node because it is stopping at the intersection. In other case, assume that the vehicle V_0 does not exist. The rest of the vehicles within the intersection zone are all candidates (i.e., $\zeta = \{V_1, V_2, V_3, V_4, V_5\}$). Among them, V_1 and V_5 are going past the intersection center while V_2, V_3, and V_4 are approaching the intersection center. We discard V_1 and V_5 from ζ because they are on the way leaving the intersection zone. Afterwards, the bridge node is selected from the remaining candidates $\zeta = \{V_2, V_3, V_4\}$. The vehicle in ζ which has the minimum velocity is then chosen as the bridge node.

When the bridge node is about to drive out of the intersection zone, a new bridge node will be selected to guarantee the connectivity in intersection scenarios.

5.2. Road Segment Evaluation. When calculating the routing path, we cannot simply adopt traditional shortest path algorithms (e.g., *Dijkstra's* algorithm) because when the packet carrier arrives at an intersection, it is not guaranteed that it can meet another vehicle moving towards the most preferred direction [39]. In order to evaluate the network condition of road segments, we put forward a distributed procedure named road segment evaluation (RSE) which is dynamically initiated by aforementioned bridge nodes.

RSE is triggered when a bridge node is selected at an intersection (I_i) by sending a light-weight control packet (P_{RSE}) to the adjacent intersection (I_j). Thereafter, P_{RSE} transverses the road segment (RS_{ij}) via relaying forwarders and gathers information regarding connectivity, delay, and hop count at each intermediate forwarder. When P_{RSE} reaches the target intersection, the bridge node at that junction calculates the delivery delay (d_{p}) as follows:

$$d_{\mathrm{p}} = t_{\mathrm{rc}} - T_{\mathrm{RSE}}, \tag{9}$$

where t_{rc} and T_{RSE} designate the received time and the generation time of P_{RSE}, respectively. d_{p} can indicate the network condition in the road because it experiences similar transmission and queuing delay in addition to interference and fading conditions in R_{ij}. Then, d_{p} is compared to a threshold $T_{M_{ij}}$ which is calculated as below:

$$T_{M_{ij}} = 2 \cdot t_{d_M} \cdot \left\lceil \frac{|\mathrm{RS}_{ij}|}{R} \right\rceil + t_{\mathrm{cf}}, \tag{10}$$

where t_{d_M} is a constant parameter representing the maximum acceptable delay per forwarder, including the transmission delay and queuing delay, $|\mathrm{RS}_{ij}|$ denotes the length of RS_{ij}, and t_{cf} is a constant parameter representing the maximum tolerable time using carry-and-forward. For a disconnected road segment, P_{RSE} is dropped when a forwarder detects that d_{p} is larger than $T_{M_{ij}}$. In case a bridge node has not received the P_{RSE} for a certain time, that is, $d_{\mathrm{p}} > T_{M_{ij}}$, the bridge node deduces that R_{ij} is disconnected. Afterwards, the bridge node assigns weight to RS_{ij}:

$$w_{ij} = \begin{cases} d_{\mathrm{p}}/T_{M_{ij}}, & d_{\mathrm{p}} \le T_{M_{ij}}, \\ \infty, & \text{otherwise}. \end{cases} \tag{11}$$

5.3. Route Construction. The protocol is designed for transmitting data with the lowest delivery delay and the highest stability in terms of connectivity. In particular, we assume that a vehicle generates a DATA_PACKET in the form of $< V_{\mathrm{src}}, M, L_{\mathrm{dest}} >$, where V_{src} denotes the identity of the sender vehicle, M is the message, and L_{dest} is the location of message destination.

As shown in Algorithm 5, the protocol consists of two phases, that is, inter- and intrasegment phases. The intersegment phase deals with the routing path decision at intersections, while the intrasegment phase focuses on packet forwarding within a road segment. When the DATA_PACKET is generated, V_{src} forwards it to the nearest bridge node.

For the intersegment phase, when the current forwarder arrives at an intersection I_i, it delivers DATA_PACKET to the bridge node at that intersection. Once received, the bridge node obtains the destination and calculates the routing metric as below:

$$M_{ij} = \frac{|\mathrm{SP}_j|}{|\mathrm{SP}_i|} \cdot w_{ij}, \tag{12}$$

where M_{ij} is combination of both geographic information and routing path delay. In (12), $|\mathrm{SP}|_j/|\mathrm{SP}|_i$ denotes the geographical process where SP_i represents the shortest path between current intersection and L_{dest}, and SP_j represents the shortest path between the next candidate intersection and L_{dest}. The road segment RS_{ij} with the minimum M_{ij} is considered to be the optimal routing path. Then, the bridge node delivers the DATA_PACKET to a vehicle on the selected road.

(1) **Initialization**: DATA_PACKET $\leq V_{\text{src}}$, M, L_{dest} >
(2) **if** the current forwarder V_{c} arrives at an intersection **then**
(3) V_{c} delivers DATA_PACKET to the bridge node V_{B_i}
(4) V_{B_i} calculates M_{ij} and selects the RS_{ij} with the minimum M_{ij}
(5) V_{B_i} delivers DATA_PACKET to a vehicle on RS_{ij}
(6) **else**
(7) **if** V_{c} is a CH **then**
(8) **if** L_{dest} inside the cluster range **then**
(9) Multicast DATA_PACKET to its CMs and **Done**
(10) **else**
(11) Unicast DATA_PACKET to a CM closest to the target intersection I_j
(12) **end if**
(13) **else**
(14) **if** DATA_PACKET is from its CH **then**
(15) Unicast DATA_PACKET to a CM or CH closest to I_j
(16) **else**
(17) Unicast DATA_PACKET to its CH
(18) **end if**
(19) **end if**
(20) **end if**

Algorithm 5: Routing protocol.

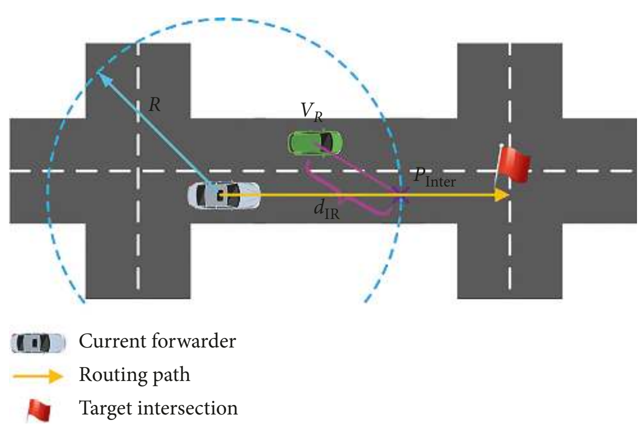

Figure 4: Computation of a good relay node.

For the intrasegment phase, if the current forwarder is a CH (Steps 6–11), it checks whether L_{dest} is inside its cluster range. If so, the CH will broadcast the DATA_PACKET to its CMs directly. If not, the CH will look for a good relay vehicle for the message propagation. The CH aims at finding a CM which is closest to the target intersection. The CH first computes the intersection point (P_{Inter}) of the route path and communication range as shown in Figure 4. Then, the CM which is closest to P_{Inter} and moves towards the target intersection is considered to be a good relay vehicle (V_R) for the message propagation. To find such a vehicle, the CH calculates the distance (d_{IR}) between P_{Inter} and its CMs. Afterwards, the CH selects the CM with minimum d_{IR} as the relay vehicle and then unicasts the DATA_PACKET to this CM.

In another case, if the current forwarder is a CM (Steps 12–17), it checks the source of the packet. If the packet does not come from its CH, the CM will unicast the packet to its CH directly. If the packet comes from its CH, the CM will be responsible for sending the packet to vehicles in nearby clusters. The CM first checks the neighbor list to find a neighbor CH which is moving towards the target intersection. If there are multiple such CHs, the CM selects the CH with the shortest distance to the target intersection. Then, the DATA_PACKET will be delivered to the selected CH. If the CM fails to find a CH in the neighbor list, the CM will select a one-hop neighbor which is closest to the next intersection as the next propagation vehicle.

6. Performance Evaluation

The proposed approach is compared with three previously proposed clustering-based schemes NHop [23], VMaSC [9], and MOSIC [24], and a nonclustering-based approach GeoSVR [40]. Among them, NHop and VMaSC are the two most cited multihop clustering algorithms and MOSIC is a latest single-hop clustering approach which has been introduced in Section 2. Since our proposed LRCA and the MOSIC are both single-hop clustering algorithms, the one-hop NHop and VMaSC are implemented in the simulation. GeoSVR is a high-cited nonclustering-based routing protocol which combines node location with the digital map. Meanwhile, it selects the routing path based on vehicular density to avoid local maximum and sparse connectivity.

The simulations are implemented in the Network Simulator NS-2 (v-2.35) [41] with the mobility of vehicles generated by SUMO [42]. As shown in Figure 5, the simulation scenario is a 5100m × 4800m ordinary urban environment which is extracted from the OpenStreetMap [43] of Shanghai China. The number of simulated vehicles is set to 1500, and we run the simulation for 100 seconds to let all the injected vehicles move around the map for a while. After 100 seconds, the simulation runs for another 500 seconds to evaluate the total performance metrics. In the simulation, we evaluate the metrics at the transmission ranges of 200 and 500 m. Meanwhile, we vary the maximum allowable velocity

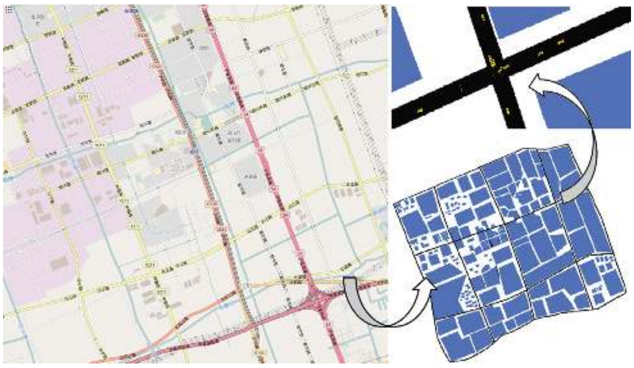

Figure 5: Simulation scenario of Shanghai China with SUMO.

of vehicles from 10 to 30 m/s and the maximum acceleration and deceleration are set to $5\,m^2/s$. The reported result is the average of 10 times repeated run. The details of simulation parameters and values are listed in Table 2.

Table 2: Simulation parameters.

Notation	Description
Simulation area	5100 m × 4800 m
Maximum velocity	10, 15, 20, 25, 30 m/s
Maximum acceleration	$5\,m/s^2$
Transmission range	200, 500 m
MAC protocol	IEEE 802.11p
Data rate	2 Mbps
MAX_CM	10
HELLO_PACKET period	200 ms
HELLO_PACKET size	64 bytes
DATA_PACKET generation rate	10, 20, 30, 40, 50 packet/s
DATA_PACKET size	1024 bytes
IN_TIMER	1 s
CH_TIMER	2 s
JOIN_TIMER	2 s
δ_s	2 s
δ_m	2 s
δ_i	1 s

6.1. Clustering Performance. To evaluate the performance of the proposed clustering algorithm, we focus on the stability of cluster, where cluster stability means lower changes in the CHs and lower changes in the CMs. Therefore, good clustering algorithm should be designed to minimize the rate of CH change and gain long term of CH duration, as well as cluster member duration.

Consequently, the following performance metrics are used for comparison:

(i) *Cluster head duration*: It is defined as the average time from a vehicle becoming a CH to transferring to another state.

(ii) *Cluster member duration*: It is defined as the average time from a vehicle joining a cluster to leaving the cluster.

(iii) *Cluster head change rate*: It is defined as $1-(|S_{CH}^{t}\cap S_{CH}^{t-1}|/|S_{CH}^{t}\cup S_{CH}^{t-1}|)$, where S_{CH}^{t} and S_{CH}^{t-1} represent the current CH set and previous CH set, respectively. $|s|$ denotes the number of elements in the corresponding set.

The results in Figures 6–8 evaluate the clustering stability for different maximum allowable velocities and transmission ranges by comparing the proposed LRCA with VMaSC, NHop, and MOSIC, from the aspects of the average CH lifetime, the average CM lifetime, and the CH change rate.

Figures 6 and 7 show the performance of CH and CM duration for different maximum allowable velocities and transmission ranges, respectively. Results show that the average CH and CM duration will decrease when the maximum allowable velocity of vehicles increases. This is because when the vehicles move faster, the vehicular network topology becomes more dynamic and eventually makes it difficult for vehicles to maintain a relatively stable condition with their neighbor vehicles for a long period. VMaSC-1hop acquires the longest CH lifetime and CM lifetime when the maximum velocity is 10 m/s. But both the CH duration and CM duration of VMaSC-1hop decrease

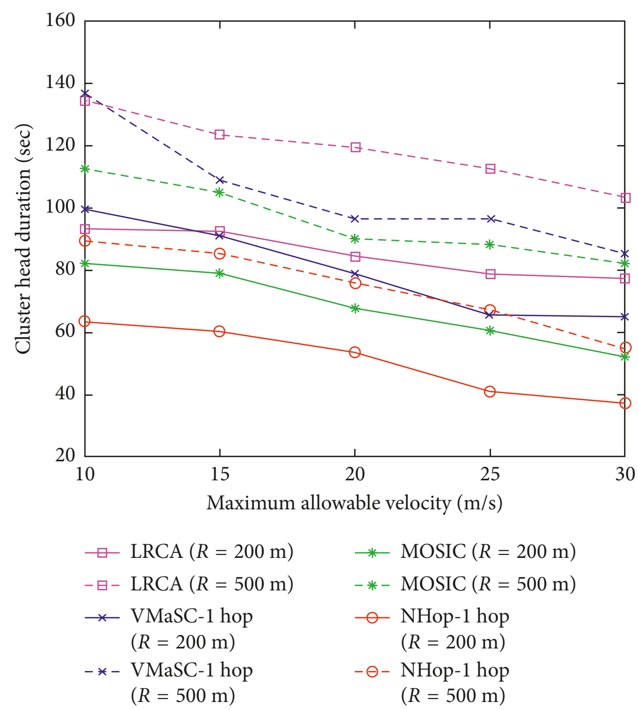

Figure 6: CH duration for different maximum allowable velocities and transmission ranges.

rapidly when the maximum allowable velocity increases. When the maximum velocity is bigger than 15 m/s, LRCA performs better against VMaSC (VMaSC-1hop), NHop (NHop-1hop), and MOSIC in terms of both CH and CM duration. This is because in our scheme, we construct stable clusters by employing LLT-based neighbor sampling scheme to pick out stable neighbors. In addition, the CH duration and CM duration are larger at a high-transmission range. The main reason is that the vehicles can communicate with more neighbor nodes and create higher correlation of connectivity behavior when the transmission range is higher. As the transmission range increases from 200 to 500 m, the

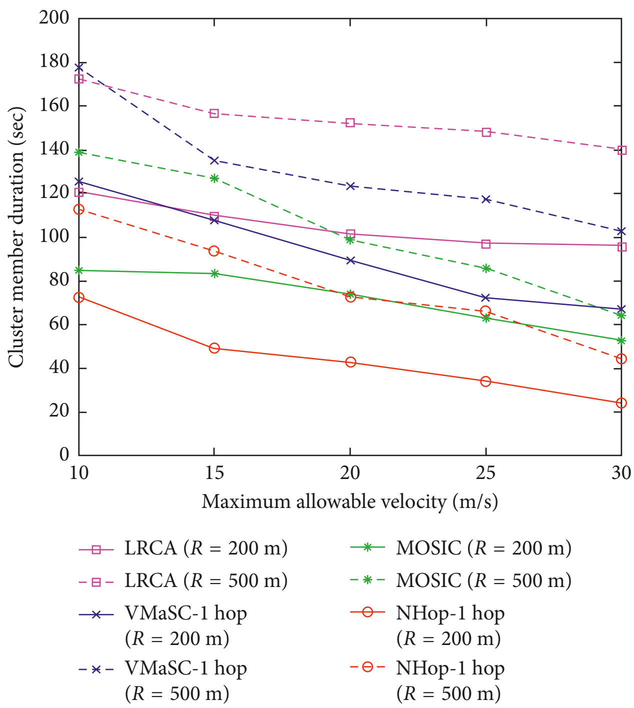

Figure 7: CM duration for different maximum allowable velocities and transmission ranges.

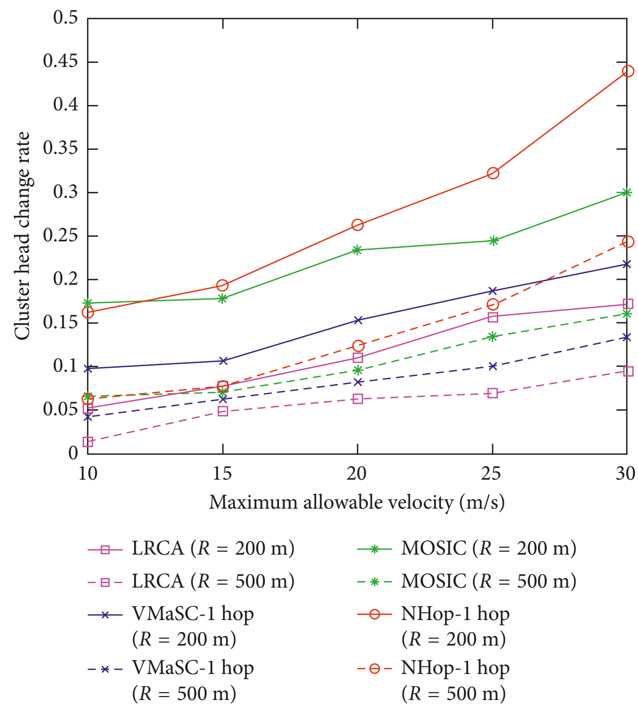

Figure 8: CH change rate for different maximum allowable velocities and transmission ranges.

link duration increases for all the simulated clustering algorithms.

Figure 8 shows the performance of the CH change rate for different maximum allowable velocities and transmission ranges. From the results, we can observe that the CH change rate increases with the increment of maximum allowable velocity and decreases as the transmission range increases. The reason is that the increment of maximum allowable velocity accelerates the change of network topology. Afterwards, some CMs may move out of the cluster or cluster merging may happen, which may result in cluster head changes. On the contrary, the higher transmission range provides much more connectivity of the vehicles within a cluster, which reduces the changes in the cluster head. The CH change rate for NHop-1hop is closer to MOSIC at both low- and high-transmission range when the maximum allowable velocity is between 10 and 15 m/s. However, the CH change rate for NHop-1hop increases rapidly as the velocity increases. Our proposed LRCA acquires the lowest CH change rate against VMaSC-1hop, NHop-1hop, and MOSIC in both cases of transmission range. We reduce the CH change rate by changing the CH state to other clustering state only when cluster merging happens or it losses all the CMs to become an isolated CH. In addition, to avoid unnecessary state transitions, we put up two time thresholds δ_m and δ_i before merging two clusters and before transferring an isolated CH to the IN state.

As shown above, mostly, the LRCA gains longer duration of both CH and CM and performs the lowest CH change rate at both low- and high-transmission range. Therefore, the proposed LRCA indicates the better performance of clustering stability than that of VMaSC (VMaSC-1hop), NHop (NHop-1hop), and MOSIC.

6.2. Routing Performance. The routing performance of our proposed LRCA is compared with three cluster-based routing mechanisms including NHop, VMaSC, and MOSIC, which have been discussed in the related work above, and a non-clustering approach called GeoSVR. We aim at achieving efficient and reliable data delivery with high packet delivery ratio and low network latency.

From the simulation results above, we can learn that clusters acquire more stability at a high-transmission range. Hence, we set the transmission range to 500 m when evaluating the routing performance here. To evaluate the effect of different parameters on the routing performance, we vary the parameters of the data packet generation rate and maximum allowable velocity in the simulation. The default value of the maximum allowable velocity and the data packet generation rate are set to 20 m/s and 30 packet/s, respectively.

Furthermore, we evaluate the routing performance using the following performance metrics:

(i) *Packet delivery ratio (PDR)*: This metric is defined as the ratio of the average number of data packets successfully received by destinations, compared to the total number of generated packets.

(ii) *End-to-end delay (E2ED)*: This metric represents the average delay between the time a packet generated by the source and the time of this packet reached to the destination.

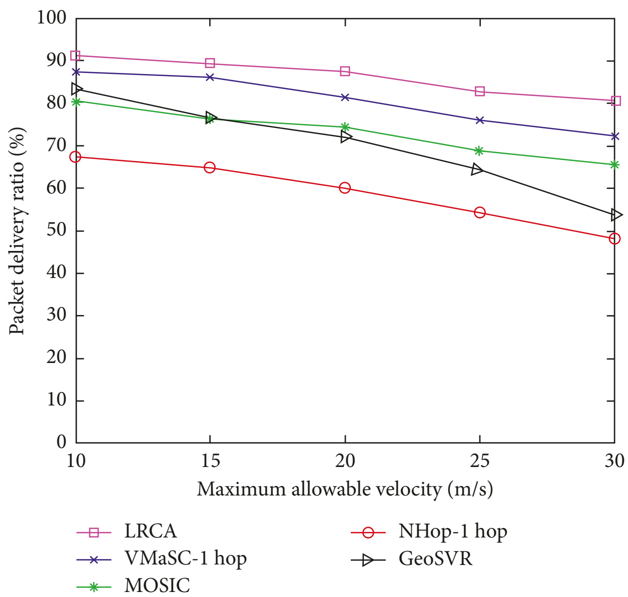

Figure 9: Packet delivery ratio for varying maximum allowable velocity.

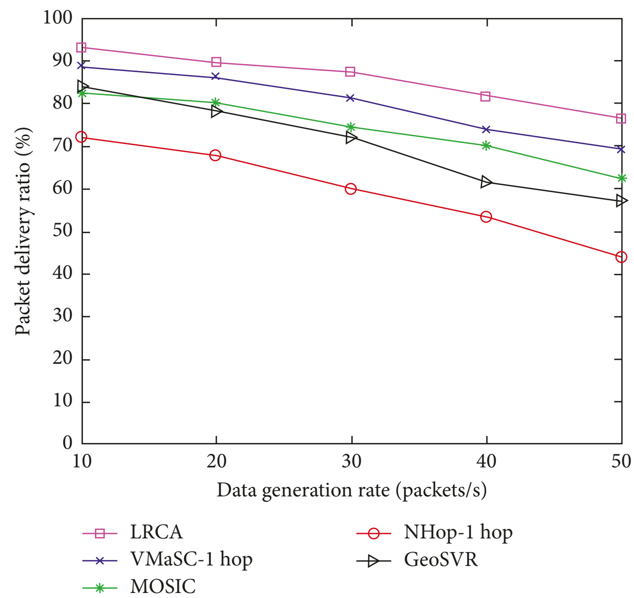

Figure 10: Packet delivery ratio for varying data generation rate.

(iii) *Normalized routing overhead (NRO)*: This metric is calculated as the ratio of the size of total generated packets to the size of the data packets successfully received by the destinations.

Figure 9 shows the performance of the packet delivery ratio for varying maximum allowable velocity. It is observed that LRCA achieves higher packet delivery ratio than other schemes. By selecting stable neighbors according to link lifetime (LLT), LRCA builds up a stable clustered virtual network which can provide the stable connections between cluster members, and meanwhile increase the bandwidth availability and reduce data collision. Moreover, the results show that the packet delivery ratio of all protocols decreases when lifting upper limitation of allowable velocity. Because the maximum allowable velocity increases, the network topology changes rapidly, leading to an increment of packet loss ratio. The packet delivery ratio of GeoSVR is very close to that of MOSIC in the low-speed scenario. However, the packet delivery ratio of GeoSVR decreases significantly when the maximum allowable velocity is greater than 20 m/s, which indicates that the clustering-based routing approaches achieve more stability than the nonclustering-based ones, especially in the high-dynamic scenario.

Figure 10 shows the performance of the packet delivery ratio for varying data generation rate. Obviously, the packet delivery ratio of all the simulated protocols decreases when the data generation rate increases. This is because the vehicles need to store and carry the data packets when encountering network partitions. However, the size of packet buffer is limited, leading to subsequent packets being dropped when the buffer is full. Results show that our proposed LRCA achieves better performance of the packet delivery ratio than the other protocols. In LRCA, the bridge node selects the optimal routing path for data delivering considering the latency of each road segment, which decreases the cases of packet carrying by

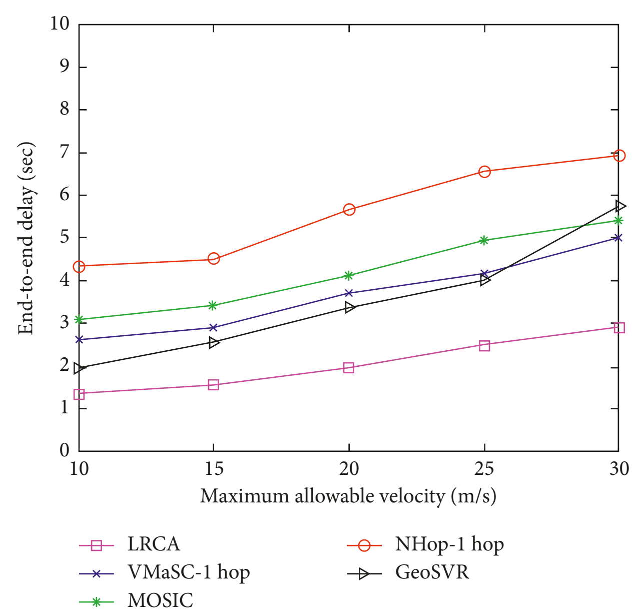

Figure 11: End-to-end delay for varying maximum allowable velocity.

setting a time threshold parameter for the carry-and-forward mode as shown in (10).

Figure 11 shows the performance of average end-to-end delay for varying maximum allowable velocity. The increment of velocity results in frequent changes of network topology. Thus, the results show that the end-to-end delay increases in high-speed scenario. The proposed scheme achieves significant reduction of end-to-end delay in comparison with other schemes. This is because LRCA builds up stable clusters which can guarantee the sufficient connectivity and reliable linking. Therefore, the retransmission times and transmission delay are reduced, which results in reduction of end-to-end delay. Another reason is that with the help of the stable connected

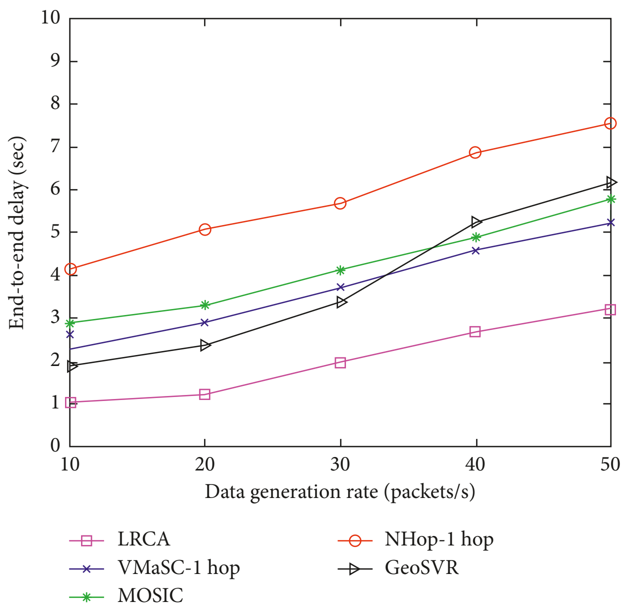

Figure 12: End-to-end delay for varying data generation rate.

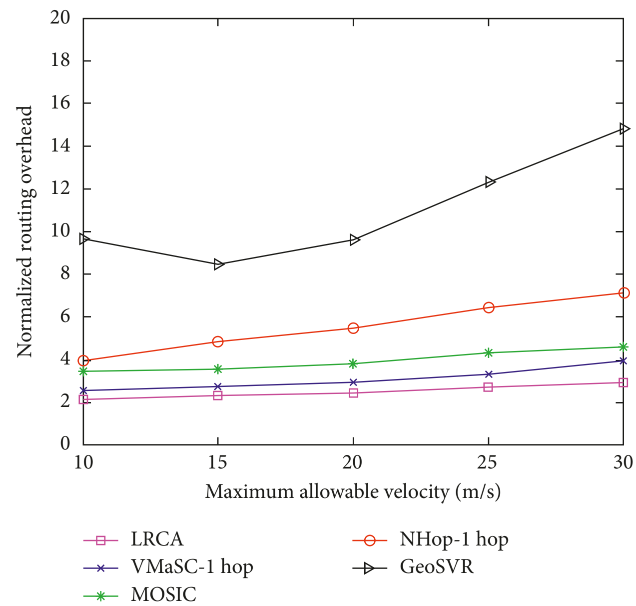

Figure 13: Normalized routing overhead for varying maximum allowable velocity.

clusters, packets can be delivered to the next hop with short MAC layer contention, which leads to short network latency. GeoSVR achieves lower delivery delay against VMaSC, MOSIC, and NHop when the maximum speed is less than 25 m/s. The GeoSVR obtains a better result by using the optimal forwarding path and the restricted forwarding algorithm. However, the end-to-end delay of GeoSVR increases significantly when the maximum velocity comes to 30 m/s. This shows that the GeoSVR may not be quite suitable for high-speed VANET scenario.

Figure 12 shows the performance of average end-to-end delay for varying data generation rate. When varying the data generation rate from 10 to 50 packet/s, the queuing delay in the buffer with relay nodes raises, which eventually affects the end-to-end delay. Consequently, the results show that the end-to-end delay of all the simulated protocols tends to increase with the increase of the data generation rate. In LRCA, the routing decision is made based on the estimated delay information over each road segment, which includes both delays due to packet carrying and packet relaying on links. LRCA acquires the lowest delivery delay by selecting the route paths with the lowest delay and highest geographical process as shown in (12). GeoSVR selects road segments with high vehicular density as routing paths to reduce the probability of store-and-carry events. That is why GeoSVR achieves lower end-to-end delay than VMaSC, MOSIC, and NHop when the data generation rate is less than 30 packet/s. But the GeoSVR suffers from data congestion when the data generation rate is high due to the strategy forwarding data over dense roads. Therefore, the end-to-end delay is much higher when the data generation rate is high.

Figure 13 shows the performance of normalized routing overhead for varying maximum allowable velocity. The proposed scheme shows the lowest overhead in comparison with others. Besides, the overhead of clustering schemes is much lower than that of the nonclustering scheme GeoSVR. This is

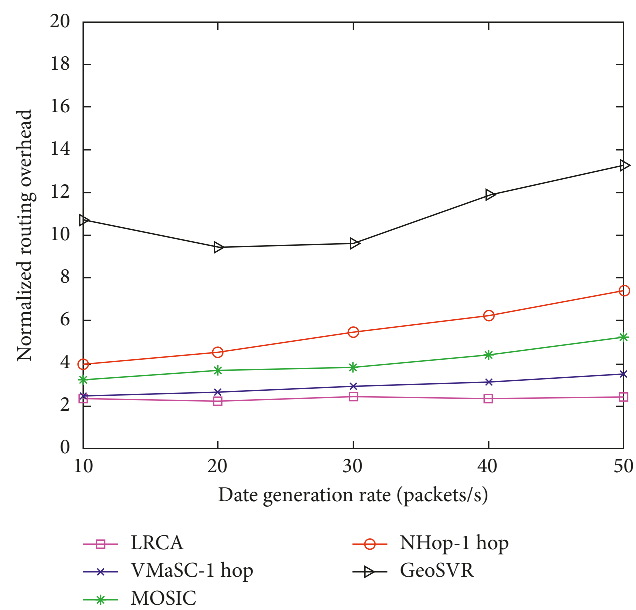

Figure 14: Normalized routing overhead for varying data generation rate.

because the clustering approaches can reduce intervehicle communications. The experiment results also show that the overhead of NHop-1hop increases when the vehicles move faster, while the overhead of LRCA does not significantly increase when the maximum velocity increases. This is benefit from the neighbor sampling strategy to filter out unstable neighbors and reduce unnecessary message exchanges. Moreover, the LRCA provides the stable connections for data delivery which can reduce the number of packet retransmissions.

Figure 14 shows the performance of normalized routing overhead for varying data generation rate. Results show that

LRCA achieves the lowest overhead and the clustering-based protocols acquire much lower overhead than the non-clustering scheme GeoSVR. The overhead of our proposed LRCA barely increases when raising the data generation rate, while other clustering methods more or less increase. LRCA reduces redundant message exchanges by selecting stable neighbor vehicles and reduces data packet retransmissions by selecting the route path with more connectivity.

7. Conclusion

In this paper, we propose a new link reliability-based clustering algorithm (LRCA) to provide efficient and reliable data transmission in urban VANET. In LRCA, vehicles are grouped with stable neighbor vehicles which are selected by the LLT-based neighbor sampling scheme. Further, we introduce a cluster-based routing protocol to provide the efficient and reliable data transmission in vehicular networks. In contrast to those protocols proposed for safety critical applications, the routing approach in this paper is designed to support infotainment services in VANET which are not stringent in delay constraints. To transmit data packets through stable paths, we introduce bridge nodes at intersections to make routing decisions. The bridge node evaluates the network condition over road segments and assigns weights to them. Then the road segment with minimum weight is selected to construct the overall routing path. The simulation results show that the proposed LRCA acquires better clustering stability in terms of long cluster head duration, long cluster member duration, and low rate of cluster head change. The proposed routing protocol performs better than the previous proposed schemes, which demonstrates the advantages of the proposed LRCA. In the future work, we will further improve the route strategy to minimize the end-to-end delay and satisfy the requirements of real-time applications in VANET.

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 there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was partially supported by the NSF of China under Grant nos. 61702334 and 61772200, Shanghai Pujiang Talent Program under Grant no. 17PJ1401900, Shanghai Municipal Natural Science Foundation under Grant nos. 17ZR1406900 and 17ZR1429700, Action Plan for Innovation on Science and Technology Projects of Shanghai under Grant no. 16511101000, Collaborative Innovation Foundation of Shanghai Institute of Technology under Grant no. XTCX2016-20, and Educational Research Fund of ECUST under Grant no. ZH1726108.

References

[1] B. T. Sharef, R. A. Alsaqour, and M. Ismail, "Vehicular communication ad hoc routing protocols: a survey," *Journal of Network & Computer Applications*, vol. 40, no. 1, pp. 363–396, 2014.

[2] R. S. Bali, N. Kumar, and J. J. P. C. Rodrigues, "Clustering in vehicular ad hoc networks: taxonomy, challenges and solutions," *Vehicular Communications*, vol. 1, no. 3, pp. 134–152, 2014.

[3] J. Jeong, H. Jeong, E. Lee, T. Oh, and D. H. C. Du, "SAINT: self-adaptive interactive navigation tool for cloud-based vehicular traffic optimization," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 6, pp. 4053–4067, 2016.

[4] C. Cooper, D. Franklin, M. Ros, F. Safaei, and M. Abolhasan, "A comparative survey of VANET clustering techniques," *IEEE Communications Surveys & Tutorials*, vol. 99, p. 1, 2017.

[5] C. R. Lin and M. Gerla, "Adaptive clustering for mobile wireless networks," *IEEE J Selected Areas in Communications*, vol. 15, no. 7, pp. 1265–1275, 1997.

[6] P. Basu, N. Khan, and T. D. C. Little, "A mobility based metric for clustering in mobile ad hoc networks," in *Proceedings of the International Conference on Distributed Computing Systems*, , Mesa, AZ, USA, April 2001.

[7] M. Fathian and A. R. Jafarian-Moghaddam, "New clustering algorithms for vehicular ad-hoc network in a highway communication environment," *Wireless Networks*, vol. 21, no. 8, pp. 2765–2780, 2015.

[8] G. V. Rossi, F. Zhong, W. H. Chin, and K. K. Leung, "Stable clustering for ad-hoc vehicle networking," in *Proceedings of the Wireless Communications and Networking Conference*, pp. 1–6, Edinburgh, UK, December 2017.

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

[10] B. Liu, D. Jia, J. Wang, K. Lu, and L. Wu, "Cloud-assisted safety message dissemination in VANET-cellular heterogeneous wireless network," *IEEE Systems Journal*, vol. 99, pp. 1–12, 2017.

[11] P. Salvo, M. D. Felice, F. Cuomo, and A. Baiocchi, "Infotainment traffic flow dissemination in an urban VANET," in *Proceedings of the Global Communications Conference*, pp. 67–72, Atlanta, GA, USA, December 2013.

[12] M. Oche, R. Md Noor, and A. Jalooli, "Quality of service management for IPTV services support in VANETs: a performance evaluation study," *Wireless Networks*, vol. 21, no. 1, pp. 315–328, 2015.

[13] K. Abboud and W. Zhuang, "Stochastic modeling of single-hop cluster stability in vehicular ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 1, pp. 226–240, 2016.

[14] X. Ji, H. Q. Yu, G. S. Fan, and W. H. Fu, "SDGR: an SDN-based geographic routing protocol for VANET," in *Proceedings of the IEEE International Conference on Internet of Things*, pp. 276–281, Exeter, UK, June 2017.

[15] O. Kayis and T. Acarman, "Clustering formation for inter-vehicle communication," in *Proceedings of the Intelligent Transportation Systems Conference (ITSC)*, pp. 636–641, Hong Kong, China, October 2007.

[16] J. Chen, C. Lai, X. Meng, J. Xu, and H. Hu, "Clustering moving objects in spatial networks," in *Proceedings of the International Conference on Database Systems for Advanced Applications*, pp. 611–623, Bangkok, Thailand, April 2007.

[17] C. Shea, B. Hassanabadi, and S. Valaee, "Mobility-based clustering in VANETs using affinity propagation," in *Proceedings of*

the Global Telecommunications Conference, pp. 1–6, FL, USA, December 2010.

[18] Z. Wang, L. Liu, M. C. Zhou, and N. Ansari, "A position-based clustering technique for ad hoc intervehicle communication," *IEEE Transactions on Systems Man & Cybernetics Part C*, vol. 38, no. 1, pp. 201–208, 2008.

[19] S. Almalag Mohammad and C. Weigle Michele, "Using traffic flow for cluster formation in vehicular ad-hoc networks," in *Proceedings of the Local Computer Networks*, pp. 631–636, Denver, CO, USA, October 2010.

[20] M. M. C. Morales, C. S. Hong, and Y. C. Bang, "An adaptable mobility-aware clustering algorithm in vehicular networks," in *Proceedings of the Network Operations and Management Symposium*, pp. 1–6, Maui, MI, USA, April 2011.

[21] M. Ren, L. Khoukhi, H. Labiod, J. Zhang, and V. Vèque, "A mobility-based scheme for dynamic clustering in vehicular ad-hoc networks (VANETs)," *Vehicular Communications*, vol. 9, pp. 233–241, 2016.

[22] G. Wolny, "Modified DMAC clustering algorithm for VANETs," in *Proceedings of the International Conference on Systems and Networks Communications*, pp. 268–273, Sliema, Malta, October 2008.

[23] Z. Zhang, A. Boukerche, and R. Pazzi, "A novel multi-hop clustering scheme for vehicular ad-hoc networks," in *Proceedings of the ACM International Workshop on Mobility Management & Wireless Access (MobiWac)*, pp. 19–26, Miami Beach, Fl, USA, October-November 2011.

[24] A. Ziagham and M. R. Noorimehr, "MOSIC: mobility-aware single-hop clustering scheme for vehicular ad hoc networks on highways," *International Journal of Advanced Computer Science & Applications*, vol. 7, no. 9, 2016.

[25] X. M. Zhang, K. H. Chen, X. L. Cao, and K. S. Dan, "A street-centric routing protocol based on microtopology in vehicular ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 7, pp. 5680–5694, 2016.

[26] M. A. Togou, A. Hafid, and L. Khoukhi, "SCRP: stable CDS-based routing protocol for urban vehicular ad hoc networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 5, pp. 1298–1307, 2016.

[27] D. Lin, J. Kang, A. Squicciarini, Y. Wu, S. Gurung, and O. Tonguz, "MoZo: a moving zone based routing protocol using pure V2V communication in VANETs," *IEEE Transactions on Mobile Computing*, vol. 16, no. 5, pp. 1357–1370, 2017.

[28] L. Rivoirard, M. Wahl, P. Sondi, M. Berbineau, and D. Gruyer, "Chain-branch-leaf: a clustering scheme for vehicular networks using only V2V communications," *Ad Hoc Networks*, vol. 68, pp. 70–84, 2018.

[29] Z. He, D. Zhang, and J. Liang, "Cost-efficient sensory data transmission in heterogeneous software-defined vehicular networks," *IEEE Sensors Journal*, vol. 16, no. 20, pp. 7342–7354, 2016.

[30] F. Zeng, R. Zhang, X. Cheng, and L. Yang, "Channel prediction based scheduling for data dissemination in VANETs," *IEEE Communications Letters*, vol. 21, no. 6, pp. 1409–1412, 2017.

[31] J. Zhu, C. Huang, X. Fan, S. Guo, and B. Fu, "EDDA: an efficient distributed data replication algorithm in VANETs," *Sensors*, vol. 18, no. 2, p. 547, 2018.

[32] T. Song, W. Xia, T. Song, and L. Shen, "A cluster-based directional routing protocol in VANET," in *Proceedings of the IEEE International Conference on Communication Technology*, pp. 1172–1175, Jinan, China, September 2011.

[33] Y. Ohta, T. Ohta, and Y. Kakuda, "An autonomous clustering-based data transfer scheme using positions and moving direction of vehicles for VANETs," in *Proceedings of the Wireless Communications and Networking Conference*, Paris, France, April 2012.

[34] M. Ren, J. Zhang, L. Khoukhi, H. Labiod, and V. Vèque, "A unified framework of clustering approach in vehicular ad hoc networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 99, pp. 1–14, 2017.

[35] S. S. Wang and Y. S. Lin, "PassCAR: a passive clustering aided routing protocol for vehicular ad hoc networks," *Computer Communications*, vol. 36, no. 2, pp. 170–179, 2013.

[36] M. H. Eiza and Q. Ni, "An evolving graph-based reliable routing scheme for VANETs," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 4, pp. 1493–1504, 2013.

[37] Z. Niu, W. Yao, Q. Ni, and Y. Song, "Link reliability model for vehicle ad hoc networks," in *Proceedings of the London Communication Symposium*, pp. 1–4, London, UK, 2006.

[38] L. C. Andrews, *Special Functions of Mathematics for Engineers*, McGraw-Hill, New York, NY, USA, 2nd edition, 1991.

[39] J. Jeong, S. Guo, Y. Gu, T. He, and D. H. C. Du, "Trajectory-based data forwarding for light-traffic vehicular ad hoc networks," *IEEE Transactions on Parallel & Distributed Systems*, vol. 22, no. 5, pp. 743–757, 2011.

[40] Y. Xiang, Z. Liu, R. Liu, W. Sun, and W. Wang, "GeoSVR: a map-based stateless VANET routing," *Ad Hoc Networks*, vol. 11, no. 7, pp. 2125–2135, 2013.

[41] *The Network Simulator: NS2*, 2015, http://www.isi.edu/nsnam/ns/.

[42] *SUMO: Simulation of Urban Mobility*, 2015, http://sumo.sourceforge.net.

[43] *OpenStreetMap*, 2017, http://www.openstreetmap.org/.