ARTICLE IN PRESS
VEHCOM:70

Vehicular Communications ••• (••••) •••–•••

Contents lists available at ScienceDirect

Vehicular Communications

www.elsevier.com/locate/vehcom

A mobility-based scheme for dynamic clustering in vehicular ad-hoc networks (VANETs)

Mengying Ren [a,*], Lyes Khoukhi [a], Houda Labiod [b], Jun Zhang [b], Véronique Vèque [c]

[a] ERA, Université de Technologie de Troyes, France
[b] INFRES, Télécom ParisTech, Université Paris-Saclay, France
[c] LSS, Université Paris-Sud, Université Paris-Saclay, France

ARTICLE INFO

Article history:
Received 1 June 2016
Received in revised form 24 November 2016
Accepted 21 December 2016
Available online xxxx

Keywords:
Vehicular ad hoc networks (VANETs)
Clustering algorithm
Wireless communication

ABSTRACT

Vehicle clustering is an efficient approach to improve the scalability of networking protocols in vehicular ad-hoc networks (VANETs). However, some characteristics, like highly dynamic topology and intermittent connections, may affect the performance of the clustering. Establishing and maintaining stable clusters is becoming one of big challenging issues in VANETs. Recent years' researches prove that mobility metric based clustering schemes show better performance in improving cluster stability. Mobility metrics, including moving direction, vehicle density, relative velocity and relative distance, etc., are more suitable for VANETs instead of the received radio strength (RSS) and identifier number metrics, which are applied for MANETs clustering. In this paper, a new dynamic mobility-based and stability-based clustering scheme is introduced for urban city scenario. The proposed scheme applies vehicle's moving direction, relative position and link lifetime estimation. We compared the performance of our scheme with Lowest-ID and the most recent and the most cited clustering algorithm VMaSC in terms of cluster head duration, cluster member duration, number of clusters, cluster head change rate and number of state changes. The extensive simulation results showed that our proposed scheme shows a better stability performance.

© 2017 Elsevier Inc. All rights reserved.

1. Introduction

Vehicular ad hoc networks (VANETs), are a vital part of Intelligent Transportation System (ITS), which aims to improve road safety and information transmission efficiency on the road. With the developments of automotive manufacturing, intelligent vehicle and wireless communication technologies, vehicles which are equipped with wireless interfaces can communicate with nearby vehicles directly through a V2V (Vehicle-to-Vehicle) communication mode, as well as with fixed equipment, called Road Side Units (RSUs), through a V2I (Vehicle-to-Infrastructure) or I2V (Infrastructure-to-Vehicle) communication manner [1].

These types of wireless communications enable vehicles to share different kinds of information, including safety related information and (non-safety related) infotainment information, corresponding respectively to road safety and non-safety applications. Safety applications mainly focus on avoiding accidents. They require low latency and high reliability, whereas non-safety applications aim to improve drivers and passengers comfort level and enhance traffic efficiency [2]. A detailed classification for road safety applications and their requirements is given in the standard of European Telecommunications Standard Institute (ETSI) [3].

VANETs have several characteristics that distinguish them from other multi-hop networks. Nodes in VANETs are highly mobile, leading to a high probability of network partitions, especially under highway scenarios. Therefore, the end-to-end communication cannot be guaranteed [4]. Intermittent connection may cause severe packet loss problem, and further influence traffic safety. Meanwhile, as a decentralized self-organizing network, VANETs is lack of a centralized management and coordination entity which is responsible for managing the bandwidth and contention operations. Moreover, VANETs is a large scale network; however, the communication range of a vehicle is limited which may also cause a weak connectivity between nodes. Therefore, maintaining a global network topology is indispensable for a node. For these reasons, a flat network topology is no longer effective for information transmission in VANETs [5]. To solve this problem, a hierarchical network topology, called cluster, has been proposed for VANETs.

A cluster is a virtual group of nodes having similar characteristics. Clustering scheme is the method to divide vehicles into different groups according to some rules. Each cluster elects at

* Corresponding author.
E-mail addresses: mengying.ren@utt.fr, mengying.ren@telecom-paristech.fr (M. Ren), lyes.khoukhi@utt.fr (L. Khoukhi), houda.labiod@telecom-paristech.fr (H. Labiod), jun.zhang@telecom-paristech.fr (J. Zhang), veronique.veque@u-psud.fr (V. Vèque).

http://dx.doi.org/10.1016/j.vehcom.2016.12.003
2214-2096/© 2017 Elsevier Inc. All rights reserved.

least one leader, called cluster head, who serves as a local central management entity, performing intra-cluster communication arrangement, local information aggregation, and local information dissemination, etc. [4]. A cluster head is followed by one or more than one cluster members. As a hierarchical network, the first level of the network is called intra-cluster communication, where a cluster member can directly communicate with its cluster head or nearby cluster members within the same cluster. The second level of the network is inter-cluster communication, by which a cluster head communicates with nearby cluster heads or road side infrastructures. Sometimes, cluster gateway node is proposed for neighboring cluster communication [6].

In [6], a detailed survey of clustering schemes for VANETs is well presented. Clustering scheme performance is usually judged by cluster stability. Generally, clustering schemes, providing high cluster stability, should ensure the following properties: (1) lower transmission overhead; (2) longer cluster head lifetime and longer cluster member lifetime; (3) less average number of state changes per vehicle; (4) less cluster head changes.

In this paper, we present a simple dynamic mobility-based clustering scheme in the purpose of establishing a stable network backbone for future data aggregation and information transmission. The proposed scheme is based on vehicles' mobility patterns, including moving direction, relative velocity, relative distance, and link lifetime. Different from previous clustering schemes of which nodes are static during cluster formation, our scheme proposes a dynamic cluster formation procedure. A "temporary cluster head" is proposed to help cluster formation. In addition, we propose a "safe distance threshold" in order to control the cluster size. The proposed clustering scheme is evaluated in terms of cluster stability, and its performance is compared with Lowest-ID [7] and VMaSC [13] algorithms.

The rest of the paper is organized as follows: Section 2 discusses the related work in VANET clustering. Section 3 presents the new proposed clustering algorithm from the aspects of cluster head selection, cluster formation and cluster maintenance. Section 4 presents the simulation environment and the performance analysis of our scheme. Section 5 concludes the paper and briefly introduces our future work.

2. Related work

Generally, clustering procedure can be separated into cluster head (CH) selection, cluster formation, and cluster maintenance. CH selection allows vehicles to choose the CH in a centralized or distributed manner; the selection criteria are presented in the following part of this section. Cluster formation process aims to establish the communication link between CH and its cluster members (CMs). Normally, a stable cluster requires a stable link between CH and CMs. However, because of the dynamic nature of VANETs, individual links may come into existence and vanish unpredictably, making the task of establishing and maintaining communication between fast-moving vehicles very challenging [8]. Cluster maintenance process focuses on solving the cluster reformation and vehicle re-affiliation problems. A good cluster maintenance scheme should generate less control overhead and should not use too much network resources. This section introduces some clustering algorithms in VANETs.

2.1. Cluster topology

In terms of cluster topology, clustering schemes can be classified into single-hop clustering and multi-hop clustering, indicating the maximum number of hops from a cluster head to its farthest cluster member.

The majority of clustering algorithms are based on single-hop clusters in which the CM is one-hop away from its CH. CH only chooses its CMs from the local vicinities. Vehicle information is broadcasted through periodic beacon messages [9]. Lowest-ID [7] is a clustering algorithm originally proposed for MANETs. The CH is the vehicle which has the lowest identifier among its neighbors. Every node determines its cluster and only one cluster in a distributed manner. Lowest-ID provides single-hop and non-overlapping clusters which reduce the use of bandwidth. One-hop cluster topology can reduce the cluster formation time and decrease cluster management overhead while fewer information exchanges are required. However, the number of vehicles in a cluster usually depends on vehicle's transmission range and vehicle density. When vehicle density is very high, collision could happen in the cluster and would cause high packet loss rate. When vehicle density is very low, a vehicle is unable to detect neighbors. To solve this problem, VWCA [10] proposed an adaptive transmission range algorithm (AART), based on the intra-cluster communication standard, Dedicated Short-range Communication (DSRC) standard [11]. Vehicle can change the transmission range dynamically from 100 m to 1000 m according to vehicle density.

In recent years, researchers focus on building up multi-hop clusters. In [12], Zhang et al. firstly proposed a multi-hop clustering algorithm for VANETs, based on vehicle's relative mobility. The cluster size is limited by the number of hops between CH and its farthest CM. The algorithm proposed in [13], known as VMaSC, is addressed as the first multi-hop clustering scheme to simulate under realistic traffic scenario, which is generated by Simulation of Urban MObility (SUMO) [14]. The scheme aims to provide more stable clusters and to reduce the number of CHs in the network. The CH election is based on the calculated relative mobility with respect to its neighbors. The performance of VMaSC was compared with [12] for 1-hop, 2-hops, 3-hops, and VMaSC shows a better performance in terms of CH duration, CM duration and CH change times, especially when the cluster size is set to 3-hop. Generally, compared to single-hop clustering schemes, a multi-hop clustering scheme requires more Beacon exchanges within the maximum number of hops, which may cause the increase in the number of connections lost and longer cluster formation time. Simulation results in VMaSC [13] show that the cluster stability decreases considerably when the maximum number of hops is above 3. Chen et al. [15] proposed a neighborhood following strategy for multi-hop clustering, in which, each vehicle finds a stable target to follow. The vehicle only needs to know the information of its local one-hop neighbors, thus, it reduces the packet loss problem.

2.2. Clustering metrics

Clustering schemes can also be classified based on clustering metrics, including CH selection metrics and cluster formation metrics. A simple and direct way to choose a CH is selecting the first vehicle moving in a certain direction. Cluster platooning in CONVOY [16], proposed for highway scenarios selects the first vehicle as a CH. Vehicles within the predefined maximum distance to CH are combined together, which construct a multi-hop cluster. MC-DRIVE [17] proposed a direction-based clustering algorithm for intersection area. The first vehicle moving in a certain direction was selected as CH; clusters are formed in one-hop based on CHs' transmission range (TR). However, this simple CH selection mechanism is only suitable for simple road topology, like straight highway.

Instead of simply choosing the first vehicle as CH, most clustering mechanisms prefer calculating the stability of a node to its surroundings. MOBIC [18] was the first article proposing aggregate mobility (it was originally proposed for ad hoc networks). Each node calculates its relative mobility to all of its neighbors based on Received Signal Strength (RSS). The node with the lowest aggregate mobility is chosen as the CH. Similar to MOBIC, the New-ALM [19]

Please cite this article in press as: M. Ren et al., A mobility-based scheme for dynamic clustering in vehicular ad-hoc networks (VANETs), Veh. Commun. (2016), http://dx.doi.org/10.1016/j.vehcom.2016.12.003

also chooses a node with less variance relative to its surroundings as a CH. Instead of using the RSS parameter, New-ALM calculated relative distance between nodes. Later, to improve the cluster stability, the paper [12] proposed a K-hop clustering. K-hop relative mobility is based on the ratio of packet delivery delay of two consecutive packets. PPC [20] is also a multi-hop clustering mechanism which is based on vehicles' speed variations and the predicted traveling time. Vehicle's "Eligibility" value, indicating cluster stability, decreases exponentially with the increased speed deviation. APROVE [21] is based on a data clustering technique, called Affinity Propagation (AP) [22]. Each vehicle sends hello messages periodically, aggregating availability and responsibility messages. Vehicles' relative distance, position and prediction position of near future are used in [21]. Vehicle with highest sum of availability and responsibility value is selected as a CH. Moreover, a cluster contention time (CCI) is proposed when two CHs encounter each other in order to reduce the unnecessary cluster reformation. SCRP [23] is a cluster-based routing protocol using Dominating Set (DS), which attempts to select a small number of mobile nodes as dominating nodes to form a stable backbone in a network.

Some other clustering mechanisms are based on Weighted Clustering Algorithm (WCA). The CH selection is based on the weighted sum operation. In [24], the author proposed a lane-based clustering algorithm based on vehicles' relative speed, relative position and traffic flow. Each lane can be distributed with a certain weight according to the traffic flow. VWCA [10] calculates the weighted clustering value based on the metrics: vehicle distrust value, entropy value, number of neighbors and relative position. The vehicle with the minimum weighted sum value in the neighbor is selected as CH. Another weighted clustering mechanism AMACAD [25] was proposed based on vehicle's final destination. In AMACAD, vehicles with similar destinations have higher possibility to stay in the same cluster. The weighted sum is calculated based on vehicles' relative destinations, final destinations, relative speed and current position.

2.3. Cluster performance evaluation

As well as we know, the majority of clustering algorithms proposed for VANETs focus on improving the cluster stability. Generally, cluster stability can be evaluated from the following aspects: CH duration, CH change rate, re-clustering frequency, and average state change rate per node, etc. Cluster stability is a crucial measure of the efficiency of clustering algorithms for VANETs. In [26], the author presented a stochastic analysis of the vehicle mobility impact on single-hop cluster stability, and a stochastic mobility model was proposed.

In this paper, a new mobility-based clustering algorithm for VANETs is presented. We evaluate cluster stability from the following aspects: average cluster head lifetime, average cluster member lifetime, average state change rate per node, cluster head change rate, and average number of clusters.

3. Proposed approach

The paper focuses on proposing a new clustering algorithm based on V2V communication for urban city scenario. It assumes that every vehicle is equipped with an On Board Unit (OBU) wireless transceiver/receiver and has a GPS receiver that can update vehicle's location on the road. Meanwhile, each vehicle can calculate the relative velocity with respect to its one hop neighbors, as well as detect the relative distance to its vicinities.

3.1. Cluster definition

We suppose that vehicles enter the road segment one by one with a predefined traffic flow rate. Each vehicle moving on the road broadcasts a *Beacon* message at every Beacon Interval (BI). According to the clustering metrics we have mentioned in Section 2.2, cluster head (CH) should be the vehicle which has higher relatively stability among its neighboring vehicles. Therefore, we choose the vehicle nearest to the central geographical position of a cluster as the CH, so that its neighbors should spend more travel time to leave the cluster, and the cluster is considered to be more stable. Cluster members (CMs) are selected from CH's one-hop neighbor set.

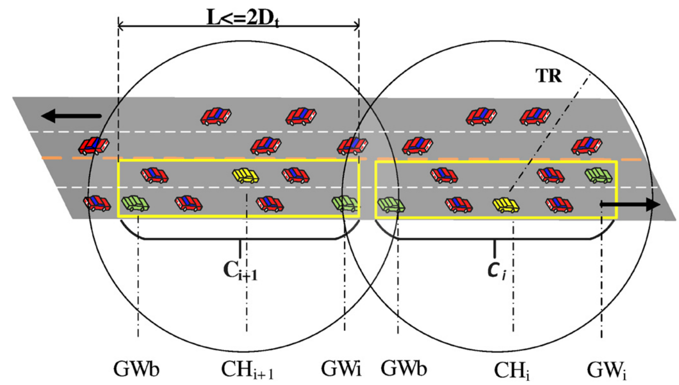


Fig. 1. Clusters (TR: Transmission Range; L: cluster length; D_t: Safe Distance threshold; GWi, GWb: Gateway node.).

Table 1
Notations.

Notation	Description
TR	Transmission range
BI	Beacon interval
D_t	Safe distance threshold
MI	Merge interval
V_i	Vehicle i
C_i	Cluster i
CH_i	Cluster head i
CM_i	Cluster member i
CHt_i	Temporary cluster head i
UN_i	Undecided node i
$Dir(i)$	Moving direction of V_i
$\triangle D_{ij}$	Relative distance between V_i and V_j
L_i	Length of cluster C_i
T_{UN}	Timer for UN transfer to CHt
T_{CHt}	Timer for CHt transfer to CH
T_{wb}	Timer for CH to monitor *Beacon* from its CM
CID	Cluster ID
CML_i	CM list of CH_i
L_{merge}	Length of the merged cluster
BL_i	Beacon list of CH_i, recording the received *Beacons*

Fig. 1 shows two clusters on a straight road, cluster C_i and cluster C_{i+1} (clusters are represented by rectangles). Cluster head is in the central position, and the length of the cluster is smaller than twice of CH's transmission range (TR). In our proposed clustering scheme, each cluster consists of two gateway nodes moving on the edge of the cluster: one is moving ahead and another one is moving in the end of the cluster.

Due to the rapid changes of vehicle mobility, vehicles on the edge of CH's transmission range are considered not being stable enough, and may cause frequent CM disconnections and CM re-clustering. To solve this problem, we introduce a "Safe Distance Threshold", denoted as D_t, which should be smaller than vehicle's TR, $D_t \le TR$. Therefore, the vehicles within D_t range of the CH are considered having more stable links with their CH. The size of the cluster is defined as $L \le 2D_t$. Table 1 lists the notations used through this study.

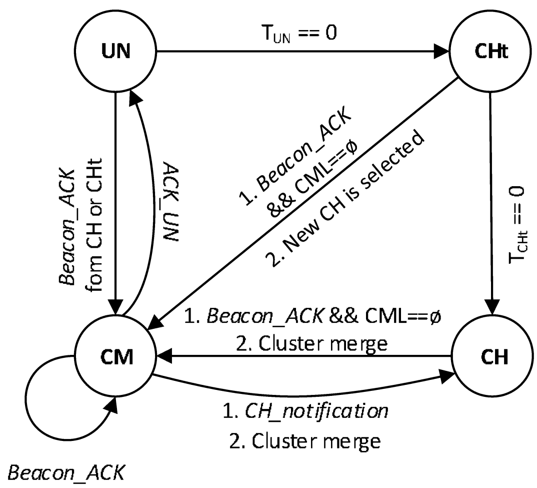


Fig. 2. State transition machine.

3.2. Cluster state transition

In the proposed clustering algorithm, a vehicle may have one of the following 4 states: Undecided Node (UN), Cluster Head (CH), Cluster Member (CM), and Temporary Cluster Head (CHt). The states are specified in the following:

- UN: Initial state of all vehicles, which means that the vehicle does not belong to any clusters.
- CH: The leader of the cluster, which can communicate with all of its members. Each cluster has only one CH and each CH maintains a CM list, CML, recording the information of its CMs.
- CM: The normal vehicle which is a one-hop neighbor of a CH. A particular type of CM is the gateway node (GW), which is responsible for inter-cluster communication and is located on the edge of the cluster. Each cluster may have two gateway nodes: GW_i, moving ahead of the cluster, and GW_b, moving in the end of the cluster.
- CHt: The temporary CH vehicle. It only appears at the beginning of cluster formation procedure and disappears when the CH is elected.

Fig. 2 illustrates the possible state transitions of a vehicle. The vehicle starts with an UN state and sets a timer T_{UN}, during which it hears *Beacon* message from a CH or a CHt. In our study, *Beacon* messages, which are broadcasted by CH or CHt vehicles, are denoted as *Beacon_ACK* message, aggregating confirmation information. Each *Beacon_ACK* message contains an *ACK_list*, a list of node identifiers. If the UN vehicle does not hear any *Beacon_ACK* message until T_{UN} expires, it changes its state to CHt; otherwise, it changes the state to CM upon receiving a confirmation beacon message, called *Beacon_ACK* message, from a CH or a CHt.

The CHt vehicle sets a timer T_{CHt} and initiates a cluster formation process which will be described in the next section. Upon *Beacon_ACK* message reception from a CH, CHt will change its state to CM if it does not have any followers, $CML = \emptyset$. In another situation, the CHt vehicle changes to CM during a CH selection procedure, described in the next section. Otherwise, the CHt vehicle changes to CH when T_{CHt} expires.

When a CH_i hears a *Beacon_ACK* message from a neighboring CH_j, it checks whether it has CMs or not. If $CML_i = \emptyset$, it changes its state to a CM of CH_j. Furthermore, when cluster merging happens, a CH vehicle can also change state to a CM of the merged cluster.

The CM vehicle will change the state to CH when it receives *CH_notification* message from its CHt, or when it is selected as CH in the merged cluster during cluster merging process. A CM vehicle can hear *Beacon_ACK* messages periodically from its CH or CHt, otherwise, it changes the state back to UN if it is no longer the CM of the current cluster, which will be described in Section 3.6.2.

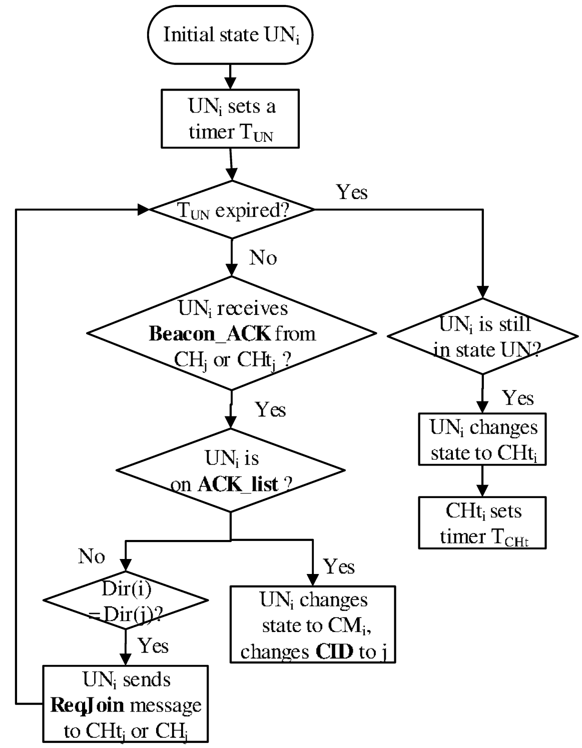


Fig. 3. Cluster formation.

3.3. Cluster formation

As been shown in Fig. 3, a vehicle i in the UN state, UN_i, tries to join an existing cluster by listening to the *Beacon* message from a CH or CHt during the time period T_{UN}. If UN_i fails to join an existing cluster when T_{UN} expires, it claims itself as a CHt node CHt_i, and sets its CID. Meanwhile, CHt_i starts a timer T_{CHt} and begins a cluster formation procedure, as described in Section 3.4.

During the time period T_{UN}, if UN_i hears a *Beacon_ACK* message from CH_j or CHt_j, it checks whether it is on the *ACK_list*. If yes, UN_i changes its state to CM directly and sets its cluster identifier $CID = j$; otherwise, it checks whether it is a CM candidate of CH_j or CHt_j. In this paper, the CM candidate should be the vehicle which are moving in the same direction with its CH, $Dir(i) = Dir(j)$. UN_i sends a $ReqJoin$ message to CH_j or CHt_j, if it is a CM candidate.

Upon receiving a $ReqJoin$ message from vehicle V_i (V_i could be in the state UN or CH), CH_j checks the following conditions to confirm that the requester is a *qualified CM*: (1) the relative distance between V_i and CH_j, ΔD_{ij}, should be smaller than the predefined D_t, $\Delta D_{ij} \le D_t$; (2) V_i is not on CML_j (the CM list of CH_j).

If V_i is a qualified CM, CH_j adds the information of V_i into its CM list CML_j. Meanwhile, CH_j adds the identifier of V_i to its confirmation list, *ACK_list*, which will be broadcasted within its next *Beacon_ACK* message. It is noticed, only vehicles in the state CH or CHt can broadcast a *Beacon_ACK* message.

3.4. Cluster head selection

Similar to a CH, CHt can also add qualified CMs according to the conditions mentioned above. However, CHt only adds CMs which

Please cite this article in press as: M. Ren et al., A mobility-based scheme for dynamic clustering in vehicular ad-hoc networks (VANETs), Veh. Commun. (2016), http://dx.doi.org/10.1016/j.vehcom.2016.12.003

are in its neighborhood. Algorithm 1 describes a CH selection procedure initiated by a vehicle CHt_j.

Upon receiving a $ReqJoin$ message from UN_i moving behind, if CHt_j detects the relative distance $\Delta D_{ij} > D_t$ and $CML_j \neq \emptyset$, CHt_j selects the farthest CM CM_k in its CML_j to be the CH, and sends a $CH\_notification$ message, containing its CM list CML_j, to inform CM_k to become CH_k. Meanwhile, CHt_j changes to CM_j state and resets $CID = k$. After receiving a $CH\_notification$ message, CH_k adds CMs and broadcasts a $Beacon\_ACK$ to inform its CMs to reset $CID = k$. CH_k continues the cluster formation process via adding new CMs. In another case, if CHt_j is still in the state CHt and $CML_j = \emptyset$ when T_{CHt} expires, CHt_j claims itself as CH_j.

Algorithm 1 CH selection process.

while $T_{CHt} \neq 0$ && CHt_j is still in state CHt **do**
  if CHt_j receives $ReqJoin$ from UN_i **then**
    if UN_i is moving behind CHt_j && $\Delta D_{ij} <= D_t$ **then**
      $CML_j \leftarrow UN_i$
      $ACK\_list \leftarrow UN_i$
      CHt_j broadcasts $Beacon\_ACK$ at next BI
    else
      if UN_i is moving behind CHt_j && $\Delta D_{ij} > D_t$ && $CML_j \neq \emptyset$ **then**
        CHt_j chooses the farthest vehicle CM_k from CML_j
        CHt_j sends $CH\_notification$ to CM_k
        $CHt_j \rightarrow CM_j$
        $CID \leftarrow k$
      end if
    end if
  end if
end while
if $T_{CHt} == 0$ && CHt_j is still in state CHt && $CML_j \neq \emptyset$ **then**
  $CHt_j \rightarrow CH_j$
end if

When CM_k receives $CH\_notification$
$CM_k \rightarrow CH_k$
$CML_k \leftarrow CML_j$
$CID \leftarrow k$
CH_k broadcasts $Beacon\_ACK$

3.5. Gateway node selection

As soon as the CH is selected and the cluster is well formed, CH_k selects two CMs, which are moving on the edge, to be the GW nodes. However, it happens sometimes that two GW candidates have the same relative distance from their CH. To solve this problem, we introduce an estimated connection time between CH and CM, called link lifetime (LLT), to evaluate the link sustainability. A higher LLT represents a more sustainable link. CH will select the GW node which has larger LLT value. The work in [27] gives the definition of LLT, shown in Eq. (1), when two vehicles are moving in the same or opposite directions. Although vehicle position should be represented by x-coordinate and y-coordinate, this study assumes the trajectory of all vehicular nodes to be a straight line, as the lane width is small. Thus, the y-coordinate can be ignored. We denote the positions of V_k and V_j by x_k and x_j, respectively.

$$LLT_{kj} = \frac{-\Delta v_{kj} * \Delta D_{kj} + \Delta v_{kj} * TR}{(\Delta v_{kj})^2} \quad (1)$$

$$\Delta D_{kj} = |x_k - x_j| \quad (2)$$

$$\Delta v_{kj} = |v_k - v_j| \quad (3)$$

Note that the TR is the transmission range of the vehicle, v_k and v_j are the velocities of CH_k and CM_j, respectively.

3.6. Cluster maintenance

Due to the high dynamic nature of VANETs, vehicles keep joining and leaving clusters frequently, thus, causing extra maintenance overhead. In our proposed scheme, Clusters are dynamically moving on the road, with their CH inside of the clusters. When CH loses all of its CMs, it becomes an UN node. Otherwise, it remains as CH until cluster merging process happens. Therefore, in our proposed scheme, the cluster maintenance procedure only deals with cluster merging and vehicle leaving steps.

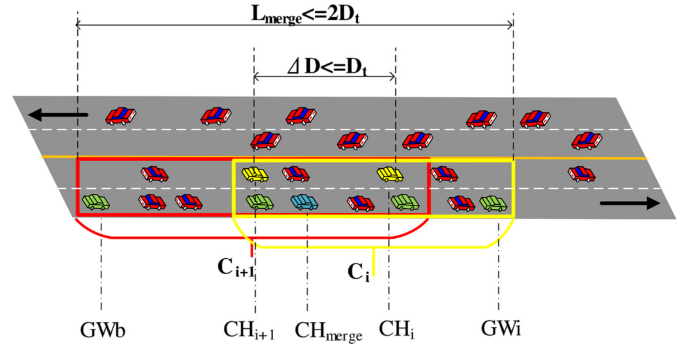


Fig. 4. Cluster merging (L_{merge}: length of the merged cluster; D_t: Safe Distance threshold.).

1) Cluster merging: The proposed algorithm allows cluster to be overlapped. However, when two neighboring clusters C_i and C_{i+1} have a big overlapping area, as presented in Fig. 4, cluster merging procedure is triggered. Instead of having two CHs, a single CH is selected. When the distance of two CHs is smaller than the predetermined threshold D_t, cluster merging procedure begins. To avoid frequent re-clustering, cluster merging is deferred. Instead of starting the cluster merging procedure immediately, the merging procedure begins if two CHs can always hear each other and are always within the range of D_t during the Merge Interval (MI). Once the cluster merging process begins, CH_{i+1}, moving behind, will send a $ReqMerge$ message to CH_i, the CH moving ahead. Cluster merging process is described in Algorithm 2.

Upon $ReqMerge$ message reception from CH_{i+1}, CH_i estimates the potential merged cluster size L_{merge}. If $L_{merge} \leq 2D_t$, cluster merging is permitted and a CH for the merged cluster, called CH_{merge}, is selected, which is the nearest node to the geographical central position of the merged cluster. After selecting CH_{merge}, previous CHs will send a ACK_merge message, containing their CMs list CML, to CH_{merge}, and claims themselves as the CMs of CH_{merge}. The CH_{merge} adds all of the CMs to its CML and broadcasts a $Beacon\_ACK$ message to inform its CMs to change their CID.

Algorithm 2 Cluster merging process.

Upon receiving $ReqMerge$
CH_i estimates L_{merge}
if $L_{merge} \leq 2 * D_t$ **then**
  CH_i selects central CM CM_m to be CH_{merge}
  CH_i and CH_{i+1} sends ACK_merge along with their CML to CM_m respectively
  $CH_i \rightarrow CM_i$
  $CH_{i+1} \rightarrow CM_{i+1}$
  $CID \leftarrow m$
end if

Upon receiving ACK_merge
$CM_m \rightarrow CH_{merge}$
$CML_m \leftarrow CML_i, CML_m \leftarrow CML_{i+1}$
$CID \leftarrow m$
CH_{merge} broadcasts $Beacon\_ACK$

2) Leaving a cluster: In the proposed approach, each CH creates and updates a CML dynamically. CH has to monitor the presence of its CMs per every waiting beacon interval, denoted as T_{wb}. Therefore, CH can detect CM disconnection as long as it does not receive the $Beacon$ message from its CM at least T_{wb} time period. Moreover, each CH creates a beacon list (BL) in order to record the

Please cite this article in press as: M. Ren et al., A mobility-based scheme for dynamic clustering in vehicular ad-hoc networks (VANETs), Veh. Commun. (2016), http://dx.doi.org/10.1016/j.vehcom.2016.12.003

Table 2
List of important messages.

Name of the message	Source	Dissemination type
Beacon	UN or CM	Broadcast
Beacon_ACK	CH or CHt	Broadcast
ReqJoin	Any single node	Towards a CH or CHt
ReqMerge	CH	Towards a CH
CH_notification	CH or CHt	Towards a new merged CH
ACK_UN	CH	Towards a CM

reception of its CMs' *Beacons*. Once a CH, for example CH_i, receives a *Beacon* message from CM_j, it checks whether CM_j is within the range of D_t. If $\Delta D_{ij} \leq D_t$, CH_i updates the information of CM_j and set $BL_i(j)$ to 1, indicating the reception of the information of CM_j; otherwise, it deletes CM_j from CML_i.

Algorithm 3 Leaving a cluster.

```
When CH_i receives Beacon from CM_j
if CM_j ∈ CML_i && ΔD_ij ≤ D_t then
    BL_i(j) ← 1
    CH_i updates the information of CM_j in CML_i
end if

Every time when T_wb == 0
for all CM_k ∈ CML_i do
    if BL_i(k) == 0 then
        CH_i deletes CM_k and BL_i(k)
    else
        if BL_i(k) == 1 then
            BL_i(k) ← 0
        end if
    end if
end for
Restart T_wb
```

3) CML and GW updating: Every time when a CH receives a *Beacon* message from its CM, it updates CM's information, for example the position, in its CML. Therefore, every CH can monitor its CML dynamically. Once the CML is updated, GW_i and GW_b selection functions are triggered immediately and cluster's gateway information will also be updated according to the process described in Section 3.5.

3.7. Important messages

Table 2 presents a set of important messages transmitted during the clustering procedure, and the message dissemination types are demonstrated. Every message must contain the following parameters: message type, source ID, source state, cluster identifier CID, x-coordination x, y-coordination y, velocity v, and direction Dir. Compared to a simple *Beacon* message, *Beacon_ACK* adds a ACK_list, and is only broadcasted by a CH or CHt.

4. Simulation

In this section, we provide a deep analysis of our proposed clustering scheme, and compare the clustering performance with two existing algorithms, Lowest-ID [7] and VMaSC [13]. Since both the proposed algorithms and Lowest-ID are based on one-hop cluster, the one-hop VMaSC is implemented in our simulation. All of the clustering algorithms are implemented on NS2 [28], and the testing scenarios are all generated by Simulation of Urban MObility (SUMO) [14].

In the testing scenarios, the road topology consists of a two-lane and two-way road of length 15 km. Vehicles are deployed in the road with a predefined traffic flow rate (vehicles per hour), denoted as TFR. The maximum vehicle velocity, being allowed on the road, is called maximum lane speed (MLS). We consider 100 vehicles, 50 vehicles for each direction.

Table 3
Simulation parameters.

Parameter	Value
Simulation time	300 s
MAC protocol	IEEE 802.11p
TR	200 m
Number of vehicles	100
Road length	15 km
Length of car	5 m
Acceleration rate	2.6 m/s^2
Deceleration rate	4.5 m/s^2
Maximum lane speed (MLS)	10–40 m/s
Traffic flow rate (TFR)	1200 vehicles/hour
D_t	100–200 m
BI	1.0 s
MI	10.0 s
T_{wb}	5.0 s
Propagation model	Two-Ray Ground
Number of iterations	10
Mobility model	Car-following model

We firstly evaluate the impacts of "Safe Distance threshold" D_t. Traffic flow rate is set to 1200 vehicles per hour, and maximum lane speed is set to 20 m/s, which is considered a regular speed on the road. The value of D_t is set to be in the range of 100–200 m, smaller than vehicle's transmission range. Therefore, cluster size is in the range of 200–400 m, as defined in our algorithm.

The second simulation evaluates the impacts of the Beacon Interval (BI) on the cluster stability with the increased maximum lane speed (MLS). The set of MLS are specified as follows: 10, 15, 20, 25, 30, 35, 40 m/s; traffic flow rate is set to 1200 vehicles per hour. The BI is set to 0.5 s, 1.0 s and 2.0 s respectively.

In the third simulation, we evaluate the impacts of the maximum lane speed (MLS) on cluster stability, and compare the clustering performance of the proposed algorithm with Lowest-ID [7], denoted as LID, and one-hop VMaSC [13], denoted as VMaSC_1hop under the same context. The set of maximum lane speed are specified as follows: 10, 15, 20, 25, 30, 35, 40 m/s. The traffic flow rate is set to 1200 vehicles per hour. To make a fair comparison, T_{wb} is set to 5.0 s, the same value as CH_TIMER when implementing VMaSC [13], and the same value of information updating interval in LID [7].

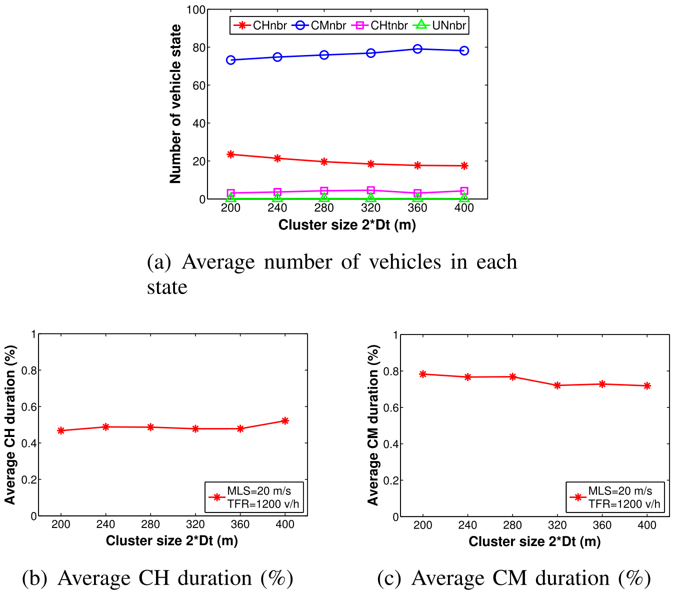
For each testing scenario, simulation runs for 800 s. The clustering process starts at time T_{start}, the time when all vehicles have entered the road, and ends at time T_{end}, before which most of vehicles are still on the road. According to the testing scenarios, we set $T_{start} = 160$ s and $T_{end} = 460$ s. Therefore, the clustering simulation time is 300 s. All of the simulations run 10 times. According to previous related works (e.g., [12,13,21]), our simulation parameters are selected as illustrated in Table 3.

4.1. Performance metrics

The clustering performance metrics, used for cluster stability evaluation and comparison, are described as follows:

- Average number of clusters: as long as the CH is alive, there is a cluster. This metric allows us evaluating the quality of cluster formation. In the worst case, each vehicle represents an independent cluster; therefore, clustering algorithm is meaningless.
- Average CH duration: this metric represents the cluster's lifetime, the time interval between a vehicle becoming a CH and changing to another state. In general, a longer duration of CH represents a more stable cluster. In this paper, the normalized average CH duration is the percentage time period of the total simulation time.

Please cite this article in press as: M. Ren et al., A mobility-based scheme for dynamic clustering in vehicular ad-hoc networks (VANETs), Veh. Commun. (2016), http://dx.doi.org/10.1016/j.vehcom.2016.12.003

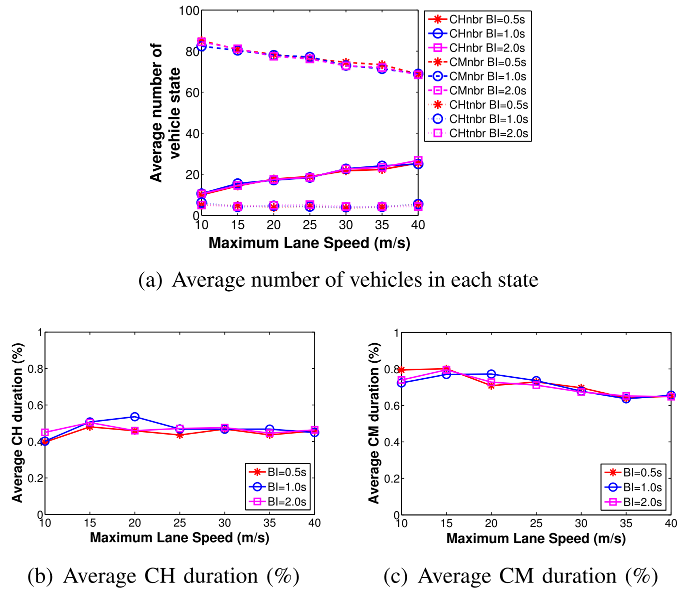


(a) Average number of vehicles in each state

(b) Average CH duration (%)

(c) Average CM duration (%)

Fig. 5. Impacts of D_t on cluster performance.



(a) Average number of vehicles in each state

(b) Average CH duration (%)

(c) Average CM duration (%)

Fig. 6. Impact of BI on cluster performance with the increased maximum lane speed (MLS).

- Average CM duration: it defines the average time interval from a node joining an existing cluster as a CM to leaving the connected cluster or to becoming a CH. The normalized average CM duration is the percentage time period of the total simulation time.
- Average CH change rate (per second): the cluster head change rate defines the number of state transitions from CH to another state per unit time.
- Average state change (per node): this metric indicates the number of state transitions in each vehicle during the clustering procedure.
- Clustering efficiency: it is defined as the percentage of vehicles participating in clustering procedure (vehicles which are not in UN state) during the simulation. A higher clustering efficiency means a better clustering performance.
- CM disconnection frequency (per second): it illustrates the total number of times that CMs lose the connections to their current CHs per unit time.

4.2. Results and analysis

1) "Safe Distance threshold" Dt: Fig. 5 presents the impacts of "Safe Distance threshold" D_t. In Fig. 5(a), with the increased D_t, less clusters are organized during the simulation. This is because that more vehicles are combined in a cluster as CMs when the cluster length increases under the same traffic density. The numbers of vehicles in CHt and UN states (both are temporary states) remain stable when cluster size is becoming larger. Fig. 5(b) shows the average CH duration, represented as the percentage of total simulation time. The average CH duration increases slightly but remains relatively stable, when D_t increases. Fig. 5(c) illustrates that the average CM duration decreases slightly with the increased cluster size. We observe that D_t has small impacts on both the CH duration and CM duration.

2) Beacon Interval (BI): According to ETSI standard [29], the Cooperative Awareness Message (CAM) is broadcasted with the frequency 1–10 Hz (0.1 s–1 s). Therefore, in our simulation, we set BI to 1.0 s as the default value, and change BI to 0.5 s and 2.0 s respectively, in order to evaluate its impacts on our proposed algorithm.

The results in Fig. 6 show the cluster performance in terms of average number of vehicles in each state (Fig. 6(a)), average CH duration (Fig. 6(b)), and average CM duration (Fig. 6(c)). From the results, we observe that BI has slight impact on the cluster performance. According to the simulation results, BI is set to 1.0 s in the rest of the simulation. In Fig. 6(a), we observe that the number of vehicles in the state CH increases with the increased maximum vehicle velocity, and meanwhile, the number of CM vehicles decreases. This is because that with the increased vehicle velocity, some CMs may move out of the cluster and may become isolated vehicles. Then, if the isolated vehicle cannot successfully reconnect to another existing cluster, a new cluster will be formed, increasing the number of CHs.

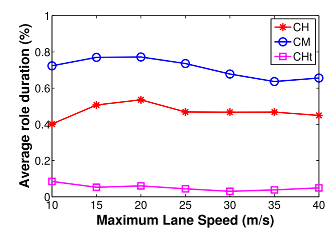
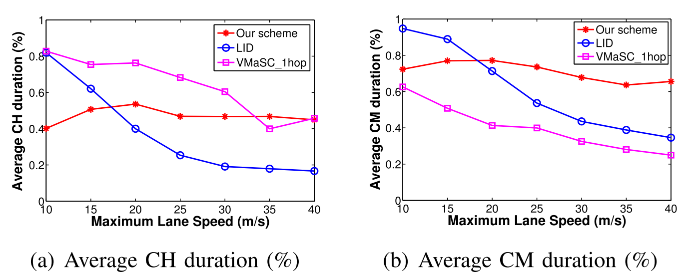


Fig. 7. Average duration of each vehicle state of the proposed scheme.



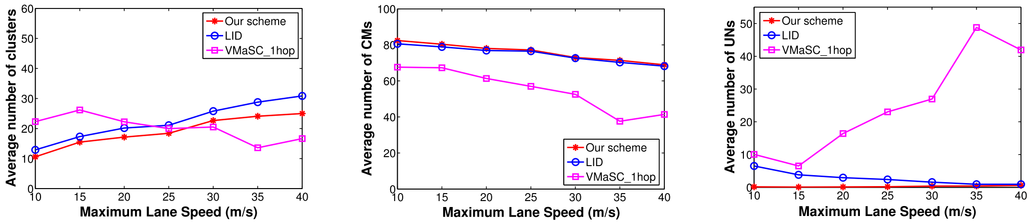
(a) Average CH duration (%)

(b) Average CM duration (%)

Fig. 8. Vehicle state lifetime comparison under the impact of vehicle's MLS.

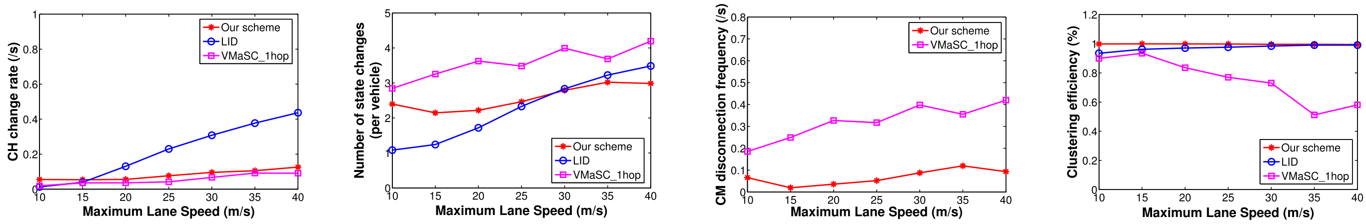
3) Impact of maximum lane speed: Fig. 7 presents the averaged lifetime of each vehicle state with the increased maximum lane speed (MLS), in the proposed algorithm. We observe that when vehicle velocity increases from 10 m/s to 40 m/s, vehicle state lifetime is relatively stable. The CHt lifetime is very small because it is a temporary state which only appears in the beginning of a cluster formation process.

Please cite this article in press as: M. Ren et al., A mobility-based scheme for dynamic clustering in vehicular ad-hoc networks (VANETs), Veh. Commun. (2016), http://dx.doi.org/10.1016/j.vehcom.2016.12.003



(a) Average cluster number (b) Average number of CMs (c) Average number fo UNs

Fig. 9. Vehicle state number comparison under the impact of vehicle's MLS.



(a) Average CH change rate (per second) (b) Average state change (per node) (c) CM disconnection frequency (per second) (d) Clustering efficiency (%)

Fig. 10. Cluster stability comparison under the impact of vehicle's MLS.

The results in Figs. 8, 9, and 10, compare the cluster stability among the proposed clustering algorithm to Lowest-ID (LID) and VMaSC (VMaSC_1 hop), from the aspects of the vehicle state duration, the number of vehicle states, and the number of state changing, respectively.

Fig. 8(a) and Fig. 8(b) reveal the impacts of the maximum vehicle velocity on the averaged CH lifetime and CM lifetime. It is obvious that both the averaged CH and CM duration of LID and VMaSC_1hop decrease rapidly when MLS increases. In Fig. 8(a), when vehicles move slowly on the road, both the mean CH duration of LID and VMaSC_1hop are higher than that of our scheme. However, their CH duration decreases rapidly with the increased MLS, especially for LID. The CH duration of VMaSC_1hop is always higher than that of our scheme until MLS becomes bigger than 33 m/s. This is because in our scheme, CHt assists cluster formation and CH is selected during the cluster formation process, while CH selection happens in the beginning in VMaSC_1hop and LID. In Fig. 8(b), the CM duration in our scheme remains the highest one when MLS is bigger than 17 m/s. When MLS becomes bigger than 30 m/s, the CM duration of our scheme is almost two times of VMaSC_1hop. The results in Fig. 9(b) and Fig. 9(c) well explain this consequence that with the increased MLS, cluster becomes less stable and many vehicles change to the temporary state in VMaSC_1hop [13], therefore, reduces the average CH and CM duration.

Results in Fig. 9 show that both the number of the CHs and the number of UNs in our scheme are slightly lower than the results of LID. Moreover, when MLS becomes larger, many CHs and CMs in VMaSC_1hop change to UN state (SE state in [13]). Therefore, the number of CHs and CMs of VMaSC_1hop in Fig. 9(a) and in Fig. 9(b) decrease and the number of UNs in Fig. 9(c) grows quickly.

Fig. 10 demonstrates the details of state transitions during the clustering process. The results in Fig. 10(a) and Fig. 10(b) reveal that both the CH change rate and vehicle state change times of LID grow quickly when MLS is increasing. In LID, CH changes its state as soon as it detects a neighbor vehicle having an identifier smaller than itself. When vehicle velocity increases, vehicle's neighbor list changes considerably, causing more frequent CH change rate. CH change rates of VMaSC_1hop and our scheme are both very low and remain relatively stable in Fig. 10(a). In our scheme, CH may change to a CM when cluster merging happens or when it loses all of its CMs, as mentioned in Section 3. In Fig. 10(b), the number of state transitions for each vehicle in VMaSC_1hop and our scheme are higher than that of LID. This is because more vehicle states are defined in these two schemes compared to LID.

The CM disconnection frequency, shown in Fig. 10(c), presents a similar growth trend compared to the results in Fig. 10(b). It is because that vehicle state transition always happens when a CM loses the link connection with its current CH. Since state transition in LID is identifier-based, Fig. 10(c) only compares our scheme and VMaSC_1hop. It is obvious that our scheme shows a very low CM disconnection frequency compared to VMaSC_1hop, indicating that our scheme provides higher cluster stability.

From the results in Fig. 10(d), we observe that both LID and our scheme perform a very high clustering efficiency, which is close to 100% when MLS increases. It means that almost all of vehicles on the road participate in clustering procedure during the simulation. However, with the growth of MLS, the clustering efficiency of VMaSC_1hop decreases significantly. It is because the number of UN nodes increases rapidly when MLS becomes high, as shown in Fig. 9(c).

5. Conclusion

In this paper, we introduce a novel dynamic mobility-based clustering scheme for VANETs. In our scheme, vehicles are clustered in a single-hop based cluster limited by a predetermined value "Safe Distance Threshold" D_t, where $D_t \leq TR$. CH is selected as the vehicle which is at the geographical center of a cluster, and CMs are within D_t range of the CH, moving in the same direction. A new vehicle state, called temporary cluster head CHt, is proposed in order to help cluster formation process. CHt vehicle only exists at the beginning of a cluster formation procedure. It changes its state to CH or CM as soon as the CH is selected. Cluster maintenance mechanisms are proposed, including cluster merging and leaving a cluster procedure.

Extensive simulations in NS2 with the vehicle mobility input from SUMO demonstrate the superior cluster performance of our scheme over LID scheme and VMaSC scheme, in terms of average cluster number, average CH duration, average CM duration, CH change rate, number of vehicle state changes, CM disconnection frequency, and clustering efficiency. The simulation results show that our proposed clustering scheme provides higher cluster stability even in a high dynamic traffic scenario.

As future work, we aim to investigate the impact of the number of hops on the cluster performance, since our algorithm is based on one-hop approach. Moreover, we will implement our proposed clustering algorithm for data aggregation and data dissemination for recent VANETs applications.

References

[1] Y.Z. Hassnaa Moustafa, Vehicular Networks: Techniques, Standards, and Applications, 2nd ed., Auerbach Publications, 2009.
[2] S. Al-Sultan, M.M. Al-Doori, A.H. Al-Bayatti, H. Zedan, A comprehensive survey on vehicular ad hoc network, J. Netw. Comput. Appl. 37 (2014) 380–392.
[3] ETSI TR 102 638 v1.1.1, intelligent transport systems (ITS); vehicular communications; basic set of applications; definitions.
[4] Y. Toor, P. Muhlethaler, A. Laouiti, A.D.L. Fortelle, Vehicle ad hoc networks: applications and related technical issues, IEEE Commun. Surv. Tutor. 10 (3) (2008) 74–88.
[5] J. Yu, P. Chong, A survey of clustering schemes for mobile ad hoc networks, IEEE Commun. Surv. Tutor. 7 (1) (2005) 32–48.
[6] R.S. Bali, N. Kumar, J.J. Rodrigues, Clustering in vehicular ad hoc networks: taxonomy, challenges and solutions, Veh. Commun. 1 (3) (2014) 134–152.
[7] C. Lin, M. Gerla, Adaptive clustering for mobile wireless networks, IEEE J. Sel. Areas Commun. 15 (7) (Sep. 1997) 1265–1275.
[8] G. Yan, S. Olariu, A probabilistic analysis of link duration in vehicular ad hoc networks, IEEE Trans. Intell. Transp. Syst. 12 (4) (Dec. 2011) 1227–1236.
[9] Y. Allouche, M. Segal, Cluster-based beaconing process for {VANET}, Veh. Commun. 2 (2) (2015) 80–94.
[10] A. Daeinabi, A.G.P. Rahbar, A. Khademzadeh, Vwca: an efficient clustering algorithm in vehicular ad hoc networks, J. Netw. Comput. Appl. 34 (1) (2011) 207–222.
[11] J.B. Kenney, Dedicated short-range communications (DSRC) standards in the United States, Proc. IEEE 99 (7) (2011) 1162–1182.
[12] Z. Zhang, A. Boukerche, R. Pazzi, A novel multi-hop clustering scheme for vehicular ad-hoc networks, in: Proceedings of the 9th ACM International Symposium on Mobility Management and Wireless Access, ser. MobiWac '11, ACM, New York, NY, USA, 2011, pp. 19–26.
[13] S. Ucar, S.C. Ergen, O. Ozkasap, Multi-hop cluster based ieee 802.11p and lte hybrid architecture for VANET safety message dissemination, IEEE Trans. Veh. Technol. PP (99) (2015) 1.
[14] D. Krajzewicz, J. Erdmann, M. Behrisch, L. Bieker, Recent development and applications of sumo-simulation of urban mobility, Int. J. Adv. Syst. Meas. 5 (3–4) (2012).
[15] Y. Chen, M. Fang, S. Shi, W. Guo, X. Zheng, Distributed multi-hop clustering algorithm for VANETs based on neighborhood follow, EURASIP J. Wirel. Commun. Netw. 2015 (1) (2015) 1–12.
[16] V. Vèque, F. Kaisser, C. Johnen, A. Busson, CONVOY: a new cluster-based routing protocol for vehicular networks, in: H. Labiod, A. Beylot (Eds.), Models and Algorithms for Vehicular Networks, ISTE Publishing Knowledge/John Wiley and Sons Inc., May 2013, pp. 91–102, Chapter 4.
[17] N. Maslekar, J. Mouzna, H. Labiod, M. Devisetty, M. Pai, Modified c-drive: clustering based on direction in vehicular environment, in: 2011 IEEE Intelligent Vehicles Symposium (IV), June 2011, pp. 845–850.
[18] P. Basu, N. Khan, T.D. Little, A mobility based metric for clustering in mobile ad hoc networks, in: 2001 International Conference on Distributed Computing Systems Workshop, IEEE, 2001, pp. 413–418.
[19] E. Souza, I. Nikolaidis, P. Gburzynski, A new aggregate local mobility clustering algorithm for VANETs, in: 2010 IEEE International Conference on Communications (ICC), May 2010, pp. 1–5.
[20] Z. Wang, L. Liu, M. Zhou, N. Ansari, A position-based clustering technique for ad hoc intervehicle communication, IEEE Trans. Syst. Man Cybern., Part C, Appl. Rev. 38 (2) (2008) 201–208.
[21] B. Hassanabadi, C. Shea, L. Zhang, S. Valaee, Clustering in vehicular ad hoc networks using affinity propagation, Ad Hoc Netw. 13 (2014) 535–548, Part B.
[22] B.J. Frey, D. Dueck, Clustering by passing messages between data points, Science 315 (5814) (2007) 972–976.
[23] M. Togou, A. Hafid, L. Khoukhi, SCRP: stable CDS-based routing protocol for urban vehicular ad hoc networks, IEEE Trans. Intell. Transp. Syst. PP (99) (2016) 1–10.
[24] M. Almalag, M. Weigle, Using traffic flow for cluster formation in vehicular ad-hoc networks, in: 2010 IEEE 35th Conference on Local Computer Networks (LCN), Oct. 2010, pp. 631–636.
[25] M. Morales, C.S. Hong, Y.C. Bang, An adaptable mobility-aware clustering algorithm in vehicular networks, in: 2011 13th Asia–Pacific Network Operations and Management Symposium (APNOMS), Sept. 2011, pp. 1–6.
[26] K. Abboud, W. Zhuang, Stochastic modeling of single-hop cluster stability in vehicular ad hoc networks, IEEE Trans. Veh. Technol. 65 (1) (Jan. 2016) 226–240.
[27] S.-S. Wang, Y.-S. Lin, Passcar: a passive clustering aided routing protocol for vehicular ad hoc networks, Comput. Commun. 36 (2) (2013) 170–179.
[28] Network simulator 2 (ns-2), http://isi.edu/nsnam/ns/.
[29] Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service, ETSI EN 302 637-2 V1.3.1, pp. 1–44, Sept. 2014.

Further reading

[30] M. Rebai, L. Khoukhi, H. Snoussi, F. Hnaien, Optimal placement in hybrid VANETs-sensors networks, in: Wireless Advanced (WiAd), IEEE, 2012, pp. 54–57.
[31] L. Khoukhi, A. El Masri, A. Sardouk, A. Hafid, D. Gaiti, Toward fuzzy traffic adaptation solution in wireless mesh networks, IEEE Transactions on Computers 63 (5) (2014) 1296–1308.

Please cite this article in press as: M. Ren et al., A mobility-based scheme for dynamic clustering in vehicular ad-hoc networks (VANETs), Veh. Commun. (2016), http://dx.doi.org/10.1016/j.vehcom.2016.12.003