A Mobility-Based Double-Head Clustering Algorithm for Dynamic VANET

Ghada H. Alsuhli, Ahmed Khattab, Yasmine A. Fahmy
Electronics and Electrical Communications Engineering Department – Cairo University, Giza, Egypt
gada.sy@hotmail.com, akhattab@ieee.org, yfahmy@cws-cufe.org

***Abstract*—Vehicular Ad Hoc Network (VANET) is a promising technology that still faces many challenges such as scalability and the highly dynamic topology. An effective VANET clustering algorithm significantly relieves the effect of these challenges. In this paper, we propose a double-head clustering (DHC) algorithm for VANETs. Our proposed approach is a mobility-based clustering algorithm that exploits the most relevant mobility metrics such as vehicles' speed, position and direction, in addition to other metrics related to the communication link quality in order to achieve stable clusters. We compare the proposed algorithm against existing clustering algorithms using different evaluation metrics under dynamic and static mobility scenarios. The proposed algorithm proves its stability and efficiency under different mobility scenarios.**

***Index Terms*—VANET; ITS; clustering algorithm; cluster head selection; mobility**

I. Introduction

The progressive development of Vehicular Ad-Hoc Networks (VANETs) paves the way for Intelligent transportation System (ITS) safety and infotainment applications. Vehicle clustering has been proposed to tackle the scalability and high dynamic nature of VANETs [1]. Clustering converts the network structure from being flat to be hierarchical by dividing the network into virtual groups of vehicles called clusters. Each cluster has a Cluster Head (CH), and a number of Cluster Members (CMs). The CH acts as an infrastructure of the cluster. Consequently, a better utilization of the network resources and scheduling of medium access are obtained [2].

In this paper, we propose a new clustering algorithm with a focus on increasing the stability of clusters and decreasing the number of clusters in the network. Our proposed technique has two functioning CHs in each cluster to avoid unnecessary re-clustering when a CM instantaneously loses the link with the primary CH. Moreover, a new multi-metric CH selection scheme is proposed. This scheme takes into consideration different selection metrics that increase the cluster stability. In order to maintain the cluster stability, we propose an All-Member-Interests-based Merging (AMIM) scheme in which the merging decision is based on the benefits of all members not only the CH.

The remainder of this paper is organized as follows. In section II, we overview the related literature of VANET clustering approaches. The proposed approach is presented in section III. The performance evaluation and simulation results are discussed in Section IV. Finally, Section V concludes the paper.

II. VANET Clustering Literature Overview

Despite the wide diversity of clustering algorithms in the literature, they share the same procedural steps [3]. The fundamental clustering procedures are: selecting a cluster leadership and adopting some schemes to maintain the formed clusters.

Clustering leadership approaches differ in terms of the number of CHs and the way they are selected. Most of the existing schemes suffice with one CH for each cluster such as in [4]–[6]. Despite the simplicity of this approach, re-clustering will be an inevitable once the CH resigns or loses its suitability to continue as a CH. In response to this, AMACAD [7], SBCA [8], FLBA [9] and SCalE [10] suggested to have a *backup* CH that takes the role of the current CH under certain circumstances. The common factor between these works is that the alternative CH has no role in the cluster unless the current CH leaves the cluster. On the contrary, our use of a second CH to be always functioning to respond to any CM that temporary loses its connection with the primary CH.

CH selection methods in the literature can be classified into competition-based [5], [11] and comparison-based [4], [10], [12], [13] CH selection methods. Competition-based algorithms, such as UFC [5] and TB [11], select the CH by having each vehicle independently setting a back-off timer. The vehicle that firstly broadcasts a CH announcement wins the competition and becomes the CH. In comparison-based methods, each node either compares the received suitability indexes from different one-hop neighbors as in SCalE [10] and APROVE [12], or it calculates and compares the neighbors' indexes by itself independently as in RMAC [13] and VMASC [4]. According to the comparison result, the node may report itself as a CH or send a request to join the most suitable neighbor. Our approach uses a comparison-based selection due to its simplicity and efficiency to minimize the number of formed clusters with relatively smaller number of control packets.

Since minimizing the number of clusters in the system is an aim, many cluster merging schemes were proposed as maintenance strategies. The decision of merging is usually taken by a CH when: 1) another stable CH stays in its range for a minimum period of time as in [5] and [14], 2) the candidate CH is more eligible than the current CH as in [4], [15] or 3) the other stable CH has more members as in [11]. The merging decision is taken in our AMIM scheme based on all members' interests not only the interest of the CH.

978-1-5386-9230-1/18/$31.00

III. Proposed Double-Head Clustering (DHC)

Our approach considers Vehicle-to-Vehicle (V2V) VANET environment. No additional infrastructure such as roadside units (RSUs) is assumed. Each vehicle is equipped with a Dedicated Short-Range Communication (DSRC) wireless unit and a Global Positioning System (GPS). The DSRC transceiver allows the vehicle to communicate with other vehicles in the network. The vehicle uses the GPS, or any other positioning system, to obtain its real time position and velocity which are necessary to be measured and exchanged periodically. Furthermore, a unique identifier (ID) is assigned to each vehicle in the network.

The proposed DHC clustering algorithm is distributed in which each vehicle runs the algorithm independent of other vehicles in the network. Each cluster has a primary CH, donated for simplicity by CH, and a secondary CH (SCH). The CH leads a number of CMs forming the cluster. During the clustering process, the vehicle goes through a number of states and different clustering procedure steps that will be described in the following subsections.

A. Vehicle States

In DHC, each vehicle operates in one of the following four states: *(1) Un-clustered Vehicle (UV) state* which is the initial state before starting the clustering process; *(2) CH state* in which a vehicle is supposed to be the most eligible vehicle among its one-hop neighbors to act as a leader; *(3) SCH state* in which the vehicle acts as a mirror of the CH in a cluster; *(4) CM state* which is the state entered by the vehicle when it joins a cluster. The transition from one state to another is triggered by the events illustrated in Fig. 1. We will explain these events in details while explaining our proposed clustering algorithm procedure in what follows.

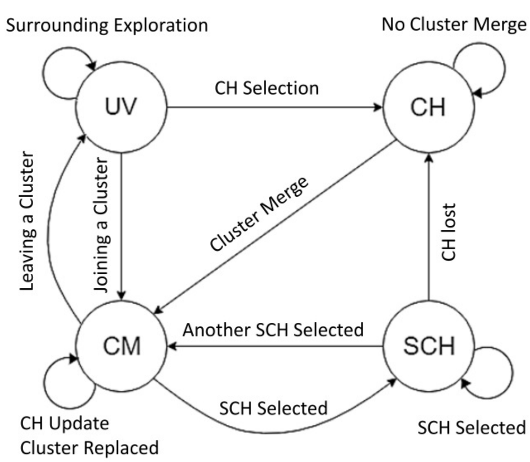

Fig. 1. Vehicle state diagram.

B. Clustering procedure

Vehicles seeking to be clustered will follow some or all of the steps described below.

1) Surrounding Exploration: Initially, all vehicles join the road in the UV state. When any vehicle decides to join the VANET, it begins to send a periodic *Hello* message to advertise its existence and to share its mobility information with the surrounding neighbors. Figure 2 shows the information carried in the broadcasted *Hello* message sent with a period T_{Hello}. This information includes: vehicle ID (VID), cluster ID (CID), two-dimensional position (x, y), two-dimensional velocity (v_x, v_y), number of consistent neighbors N and the vehicle's eligibility E to be a CH.

Once a *Hello* message is received, the consistency is examined for this neighbor. The sender's neighbors are classified into consistent or inconsistent neighbors. The consistent neighbors CNs must have the same road direction with a speed difference Δs less than a predefined limit Δs_{max}. The consistent neighbor information is stored in the Consistent Neighbors Table (CNT) alongside the receiving time (RT), signal-to-noise ratio (SNR) of the received packet and expected link expiration time (LET) with this neighbor. The content of the CNT table is illustrated in Fig. 2.

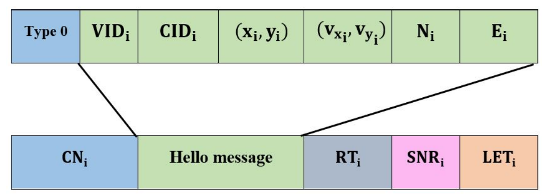

Fig. 2. *Hello* message and CNT table content.

2) CH Selection: Each vehicle calculates and advertises its eligibility periodically within a *Hello* message during an exploration time T_E. When the T_E timer expired, a vehicle automatically enters the CH selection step in which it compares its eligibility with consistent neighbors' eligibility. The vehicle that has the highest E_c announces itself as a CH by sending a *CHHello* message immediately and then periodically each T_{Hello}. The received *CHHello* message is stored in a Cluster Head Table (CHT) that is similar to the CNT table.

Each node estimates its eligibility depending on its popularity (i.e., the number of consistent neighbors N), relative position with respect to the mean position of the CNs, relative speed with respect to the mean speed of the CNs, the average signal-to-noise ratio, and the average link expiration time.

LET is the average communication link lifetime between n_c and its consistent neighbors. We compute LET between the current vehicle and neighbor n_i assuming that the road width can be ignored as:

$$LET_{ci} = \frac{R - \Delta d_{ci} \cdot sign(\Delta s_{ci})}{|\Delta s_{ci}|} \quad \forall n_i \in CNT_c \qquad (1)$$

where R is the service channel range, Δd_{ci} and Δs_{ci} are the distance and speed difference between n_c and n_i, respectively. $sign(.)$ is the sign function, i.e., $sign(\Delta s_{ci}) = 1$ when n_c is faster than n_i and -1 otherwise.

Suitability of the current vehicle to be the CH is specified by its eligibility value, E_c, which is computed in Eq. (2) as a weighted sum of the previous metrics.

$$\begin{aligned} E_c = \quad & w_d \cdot f(\Delta d_{cm}) + w_s \cdot f(\Delta s_{cm}) + w_N \cdot g(N_c) \\ & + w_{LET} \cdot g(LET_c) + w_{SNR} \cdot g(SNR_c) \end{aligned} \quad (2)$$

where w_d, w_s, w_N, w_{LET} and w_{SNR} are the weights of corresponding metrics which indicate their impact on E_c. Since the values of these metrics have different ranges, two normalization functions are proposed to extract a score between 0 and 1 from each metric separately, $f(z) = e^{-(\frac{z}{\sqrt{2}\sigma})^2}$ and $g(u) = \frac{1}{1+e^{\alpha(\mu_u - u)}}$, where σ is the speed and distance standard deviation of CNs from the mean, α and μ_u are constants. Then, E_c value ranges between 0 and 5, such that the higher the E_c the more the vehicle is qualified to be a CH.

3) Joining a Cluster: Once an UV vehicle in the cluster head selection step receives a *CHHello* message, it checks whether it is from a CN neighbor and whether it is from a more eligible vehicle. By fulfilling these two conditions, the vehicle unicasts a join request *JoinReq* message to the corresponding CH and changes its state to a CM. On receiving a *JoinReq* message, the CH in turn checks the consistency of the sender to be added to its CM Table (CMT). The CH also makes sure that the number of CMs does not exceed the maximum limit N_{max} before adding any vehicle.

4) Cluster Maintenance: After selecting the cluster heads and forming the clusters, a set of procedures are proposed to maintain a relatively stable topology inside the cluster.

a) SCH Selection: After the selection of a CH, this CH selects its representative in the cluster that we call the SCH. The selection is based on the eligibility E_c of each CM. A CM with the highest eligibility will be declared in the next $CHHello$ as the SCH.

b) All-Member-Interests-based Merging: The purpose of cluster merging is to combine two clusters to form a larger one. Cluster merging threatens the stability for the sake of decreasing the number of clusters. In order to benefit from merging without suffering from instability, cluster merging in our scheme is based on all CMs interests instead of CH's as in the existing works discussed in Section II. When two consistent CHs are moving within each other's range, the two CHs begin the cluster merging procedure. The merging conditions that must be checked by each CH are: First, all members of the current CH are consistent with the target CH. Second, the total number of members after merging is less than the maximum limit N_{max}. Third, the current CH is less eligible than the target CH. Whenever these conditions are met for a CH, it resigns from the CH role and sends *JoinReq* to be a CM in the new cluster.

5) Leaving a Cluster: From the CM perspective, the CM checks the consistency and updates its CHT upon receiving a *CHHello* message from its CH. If the CH becomes inconsistent with the CM or the cluster member missed a *CHHello* message from the CH, this indicates loosing the connection with the CH. Due to the presence of the SCH in our scheme, the CM does not have to leave the cluster unless the SCH is out of rage or inconsistent as well. From the CH perspective, the CH believes that a CM leaves the cluster, if the CH stops receiving the CM periodic *Hello* messages. Another reason is that the CM becomes inconsistent with the CH.

TABLE I
SIMULATION SCENARIOS

Scenario	Mobility Parameters				Lane Max. Speed
	Acceleration $[m/s^2]$	Deceleration $[m/s^2]$	Speed Factor	Speed Deviation	
S1	2.9	7.5	1.3	0.2	13.8-30 m/s
S2	1.85	0.9	1.13	0.11	30.5 m/s

TABLE II
NS3 SIMULATION PARAMETERS

Parameter	Value
NS3 version	3.26
Simulation time	100 s
R	300 m
MAC protocol	IEEE 802.11p
Propagation Model	Two-Ray Ground
T_{Hello}	0.1 s
ΔS_{th}	6 m/s
T_E	1.1 T_{Hello}
θ, α	$\pi/2$, 2
μ_{LET}, μ_{SNR}	20
$w_d, w_s, w_{N_c}, w_{LET}, w_{SNR}$	1

IV. PERFORMANCE EVALUATION

A. Simulation Environment and Scenarios

The simulation environment is a synthetic straight 5 km length highway. This highway has two moving directions with 6 lanes per direction, each lane is divided into a number of edges. Vehicles traffic was generated using Simulation of Urban Mobility (SUMO) [16]. Two vehicles types were used to generate two testing scenarios, illustrated in Table I. The first scenario S1 is a very dynamic traffic scenario in which the vehicles change their speed too much during simulation. This is because the successive highway edges have different maximum speeds, ranging randomly between 13.8-30 m/s, in addition to the high acceleration and deceleration of the vehicles. In contrast to S1, S2 is considered a relatively static, with high speed vehicles scenario. Our proposed clustering algorithm was implemented on Network Simulator NS3.26 [17] with the simulation parameters shown in Table II. The following metrics are evaluated: CH Life-Time (CHL), CM Life-Time (CML), Total Number of Clusters (TNC), and Control Packet Overhead (CPO).

B. Simulation Results

The performance of our proposed DHC algorithm is compared to the highly cited Threshold Based (TB) clustering algorithm [11] in terms of the previously mentioned metrics in four different densities and two simulation scenarios.

The average CH lifetime of the DHC and TB algorithms are illustrated in Fig. 3(a). For both algorithms, the lifetime of a CH starts from the moment at which the vehicle announces itself as a CH to the time it merges with another cluster. As shown in Fig. 3(a), increasing the vehicles density leads to a smaller CH duration as higher density means larger number of clusters, and thus, increased probability of merging. DHC CHL superior performance is attributed to its AMIM merging scheme that makes cluster merging less frequent, and the presence of SCH that reduces unnecessary re-clustering. DHC results in longer CHL, by up to 30s in S1, compared to TB.

Likewise, DHC is superior to TB in terms of the CML as depicted in Fig. 3(b) because the CMs connect with the SCH whenever the link with their CH is lost, to prevent leaving the cluster during transit situations. Figure 3(b) shows that the CML is smaller in dynamic scenarios, than in static ones.

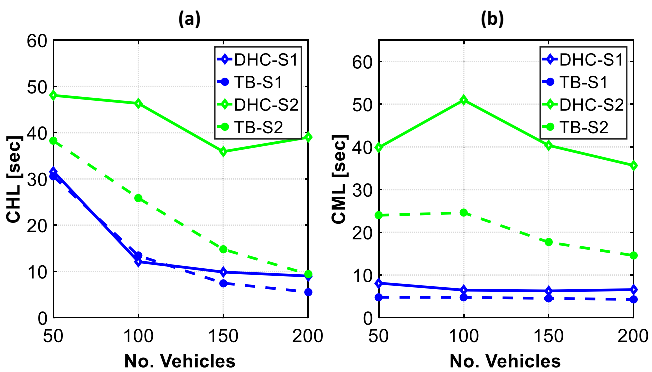

Fig. 3. Comparison between DHC and TB in terms of CHL and CML. (a) CH lifetime. (b) CM lifetime.

To compare how much the re-clustering occurs in both TB and DHC, the total number of formed clusters is plotted in Fig. 4(a). This number is higher for TB in all cases especially when frequent changing mobility patterns are considered. The frequent re-clustering encountered by TB causes the total number of formed clusters to dramatically increase compared to our stable DHC. Likewise is the CPO which is also directly influenced by the instability problem and unnecessary re-clustering, and thus, many control packets are to be sent. Even though both TB and DHC have the same maintenance overhead, cluster merging is more frequent in TB than in DCH. This causes the CPO of DHC to be less than that of TB as shown in Fig. 4(b).

V. Conclusions

In this paper, we have presented a mobility-based stable clustering approach that creates the minimum number of stable clusters in both urban and highway scenarios. Different vehicles mobility and link quality parameters, such as the relative position, speed, direction, popularity, SNR and link expiration time, are used for cluster head selection. The proposed DHC algorithm outperforms existing approaches in different traffic scenarios and vehicle densities, especially in dynamic mobility environments.

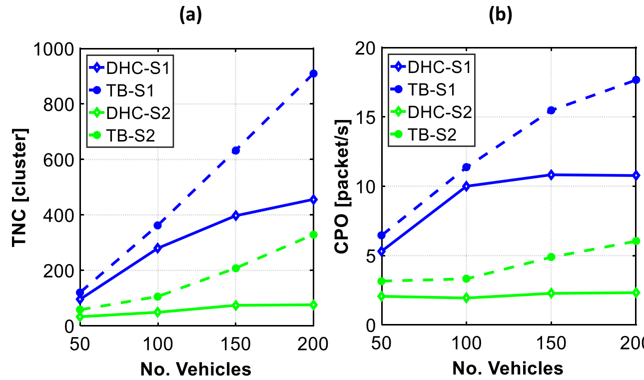

Fig. 4. CPO and TNC comparison between DHC and TB. (a) Total number of clusters. (b) Control packet overhead.

References

[1] L. Liu, C. Chen, T. Qiu, M. Zhang, S. Li, and B. Zhou, "A data dissemination scheme based on clustering and probabilistic broadcasting in VANETs," *Vehicular Communications*, vol. 13, pp. 78–88, 2018.

[2] A. Mehmood, A. Khanan, A. H. H. M. Mohamed, S. Mahfooz, H. Song, and S. Abdullah, "ANTSC: an intelligent naïve bayesian probabilistic estimation practice for traffic flow to form stable clustering in vanet," *IEEE Access*, vol. 6, pp. 4452–4461, 2018.

[3] C. Cooper, D. Franklin, M. Ros, F. Safaei, and M. Abolhasan, "A comparative survey of VANET clustering techniques," *IEEE Communications Surveys & Tutorials*, vol. 19, pp. 657–681, 2016.

[4] S. Ucar, S. C. Ergen, and O. Ozkasap, "Multihop-cluster-based IEEE 802.11p and LTE hybrid architecture for VANET safety message dissemination," *IEEE Trans. Veh. Technol.*, vol. 65, pp. 2621–2636, 2016.

[5] M. Ren, J. Zhang, L. Khoukhi, H. Labiod, and V. Vèque, "A unified framework of clustering approach in vehicular ad hoc networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 5, pp. 1401–1414, 2018.

[6] L. Zhao, Y. Li, C. Meng, C. Gong, and X. Tang, "A SVM based routing scheme in VANETs," in *IEEE International Symposium on Communications and Information Technologies (ISCIT)*, 2016.

[7] M. M. C. Morales, C. S. Hong, and Y.-C. Bang, "An adaptable mobility-aware clustering algorithm in vehicular networks," in *IEEE Asia-Pacific Network Operations and Management Symposium (APNOMS)*, 2011.

[8] A. Ahizoune and A. Hafid, "A new stability based clustering algorithm (SBCA) for VANETs," in *IEEE Conference on Local Computer Networks (LCN) Workshops,*, 2012.

[9] K. A. Hafeez, L. Zhao, Z. Liao, and B. N.-W. Ma, "A fuzzy-logic-based cluster head selection algorithm in VANETs," in *IEEE International Conference on Communications (ICC)*, 2012.

[10] G. V. Rossi, Z. Fan, W. H. Chin, and K. K. Leung, "Stable clustering for ad-hoc vehicle networking," in *IEEE Wireless Communications and Networking Conference (WCNC)*, 2017.

[11] Z. Y. Rawashdeh and S. M. Mahmud, "A novel algorithm to form stable clusters in vehicular ad hoc networks on highways," *EURASIP J Wirel Commun Netw.*, vol. 2012, no. 1, p. 15, 2012.

[12] C. Shea, B. Hassanabadi, and S. Valaee, "Mobility-based clustering in vanets using affinity propagation," in *IEEE Global telecommunications conference (GLOBECOM)*, 2009.

[13] R. Goonewardene, F. Ali, and E. Stipidis, "Robust mobility adaptive clustering scheme with support for geographic routing for vehicular ad hoc networks," *IET Intell. Transp. Syst.*, vol. 3, no. 2, pp. 148–158, 2009.

[14] E. Souza, I. Nikolaidis, and P. Gburzynski, "A new aggregate local mobility (ALM) clustering algorithm for VANETs," in *IEEE International Conference on Communications (ICC)*, 2010.

[15] R. Ghosh and S. Basagni, "Mitigating the impact of node mobility on ad hoc clustering," *Wireless Communications and Mobile Computing*, vol. 8, no. 3, pp. 295–308, 2008.

[16] *Simulation of Urban MObility, SUMO. [Online]. Available: https://sourceforge.net/projects/sumo/. Accessed: November, 2018.*

[17] *Network simulator, ns-3. [Online]. Available: http://www.nsnam.org/. Accessed: November, 2018.*