A Distributed D-hop Cluster Formation for VANET

Meysam Azizian, Soumaya Cherkaoui and Abdelhakim Senhaji Hafid†
Department of Electrical and Computer Engineering, Université de Sherbrooke, Canada
Email: {meysam.azizian, soumaya.cherkaoui}@usherbrooke.ca
†Department of Computer Science and Operations Research, University of Montreal, Canada
†Euromed University, Fes, Morocco
Email: ahafid@iro.umontreal.ca, a.senhaji@ueuromed.org

***Abstract*— A major challenge in vehicular ad-hoc networks (VANETs) is the ability to account for resource sharing and location management so that multicasting/routing functions and bandwidth reservations can be organized efficiently. By creating clusters of vehicles we are able to control resource sharing and management functions in VANETs that are highly dynamic. In this paper, a D-hop clustering algorithm, called DHCV is presented which organizes vehicles into non overlapping clusters which have adaptive sizes according to their respective mobility. The D-hop clustering algorithm creates clusters in such a way that each vehicle is at most D hops away from a cluster head. To construct multi-hop clusters, each vehicle chooses its cluster head based on relative mobility calculations within its D-hop neighbours. The algorithm can run at regular intervals or whenever the network formation changes. One of the features of this algorithm is tendency to re-elect the surviving cluster heads whenever the network structure changes. Extensive simulation results have been done under different scenarios to show the performance of our clustering algorithm.**

Keywords—clustering; vehicular ad-hoc networks; cluster stability; multi-hop; vehicular cloud; cloud formation

I. INTRODUCTION

VANETs are ad hoc networks that are established in order to enable vehicles to communicate with each other for specific purposes, such as safety, road traffic management and data sharing [1]. Dynamic and dense network topology characteristics in VANETs cause problems, such as congestion, rerouting and the hidden terminal problem. Clustering techniques have been proposed [4, 5] to make communication in VANETs more robust and scalable by grouping nodes in a geographical vicinity together to improve communication efficiency. Clustering can also be used for frequency reuse.

Most of existing clustering algorithms in VANETs are based on one-hop communication where cluster head (CH) and cluster members (CM) can communicate directly [1-3].Small coverage of one-hop clusters leads to an increase in the number of CHs. This can decrease routing efficiency and raise communication cost when communication beyond the cluster small range is necessary. Multi-hop clustering algorithms extend the communication coverage in the same cluster and reduce the number of CHs and subsequently the number of clusters.

To build multi-hop clusters in VANETs, we propose a scheme based on D-clustering [4]. Different from other D-cluster algorithms which are proposed for general wireless ad hoc networks [5, 6], our proposed scheme is designed to cater to specific characteristics of VANETs such as high mobility. D-cluster algorithms, in general, form clusters such that CHs form a D-hop dominating set where each node is at most D hops away from CH. Existing D-hop algorithms proposed for ad hoc networks construct clusters without regard to high mobility factors in the clustering process [5, 6].

In this paper, we propose a distributed D-hop clustering algorithm, called DHCV (D-HOP Clustering for VANETs) using relative mobility information to construct stable clusters for VANETs. In our algorithm we use speed and location differences of vehicles as metrics to model relative mobility in D-hop communication range. We suppose that vehicles use the WAVE standard [7] to periodically broadcast their speed and location, acquired through GPS, to vehicles in their one-hop vicinity. Clusters will be dynamically formed by vehicles in a distributed manner, based on this information. Due to mobility of vehicles, clusters may be reconstructed in a timely fashion, as vehicles move away from each other. However, we will show that, compared to other VANET multi-hop clustering algorithms, our clustering algorithm is able to form clusters with higher stability.

The rest of this paper is organized as follows. Section 2 presents related works. Section 3 describes our D-hop clustering algorithm. Section 4 evaluates, via simulations, the proposed algorithm. Finally, section 5 concludes the paper and suggests future works.

II. RELATED WORKS

Clustering has been extensively researched in general wireless ad-hoc networks. Vehicle movement, high topology changes, and availability of energy sources in VANETs make clustering algorithms proposed for other kinds of ad hoc networks such as sensor networks, not suitable to be used in VANETs. Conversely, different from general ad hoc networks, some characteristics of vehicles are helpful to design clustering algorithms [7]. For example, vehicle movement patterns are predictable and can be retrieved from road structure and drivers' behavior. Also, GPS can be used to retrieve the location of vehicles and digital maps can be helpful for tracking purposes.

978-1-4673-9814-5/16/$31.00 ©2016 IEEE

Two main approaches have been proposed for clustering in VANETs [8]. One is location service-dependent which uses location, speed, and movement direction for clustering. Another one is based on collective computable parameters, such as radio transmission, connectivity, vehicles density, etc. However, the vast majority of existing clustering algorithms form one-hop clusters. Examples are DCA [9], SBCA [1] and DCCT [3]. VANET One-Hop clusters have a smaller coverage range and many clusters may need to be constructed and re-constructed because of vehicle movements. Multi-hop clusters can achieve cluster stability, reduce maintenance cost, and increase routing efficiency. To the best of our knowledge, only five VANET multi-hop clustering algorithms have been proposed so far and are briefly described hereafter.

HCA [10] is a fast randomized clustering algorithm which collects connectivity information from neighboring vehicles. In HCA, Cluster size in is limited to 2 or less hops. In [11] Modified DMAC adapts DMAC [2] to VANET scenarios. DMAC does not take into account mobility. Modified DMAC avoids forming clusters with vehicles which are moving in opposite directions. Although this clustering technique showed efficiency in some scenarios, it does not make stable clusters. In [12], the authors proposed DMCNF, a distributed multi-hop clustering scheme for VANETs based on a neighborhood concept called "neighborhood follow". In DMCNF, clusters tends to be rather large, which may impact negatively the network performance inside a single cluster.

Zhang et al [13] introduced a multi-hop clustering solution which uses relative mobility between vehicles in multi-hop communication range as a metric. The main drawback of this algorithm is the network overhead generated, in terms of clustering control messages, which might degrade the network performance. In [14], a vehicular multi-hop algorithm for stable clustering (VMaSC) is presented which also uses relative mobility to form clusters. This algorithm, has the same drawback as the scheme in [13] but with better stability. In this paper, we compare the results of DHCV to those of VMaSC [14] in terms of cluster stability.

III. PROPOSED SCHEME

In this section, we describe DHCV clustering algorithm for VANETs. Using WAVE, vehicles periodically send their position information using beacon messages. When the algorithm is run by vehicles, clusters are formed such as a vehicle (node) is either CH or is at most D communication hops away from its CH. This is what we call a D-hop cluster. Our D-hop clustering will result in variable size clusters depending on vehicles' mobility.

To group vehicles into different clusters, each vehicle selects it's appropriate CH in at most D hops distance. A vehicle selects its CH by first choosing a vehicle in its one-hop communication range. Choosing here means selecting a next vehicle as a next-hop neighbor toward a CH. The chosen vehicle which is selected among neighboring vehicles, is the one which has the least relative mobility with the chooser vehicle. Then, the chooser vehicle chooses another vehicle in the 2nd hop which has the least relative mobility with the vehicle that has been chosen in the first hop. The vehicle continues this process for D hops to select its appropriate CH. The main idea of selecting vehicles one by one in each hop is to have precise details of vehicles in multi-hop communication range and decrease the exchange of messages in large scale VANETs to avoid congestion. In the last step, the vehicle selects its CH among the vehicles which it has chosen in D hops. The selected vehicle (CH) is the one with the highest degree among the chosen vehicles (see definition below).

For the purpose of the algorithm, we define the following terms:

- **Current node (CN)**: A node that tries to find and select a CH up to D-hops away.
- **Chosen Neighbor node i (CNN_i):** it is a direct neighbour selected by CNN_{i-1}. CNN_i is selected as the direct neighbor with least relative mobility to CNN_{i-1}. CNN_0 is CN.
- **Degree:** degree of connectivity of a CNNi.
- **Potential Cluster Head i (PCH_i):** it is the CNNj with the maximum degree among the set $\{CNN_j\}$:

$$PCH_i = \max_{degree(CNN_j)} \left(\{CNN_j\}_{j \le i} \right)$$

The algorithm runs for D-hops of information exchange. Each CN constructs two arrays of information, CNN_i and PCH_i, as shown in Figure 1. In the graph, the table values are node identities as identified in the graph; formed clusters are shown circled. The algorithm consists of 3 phases. Namely initialization, relativeMax and maintenance.

Initialization:

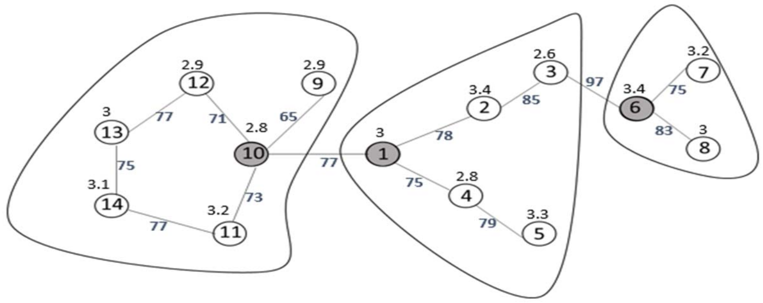
At the beginning, each node sets its own node id to be equal to its PCH_0 id. For example, node 3 at the initialization phase in Figure 1, selects its own id as the PCH_0 id.

RealtiveMax:

Each vehicle broadcasts a beacon message including its speed and location information to all of its 1-hop neighbors. Once a node has receive beacons from all neighboring nodes, it calculates its relative mobility with them and chooses the vehicle with the least relative mobility. The node also checks the chosen node's (CNN_1) degree of connectivity (degree). If CNN_1 has better degree, then it can be a potential CH; in this case, the node changes its PCH_0 id to the id of CNN_1 .Whenever the degree of CNN_i is equal to CNN_{i-1} and CNN_i has a lower speed compared to CNN_{i-1}, CNN_{i-1} will change its PCH id to the id of CNN_i, unless no change in PCH id is needed. As 2nd hop, CN selects the node (as CNN_2) which has the least relative mobility with the already selected 1st hop. The node compares its PCH's degree with the degree of the 2nd hop. If the degree of the 2nd node is higher (has better connectivity), then CN will change its PCH id to the id of the 2nd hop. This process continues for D hops. Each node keeps a logged entry of results of each hop and at the conclusion of the RelativeMax, the PCH nodes are elected CHs in the network.

There are certain scenarios where the following exceptions should be considered:

Exception 1: the algorithm generate CM that has no link to its desired CH without passing through another cluster. In this case, CM selects the 2nd PCH node from its array to be its CH.



Nodes	1		2		3		4		5		6		7		8		9		10		11		12		13		14	
	PCH	CNN	PCH	CNN	PCH	CNN	PCH	CNN	PCH	CNN	PCH	CNN	PCH	CNN	PCH	CNN	PCH	CNN	PCH	CNN	PCH	CNN	PCH	CNN	PCH	CNN	PCH	CNN
initialization	1	1	2	2	3	3	4	4	5	5	6	6	7	7	8	8	9	9	10	10	11	11	12	12	13	13	14	14
First hop	1	4	1	1	2	2	1	1	4	4	6	7	6	6	6	6	10	10	10	9	10	10	10	10	13	14	13	13
2nd hop	1	5	1	4	1	1	10	10	1	1			6	8	6	7	10	12			10	9	10	9	11	11	12	12
3rd hop			1	5	1	4	10	9	10	10							10	13							10	10	10	10
PCH Results	1		1		1		1		1		6		6		6		10		10		10		10		10		10	

Figure 1 D-hop formation in a network of 14 nodes (D=3).

If there is no connection to the 2nd PCH, the node selects CNN_1's CH, to be its CH. Nodes 4 and 5, in Figure 1 are examples for this exemption.

Exception 2: the algorithm generates CH which has already been selected as CM in another cluster. This means CH does not select itself as CH. In this case, CH and its connected CMs introduce themselves as CMs and select CH of the constructed cluster as their own CH. If the D-hop condition cannot be satisfied with switching CH to CM, CH remains as such and makes a cluster.

Exception 3: the algorithm select CH that has no CMs. In this case, the CH introduces itself as CM and selects CNN_1 which has the least relative mobility with it. If CNN_1 is CH, CM selects it as CH, unless it selects the CNN_1's CH as its own CH.

Figure 1 shows the clusters constructed by the proposed algorithm in a sample configuration of 14 nodes; three CHs are selected, namely nodes, 1, 6, and 10. In Figure 1, nodes represent vehicles, the number next to a node represents the normalized speed of the node, and the weight under a link represents the normalized relative mobility between the two end nodes of the link. As shown in the CNN arrays in Figure 1, node 3 chooses node 2 in the first hop since node 2 has the least relative mobility with node 3. Also, as shown in PCH array, node 3 selects node 2 as its own PCH, as node 2 has a lower speed compared to itself. Then, node 3 chooses node 1 as 2nd hop and selects node 1 as its own PCH, since node 1 has a higher degree compared to the degree of node 2. In the 3rd hop, node 3 chooses node 4 but does not change its PCH id as the degree of node 1 is higher than the degree of node 4.

Maintenance:

Vehicles execute the procedure of selecting CH at renewal intervals (RI). RI can be chosen empirically. The maintenance phase is processed between RIs to maintain cluster membership. Vehicles perform a verification task during maintenance for a verification interval (VI) [2]; in this interval CMs verify that connections with their corresponding CH are still there. If the connection is lost for a VI time, then CM selects its 2nd CH choice as its primary CH. Whenever the 2nd choice is not available, the vehicle choses a neighboring node which has the least relative mobility with it (CNN_1). When choosing CNN_1, the D hop limitation should also be considered. In the worst-case scenario, where no maintenance is possible, the vehicle runs the algorithm from the start (initialization phase). It is worth noting that only vehicles which have lost connections need to do re-runs of the clustering algorithm.

A. *Mobility metrics*

Mobility metrics can be used to characterize the mobility level of vehicles. We assume each vehicle knows its speed and position through GPS. Every vehicle broadcasts a beacon message with speed and location information to its neighboring nodes at every beacon interval.

Whenever a neighboring vehicle gets the beacon frame, it determines the relative mobility and chooses the node that has the least relative mobility with it.

In our algorithm, to calculate the relative mobility between vehicles, each node calculates the speed and location differences with all of its neighbors. For example, vehicle X calculates its location difference $\boldsymbol{D_{XY_t}}$ (see Equation 1) and speed difference $\boldsymbol{\overline{V}_{XY_t}}$ (see Equation 5) with each vehicle $\boldsymbol{Y}$, in its neighborhood $(\boldsymbol{X})$ **,** that is moving in the same direction. In all equations, t denotes the time instant when the algorithm runs.

Let $\boldsymbol{D_{XY_t}}$, denotes the distance between node X and node Y (one of the neighboring nodes of X).

$$D_{XY_t} = \left| G_{X_t} - G_{Y_t} \right|, \ where\ Y \in N\ (X) \qquad (1)$$

where $\boldsymbol{G_{X_t}}$ and $\boldsymbol{G_{Y_t}}$ represent the locations of node X and Y at time t acquired through GPS, respectively.

$\boldsymbol{D_{XYN_t}}$ is the normalized value of $\boldsymbol{D_{XY_t}}$:

$$D_{XYN_t} = \frac{D_{XY_t}}{CD} \qquad (2)$$

In Equation (2) we use a value, named Communication Distance (CD). Where CD is the maximum effective communication range (e.g. 300m).

$\overline{\boldsymbol{V}}_{\boldsymbol{X_t}}$ and $\overline{\boldsymbol{V}}_{\boldsymbol{Y_t}}$ differentiate nodes moving in different directions:

$$\overline{V}_{X_t} = V_{X_t} COS\theta, \tag{3}$$

$$\overline{V}_{Y_t} = V_{Y_t} COS\theta. \tag{4}$$

In Equations (3) and (4), $\boldsymbol{V_{X_t}}$(resp. V_{Y_t}) represents the speed of node X (resp. Y) at time t and $\boldsymbol{\theta}$ is the velocity vector angels between X and Y. $\boldsymbol{\theta}$ is zero and 180 degrees if they are moving in the same direction and opposite direction, respectively. $\overline{\boldsymbol{V}}_{\boldsymbol{XY_t}}$ is the speed difference between node X and node Y :

$$\overline{V}_{XY_t} = \left|\overline{V}_{X_t} - \overline{V}_{Y_t}\right|, where\ Y \in N(X) \tag{5}$$

$\overline{\boldsymbol{V}}_{\boldsymbol{XYN_t}}$ is the normalized value of $\overline{\boldsymbol{V}}_{\boldsymbol{XY_t}}$:

$$\overline{V}_{XYN_t} = \frac{\overline{V}_{XY_t}}{LSR} \tag{6}$$

In Equation (6), we use a value called Legal Speed Range (LSR) for normalization. LSR is the difference between maximum and minimum allowed speeds on the road. Vehicles can have LSR information through their integrated GPS.

After $\overline{\boldsymbol{V}}_{\boldsymbol{XYN_t}}$ and $\boldsymbol{D_{XYN_t}}$ computation, vehicle X calculates the relative mobility with its neighboring nodes as follows:

$$RM_{XY\,t} = \alpha\, D_{XYN_t} + \beta\, \overline{V}_{XYN_t}, where\ Y \in N \tag{7}$$

where $\boldsymbol{RM_{XY\,t}}$ is the relative mobility between X and Y. α, β values are defined as distance and speed factors, respectively. These values are weights to control efficiency of the metrics.

Then vehicle X chooses the vehicle (denoted by $\boldsymbol{C_X}$) that has the least relative mobility (minimum value) with it. This value has to be lower than a threshold, $\boldsymbol{Th_m}$ of relative mobility. If this threshold is not satisfied, then no chosen node is selected as in Eq. 8.

$$C_X = arg\ min\left(RM_{XY\,t}\right) \wedge RM_{XY\,t} < Th_m \tag{8}$$

$$\alpha + \beta = 1$$

$$Th_m = (a_{mob} + k\delta_{mob})/1000$$

where $\boldsymbol{a_{mob}}$ is the average value of the speed information that node X received, from its neighboring nodes, and $\boldsymbol{\delta_{mob}}$ is the corresponding standard deviation. K is constant and can be changed based on the desired cluster stability.

IV. NUMERICAL RESULTS

In this section, we present the performance evaluation and simulation results for the proposed D-hop clustering. The mobility model used in the simulation is the freeway mobility model generated by SUMO [15]. We consider a vehicular network of 50 vehicles with varying speeds. The simulation scenario uses four highway lanes. The other simulation parameters are illustrated in Table 1. A safety distance is considered so that a vehicle does not exceed the speed of the vehicle in front of it. The overtaking choice of the vehicle is determined by considering the speed limit, distance to the vehicle in front, density of vehicles and acceleration–deceleration characteristics of the vehicle. Moreover, α and β selection is based on the network under consideration characteristics such as transmission range, legal speed range and average velocity. Our proposed D-hop clustering scheme is compared to VMaSC [14, 16]. VMaSC is a novel multi-hop clustering algorithm which uses relative mobility to construct clusters. The relative mobility in VMaSC is computed as the average of relative speed of all vehicles in the same direction.

To evaluate our D-hop clustering scheme, we consider the following metrics: (1) CM duration: is the period from when a vehicle chooses to be CM of a cluster for which CH has been chosen; (2) CH duration: is the period from when a vehicle gets selected to be CH to when the vehicle leaves its CH role; (3) CH change number: is the number of vehicles whose role changes from CH to CM during one simulation experiment; (4) Number of clusters: is the number of constructed clusters when the clustering algorithm terminates.

In Figure, Figure3 and Figure 4 bellow, the results for DHCV are shown in the plots named "X-hop D-hop" where X takes values of 2, 3, or 4. Results of VMaSC hold that name on the corresponding plots.

Table 1 Simulation parameters

Parameters	Value
Simulation Time	300 s
Number of vehicle	50
α	0.9
β	0.1
k	2
VI	10 s
RI	150 s
Transmission Range	300 m

A. *Cluster head duration*

Figure 2 shows the variation of the average CH duration with velocity for different values of D (i.e. D=2, 3 and 4). This variation is one of the metrics to demonstrate the stability of the clusters. We observer that, the average CH duration increases with the maximum number of hops (D). This can be explained by the fact that CH will have a larger coverage area as D increases; this coverage area leaves flexibility for CMs to move without leaving the cluster. For example, when D increase from 2 to 4, the average CH duration increases by about 20%.

We also observe that the average CH duration decreases with the increases in maximum speed of vehicles. When vehicles move faster, the network topology becomes more dynamic; this makes it difficult for CHs to keep stable connections within their cluster. If the condition for staying CH is no longer satisfied, CH becomes CM or CH of another cluster. CHs in VMaSC are vehicles which have the least average speed with their neighbors in the predefined number of hops. This condition may lead CHs in VMaSC to have weak connectivity with corresponding CMs in their cluster because VMaSC uses the notion of aggregated mobility to choose CHs. Nevertheless, with DHCV algorithm, each CM tries to establish the best individual connection links to its desired CH. Therefore, as shown in the figure, DHCV clustering algorithm outperforms VMaSC under different mobility conditions.

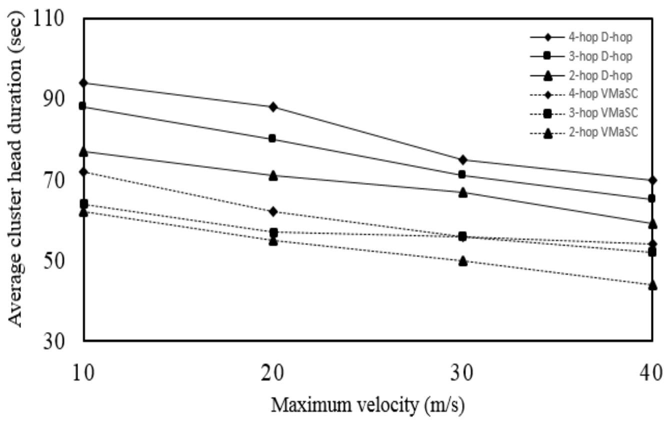


Figure 2 Average cluster head duration.

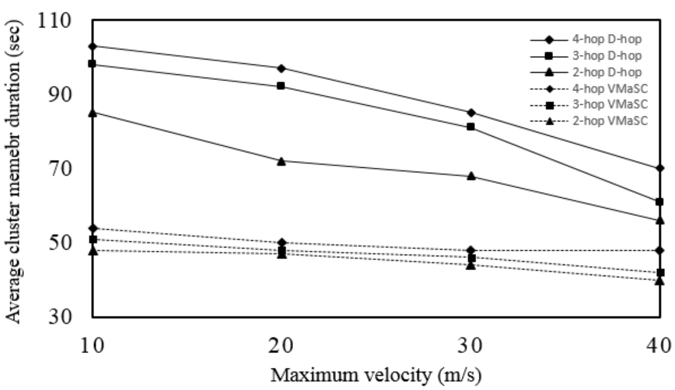


Figure 3 Average cluster member duration.

B. Cluster member duartion

The average CM duration can also demonstrate the stability of the constructed clusters. The average CM duration of DHCV is shown in Figure 3. We observe that the average CM duration increases as the maximum hop number increases. This can be explained by the fact that CH will have a larger coverage area as D increases; this coverage area leaves flexibility for CMs to move without leaving the cluster.

With VMaSC, the average cluster member duration is lower than with DHCV. CMs in VMaSC select their desired CHs from the ordered list of CHs based on the average relative speed. CHs are advertising themselves. These advertising CHs have the least relative speed with other neighboring vehicles which are in their predefined allowed hop distance. However, vehicles with the least average relative speed might not be CHs which have the least relative mobility with their assigned CMs. This explains why DHCV has a better performance. In DHCV, CMs select CHs which have the least relative mobility with them.

We also notice that the average CM duration decreases when the maximum speed of vehicles increases. As described for the average CH duration, vehicles change their assigned clusters faster whenever they have higher mobility. If CMs do not satisfy the condition for being CM, then they follow the maintenance procedure.

C. Cluster head change number

CH change number also determines the stability of the constructed clusters. A lower CH change number demonstrates a more stable cluster structure. The average CH change number of DHCV is shown in Figure 4. CH change number increases as the maximum allowed number of hops decreases because CMs have less flexibility moving while still belonging to the same cluster. We also notice that the CH change number increases with maximum velocity of vehicles. This is because of the frequent role changes among members in clusters with high speed vehicles node.

As shown in the Figure, CH change number with VMaSC is higher than with DHCV algorithm. Then reason is similar to the one described before; with DHCV D-hop algorithm, the relative mobility between CMs in a cluster is individually chosen to be the best when choosing CH and forming a cluster. Moreover, D-hop considers the location differences along with speed differences in relative mobility for selecting the chosen nodes. This feature increases the cluster life time and decrease the cluster head change number.

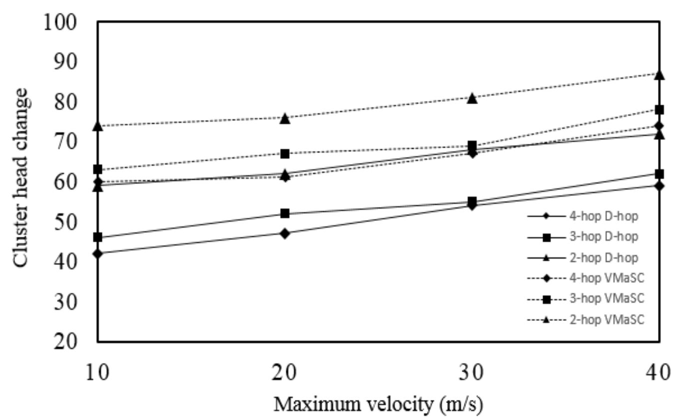


Figure 4 Cluster head change number.

D. Number of clusters

As shown in figure 5, the number of constructed clusters decreases as the maximum allowed hops (D) increases.

The number of constructed clusters with VMaSC is lower than with DHCV. In DHCV, each CM selects its desired CH with the minimum relative mobility based on its neighbors' best relative mobility. Selected CHs in D-hop can be even in direct communication range of each other. In VMaSC, CHs cannot be in the communication range of each other and need rather to be

separated by at least D hops. Therefore, the number of selected CHs (possibilities of clusters) in DHCV is higher in comparison to VMaSC, which results in a higher number of clusters.
Although as mentioned before, larger clusters can, in general, improve cluster stability, the higher cluster size in VMaSC is rather a result from the limited possible choices of CHs, rather than from choices related to better cluster stability (i.e. mobility metrics factors). In comparison, DHCV forms in the scenarios simulated, smaller, more stable clusters based solely on mobility factors as defined with its metrics. In fact, DHCV can construct variable size clusters based on vehicles mobility. Clusters are more stable in DHCV (as shown in Figures 2-4) as it construct variable-size clusters of up to D-hops.

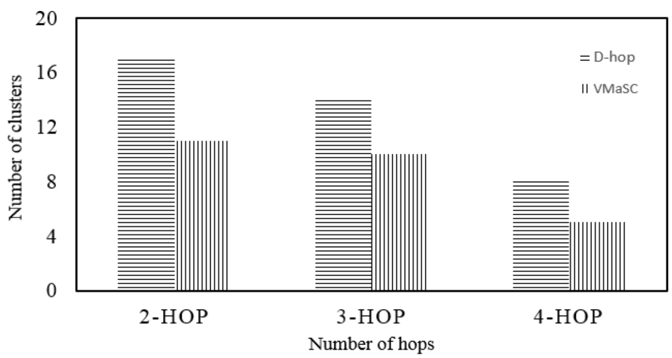


Figure 5 Number of clusters.

V. CONCLUTION

In this paper, a D-hop clustering algorithm is presented which constructs variable size clusters depending of vehicles mobility and distribution on the road. This algorithm organizes vehicles into D-hop non overlapping clusters according to their relative mobility. Compared to other multi-hop clustering algorithms, the proposed algorithm constructs more stable clusters in terms of cluster membership duration and cluster head changes. This will allow for a better management and scheduling of wireless networking resources among vehicles.

VI. BIBLIOGRAPHY

[1] A.Ahizoune and A. Hafid, "A new Stability Based Clustering Algorithm (SBCA) for VANETs," in *IEEE 37th Conference on Local Computer Networks Workshops (LCN Workshops)*, Clearwater, FL, 2012.

[2] S. Basagni, "Distributed clustering for ad-hoc networks," in *Fourth International Symposium on Parallel Architectures, Algorithms, and Networks*, Perth/Fremantle, WA, 1999.

[3] S.Khakpour, K. El-Khatib and R. Pazzi, "A Distributed Clustering Algorithm for Target Tracking in Vehicular Ad-hoc Networks," in *13 Proceedings of the Third ACM International Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications*, barcelona, spain, 2013.

[4] X. Gao, W. Wu, X. Zhang and X. Li, "A constant–factor approximation for d–hop connected dominating sets in unit disk graph," *International Journal of Sensor Networks,* vol. 12, no. 3, pp. 125-133, 2012.

[5] A. Amis, R. Prakesh, T. Vuong and D. Huynh, "Max-Min D-Cluster Formation in Wireless Ad Hoc Networks," in *INFOCOM 2000*, Tel Aviv, 2000.

[6] M. Rieck, S. Pai and S. Dhar, "Distributed Routing Algorithms for Wireless Ad Hoc Networks using d-Hop Connected d-Hop Dominating Sets," *Computer Networks and ISDN Systems,* vol. 47, no. 6, pp. 785-799, 2005.

[7] M.Azizian, E. Ndih and S. Cherkaoui, "Improved Multi-Channel Operation for Safety Messages Dissemination in Vehicular Networks," in *Divanet14*, Montreal, 2014.

[8] s.Vodopivec, J. Bester and A. Kos, "A survey on Clustering Algorithms for Vehicular Ad-Hoc Networks," in *35th International Conference on Telecommunications and Signal Processing*, Prague, 2012.

[9] W. Fan, Y. Shi, S. Chen and L. Zou, "A Mobility Metrics Based Dynamic Clustering Algorithm for VANETs," in *ICCTA*, Beijing, 2011.

[10] E. Dror, C. Avin and Z. Lotker, "Fast Randomized Algorithm for Hierarchical Clustering in Vehicular Ad-Hoc Networks," in *Ad Hoc Networking Workshop (Med-Hoc-Net), 2011 The 10th IFIP Annual Mediterranean*, Favignana Island, Sicily, 2011.

[11] G. Wolny, "Modified DMAC Clustering Algorithm for VANETs," in *3rd International Conference on Systems and Networks Communications*, Sliema, 2008.

[12] Y. Chen, M. Fang, S. Shi, W. Guo and X. Zheng, "Distributed multi-hop clustering algoritm for VANETs based on neighborhood follow," *journal on wireless communications and networking,* vol. 2015, no. 98, pp. 1-12, 2015.

[13] Z. Zhang, A. Boukereche and R. Pazzi, "A novel multi hop clustering scheme for vehicular ad-hoc networks," in *DIVANET*, Miami, 2011.

[14] S. Ucar, S. Ergen and O. Ozkasap, " VMaSC: Vehicular Multi-hop Algorithm for Stable Clustering in Vehicular Ad Hoc Networks," in *Wireless Communications and Networking Conference (WCNC)*, Shanghai, 2013.

[15] "Simulation of urban mobility," [Online]. Available: http;//sumo.sourceforge.net.

[16] S. Ucar, S. Ergen and O. Ozkasap, "Multi-Hop Cluster based IEEE 802.11p and LTE Hybrid Architecture for VANET Safety Message Dissemination," *IEEE Transactions on Vehicular Technology,* vol. pp, no. 99, pp. 1-17, 2015.

[17] H.Su and X. Zhang, "Clustering-Based Multichannel MAC Protocols for QoS Provisionings Over Vehicular Ad Hoc Networks," *IEEE Transactions on Vehicular Technology,* vol. 56, no. 6, pp. 3309 - 3323, 2007.