# VMaSC: Vehicular Multi-hop algorithm for Stable Clustering in Vehicular Ad Hoc Networks

Seyhan Ucar*, Sinem Coleri Ergen† and Oznur Ozkasap*
Department of Computer Engineering*
Department of Electrical and Electronics Engineering†
Koc University, Istanbul, Turkey
sucar@ku.edu.tr, sergen@ku.edu.tr, oozkasap@ku.edu.tr

*Abstract*—**Clustering is an effective mechanism to handle the fast changes in the topology of vehicular ad hoc networks (VANET) by using local coordination. Constructing stable clusters by determining the vehicles sharing similar mobility pattern is essential in reducing the overhead of clustering algorithms. In this paper, we introduce VMaSC: Vehicular Multi-hop algorithm for Stable Clustering. VMaSC is a novel clustering technique based on choosing the node with the least mobility calculated as a function of the speed difference between neighboring nodes as the cluster head through multiple hops. Extensive simulation experiments performed using ns-3 with the vehicle mobility input from the Simulation of Urban Mobility (SUMO) demonstrate that novel metric used in the evaluation of the least mobile node and multi-hop clustering increases cluster head duration by 25% while decreasing the number of cluster head changes by 10%.**

## I. INTRODUCTION

Vehicular ad hoc networks (VANET) enable numerous applications including safety message dissemination [1], [2] dynamic route discovery [3], gaming and entertainment [4]. The strict requirements for the delay and reliability of the message transfer for such safety dissemination and entertainment applications makes the random access protocols such IEEE 802.11p [5] unsuitable for VANET. On the other hand, the mere usage of cellular networks such as Long Term Evolution (LTE) [6] is costly. This has lead to the investigation of heterogeneous architectures based on clustering where IEEE 802.11p random access protocol is used within each cluster whereas LTE is used for the communication among cluster head nodes. Stable and efficient cluster formation with minimum number of cluster heads is essential in such heterogeneous architectures to minimize the overhead of fast topology changes and the amount of information transfer among cluster heads through cellular network.

The metrics used in determining the cluster head in the MANET literature are; node unique id where lowest-id is elected as cluster head [7]; received signal strength where mobility is estimated by comparing received power of consecutive messages and less mobile one is elected as cluster head [8]; enhancement of lowest id where re-clustering is invoked in only two cases; when two cluster heads move into the reach range of each other and when a mobile node cannot access any cluster head [9], [10]; node's movement, where node's placement, which is greater than the predefined threshold, used and mobile node with less displacement becomes a head [11]; without any metric, where mobile node becomes cluster head when it has something to send [12]. In particular, these proposed metrics and algorithms are not suitable for VANET because [8] and [11] are only feasible and effective with group mobility behaviour and their performance may be degraded in VANET where mobile node moves randomly with high speed and changes speed time to time. Another reason is the stationary assumption where mobile nodes are assumed to be static in the cluster formation [9], [10], [12] which contradicts with highly mobile characteristics of VANETs.

Investigation of the clustering mechanisms in VANET on the other hand focuses on one-hop clustering algorithms that use metrics such as similarity values, where a function of the distance between the position of the nodes and its neighbours are used [13]; region of vehicles, where road is split up into regions and first entered vehicle among same direction is elected as cluster head [14]; direction, where same direction vehicles are grouped together and vehicle advertises itself as a cluster head if it cannot receive invitation message from another cluster head within predefined amount of time [15]. Overall, deficiencies of these metrics are; they all form one-hop clusters where only direct communication is allowed and they do not aim to provide stability of extracting vehicle mobility in a highly dynamic environment for multi-hop clusters. Multi-hop clustering algorithm proposed for VANET [16] on the other hand, uses the changes in the packet delivery delay to calculate the relative mobility among the nodes. However, calculating packet delivery delay requires very accurate synchronization among the vehicles, which is not feasible for such dynamic networks.

The goal of this study is to develop an algorithm to construct stable multi-hop clusters with minimum number of cluster heads in VANET. The original contributions of this paper are three. First, we propose a novel mobility metric, that is periodically exchanged and used for similarity calculation among vehicles. Second, our work envisions multi-hop clustering with stable mobility metric in highly dynamic scenario. Third, to the best of our knowledge, the

978-1-4673-5939-9/13/$31.00 ©2013 IEEE

proposed approach VMaSC is the first work to simulate multi-hop clustering under realistic vehicle mobility which is generated by realistic mobility generator SUMO [17].

The rest of the paper is organized as follows. Section II describes the proposed multi-hop stable clustering algorithm. Section III presents the simulation results. Finally, main results are summarized and future work is given in Section IV.

## II. SYSTEM MODEL

The nodes in VANET aim to form clusters such that each has one cluster head and all nodes in a cluster can communicate with the cluster heads in a number of hops that is less than a maximum pre-determined value. Figure 1 shows three example clusters, namely 1-hop, 2-hop and 3-hop, where in each case middle vehicle is the cluster head (CH), and vehicles that are n-hop far away, are n-hop cluster members (CM). The cluster formation algorithm should be designed with the goals of minimizing the number of cluster heads in the network to decrease the cost of communication over cellular network, maximizing the duration of cluster head and cluster member to provide the stability and minimizing the overhead of forming the clusters. In this section, we describe the states of the vehicles, the algorithm for cluster formation and maintenance, and multi-hop clustering mechanism. The notation used is presented in Table I.

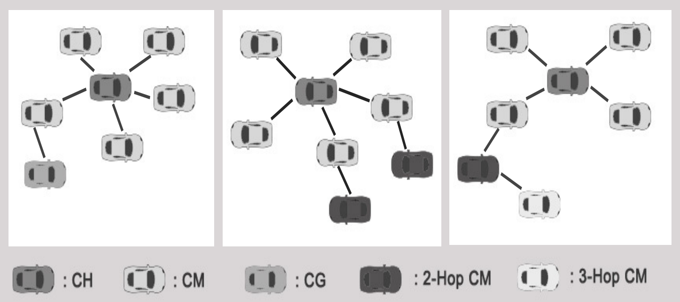

Fig. 1. Clustered network topology.

### A. States of Vehicles

Each vehicle can operate under one of the five states as described below.

- INITIAL is the starting state of the vehicle. Vehicles stay in this state and start to receive and send HELLO_PACKETs with clustering related attributes.
- STATE_ELECTION is the state where the vehicle makes decision about the next state based on state election algorithm (Algorithm-1) by using LOCAL_KNOW which is constructed upon reception of packets.
- CLUSTER_HEAD is the state of the vehicle which is less mobile with respect to its neighbours.
- CLUSTER_MEMBER is the state where the vehicle is connected to a constructed cluster.
- CLUSTER_GUEST is the state which is enabled only in one-hop scenarios and used for preventing system from unnecessary cluster head election in case when a vehicle cannot hear head related message, it declares itself as new cluster head. Vehicle in this state is regarded as a cluster member which accesses to cluster with the help of a cluster member.

TABLE I
NOTATION

| Notation | Description |
|---|---|
| CH | Cluster Head |
| CM | Cluster Member |
| CG | Cluster Guest |
| $V_{timer}$ | Vehicle's Timer |
| $V_{state}$ | Vehicle's Current State |
| $AVGREL_{speed}$ | Vehicle's Average Relative Speed |
| $MEMBER_{ch}$ | CH's connected member counter |
| $GUEST_{cm}$ | CM's connected guest counter |
| CH_ADV | CH's Advertisement Packet |
| MAX_HOP | Max. Hop Between CH and CM |
| CLUSTER_INFO | Constructed Cluster Information |
| MERGE_REQ | CH's Merge Request |
| LOCAL_KNOW | Vehicle's Local Knowledge Base |
| JOIN_REQ | Vehicle's Join Request Packet |
| JOIN_RESP | Join Response for Vehicle's Join Request |
| HELLO_PACKET | Vehicle's Periodic Hello Packet |

### B. Cluster Formation and Maintenance

Upon collecting the clustering related metrics; direction, current state, current speed, current hop counter, $AVGREL_{speed}$, MAX_HOP, connected cluster head id, $MEMBER_{ch}$ and $GUEST_{cm}$ in INITIAL, LOCAL_KNOW is updated. When the vehicle timer is expired, vehicles change state to STATE_ELECTION and clustering process is triggered. Via using LOCAL_KNOW, $AVGREL_{speed}$ is calculated as follows: vehicle first checks the LOCAL_KNOW for vehicles which are in the same direction. The reason for checking only same direction vehicles is to maximize the duration of the cluster heads. The relative mobility of the vehicle is then calculated by finding the average of the relative speed of all the same direction neighbours as

$$AVGREL_{speed} = \frac{\sum_{j=1}^{n} |S_{current}(t) - S_j(t))|}{n} \quad (1)$$

where $n$ is the number of same direction neighbours, $current$ is the index of the vehicle evaluating the relative mobility, $S_j(t)$ is the speed of the $j$-th same direction neighbour.

In STATE_ELECTION, the decision to become cluster head, cluster member and cluster guest is made as described in Algorithm-1. Since the main goal of clustering scheme is electing minimum number of cluster heads, Algorithm-1 first tries to set up a connection between existing cluster heads (Lines $3-11$). Via using LOCAL_KNOW, vehicle checks CH existence and its $MEMBER_{ch}$. After CH control, CHs are ordered based on $AVGREL_{speed}$ and comparison of $AVGREL_{speed}$ is done between CH and current vehicle (Line $5-6$). To extend the CM lifetime, CH whose relative mobility resembles current vehicle the most is elected from

**Algorithm 1** State Election Algorithm

```
1: Start $V_{timer}$;
2: while $V_{timer}$ is not expired do
3:   if LOCAL_KNOW contains CH then
4:     for each CH in LOCAL_KNOW do
5:       Control $AVGREL_{speed}$ and $MEMBER_{ch}$;
6:       if $AVGREL_{speed}$ and $MEMBER_{ch}$ are satisfied then
7:         Send JOIN_REQ and set $V_{timer}$ for reply;
8:         Wait for JOIN_RESP;
9:         if JOIN_RESP is received then
10:          Connect to CH;
11:          Change state to CLUSTER_MEMBER;
12:        else
13:          Set $V_{timer}$;
14:          Change state to STATE_ELECTION;
15:  if MAX_HOP $\equiv 1$ then
16:    if no CH found in LOCAL_KNOW then
17:      if LOCAL_KNOW contains CM then
18:        for each CM in LOCAL_KNOW do
19:          Control $AVGREL_{speed}$, $GUEST_{cm}$ and MAX_HOP;
20:          if $AVGREL_{speed}$, $GUEST_{cm}$ and MAX_HOP are satisfied then
21:            Send JOIN_REQ packet and set a $V_{timer}$;
22:            Wait for JOIN_RESP;
23:            if JOIN_RESP is received then
24:              Connect to CM;
25:              Change state to CLUSTER_GUEST;
26:            else
27:              Set $V_{timer}$;
28:              Change state to STATE_ELECTION;
29:  if MAX_HOP $\geq 2$ then
30:    Apply MultiHopClustering();
31:  if $V_{state}$ is not determined then
32:    if $AVGREL_{speed}$ is smallest in LOCAL_KNOW then
33:      Broadcast CH_ADV packet;
34:      Change state to CLUSTER_HEAD;
35:    else
36:      Set $V_{timer}$, wait for CH_ADV_PACKET;
37: if $V_{timer}$ is expired and CH_ADV is not received then
38:   Set $V_{timer}$;
39:   Change state to STATE_ELECTION;
40:   Trigger cluster forming process again;
```

ordered list and vehicle sends JOIN_REQ packet to inform the CH about the connection request (Line 7). If vehicle receives JOIN_RESP from cluster head, vehicle changes state to CLUSTER_MEMBER (Lines $10-11$). Response waiting is controlled via timer where if vehicle does not receive JOIN_RESP in given amount of time, it sets the timer and waits in the STATE_ELECTION (Lines $13-14$) and apply clustering process again.

If clustering scheme is 1-hop, where the MAX_HOP is 1 (Line 15), second step of Algorithm-1 tries to set up a connection with cluster member to be a cluster guest (Lines $16-26$). To prevent system from unnecessary cluster head in 1-hop, cluster guest state is initiated. Like in the first step, vehicles check the LOCAL_KNOW for CM and control GUEST$cm$, MAX_HOP and form ordered list by comparing $AVGREL_{speed}$ (Line 19). If GUEST$cm$ and MAX_HOP are smaller than the predefined threshold (Line 20) then via using ordered list comparison results, it tries to find the most similar one and send JOIN_REQ to that member (Line 21). After that, vehicle waits for JOIN_RESP for becoming a cluster guest. Until the timer expiration, if no response packet is received, it sets the timer again and stays in the STATE_ELECTION (Line 28).

Next step of Algorithm-1 differs depending on MAX_HOP between CH and CM. If the hop number is 1 which means 1-hop clustering is in progress, next step is cluster head election.

The cluster head election is based on calculated relative mobility with respect to its neighbours (Lines $31-36$). We believe that selecting less mobile vehicle in regard to its neighbour can extend the life time of cluster. Therefore, vehicles which has the smallest $AVGREL_{speed}$ are elected as cluster head. Elected cluster heads advertise themselves via broadcasting CH_ADV packets (Line 33). Other vehicles, which are in STATE_ELECTION, waits for CH_ADV packets and if advertisement packet is received, it follows the procedures JOIN_REQ and JOIN_RESP to get authorization from CH.

In cluster maintenance part, timers are used for controlling the connections between CH and CM. After state decision, if CH does not get any packet from connected members in predefined amount of time, it assumes that member vehicles are lost. CM controls the connection like CH where if it does not receive any packets from connected CH in given amount of time, it changes state to STATE_ELECTION for applying cluster forming process again.

If MAX_HOP is greater than or equal to 2 then constructed clusters are in multi-hop. The main logic behind multi-hop clustering is re-broadcasting which is controlled by one of HELLO_PACKET attributes, current hop counter, in order to prevent system from flooding. Vehicles which receive HELLO_PACKET first increase the current hop counter by one and compare it with MAX_HOP. If current hop counter is less than MAX_HOP, vehicle attaches its id, $AVGREL_{speed}$ and current state into packet as a sender information and rebroadcasts it. Via applying the hop counter approaches, vehicles in MAX_HOP distance are reached.

After reaching MAX_HOP distance vehicles, Algorithm-1 is executed as follows. Vehicles again attempt to connect to existing cluster heads and try to use the CH as much as possible (Lines $3-11$). Hence CLUSTER_GUEST is disabled in multi-hop clustering, if no CH is found then it goes to multi-hop clustering part where first trial takes aim at cluster members (Line 30). Like in the previous steps, the comparison of $AVGREL_{speed}$ is done and average relative speed based ordered list is formed. By using the ordered list, vehicles try to find the suitable 1-hop CM whose mobility is the most similar to itself and apply JOIN_REQ and JOIN_RESP procedure. If no 1-hop cluster members found that satisfies the most similar mobility conditions, next step is controlling the MAX_HOP and trying to find a vehicle which is not more than MAX_HOP distance from CH. If no vehicles found then a new cluster is constructed by selecting new cluster head (Lines $31 - 36$).

## III. SIMULATION RESULTS

We implemented our algorithm VMaSC on Network Simulator - ns3 (Release 3.13) [18] and used the topology of the network generated by SUMO [17]. Extensive simulations are performed and analysis results are presented in this section. The acceleration and overtaking decision of the vehicles are determined by using the distance to the leading vehicle, travelling speed, dimension of vehicles and profile of acceleration-deceleration.

Our scenarios consist of a two lane and two way road which is used to simulate the microscopic mobility of vehicles. For each scenario, simulation runs 600 seconds, and the clustering process starts at 300 second where all vehicles are on the road. Our proposed scheme multi-hop clustering VMaSC is compared with N-hop clustering where relative mobility is computed based on the variation of the packet delay of two consecutive messages [16].

General simulation parameters are illustrated in Table II. The performance of the clustering mechanism is evaluated by using the metrics of cluster head duration, cluster member duration and cluster head change.

TABLE II
SIMULATION PARAMETERS

| Parameters | Value |
|---|---|
| Simulation Time | 300 s |
| Area range | 1000 m * 1000 m |
| Maximum Velocity | 10 - 35 m/s |
| Number Of Vehicles | 100 |
| Transmission Range | 100-300 m |
| Max. Head Member Number | 5 |
| Max. Member Guest Number | 1 |
| HELLO_PACKET period | 200 ms |
| $V_{timer}$ value | 2 s |

### A. Cluster Head Duration

Cluster head duration is the time period from when vehicle changes state to CLUSTER_HEAD to when vehicle leaves this state and goes to another state (e.g STATE_ELECTION). Average cluster head duration is computed by dividing total cluster head duration into total number of state changes from CLUSTER_HEAD to another state. The average cluster head durations under different range and velocity scenarios are given in Figure 2. The effect of maximum hops in clustering process is taken into account by varying the hop numbers as MAX_HOP = 1, 2 and 3.

Fig.2 shows the comparative results for different transmission ranges. The average cluster head duration decreases as the vehicle velocity increases. This is because when the vehicle velocity increases, the topology of network becomes more dynamic. The effect of transmission range is observed such that when the transmission range increases, the average head duration also increases. This can be explained as in large transmission range, CH can find at least a member to serve so when CH's timer expires, CH continues to reside in CLUSTER_HEAD. Other metric that significantly affects the average head duration is MAX_HOP to the cluster head. Average cluster head duration increases as MAX_HOP increases. From numerical results, it can be said that the average cluster head duration in multi-hop scenarios is increased by 25% compared to 1-hop scenarios.

On the other hand, in multi-hop scenarios, when MAX_HOP and transmission range get larger, cluster head duration decreases. This can be explained by increase in cluster head changes caused by re-broadcasting and packet collision. Vehicles in multi-hop distance cannot hear head related packets and advertise themselves as new cluster heads. However, after some time vehicles either hear another cluster head or their timers are expired and change state to STATE_ELECTION which increases total cluster head change and decreases the average cluster head duration. The results show that our approach VMaSC outperforms N-hop clustering where the parameter packet delay is used for cluster head election. To extend the lifetime of cluster heads, not only cluster heads but also members must be elected based on criteria which enables head-member pair to have strong connectivity. In N-hop clustering, more head changes occur in comparison to our system VMaSC, thus the average cluster head duration decreases.

### B. Cluster Member Duration

Cluster member duration is defined as the time interval from joining specified cluster as CM in CLUSTER_MEMBER state to leaving the connected cluster by changing the state. By dividing the total cluster members into total cluster member changes, average cluster member duration is calculated.

As shown in Fig.3, average cluster member duration increases as the vehicle velocity increases. This can be explained by state changes from STATE_ELECTION to CLUSTER_MEMBER. Due to high dynamic network, vehicles cannot hear head related packets and advertise themselves as CH. However, after some time either vehicles hear another more suitable cluster head or they do not find any CM to serve. When the timer is expired, vehicles go to STATE_ELECTION and try to connect existing CH and CM. Eventually, vehicles either become CM or new cluster is constructed where in both cases total cluster member duration is increased. Another metric that plays role on member duration is maximum hops between CH and CM. When the MAX_HOPs increase, average member duration also increases. Vehicles connect to existing cluster by controlling the allowable MAX_HOPs and become a member in multi-hop distance. However, in N-hop clustering member election is based on cluster information

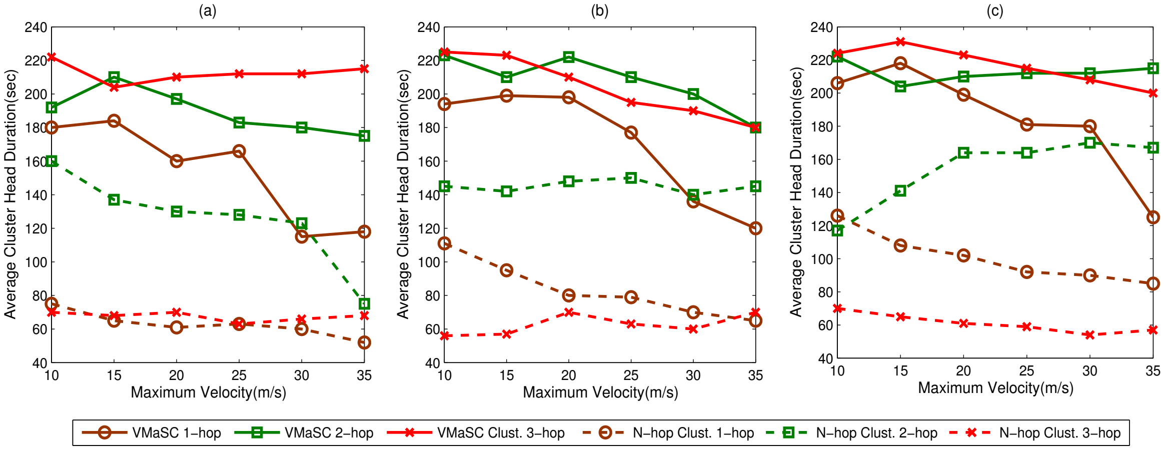

Fig. 2. Average cluster head duration, for transmission range (a) 100m (b) 200m (c) 300m

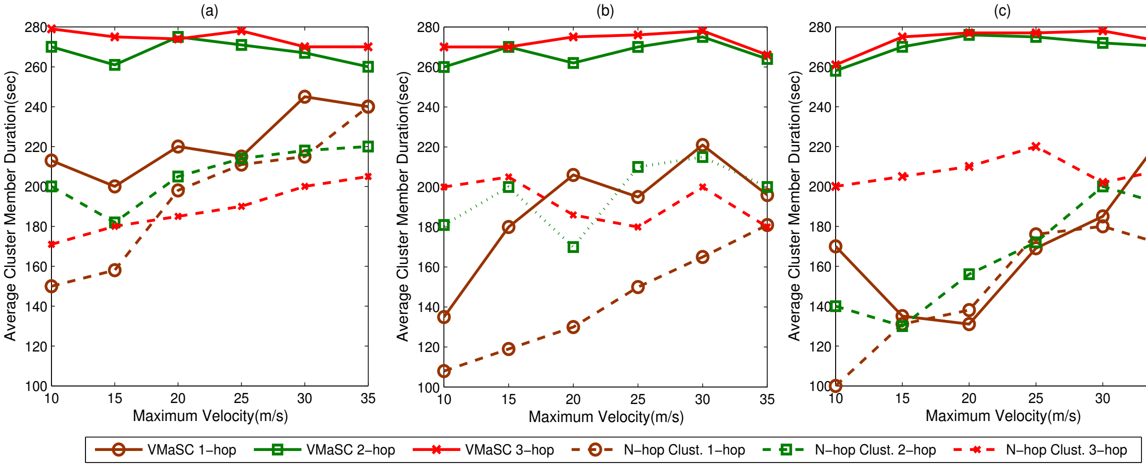

Fig. 3. Average cluster member duration, for transmission range (a) 100m (b) 200m (c) 300m

reception, so average cluster member duration is lower than our approach VMaSC.

### *C. Cluster Head Change*

Cluster head change number is a metric that can be used for demonstrating the cluster stability. It is defined as the number of state changes from CLUSTER_HEAD to another state (e.g. STATE_ELECTION or CLUSTER_MEMBER). The cluster head change numbers are shown in Fig.4. When the vehicle velocity increases, head change numbers also increase. This is because of the network dynamics where the more velocity the vehicles have, the more dynamic the network topology is. Another metric that significantly affects head change is the transmission range, where the head change number differs in hop number. In 1-hop scenarios, as the transmission range increases head change numbers decrease. Main reason behind this is in large transmission range more vehicles can hear CH_ADV packet and become member. When the CH's timer is expired, it has at least one member to serve. In contrast to 1-hop scenarios, in multi-hop scenarios when the transmission range increases, head change number also increases. However, compared to 1-hop scenarios, in multi-hop scenarios head change is decreased by 10%. Head change increase in multi-hop scenarios is due to packet collisions described above. In contrast to multi-hop clustering, in N-hop clustering head change number is larger than multi-hop clustering. This can be explained by state changes from cluster head to another state. In N-hop clustering, member election is based on cluster information reception, and thus the connection between head-member pair is weaker than our multi-hop clustering VMaSC where the average relative speed is used as the key metric.

## IV. CONCLUSION

We introduced a stable multi-hop clustering technique based on the changes in the relative mobility of the vehicles which is calculated by finding the average of the relative speed of all the

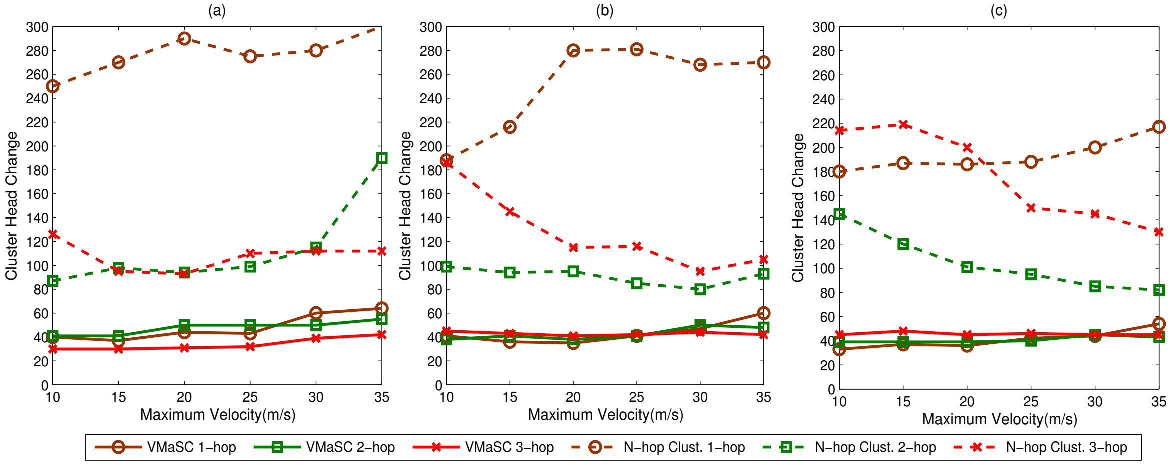

Fig. 4. Cluster head change, for transmission range (a) 100m (b) 200m (c) 300m

same direction neighbors. We modeled our approach VMaSC on ns-3 using the realistic mobility traces of SUMO and compared its performance to previously proposed multi-hop cluster approach called N-hop clustering that uses the variation in the packet delay metric. Simulation results show that the clustering of VMaSC outperforms the N-hop clustering in terms of cluster head duration, cluster member duration and cluster head change metrics at various transmission range and vehicle velocity scenarios. Our ongoing work involves adapting new road scenarios and testing our approach, VMaSC, in these roads with different velocity and direction to do more performance analysis. In the future, we are planning to integrate our clustering approach to the heterogeneous architecture of IEEE 802.11p and LTE.

## ACKNOWLEDGEMENT

Our work was supported by Turk Telekom under Grant Number 11315-07.

## REFERENCES

[1] R. Chen, W.-L. Jin, and A. Regan, “Broadcasting safety information in vehicular networks: issues and approaches,” *Network, IEEE*, vol. 24, no. 1, pp. 20 –25, 2010.

[2] T. Tielert, D. Jiang, Q. Chen, L. Delgrossi, and H. Hartenstein, “Design methodology and evaluation of rate adaptation based congestion control for vehicle safety communications,” in *Vehicular Networking Conference (VNC), 2011 IEEE*, pp. 116 –123.

[3] J. Nzouonta, N. Rajgure, G. Wang, and C. Borcea, “Vanet routing on city roads using real-time vehicular traffic information,” *Vehicular Technology, IEEE Transactions on*, vol. 58, no. 7, pp. 3609 –3626, 2009.

[4] Y. Toor, P. Muhlethaler, and A. Laouiti, “Vehicle ad hoc networks: applications and related technical issues,” *Communications Surveys Tutorials, IEEE*, vol. 10, no. 3, pp. 74 –88, 2008.

[5] A. International, *Standard Specification for Telecommunications and Information Exchange Between Roadside and Vehicle Systems 5 GHz Band Dedicated Short Range Communications (DSRC) Medium Access Control (MAC) and Physical Layer (PHY) Specifications,*, 2009.

[6] A. Damnjanovic, J. Montojo, Y. Wei, T. Ji, T. Luo, M. Vajapeyam, T. Yoo, O. Song, and D. Malladi, “A survey on 3gpp heterogeneous networks,” *Wireless Communications, IEEE*, vol. 18, no. 3, pp. 10 –21, 2011.

[7] M. Gerla and J. T.-C. Tsai, “Multicluster, mobile, multimedia radio network,” *Wirel. Netw.*, vol. 1, no. 3, pp. 255–265, 1995. [Online]. Available: http://dx.doi.org/10.1007/BF01200845

[8] P. Basu, N. Khan, and T. Little, “A mobility based metric for clustering in mobile ad hoc networks,” in *Distributed Computing Systems Workshop, 2001*, pp. 413 –418.

[9] C. chuan Chiang, H.-K. Wu, W. Liu, and M. Gerla, “Routing in clustered multihop, mobile wireless networks with fading channel,” 1997.

[10] J. Yu and P. Chong, “3hbac (3-hop between adjacent clusterheads): a novel non-overlapping clustering algorithm for mobile ad hoc networks,” in *Communications, Computers and signal Processing. PACRIM. 2003 IEEE Pacific Rim Conference on*, vol. 1, pp. 318 – 321 vol.1.

[11] Y. Zhang and J. M. Ng, “A distributed group mobility adaptive clustering algorithm for mobile ad hoc networks,” in *Communications. ICC ’08. IEEE International Conference on*, pp. 3161 –3165.

[12] T. J. Kwon, M. Gerla, V. Varma, M. Barton, and T. Hsing, “Efficient flooding with passive clustering-an overhead-free selective forward mechanism for ad hoc/sensor networks,” *Proceedings of the IEEE*, vol. 91, no. 8, pp. 1210 – 1220, 2003.

[13] C. Shea, B. Hassanabadi, and S. Valaee, “Mobility-based clustering in vanet using affinity propagation,” in *Global Telecommunications Conference. GLOBECOM 2009. IEEE*, pp. 1 –6.

[14] N. Maslekar, M. Boussedjra, J. Mouzna, and L. Houda, “Direction based clustering algorithm for data dissemination in vehicular networks,” in *Vehicular Networking Conference (VNC), 2009 IEEE*, pp. 1 –6.

[15] H. Su and X. Zhang, “Clustering-based multichannel mac protocols for qos provisionings over vehicular ad hoc networks,” *Vehicular Technology, IEEE Transactions on*, vol. 56, no. 6, pp. 3309 –3323, 2007.

[16] Z. Zhang, A. Boukerche, and 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. New York, NY, USA: ACM, pp. 19–26.

[17] Simulation of urban mobility, sumo @ONLINE. [Online]. Available: http://http://sumo.sourceforge.net/

[18] Network simulator, ns-3 @ONLINE. [Online]. Available: http://www.nsnam.org/