An Adaptable Mobility-Aware Clustering Algorithm in Vehicular Networks

Mildred M. Caballeros Morales, Choong Seon Hong
Networking Lab. Dept of Computer Engineering
Kyung Hee University
South Korea
madai@networking.khu.ac.kr, cshong@khu.ac.kr

Young-Cheol Bang
Dept of Computer Engineering
Korea Polytechnic University
South Korea
ybang@kpu.ac.kr

Abstract—**The forthcoming Intelligent Transportation System aims to achieve safety and productivity in transportation using vehicular ad hoc networks (VANETs) to support the communications system required. Currently, some clustering approaches have been proposed to improve the performance of VANETs due to their dynamic nature, high scalability and load balancing results. However, the host mobility and the constantly topology change continue to be main problems of this technique due to the lack of models which represent the vehicular behavior and the group mobility patterns. Therefore, we propose an Adaptable Mobility-Aware Clustering Algorithm based on Destination positions (AMACAD) to accurately follow the mobility pattern of the network prolonging the cluster lifetime and reducing the global overhead. In an effort to show the efficiency of AMACAD, a set of simulation was executed. The obtained results reveal an outstanding performance in terms of the lifetime of the cluster heads, lifetime of the members and the re-affiliation rate under varying speeds and transmission ranges.**

*Keywords:**Clustering Algorithm; Vehicular Networks; VANET; Intelligent Transportation System ITS; Mobility Aware; Destination-Based.***

I. INTRODUCTION

The vehicular network VN has arisen as a profitable and auspicious network that improves the ubiquitous communication services across the moving vehicles. However, nowadays the critical issues in this area are the lack of reliable communication channels and the high delay of the communications, due to the heterogeneous nature and mobility of the environment.

The relevance of these problems is shown practically in urban scenario in which using an Intelligent Transportation System (ITS), the vehicles share information with each other and with a local server through base stations (BSs) to avoid traffic congestion on the road. Moreover, a communication link exists between the local server and the traffic lights to control and divert traffic in case of an accident or traffic congestion occurs. In this scenario the low delivery rate, the temporary fragmentation and the congestion at the BS and local server are evident. To overcome these challenges we propose a mobility-aware clustering algorithm to decrease the delay of the communications, the bottlenecks, the overhead and the temporary fragmentation, considering the vehicular behavior and group mobility patterns.

Some cluster-based approaches have been applied in VANETs, because the clusters reduce the overhead and delay, solving the scalability problem, providing an efficient resource consumption and load balance in large scale networks. However, in a high mobility environment the clusters usually are unstable and the clustering/de-clustering is constantly executed. The unstable issue is addressed in MobHiD [1], by proposing a predictive approach to arrange the clusters by selecting the node which is relatively stable as cluster head (CH), and using trees of neighborhood. It does not consider the fast mobility that occurs within VANETs. In the other hand the selection of the CH could be determined by the transmission speed and the quantity of mobile nodes [2]. Nevertheless, the authors do not consider the patterns and behavior of the vehicles. DGMA [3] is the approach that better considers the behavior of the vehicles, using the speed and direction parameters, but without regard of their destination.

The destination of the vehicles is a key feature to model the group mobility and the behavior of the vehicles, enhancing the cluster stability and cluster lifetime.

Therefore, our clustering algorithm takes into account the destination of vehicles, including the current location, speed, relative destination and final destination of vehicles as parameter to arrange the clusters. In this manner, the clustering algorithm resembles a natural model of location references, which helps to manage the mobility by improving the lifetime of the cluster and decreasing the number of cluster head changes and the number of cluster re-affiliations. The information is disseminated by groups enhancing the communication delay, reliability, low data delivery and congestion issues, making the vehicular networks accurate and efficient.

The rest of the paper is structured as follows. Section II gives an overview of the approaches related to the clustering algorithms within a mobile environment. In section III we explain the scenario, the network model and the assumptions of the proposal. In section IV, the destination-based clustering algorithm is described. Section V shows the performance and the evaluation of the proposal. Finally, the conclusions of our work are presented in section VI.

II. RELATED WORK

Clustering algorithms have been proposed for vehicular networks. MobHiD [1] estimates the future mobility of nodes predicting the probability that the current neighborhood of a mobile node will remain the same through the received signal power to estimate the distance between two nodes. The drawback of the prediction method is the lack of accuracy in some cases. MOBIC [4], calculates the variance of relative mobility of a mobile node with each of its neighbors, where a small value of variance indicates the mobile node is moving

This work was supported by the IT R&D program of MKE/KEIT [2001-S-014-01, On the development of Sensing based Emotive Service Mobile Handheld Devices]. Dr. CS Hong is the corresponding author.

relatively less than its neighborhood. However, in the case in which few neighbor nodes move differently, the method still results in dramatic increase in the variance. DGMA [3] is a distributed and adaptive clustering algorithm, which is implemented within a mobile environment called Reference Region Group Mobility model. It can be adapted to high speed environment; it also makes difference between micro and macro changes, taking actions only when a macro change arises. DGMA designs a Linear Distance based Spatial Dependency (LDSD) metric for clustering, considering distance, direction and speed as parameters. Despite all the efficient considerations of the DGMA, it does not take into consideration the destination of the vehicles, and this parameter is very important to prolong cluster lifetime because vehicles with the same destination have the same route and can easily travel in groups. UFCM [2] and VWCA [5] enforce a weight cluster mechanism with a backup manager. These algorithms operate in similar way. UFCM considers the position, direction, speed and range of the nodes to perform the algorithm. On the other hand, VWCA takes into consideration the number of neighbors based on the dynamic transmission range, the direction of vehicles, the entropy, and the distrust value parameters. VWCA works with an adaptive allocation of transmission range (AATR) technique, where hello messages and density of traffic around vehicles are used to adaptively adjust the transmission range among them. However, both proposals, as DGMA, do not consider the destination of the vehicles as a determinant parameter to arrange clusters.

A suitable solution to prolong the cluster lifetime, avoid congestion and overhead considering the vehicular behavior is essential. Thus, we propose a clustering algorithm that uses the destination of the vehicles besides the common parameters i.e., speed, direction and density.

III. Scenario, Network Model And Assumptions

A. Scenario

We considered an urban scenario with an Intelligent Transportation System (ITS) as shown in Fig. 1, in which vehicles request and receive information from Base Stations (BSs) or from other vehicles. The vehicles are interested in information to avoid accidents (road warnings), reduce congestion, update local maps, and provide entertainment (web browsing, email, media streaming). All the vehicles employ Global Position System (GPS) or Navigation System (NS).

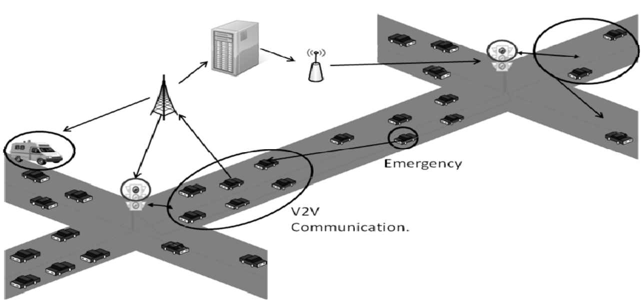

Figure 1. Urban scenario

The local servers gather information of the road conditions, disseminate infotainment messages to the vehicles and address the traffic light system. When an emergency occurs such as an accident, a driver's bad health or a police chase takes place, the dissemination of information among vehicles, the ambulance and the traffic light system near by the affected area starts to divert traffic and avoid vehicular congestion on the road.

B. Network Model and Assumptions

Our proposal selects a part of the city compounded by streets and avenues and divides this part into smaller segments. Each segment is called region. The destination of the vehicles is categorized into 2 types: 1) Relative destination: it is the nearest destination according to the current region. 2) Final destination: it is the final destination which is registered in the GPS or NS.

When a new vehicle arrives to a region, the vehicle updates its relative destination in accordance with the current region. The relative positions of vehicles are managed by mapping the vehicle's latitude and longitude coordinates to points on the road. All the points that belong to a region are mapped to a coordinate system (x, y).

In the network there are three entities: Mobile Node (MN), Cluster Head (CH), and Base Station (BS). We assume that:

- Each vehicle has onboard units with GPS or NS.
- The destination of each vehicle is known. This assumption is realistic in case of real-world systems because the GPS or NS allows drivers to enter their travel destination.
- There is historical information of the traffic and density of each street.
- The communication is content-based.
- The routing is geographic-based.
- To transmit data efficiently, a combination of broadcast data transmission and store and forward is used.
- An exchange messages mechanism is performed to gather information among vehicles, such as statistics or parameter's values.
- The position and the speed of the vehicle are calculated periodically during a time interval Δt .
- The appearance of relevant events triggers the exchange message mechanism.
- The CH is responsible of request information to the BS.

IV. Clustering Algorithm

The feasibility of a clustering method is determined by the stability of the cluster structure. An efficient clustering method should seriously take into account the movements of mobile host in order to form clustering structures resistant to the host mobility. Thus, our clustering algorithm, AMACAD operates with the final destination, relative destination, speed, and current location as parameters to calculate the metric function, within an urban scenario.

AMACAD works in a distributed way, where each node can execute the algorithm.

A. Message exchange mechanism

The vehicles exchange information periodically to adapt the clusters to the vehicular behavior in real time. This mechanism is performed in two cases. 1) When the mobility changes are calculated, at each Δt. The Δt value is dynamically assigned depending on the minimum block size of the streets and the speed limit of the road in a region. 2) When special events occur such as significant changes of speed or bandwidth, triggering the message exchange mechanism.

The CHs keep a parameter table, which stores the vehicle's id, the CH id, the current location in terms of (x, y) coordinates, speed, relative destination and the final destination in (x, y) coordinates. The CHs and MNs have a list of other nearby CHs. These CHs are the candidates to be CH in case of the current CH is no longer a good option. The MNs hold a list of the CHs of which they are members.

All the nodes maintain a table storing historical information related to the location (x, y), speed, relative destination (x, y), final destination and bandwidth. This information is collected over periods of time, Δt, discarding the oldest information.

The algorithm employs different types of messages. Table 1 shows the basic information of the messages.

TABLE I. MESSAGES

Name	*Type*	*Goal*
Affiliation	Request	Initialize the list of nearby CHs.
Affiliation_ack	Acknowledge	Send the current CH id.
Hello	Request	Start the hello message exchange.
Hello_ack	Acknowledge	Information exchange to control the mobility changes.
Add	Request	Join to a CH.
Member_ack	Acknowledge	Add the CH id as current CH.
CH_ack	Acknowledge	Add the MN to the parameter table of the CH.
Member_update	Acknowledge	Update the CH of the cluster.
Delete	Request	End the subscription with the current CH
Warning (speed or bandwidth)	Information	Report significant events.
Reclustering	Request	Start the re-clustering process.
Reclustering_ack	Acknowledge	Confirm participation in the re-clustering process.

B. Forming cluster

The main goal is to find out the members for each cluster. The process outline of our algorithm is shown in Fig. 2. When a vehicle V wants to find a cluster head, it sends an "affiliation" message to all its neighbors, if it does not receive any reply; it starts the cluster formation process. The first step is exchanging hello messages with its neighbors, gathering information related to the parameter table. As a second step V computes relative values using its parameter table.

- At an instant of time *t*, the distance between V and its neighbor node Z from its current location is: ΔLv,z

$$\Delta L_{v,z} = \sqrt{((Lx_v - Lx_z)^2 + (Ly_v - Ly_z)^2)} \quad (1)$$

  Where (Lx_v, Ly_v) and (Lx_z, Ly_z) are the (x,y) nodes locations V and Z respectively.

- The difference in speed between V and its neighbor node Z is: ΔSv,z

$$\Delta S_{v,z} = |Speed_v - Speed_z| \quad (2)$$

- The distance between V's relative destination and Z's relative destination at time t, is: ΔRDv,z

$$\Delta RD_{v,z} = \sqrt{((RDx_v - RDx_z)^2 + (RDy_v - RDy_z)^2)} \quad (3)$$

  Where (RDx_v, RDy_v) and (RDx_z, RDy_z) are the (x,y) relative destination of nodes V and Z respectively.

- Finally, V computes,

$$F_{v,z} = w_1 \times \Delta L_{v,z} + w_2 \times \Delta S_{v,z} + w_3 \times \Delta RD_{v,z} \quad (4)$$

  Where w_1, w_2, w_3 represent the weights for each variable.

The value $F_{v,z}$ should be calculated for all vehicles Z within V's transmission range (m). The third step is the construction of the cluster, computing all the combinations of $F_{v,z}$ among the m vehicles, through a matrix. The selection criterion is pondered as best the minimum value of $F_{v,z}$.

The matrix is symmetric. Therefore, we focus only in the upper or lower triangular matrix, selecting the minimum value of this triangular matrix, and then two vehicles are involved. These vehicles belong to the same group from now on, being represented by a single value, which is the lower value of the two vehicles. The process is executed up to either way, the number of formed clusters is less than or equal to the threshold N_{max} or the difference between the minimum $F_{v,z}$ value of the previous round and the current is greater than the threshold of change (ΔMC_{max}). The default value of N_{max} is 1. However, the N_{max} and ΔMC_{max} thresholds can be manually set or dynamically adjusted according to the vehicular density of the road.

Then V starts the process to find the CHs.

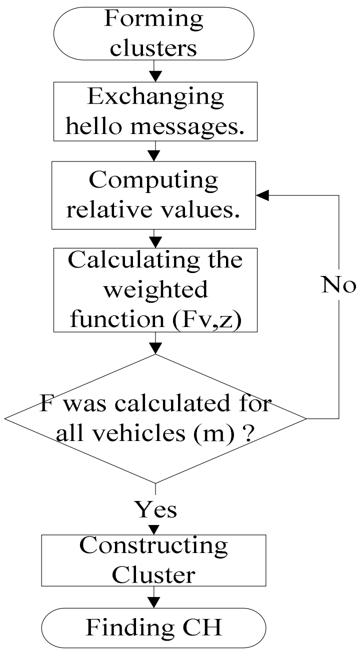

Figure 2. AMACAD Algorithm

C. *Finding CH*

After V performs the cluster formation, the CH for each cluster is selected, the CH is the node with the minimum value of F_v, where, n is the number of vehicles that belong to a specific cluster.

$$F_v = \sum_{i=1}^{n} F_{v,z} \tag{5}$$

V sends the updated parameter table with the "CH_ack" message to each CH. Each CH sends the "member_ack" message to all its members.

D. *Cluster maintenance*

When the groups and CHs are already selected, an MN might need to be moved to another cluster or a new CH has to be selected because the current CH is not appropriate anymore. Hence, five operations are required: add member, delete member, update mobility changes, change of CH and re-clustering. AMACAD performs these operations considering thresholds related to the density, speed and bandwidth of vehicles.

- Density: The cluster should not be big because too many node members may cause overhead or too small increasing the number of re-affiliations.
  - d_{max}: Maximum number of vehicles allowed in a cluster.
  - d_{min}: Minimum number of vehicles allowed in a cluster.
  - CH_{max}: Maximum amount of CHs that will form the neighboring cluster in the re-clustering operation.
- Speed:
  - S_{max}: Maximum limit for speed variance.
- Bandwidth:
  - bw_{min}: Minimum bandwidth or QoS level required.

V. PERFORMANCE AND EVALUATION

In order to evaluate our proposal we implemented a testbed using Java JDeveloper 10G, taking into account the data dissemination problem described in our scenario and comparing the results with the performance of other clustering algorithm such as Lowest-ID [6], MOBIC, DGMA, MobHiD.

The time of each simulation was 1500 seconds in a space of 1500 by 1500 meters (m). This space is divided by blocks of 100 m. The basic simulation parameters were transmission range between 100 to 300 m, vehicle density between 0 to 5 vehicles per 100 m^2 and speed between 5 to 20 meters/second (m/s). To analyze the performance of the AMACAD method and the existing approaches we employed the mobility model, evaluation criteria, parameter values and simulation results presented in the next sections.

A. *Mobility model*

MobHiD [1] implements a Reference Point Group Mobility model (RPGM) in which all the nodes belonging to the same group are moving randomly within a certain distance of a reference point. However, it does not describe grouping behaviors such as group partition and group merges. DGMA [3] simulation utilizes the References Region Group Mobility (RRGM), in which each node belonging to the group randomly picks a location inside the reference region and moves towards to the group's destination. This model enhances the RPGM, allowing a group mobility environment, including, the phenomenon of group partition and group merge. However, RRGM does not represent our scenario, due to the fact that each vehicle has its own destination and we need to simulate routes with realistic mobility traces. Therefore, we adjust the RRGM scenario including roads; each vehicle is initially placed at a random position within the roads and has their final destination within the region. As the simulation progress, each vehicle moves on the road to its destination with a speed between 5 to 20 m/s, speed changes exist in some cases. Each vehicle continues this behavior until its destination is reached or for the simulation period.

The vehicles positional system of the vehicles manages the x and y coordinates within a region, following in this way the routes toward the vehicle destination.

B. *Evaluation criteria*

To measure the performance of AMACAD, the following indicators are used:

- CH lifetime: It is the mean time duration of the nodes, remaining its leadership role as CHs. Long CH lifetime implies, few changes and good stability.
- Membership lifetime: It is the mean time duration of the nodes remaining as members of a cluster, before moving to other cluster or becoming a CH.
- Number of cluster: It is the number of the formed clusters/CHs during the simulation.
- Re-affiliation rate: It is the number of CH elections and MN affiliations per second.

- Link capacity: It is the available bandwidth for each member, which is evaluated through the membership lifetime because when the members detects a low link capacity looks for other CH. Good quality link capacity is determined by long membership lifetime.

C. Parameter values

The dynamic parameters of AMACAD, are the weights (w1, w2, and w3) and the Δt. The values of these parameters are dynamically assigned according to each scenario. Based on the outcome of the set of simulations, we obtained significant data to determine the weights and Δt. The value of Δt depends on the block size and the maximum speed allowed. Therefore, Δt=4 seconds ensures a response in real time, due to the fact that the block size is 100m, the maximum speed is 20m/s and the road intersections to change direction are separated at least by one block. Another finding was that the number of cluster was determined by the ΔMC_{max} threshold because it arranged the clusters more efficiently than the N_{max}. In addition, the weights with best results in terms of CH lifetime were w1=0.2, w2=0.4 and w3=0.4 as shown in table 2 (set 1). Therefore, these values were used to run the simulations.

TABLE II. WEIGHTS RESULTS

Set	***Location w1***	***Speed w2***	***Destination w3***	***Number of cluster***	***CH lifetime***
1	0.2	0.4	0.4	9	118
2	0.4	0.2	0.4	4	50
3	0.4	0.4	0.2	12	80
4	0.33	0.33	0.33	7	100
5	0.01	0.7	0.29	5	70
6	0.7	0.01	0.29	7	90
7	0.29	0.01	0.7	8	35
8	0.01	0.29	0.7	9	100
9	0.7	0.29	0.01	4	35
10	0.29	0.7	0.01	14	60

D. Simulation results

The simulation environment includes 50 vehicles with a predefined destination. We executed two sets of experiments. The first set evaluates how the variation of the transmission range affects the AMACAD performance. The second set evaluates the speed effects on the performance. The performance is measured by the CH lifetime, membership lifetime and re-affiliation rate.

1) Transmission range effects

The speed was set to 10 and 20 m/s; in each set, the transmission range varies from 100 to 200 m. Fig. 3, Fig. 4 and Fig. 5 illustrate the behavior of the existing algorithms and our proposal over the transmission range values. An interesting fact is that while the transmission range increases, the CH and membership follow the same tendency as shown in Fig. 3 and Fig. 4 This effect is due to larger transmission range allows more diversity of nodes in term of destinations and speed, improving the cluster stability. However, larger transmission range makes possible to arrange clusters with many members, if the algorithm does not address this problem, the outcome could be congestion or continuous changes, such as, MOBIC and MobHid behavior. The CH lifetime of AMACAD performs better than the rest of algorithms, because the Lowest-ID does not take into account the mobility to the CH election, MOBIC and DGMA only consider the current mobility meanwhile MobHiD predicts the future mobility. However, AMACAD exactly knows the vehicles destination to manage precisely the clusters of vehicles in real time, avoiding negative impact of the mobility and the speed changes.

Fig. 4 exhibits the stability of the link capacity because the members remain in the same cluster, which means members are receiving the necessary amount of bandwidth.

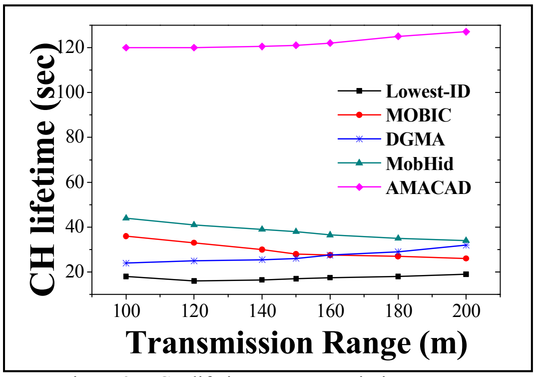

Figure 3. CH lifetime vs. Transmission Range.

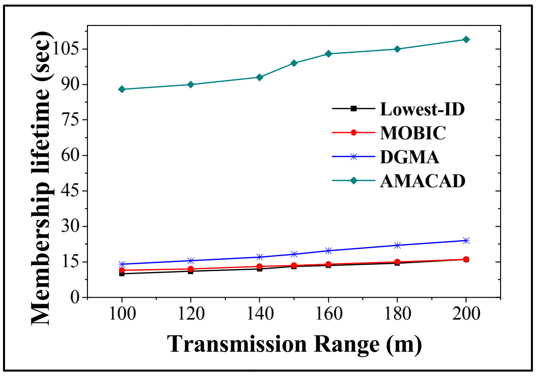

Figure 4. Membership lifetime vs. Transmission Range

Fig. 5 shows an inverse tendency, because if the lifetime is short the number of re-affiliation per second increases.

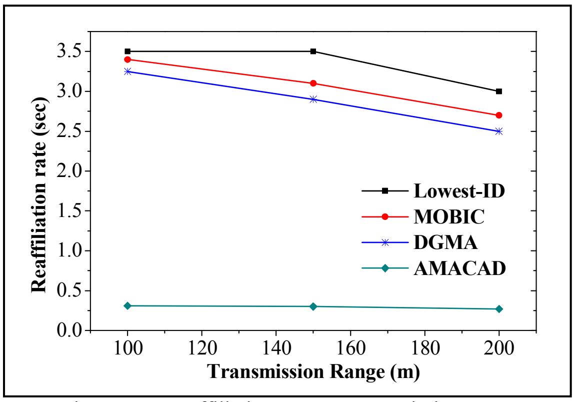

Figure 5. Re-affiliation rate vs. Transmission Range

2) Speed effects

The transmission range was set to 100 m, the speed ranged from 5 to 20 m/s, including speed variations. Fig. 6, Fig. 7 and Fig. 8 present the performance of AMACAD compared to other techniques in terms of CH lifetime, membership lifetime and re-affilation rate respectively. When the speed increases the stability decreases as well, this behavior is caused by: 1) the vehicles move at high speed; therefore, they cross the distance in less time. 2) When vehicles travel at high speed the vehicles have more chance to contact with others CH, causing a high re-affiliation rate. Fig. 6 shows the CH lifetime of the algorithms in high speed, it is observed that Lowest-Id has a good performance with a low speed, but as the speed increases the lifetime CH drastically decreases.

The AMACAD performs well at high speed if most of the members maintain similar average speed, but with a meaningful speed difference among the members, the performance may decrease. However in urban scenarios the vehicles travel with similar speeds.

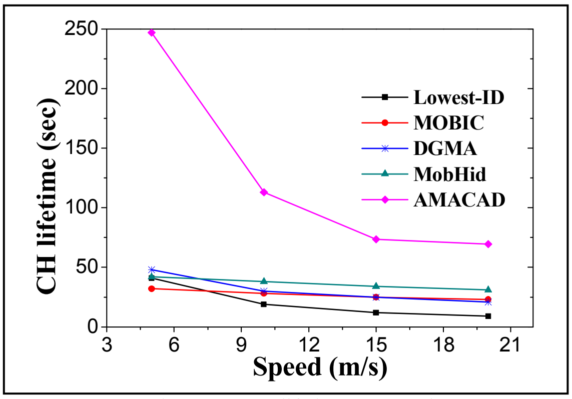

Figure 6. CH lifetime vs. Speed

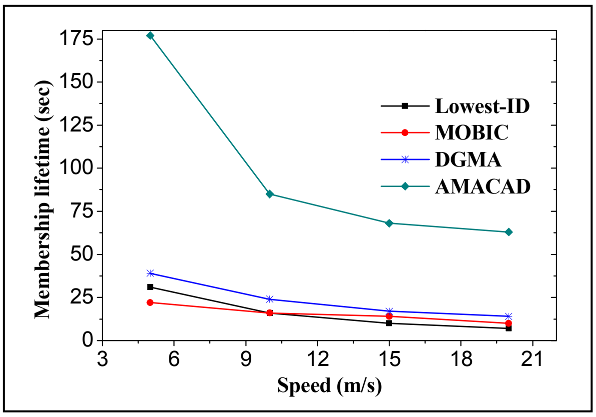

Figure 7. Membership lifetime vs. Speed

The key difference between the AMACAD and other clustering algorithms performance is that, in our scenario the vehicles have similar average speed and several destinations with different paths due to realistic mobility traces. Therefore, other approaches cannot select the clusters that travel together for long distances. Thus, their performance in real scenarios would be poor.

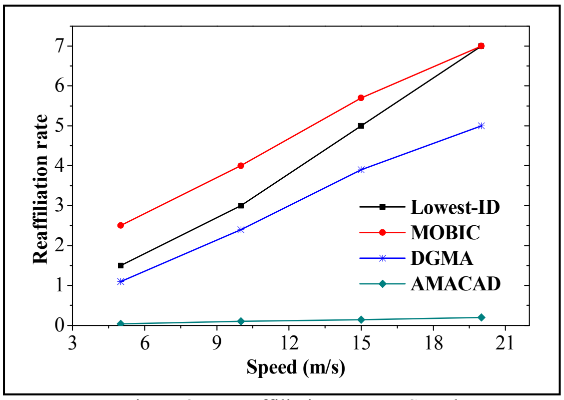

Figure 8. Re-affiliation rate vs. Speed

VI. CONCLUSION

The VANET has to overcome the challenges of communication delay, low delivery rate, reliability, scalability and congestion. The clustering dissemination is a prominent solution to overcome these challenges. However the current clustering algorithms do not exploit the vehicular behavior and group mobility patterns taking into account the final destination of the vehicles to provide cluster stability despite of the mobility. Therefore, we propose an Adaptable Mobility-Aware Clustering Algorithm based on Destination in vehicular networks (AMACAD) to enhance the clustering stability for VANETs scenarios. AMACAD takes into account the destination of the vehicles to arrange the clusters and implements an efficient message mechanism to respond in real time and avoid global re-clustering. The AMACAD performance has been tested through a set of experiments. The results of the experiments revealed a very good performance compared with other approaches such as MOBIC, DGMA, etc. The benefits of consider the current location, the speed and the vehicles destination in the CH selection are evident.

REFERENCES

[1] C. Konstantopoulos, D. Gavalas, G. Pantziou, “Clustering in mobile ad-hoc networks through neighborhood stability-based mobility prediction” Journal of Computer Networks, 2008.

[2] S-C. Wang, H-H. Pan, K-Q. Yan, Y-L. Lo, “A unified framework for cluster manager election and clustering mechanism in mobile ad-hoc networks” Computer Standards & Interfaces, 2008.

[3] Y. Zhang, J. Mee Ng, C. Ping Low, “A distributed group mobility adaptive clustering algorithm for mobile ad hoc networks” Journal of Computer Communications, 2009.

[4] P. Basu, N. Khan, T. D.C Little, “A mobility based metric for clustering in mobile ad hoc networks” In: IEEE ICDCS 2001 Workshop on Wireless Networks and Mobile Computing, 2001

[5] A. Daeinabi, A. G. P. Rahbar, A. Khademzadeh, “VWCA: An efficient clustering algorithm in vehicular ad hoc networks”, Journal of Network and Computer Applications, 2010.

[6] A. Ephremides, J. E. Wieselthier, D. J Baker, “A design concept of reliable mobile radio networks with frequency hopping signaling” In: Proceedings of the IEEE, 1987, vol.75. no.1. pp.56-73.

[7] S. Basagni, “Distributed clustering for ad-hoc networks” In: Proceedings of the 1999 International Symposium on Parallel Architectures, Algorithms and Networks, 1999, pp.310-315

[8] H. Dang, H. Wu, “Cluster-Based Data Transmission Protocol in Delay-Tolerant Mobile Networks. In LA” IEEE, 2008.