



Wayne State University

---

Wayne State University Associated BioMed Central Scholarship

---

2012

# A novel algorithm to form stable clusters in vehicular ad hoc networks on highways

Zaydoun Y. Rawashdeh

Wayne State University, [zaydounr@wayne.edu](mailto:zaydounr@wayne.edu)

Syed Mahmud

Wayne State University, [smahmud@eng.wayne.edu](mailto:smahmud@eng.wayne.edu)

---

## Recommended Citation

Rawashdeh and Mahmud *EURASIP Journal on Wireless Communication and Networking* 2012, **2012**:15

doi:[10.1186/1687-1499-2012-15](https://doi.org/10.1186/1687-1499-2012-15)

Available at: <http://digitalcommons.wayne.edu/biomedcentral/48>

This Article is brought to you for free and open access by DigitalCommons@WayneState. It has been accepted for inclusion in Wayne State University Associated BioMed Central Scholarship by an authorized administrator of DigitalCommons@WayneState.

RESEARCH

Open Access

# A novel algorithm to form stable clusters in vehicular ad hoc networks on highways

Zaydoun Y Rawashdeh\* and Syed Masud Mahmud

## Abstract

Clustering in vehicular ad hoc networks (VANET) is one of the control schemes used to make VANET global topology less dynamic. Many of the VANET clustering algorithms are derived from mobile ad hoc networks (MANET). However, VANET nodes are characterized by their high mobility, and the existence of VANET nodes in the same geographic proximity does not mean that they exhibit the same mobility patterns. Therefore, VANET clustering schemes should take into consideration the degree of the speed difference among neighboring nodes to produce relatively stable clustering structure. In this paper, we introduce a new clustering technique suitable for the VANET environment on highways with the aim of enhancing the stability of the network topology. This technique takes the speed difference as a parameter to create relatively stable cluster structure. We also developed a new multi-metric algorithm for cluster-head elections. A simulation was conducted to evaluate our method and compare it with the most commonly used clustering methods. The simulation results show that our technique provides more stable cluster structure on the locale scale which results in a more stable network structure on the global scale. The proposed technique reduces the average number of clusters changed per vehicle by 34-46%, and increases the average cluster lifetime by 20-48% compared to the existing techniques.

**Keywords:** Vehicular networks, V2V, clustering schemes in VANET, CH election

## 1. Introduction

Recent advances in wireless networks have led to the introduction of a new type of networks called vehicular ad hoc networks (VANETs). This type of networks has recently drawn significant research attention since it provides the infrastructure for developing new systems to enhance drivers' safety [1-3]. Equipping vehicles with various kinds of sensing devices and wireless communication capabilities help drivers to acquire real-time information about road conditions allowing them to react on time. For example, warning messages sent by vehicles involved in an accident enhances traffic safety by helping the approaching drivers to take proper decisions before entering the crash dangerous zone [4,5]. Moreover, information about the current transportation conditions facilitate driving by taking new routes in case of congestion, thus saving time and adjusting fuel consumption [6,7]. In addition to safety concerns, VANET can also support other non-safety applications that require a quality of service (QoS) guarantee.

This includes Multimedia (e.g., audio/video) and data (e.g., toll collection, internet access, weather/maps/information) applications.

Vehicular ad hoc networks (VANETs) are characterized by high vehicle mobility. Due to high mobility, VANET topology changes rapidly, thus, introducing high communication overhead for exchanging new topology information [8,9]. Several control schemes for media access and topology managements have been proposed [8,10,11]. One of these schemes is establishing a hierarchical clustering structure within the network. The clustering allows the formation of dynamic virtual backbone used to organize media access, to support QoS and to simplify routing [8,12]. Mainly, nodes are partitioned into clusters, each with a cluster head (CH) node that is responsible for all management and coordination tasks of its cluster.

Ensuring stability is the major challenge for clustering algorithms especially in a highly dynamic environment. Thus, efficient clustering algorithms should not only focus on forming a minimal number of clusters as many existing algorithms do, but also maintain the current

\* Correspondence: zaydounr@wayne.edu

Electrical and Computer Engineering Department Wayne State University, Detroit, MI 48202, USA

cluster structure and keep the overhead at the minimum level. Most of the existing VANET clustering algorithms are derived from the MANET clustering schemes [8,13-17]. However, these algorithms lack a technique to capture the mobility characteristics of VANET nodes and fall in a major drawback of forming clusters considering only position and direction of vehicles located in geographic proximity regardless of their high relative speed. We believe that the existence of group members in the same geographic area does not mean that they exhibit the same mobility patterns, e.g., vehicles on the left lanes move faster than the vehicles on the right lanes, and thus their relative speed might be very high.

Since the main goal of clustering is to make global topology less dynamic, we believe that, changes in the network topology on the global scale are directly related to the stability of local clustering structure. Therefore, in order to enhance their stability, clustering models need to be redefined so that they are characterized based on the full status elements: speed difference, location, and direction rather than considering only position and direction. Some clustering techniques took mobility into consideration for cluster head (CH) elections, but not for cluster formation. For example, when the CH leaves its cluster due to merging with other clusters or mobility, the cluster members use a CH election algorithm that considers mobility to elect a new CH out of the cluster members [14].

In this work, we introduce a new clustering approach with the aim of increasing the stability of the network topology and making it less dynamic. This approach takes the speed difference, in addition to the location and direction, into consideration during the clustering process. But, with the inclusion of the speed difference as a new parameter, a new challenge arises as follows: how to partition the network into minimum number of clusters, such that when the clusters are finally formed, the distribution of the vehicles among them based on their mobility patterns is achieved with high probability. In short, we need an algorithm to accurately identify nodes showing similar mobility patterns and group them in one cluster. In this paper, our main contributions are as follows: first, developing a new clustering algorithm that runs on all nodes in a fully distributed fashion. This algorithm is used to divide the network nodes into clusters such that when the network is finally partitioned (clustered), the probability of partitioning along cluster boundaries is achieved with high probability. This means that vehicles with high mobility are grouped in one cluster and vehicles with low mobility are grouped in another cluster. Second, developing a new multi-metric election method that can be used by network nodes to determine their suitability to become cluster heads.

The rest of the paper is organized as follows: Section 2 presents VANET clustering algorithms. Section 3

introduces the system overview and assumptions. Section 4 describes the clustering process and the protocol structure. Section 5 shows the simulation results and the performance evaluation. Section 6 concludes the paper.

## 2. VANET-clustering algorithms

Several clustering techniques for VANET have been proposed in the literature. While most of these techniques focus on the media access organization for cluster members and use the MANET clustering techniques to form the clusters, none of them took speed difference into consideration for cluster formation in VANET. As a result, these techniques do not produce a stable clustering structure. Some of these proposed techniques are summarized below.

In [13], the authors proposed the cluster-based location routing (CBLR). Nodes use HELLO messages to distribute their states. When a node enters the system, it enters the undecided state and then announces itself as a CH if it does not receive a HELLO message within a period of time from other nodes; otherwise it registers at a CH as a member node. To cope with the VANET topology changes, nodes maintain a table containing a list of the neighboring nodes with which they can exchange information. The protocol mainly focuses on improving routing efficiency in VANET. The nodes are supposed to know their position and the position of their destination and therefore, the packets are forwarded directly toward the destination.

In [14], the authors adopted the same algorithm used in the CBLR for the cluster formation. Nodes can be members in more than one cluster. In this case they are called Gateways and used to route packets to their destination. Nodes track changes in the topology and adapt their states to the situation using two tables; one for the neighboring nodes and the other one for the adjacent clusters. When two cluster heads come into a direct communication range, one should give up its cluster-head role and merge with the other. The decision about which one keeps its state and which one loses its CH role is based on a weighted factor  $W_v$ , which takes into consideration the mobility, the connectivity, and the distance to the neighbors. These parameters are multiplied by their given weights and then summed to produce the total weight  $W_v$ . The smaller the  $W_v$ , the more qualified the node is to become a cluster head. The work also focuses on the media access control in the cluster-based VANET environment to improve the QoS support. The time division multiple access (TDMA) technique is used to divide the medium into time slots, which are then grouped into frames. The time slots are assigned to cluster members according to their needs.

Another clustering algorithm was proposed in [15]. The proposed algorithm is basically the lowest ID used

in MANET with a new modification. The authors included the leadership duration as well as the direction in the lowest ID algorithm to determine the node to be a cluster head. The leadership duration (LD) is defined as the period the node has been a leader since the last role change. The higher the leadership duration, the more qualified the node is to be a cluster head. Therefore, the cluster-head rule is: choose the node with the longest leadership duration and then choose the one with the lowest ID. The formation of clusters is based on beacon signals broadcasted by the VANET nodes. Each node announces itself as a cluster head and broadcasts this to all neighbors. If it receives a reply from a neighboring node with a lower ID and a higher leadership duration, then the node changes its state to a cluster member. When a node leaves its cluster, it looks for another cluster in the neighborhood to join. If none of the neighboring nodes or the neighboring cluster head satisfy the cluster head election rules, then the node claims itself as a cluster head.

The work in [15] was modified and presented in [16]. In addition to the LD and the moving direction (MD), the authors introduced the projected distance (PD) variation, which means distance variation of all neighbors over a period of time. Each node is associated with a utility weight (uW) of three parameters (LD, PD, and ID), where the ID is the identifier of the node. The LD parameter is given the highest weight. To define the total utility weight, a lexicographical ordering of the three parameters (LD, PD, and ID) is used. For example, the utility weight (LD1, PD1, ID1) is greater than (LD2, PD2, and ID2) if either  $LD1 > LD2$  or  $(LD1 = LD2 \text{ and } PD1 < PD2)$  or  $(LD1 = LD2 \text{ and } PD1 = PD2 \text{ and } ID1 < ID2)$ . Based on this, the LD value has maximum importance and its value is the primary factor to determine the total uW. However, in both works [15,16], the node that has higher connectivity degree might not be elected to lead the cluster if there is another node that has longer leadership duration. This will produce less stable cluster structure, because having longer leadership duration does not mean that the node has high connectivity degree that gives it the ability to lead the cluster.

In [17], the authors proposed a distributed cluster-based multi-channel communications scheme for QoS provisioning over V2V-based VANET. The goal is supporting the QoS for timely delivery of the real-time data (e.g., safety messages, road condition, etc.) and increasing the throughput for the non-real-time traffic over the V2V networks. The formation of the clusters is implemented using the traditional algorithms mentioned earlier, e.g., when a vehicle enters the road, it checks for nearby clusters to join. If there are no clusters, then the vehicle announces itself as a cluster head and forms a new cluster. The cluster merging can happen only when two

cluster heads come within the transmission range of each other. The cluster with less members is dismissed and its cluster head joins the neighboring cluster, while the other members start cluster formation process if they cannot join any nearby clusters. The proposed scheme assumes that each vehicle is equipped with two sets of transceivers, which can operate simultaneously on different channels. The cluster members use one transceiver to exchange safety messages and stay connected with the cluster head over the service channel; and use the other one to communicate with other members to exchange non-safety data. The cluster head communicates with its members via the service channel using one transceiver; and uses the other one to communicate with the neighboring clusters via the control channel.

In [18], the authors proposed a heuristic clustering approach for cluster-head elections that is equivalent to the computation of the minimum dominating sets (MDS) used in graph theory. This approach is called position-based prioritized clustering (PPC) and uses geographic position of nodes and the priorities associated with the vehicles traffic information to build the cluster structure. For clustering purposes, each node is assumed to broadcast a small amount of information of itself and its neighbors, which is referred by five tuples (node ID, cluster-head ID, node location, ID of the next node along the path to the cluster-head, and node priority). A node becomes a cluster-head if it has the highest priority in its one-hop neighborhood and has the highest priority in the one-hop neighborhood of one of its one-hop neighbors. The priority of the node is calculated based on the node ID, current time and the eligibility function. A Node having longer travel time has higher eligibility value, and this value decreases when the velocity of the node deviates largely from the average speed.

A new clustering algorithm was proposed in [19]. This technique basically classifies vehicles into groups based on the speed range of vehicles. Vehicles that fall in the same speed group belong to the same cluster. The authors defined seven groups based on the minimum and maximum value of the speeds that the vehicles can use. The range of the speed difference is 15 kmph for all groups except groups 0 and 6, which is 30 and 10 kmph respectively. The authors adopted the "First Declaration Wins rule", which is basically a node that first claims to be a cluster-head remains as a cluster-head and rules the rest of nodes in its clustered area. According to the authors' definition, if a cluster member speed changes such that the node travels at a speed that is different from the group speed for a period of time, then, the node must update its clustering group and should seek for a new cluster even though the node is still under the transmission range of its current cluster-head. The authors proposed that the cluster-head adjust its transmission

range when the density of the vehicles is very high. The cluster-head can reduce its transmission range to include less number of vehicles to reduce the management overhead. One of the drawbacks of this technique is that the first vehicle that claims to be the cluster-head may have its speed and location on the boundaries of both parameters. This cluster-head might lose the communications with its members soon. Moreover, having the cluster-head adjust its transmission range according to the speed of the group, makes the cluster members on the cluster boundary out of the transmission range of the cluster-head. Thus, these nodes will leave the cluster, which results in an increase of the cluster change rate.

The authors of [20] proposed a cluster formation technique where nodes use the affinity propagation (AP) method to pass messages to one another. Basically, the proposed algorithm takes an input function of similarities,  $s(i, j)$ , which reflects how well suited data point  $j$  is to be the exemplar of data point  $i$ . Nodes exchange two types of messages: responsibility,  $r(i, j)$ , indicating how well suited  $j$  is to be  $i$ 's exemplar, and availability,  $a(i, j)$ , indicating the desire of  $j$  to be an exemplar to  $i$ . The nodes use the self responsibility,  $r(i, i)$ , and self availability  $a(i, i)$ , to reflect the accumulated evidence that node  $i$  is an exemplar. When a node's self responsibility and self availability become positive, that node becomes a cluster-head. The authors proposed that a clustering decision is made periodically every clustering interval (CI) period, and a clustering maintenance is performed in between CI. However, having cluster members make clustering decision every CI will increase the probability of re-clustering. Also the authors did not take into consideration the speed difference among neighboring nodes.

In [21], the authors proposed a clustering technique for MANET applications. They introduced an aggregate local mobility (ALM), which is a relative mobility metric that used the received signal strength (RSS) at the receiving node as an indication of the distance between the sender and the receiver. However, the use of RSS is highly unreliable, especially in VANET environment, as indicated by other researchers [22]. The paper [21] also did not take the speed difference as a parameter to form clusters.

In [22], the authors basically uses the ALM proposed in [21], with some modifications, as a criterion for triggering cluster re-organization. Originally, the ALM is a relative mobility metric that uses the RSS at the receiving node as an indication of the distance between the sender and the receiver [21]. The ratio of the RSS of two successive periodic *hello* messages indicates the relative mobility between the two nodes. In [22], the authors used the location information embedded in the periodic hello

messages to determine the relative mobility of the nodes instead of using the signal strength. In this technique, if two cluster heads come into direct communication range, they exchange more than one packets in a predefined period of time in order to consider the merging between the two clusters. In case merging takes place, the cluster-head with the lower ALM value maintains its role while the other gives up its role and becomes a member node in the new cluster. However, the nodes that lost their cluster-head due to merging or mobility and cannot find nearby clusters to join, they will all become cluster heads almost at the same time. There will be a period where they will organize their minds as to who will be the new cluster-head. However, the authors did not take the speed difference of neighboring nodes into consideration.

### 3. System overview and assumptions

The degree of the speed difference among neighboring vehicles is the key criterion for constructing relatively stable clustering structure. Neighboring vehicles cooperate with each other to form clusters. In general, vehicles build their neighborhood relationship using the position data embedded in the periodic messages. Usually, vehicles broadcast their current state to all other nodes within their transmission range  $r$ . Therefore, two vehicles are considered *r-neighbors* if the distance between them is less than  $r$ . The total number of *r-neighbors* of a given vehicle is called the nodal degree of the vehicle. All notations used for analysis are presented in Table 1.

Clusters are formed by vehicles traveling in the same direction (one way). Therefore, all *r-neighboring* nodes used in our analysis are limited to those vehicles traveling in the same direction. However, the speed levels among the *r-neighbors* vary and this variation might be very high; thus, not all *r-neighbors* are suitable ones to be included in one cluster, and therefore, they are not good Candidate Cluster Member. In order to build relatively stable clustering structure, vehicles should consider only *r-neighbors* that are good candidate cluster member (CCM). Therefore, in this work, vehicles are required to classify their *r-neighbors* into stable neighbors (SN) and non-stable neighbors. Two vehicles are considered stable *r-neighbors* if their relative speed is less than  $\pm \Delta v_{th}$ . Hence, only stable neighbors of the vehicle initiating the cluster formation request participate in the cluster formation process.

To show how the degree of the speed difference is used in our technique, we first introduce the statistical distributions of the vehicles' velocity. According to [23-25], the velocity can be modeled using the normal distribution with mean,  $\mu$ , and variance,  $\sigma^2$ , and its probability density function (pdf) is given by:



**Table 1 Notation**

Notation	Description
$r$	Service channel transmission range
$R$	Control channel transmission range
$d$	Nodal degree
SN	Stable neighbors
CCM	Candidate cluster member
$\Delta v_{th}$	The threshold used to classify neighboring nodes as stable or non stable neighbors
$\mu$	The mean value of the speed
$\sigma$	The standard deviation of the speed
$u$	The suitability value to become a cluster-head
$p$	The position of the vehicle
$p_{norm}$	The normalized position
$v_{norm}$	The normalized speed
CH	Cluster-head
CM	Cluster member
COV	Cluster originating vehicle
$T_{wait}$	The waiting time before announcing the eligibility to become a CH
$\Gamma(t)$	A set containing the IDs of the $2r$ -stable neighbors whose velocities are greater than the velocity of the current vehicle
$\Lambda(t)$	A set containing the IDs of the $2r$ -stable neighbors whose velocities are less than the velocity of the current vehicle
$N_{max}$	The total number of vehicles in $\Gamma(t)$ at time $t$
RT	The residual time of the vehicle in the transmission range of the CH
NCC	The number of cluster changes per vehicle
$\lambda_{tr,mean}$	The average transition rate between clusters
$N_{i,life}$	The lifetime of vehicle $i$ during the simulation
$K$	The total number of vehicles in the system
$C_{avg}^{total}$	The average number of the total number of the created clusters

$$p_v(v) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(v-\mu)^2}{2\sigma^2}} \quad (1)$$

The speed difference,  $\Delta v$ , between a vehicle and its  $r$ -neighbor follows normal distribution with pdf given as:

$$p_{\Delta v}(\Delta v) = \frac{1}{\sigma_{\Delta v}\sqrt{2\pi}} e^{-\frac{(\Delta v - \mu_{\Delta v})^2}{2\sigma_{\Delta v}^2}} \quad (2)$$

Where  $\Delta v = v_1 - v_2$ ,  $\mu_{\Delta v} = \mu_1 - \mu_2$ , and  $\sigma_{\Delta v}^2 = \sigma_1^2 + \sigma_2^2$ . The probability that the speed difference between two  $r$ -neighbors falls within the threshold  $\Delta v_{th}$  can be obtained by:

$$p_{\Delta v}(-\Delta v_{th} < \Delta v < \Delta v_{th}) = \frac{1}{\sigma_{\Delta v}\sqrt{2\pi}} \int_{-\Delta v_{th}}^{\Delta v_{th}} e^{-\frac{(\Delta v - \mu_{\Delta v})^2}{2\sigma_{\Delta v}^2}} .d\Delta v \quad (3)$$

Note that, in (3), for a given  $\Delta v_{th}$ , the  $p_{\Delta v}$  value decreases as  $\sigma_{\Delta v}$  increases. Thus, the expected number of stable neighbors (SN) will vary. So, in order to avoid having high variation of this number, the threshold can be set as a function of the standard deviation, e.g.,  $\Delta v_{th} = \beta\sigma$ . Thus, the threshold is a dynamic parameter which

depends on the speed characteristics of the vehicles within the vicinity (Table 1).

The stable neighbors of a given vehicle might not be stable with respect to each others; thus they can't belong to the same cluster. Therefore, in order to partition the network into minimum number of clusters such that all cluster members are stable with respect to each other (fast moving vehicles in one cluster and slower moving vehicles in another cluster), not all vehicles are allowed to initiate the cluster formation process even though each vehicle can determine its stable neighbors. In the following section, we discuss which vehicle is a preferable one to initiate the clustering process.

#### 4. Clustering process and protocol structure

The inter-vehicle communication (IVC) operates in the 5.9 GHz band to support safety and non-safety applications. The dedicated short range communications (DSRC) uses 75 MHz bandwidth (5.850-5.925 GHz) which is divided into seven channels. One of the channels is called the control channel, and the remaining six are called service channels [26]. Vehicles are assumed to utilize the control channel to exchange periodic messages and gather information about their neighborhood, and use one service channel to define the cluster radius

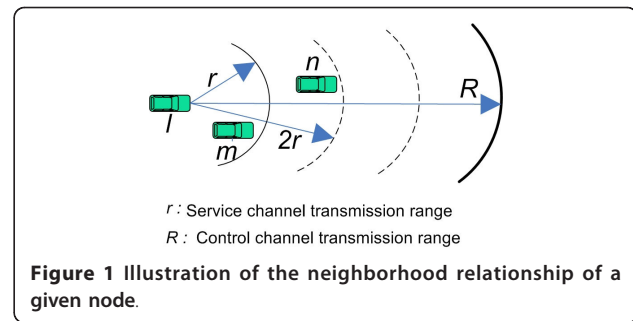
and perform all intra-cluster communication tasks. According to the DSRC specifications [26], the data link layer can provide a transmission range of up to 1,000 m for a channel. VANET applications can use a longer range,  $R$ , for the control channel so that a cluster-head can communicate with neighboring cluster-heads for safety message disseminations, and a shorter range,  $r$ , for a service channel that is used for intra-cluster managements. Using the control channel, vehicles can gather status information of other neighboring vehicles and then can build a complete picture about their neighbors which can even go beyond the cluster boundaries.

Since in our technique, slower vehicles will be in one cluster and faster vehicles will be in a different cluster, we can start the cluster formation process either from the slowest or fastest vehicle. For example, if we start with the slowest vehicle, then all the neighboring vehicles of this slowest vehicle that satisfy the speed threshold will be in the first cluster. The remaining vehicles will then go through the same cluster formation process to create other clusters. By extracting the velocity data embedded in the periodic messages, any vehicle can determine whether it has the *slowest velocity* among all its neighbors within  $R$  communication range. The *slowest vehicle*, in our method, is supposed to initiate the cluster formation process by sending a cluster formation request and *only* its stable neighbors participate in this process. The neighboring vehicles whose relative velocity, with respect to the slowest vehicle, is greater than the threshold,  $\Delta v_{th}$ , will not be grouped in the same cluster.

#### 4.1. Neighborhood relationship

The neighborhood term is directly associated with the transmission zone of the node. But, the DSRC is a multi-channel interface with different transmission ranges. Therefore, the neighborhood term needs to be re-defined according to the channel being used for the communications. To illustrate this, consider Figure 1 in which three vehicles  $l$ ,  $m$  and  $n$  are located within geographical area. For node  $l$ , node  $n$  is considered a neighbor from the perspective view of the control channel, but not a neighbor from the perspective view of the service channel because the distance to  $l$  is greater than  $r$  which is the maximum range of the service channel. Node  $m$  is considered a neighbor from the perspective view of both service and control channels.

As nodes exchange their status information via the control channel, it would be easy for node  $l$  to identify that  $n$  is within  $2r$  distance. Although neighborhood relationship is built using the *control channel*, it will be represented using *r-neighborhood* terminology. For example, node  $n$  is called a *2r-neighbor* because it's within  $2r$  distance.



#### 4.2. Cluster-head election parameters

The mobility information (velocity, location, node degree, and direction) of the nodes is exchanged via the control channel whose coverage area,  $R$ , is larger than that of the service channel,  $r$ , used to define the cluster boundary (radius). The mobility information of the  $2r$ -stable neighbors is needed for the vehicle to initiate the cluster formation request, while cluster-head election information for any node is limited to the nodes that are within  $r$  distance from the node itself.

The priority of a node to become a cluster-head is determined by its suitability value,  $u$ , which is computed based on the mobility information of its neighborhood. Thus,  $u = f(\mathbf{d}, \mathbf{v}, \mathbf{p})$  is a function defined according to the following criteria:

- The suitability value of the vehicle is calculated by considering the mobility information of its stable neighbors only.
- Nodes having higher number of stable neighbors, maintaining closer distances to their stable neighbors, and having closer speed to the average speed of their stable neighbors should have higher suitability value, thus they are more qualified to be elected as cluster-heads.

To calculate  $u$ , each vehicle has to find how close its position is to the mean position of all its  $\mathbf{d}$  stable neighbors. The vehicle also determines how close its velocity is to the mean velocity of all its  $\mathbf{d}$  stable neighbors. Since the distance of the vehicle to the mean position of its  $\mathbf{d}$  stable neighbors can have large values, it is necessary to use the normalization technique to avoid having this parameter dominate the results of the calculation. The normalized mean distance,  $p_{norm}$ , of a node to its  $\mathbf{d}$  stable neighbors can be found by having each node calculate the mean position,  $\mu_p$ , and the standard deviation,  $\sigma_p$ , of its all  $\mathbf{d}$  stable neighbors, thus, the  $p_{norm}$  is calculated by:

$$p_{norm} = \frac{n_i^{pos} - \mu_p}{\sigma_p} \quad (4)$$

Where  $n_i^{\text{pos}}$  is the position of the vehicle. The smaller the  $p_{\text{norm}}$  value, the closer the position of the vehicle to the mean position of its stable neighbors. The normalized mean speed  $v_{\text{norm}}$  can be calculated using the same way. The smaller the  $v_{\text{norm}}$  value, the closer the speed of the vehicle is to the mean speed of its neighbors. Finally,  $u$  can be calculated as follows:

$$u = d * e^{-\alpha w} \quad (5)$$

Where  $w = |p_{\text{norm}}| + |v_{\text{norm}}|$  and  $0 < \alpha \leq 1$  indicates the sensitivity of  $u$  to  $w$ . The higher the  $u$  the more qualified the node is to become a cluster-head. Figure 2 shows the impact of the mobility parameters on the suitability. The figure shows that the suitability of the node to win the cluster-head role decreases as the distance and speed to the  $d$  neighbors deviates very large from the mean.

### 4.3. The cluster formation algorithm

In order to execute the algorithm, each vehicle is assumed to maintain and update the two sets  $\Gamma(t)$  and the  $\Lambda(t)$  that contain the IDs of the  $2r$ -stable neighbors. At any time, there should be a vehicle whose speed is the slowest among its  $2r$ -stable neighbors, and as a result, the  $\Lambda(t)$  list maintained by this vehicle is empty. The pseudo code of the algorithm is shown in Algorithms 1-3. The algorithm basically requires that the slowest vehicle or the vehicle whose  $\Lambda(t)$  members belong to other clusters originates the cluster formation process. This vehicle is called the cluster originating vehicle (COV). Line 3 in Algorithm 1, shows that COV sends the Initiate Cluster(CID<sub>tmp</sub>) with its ID as a temporary cluster ID to all  $\Gamma(t)$ . Then, as shown in Algorithm 2, all  $\Gamma(t)$  non-clustered members react upon receiving this message by setting their cluster ID

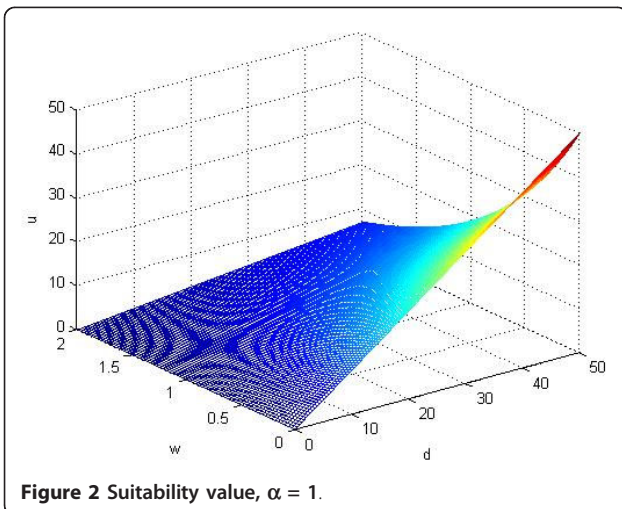


Figure 2 Suitability value,  $\alpha = 1$ .

temporarily to be the ID of the COV as shown in line 3. Vehicles start calculating their suitability to become a CH as shown in line 4. Then, the vehicle calculates  $T_{\text{wait}}$  before announcing its eligibility to become a cluster-head as shown in line 5. The vehicle waits for  $T_{\text{wait}}$  that is proportional to the suitability value of the vehicle. The higher the suitability value, the less the value of  $T_{\text{wait}}$ . This can be seen in lines 6 through 15. If the vehicle receives a Form Cluster(CH<sub>id</sub>) message from any other vehicle that belongs to  $\Gamma(t)$  before its waiting time,  $T_{\text{wait}}$  expires, then the vehicle determines that there are other vehicles belong to  $\Gamma(t)$  that are more suitable to win the CH role. Therefore, the vehicle quits the competition and processes the received message.

#### Algorithm 1 Initiating clustering process

1: **if** ( $\Lambda(t)$  is empty) || ( $\Lambda(t)$  members  $\in$  other clusters)  
**then**

2: CID<sub>tmp</sub>  $\leftarrow$   $v_j$ .id  
 3: send Initiate Cluster(CID<sub>tmp</sub>)

4: **end if**

#### Algorithm 2 CH competition and determination

1: **if**  $v_j \in \Gamma(t)$  **then**  
 2: **On Receiving** InitiateCluster(CID<sub>tmp</sub>)  
 3:  $v_j$ .CID  $\leftarrow$  CID<sub>tmp</sub>  
 4:  $v_j$ .Suitability() //w.r.t its  $r$ -neighbors that  $\in \Gamma(t)$   
 5:  $v_j$ . $T_{\text{wait}}$   $\leftarrow$   $v_j$ .DeferTime() //calculate the waiting time

6: **while**  $v_j$ . $T_{\text{wait}} > 0$  **do**  
 7: **if** FormCluster(CH<sub>id</sub>)isreceived **then**

8: **if** received CH<sub>id</sub>  $\in \Gamma(t)$  **then**  
 9: QuitCompetition() //give up CH competition

10: **Process** FormCluster(CH<sub>id</sub>) //process received message

11: **end if**

12: **else**

13: **Decrement**  $v_j$ . $T_{\text{wait}}$

14: **end if**

15: **end while**

16:  $v_j$ .STATUS  $\leftarrow$  CH

17: CH<sub>id</sub>  $\leftarrow$   $v_j$ .id //  $v_j$  declares itself as a CH

18:  $v_j$ .CID  $\leftarrow$  CH<sub>id</sub> //  $v_j$  sets its cluster id

19: **Send** FormCluster(CH<sub>id</sub>) //sends its cluster id to all vehicles

20: **end if**

This is shown in lines 7 through 11. If  $T_{\text{wait}}$  of the vehicle expires before any other vehicle sends the FormCluster(CH<sub>id</sub>) message, then the current vehicle wins the cluster-head competition, changes its state to a cluster-head, and sets the cluster ID to be its own ID. This is shown in lines 16 through 18. Finally, the vehicle sends the FormCluster(CH<sub>id</sub>) message with its own ID as the new cluster ID as shown in line 17. Algorithm 3,



shows the final stage of the clustering process. All vehicles in the  $\Gamma(t)$  of the COV receive the FormCluster ( $CH_{id}$ ) as shown in line 2. But, only  $r$ -stable neighbors of the winner (since the cluster boundary is defined by  $r$ ), which belong to the  $\Gamma(t)$  of the COV change their state to a CM and change their temporary cluster ID to be the

**Algorithm 3** Finalizing cluster formation process

```

1: if  $v_k \in \Gamma(t)$  then
2:   On Receiving FormCluster( $CH_{id}$ )
3:   if  $v_k \in \Gamma(t) \cap v_j.SN$  then
4:      $v_k.STATUS \leftarrow CM$  //become a Cluster Member
5:      $v_k.CID \leftarrow CH_{id}$ 
6:   else
7:      $v_j.CID \leftarrow default$  //the vehicle's id
8:     Reconstruct  $\Gamma(t)$ 
9:   end if
10: end if
    
```

new cluster ID embedded in the received FormCluster ( $CH_{id}$ ) as shown in lines 4 and 5. After that, the vehicle becomes a cluster member of the corresponding cluster. Vehicles that belong to  $\Gamma(t)$  of the COV and could not associate with the cluster being formed, set their temporary cluster ID to the default (their own ID), modify their  $\Gamma(t)$  and start the cluster formation process again, this is shown in lines 7 and 8. According to the proposed algorithm, vehicles wait for a period of time before accessing the media to announce their eligibility to be a cluster-head. Media access is controlled by the distributed coordination function (DCF) on the media access control (MAC) layer [27]. Usually, vehicles use the minimum Contention Window ( $CW_{min}$ ) size value before accessing the media, and they double this size for each unsuccessful transmission until they reach the max Contention Window size ( $CW_{max}$ ). In this work, vehicles wait for a period of time that is proportional to their suitability value,  $u$ , before announcing their suitability to be a cluster-head as follows:

$$T_{wait} = \left\lfloor \frac{N_{max} - u}{N_{max}} * (CW_{max} - CW_{min}) + CW_{min} \right\rfloor \quad (6)$$

where  $N_{max}$  is the total number of vehicles in  $\Gamma(t)$ ,  $u$  is the suitability value of the vehicle, and  $CW_{max}$  and  $CW_{min}$  are the maximum and the minimum contention window sizes respectively [27]. When there is more than one vehicle having the same  $T_{wait}$ , they will send the FormCluster( $CH_{id}$ ) to announce their eligibility to become a CH at the same time. As a result, a collision occurs and none of them wins the competition. In this case, only those collided vehicles start new iterations of competition until one of them wins or the maximum number of iterations is completed. If the max number of iterations is used and nodes still collide, then each

node picks a random number and the one with the smaller value wins the competition. The length of  $T_{wait}$  in iteration  $i$  is calculated as follows:

$$T_{wait} = \left\lfloor \left( 10^i \left( \frac{N_{max} - u}{N_{max}} \right) - \left\lfloor 10^i \left( \frac{N_{max} - u}{N_{max}} \right) \right\rfloor \right) * (CW_{max} - CW_{min}) + CW_{min} \right\rfloor \quad (7)$$

If the maximum number of iterations is used and nodes still collide, then each node picks a uniformly distributed random number between 0-9 and the one with the smaller value wins the competition. If the random numbers are the same, then the nodes will generate another pair and so on. Let  $s$  be the probability that a node will be able to announce its eligibility first time it generates a random number. The probability that a node will be able to announce its eligibility during the second time given the fact that it failed to announce during the first time is  $(1 - s)s$ . Similarly, the probability that a node will be successful during the third time given the fact that it failed during the first and second times is  $(1 - s)(1 - s)s$  and so on. The node has to generate random numbers  $\frac{1}{s}$  times before it can announce its eligibility. Therefore, if the node went through  $i$  iterations using Equation 7 before it started generating random numbers, the average number of trials for eligibility announcement is  $i + \frac{1}{s}$ .

#### 4.4. Cluster maintenance

Due to the high dynamic nature of the VANET, vehicles keep joining and leaving clusters frequently, thus, causing extra maintenance overhead. The events that trigger the maintenance procedure can be summarized as follows:

- *Joining a cluster:* when a standalone (non-clustered) vehicle comes within  $r$  distance from a nearby cluster-head, the cluster-head and the vehicle check whether their relative speed is within the threshold  $\pm \Delta v_{th}$ . If the speed difference is within  $\pm \Delta v_{th}$ , then the cluster-head will accept the vehicle and will add it to the cluster members list. If there are more than one cluster-heads in the vicinity that can be joined, the vehicle calculates the time, RT, it will remain in the transmission range  $r$  of these cluster-heads. The vehicle joins the cluster-head where it will stay for the longest period of time. The RT could be computed from the information about the relative speed, current location, and the transmission range  $r$  as follows:

- If the standalone vehicle is following the cluster-head and its velocity at time  $t$  is less than that of the cluster-head, then

$$RT(t) = \frac{r - \text{dis}(n, CH)}{\Delta v}$$

Where  $\Delta v$  is the speed difference, and  $\text{dis}(n, CH)$  is the distance between the standalone vehicle,  $n$ , and the cluster-head, CH. The above formula can also be used when the standalone vehicle is followed by the cluster-head but its velocity is greater.

- If the standalone vehicle is following the cluster-head and its velocity at time  $t$  is greater than that of the cluster-head, then

$$RT(t) = \frac{r + \text{dis}(n, CH)}{\Delta v}$$

This formula can also be used when the standalone vehicle is followed by the cluster-head but its velocity is less.

- *Leaving a cluster*: when a cluster member moves out of the cluster radius, it loses the contact with the cluster-head over the service channel,  $r$ . As a result, this vehicle is removed from the cluster members list maintained by the cluster-head. The vehicle changes its state to a standalone if there is no nearby cluster to join or there is no other nearby standalone vehicle to form a new cluster according to our cluster formation algorithm.
- *Cluster merging*: when two cluster heads come within each others transmission ranges and their relative speed is within the predefined threshold  $\Delta v_{th}$ , the cluster merging process takes place. The cluster-head vehicle that has less number of members gives up its cluster-head role and becomes a cluster-member in the new cluster. The other cluster members join that neighboring cluster if they are within the cluster-head's transmission range and the speed is within the threshold. If there is any other nearby clusters, then vehicles calculate their RT and join the cluster where they can stay for the longest period of time. Finally, vehicles that cannot merge with the cluster nor can join a nearby cluster, start clustering process to form a new cluster according to our algorithm.

## 5. Simulation and performance evaluation

An extensive simulation study was conducted to evaluate the performance of our protocol. The C++ was used to develop the simulation. In our simulation, we consider different road traffic and different network data parameters.

### 5.1. The mobility model

The mobility model used in this paper is built based on the car following model presented in [28]. In the car

following models, the behavior of the car is described based on the relation with respect to the car ahead. The speed,  $v_i(t)$ , and the acceleration,  $a$ , of the vehicle is a function of different factors, mainly the distance,  $\Delta x(t)$ , to the front vehicle and the speed of both vehicles at time  $t$ . In this model, the speed of the vehicle, called  $v_{i, safe}(t)$ , is computed such that a safety distance is maintained. The desired new speed  $v_{i, des}(t + \Delta t) = \min[v_{i, max}, v_i(t) + a\Delta t, v_{i, safe}(t + \Delta t)]$ , whereas  $v_{i, max}$  is the max allowed speed, and  $\Delta t$  is a time step. The speed of the following vehicle is determined according to the  $v_{i, des}$ .

For lane change, the methods proposed a safety requirement that must be satisfied as follows: if  $(|v_i(t)|_{L-b\Delta t} \leq |v_{i, safe}(t)|_L)$ , then it is safe to change to  $L$ , where  $L \in [\text{right}, \text{left}]$  lanes, and  $b$  is the deceleration. The equation means if the lane change is taking place, each vehicle should be able to brake with a finite  $b$  to avoid colliding with the vehicle ahead.

We simulated a 5-lane per direction highway. In the simulation, we monitor 400 vehicles on a highway of 15 km length for 650 s. The arrival rate of the vehicles follows the Poisson process. We simulated three types of vehicles' speed taken from statistical measurements [23-25]. The speed assigned to the vehicles follows the normal distribution with average  $\mu$  and standard deviation  $\sigma$  as shown in Table 2. In our simulation, the speed of the vehicle is determined according to the rules mentioned above. Vehicles can change their lanes if there is a room in the next lane and the rule for lane change mentioned early is satisfied. The density of the vehicles varies between (13-21 vehicle/km/Lane) depending on the speed being used. For all simulation scenarios, the  $\Delta v_{th} = \sigma$ , e.g., for  $\mu = 70$  km/h and  $\sigma = 21$  km/h, the  $\Delta v_{th} = 21$ . The performance of different  $\Delta v_{th}$  values can be found in [29].

### 5.2. Network parameters

We used different network parameters in the simulation. The data rate is set to 6 Mbps and the periodic messages are sent every 100 ms, the size of the message including the mobility information is 100 bytes. DSRC standard supports data rate in the range 6-27 Mbps [26]. However, various members of the vehicle infrastructure integration (VII) Consortium use 6 Mbps data rate [30,31] for road testing. Thus, we also decided to

**Table 2 The average and the standard deviation of the speed**

$\mu(Km/h)$	$\sigma(Km/h)$
70	21
90	27
110	33

use 6 Mbps data rate. To study the performance of the clustering techniques for different cluster sizes, we used different transmission ranges for  $r$  and  $R$ . The transmission range for  $r$  was varied between 150 and 300 m, while it is between 800 and 1,000 m for  $R$ . For media access, we used the IEEE802.11 standard [27]. We set the  $CW_{min} = 15$ ,  $CW_{max} = 1023$ , a Slot Time =  $16\mu s$ , SIFS =  $32\mu s$ , and DIFS =  $64\mu s$ .

### 5.3. Evaluation criteria

To show the performance of our proposed threshold-based (TB) technique, we compare it with the weight-based (WB) and the position-based (PB) methods proposed in [14] and [18] respectively. Originally, the WB method for MANET was proposed by Chatterjee et al. [32] by introducing the combined weight metric. The algorithm assigns *node weights* based on the suitability of a node being a cluster-head. This algorithm basically takes into consideration the nodal degree, the transmission power, mobility, and battery power of the mobile nodes. Each one of these parameters is assigned a weight; the sum of these weights is 1. Then, the value of each parameter is multiplied by its weight and all the values are finally summed to produce the combined weight. The node with the lower combined weight is more suitable to become a cluster-head. The same algorithm was adopted by VANET clustering techniques [14], but without considering the battery power factor since it is not a crucial problem in VANET. In the simulation, we assigned all WB method parameters equal weights. For the PB method, the priority of the node is calculated based on the eligibility function. A Node having longer travel time has higher eligibility value, and this value decreases as the velocity of the node deviates largely from the average speed. We compare the three methods under the same environment variables. Each simulation run was repeated 10 times with different random seeds and the collected data was averaged over those runs.

### Cluster stability

A clustering structure should be stable with respect to the nodes' motion, i.e., the cluster configuration should not change too much while the topology changes. In a high dynamic VANET, vehicles keep joining and leaving clusters along their travel route, and the number of cluster changes (NCC) of the vehicle will vary depending on the clustering algorithms being used. Good clustering algorithms should be designed to minimize the number of cluster changes of the vehicle by minimizing vehicle transitions between clusters. The *NCC* of the vehicle during its lifetime can be used to evaluate the cluster stability. To find the *NCC* of the vehicle, we first introduce the basic transition events the vehicle encounters during its lifetime:

- e1: A vehicle leaves its cluster and forms a new one.
- e2: A vehicle leaves its cluster and joins a nearby cluster.
- e3: A cluster-head merges with a nearby cluster.

For each vehicle, the sum of all transition events (e1, e2, and e3) defines the *NCC* of the vehicles over its lifetime. We compare the average *NCC* of the vehicles for the TB, WB, and PB methods when different speeds and different transmission ranges are used. In Figure 3a-c, the  $x$ -axis represents the transmission range, while the  $y$ -axis represents the average *NCC* of the vehicle. From Figure 3a-c, we can see that the average *NCC* produced by our TB technique is smaller compared to that produced by the WB and PB methods. This means our technique causes less number of cluster transitions for all different velocities and different transmission ranges. The figure shows that the average *NCC* of a vehicle is reduced by 34-46% compared to the WB and PB methods. We can see that the TB method performs even much better when the average speed becomes higher. Note also that the average speed

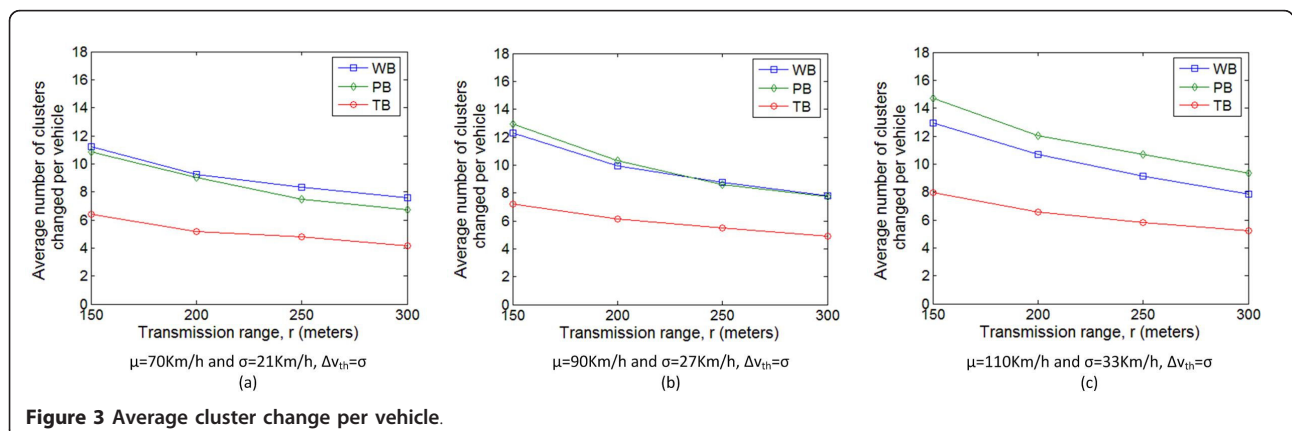


Figure 3 Average cluster change per vehicle.

increase has little impact on the number of clusters changed per vehicle when the TB method is used. This is because the threshold is a function of the speed deviation and it is always proportional to the speed regardless of its average value. The figures show that the average  $NCC$  of the vehicle decreases as the transmission range increases. This is because increasing the transmission range  $r$ , increases the probability that a vehicle stay connected with its cluster-head. The cluster stability can also influence the signaling overhead. A frequently changing clustering structure results in an increase in maintenance messages and thus increasing the load on nodes. From the figure, we can conclude that the TB method reduces the signaling overhead and the traffic load since it causes less number of transition between the clusters. We can also calculate  $\lambda_{tr,mean}$  between clusters as follows:

$$\lambda_{tr,mean} = \frac{1}{K} \sum_{i=1}^K \frac{NCC_i}{N_{i,life}}$$

where  $NCC_i$  is the number of clusters vehicle  $i$  changes during its lifetime.

**Average cluster lifetime**

The average cluster lifetime is an important metric that shows the performance of the clustering algorithm. The cluster lifetime is directly related to the lifetime of its cluster-head. The cluster-head lifetime is defined as the time period from the moment when a vehicle becomes a cluster-head to the time when it is merged with a nearby cluster.

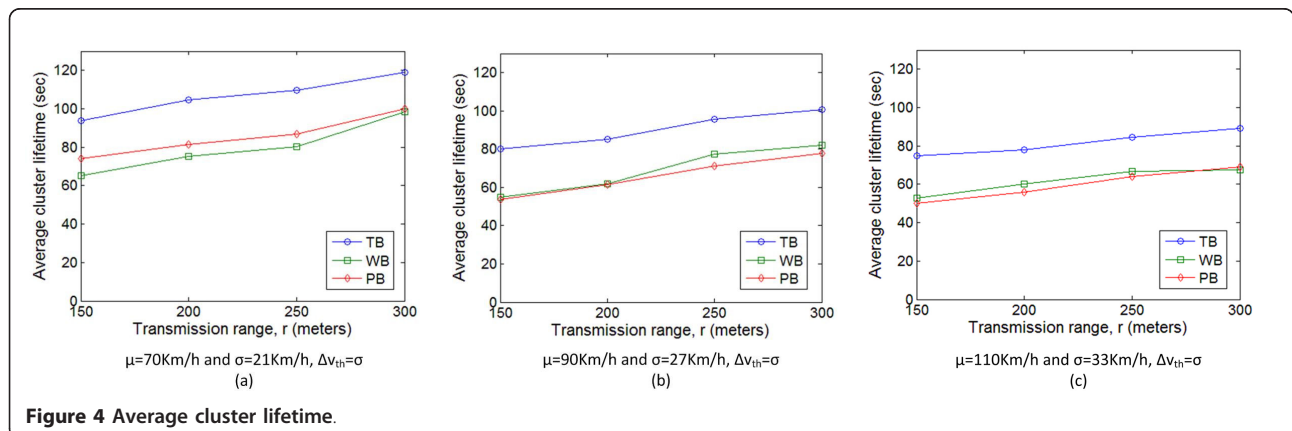
The average cluster lifetime produced by the TB, the WB and the PB methods is compared in different speed scenarios with different transmission ranges. Figure 4a-c shows that the average cluster lifetime is increased by 20-48% when the TB method is used compared to the WB and PB methods. This is due to the high variation of the speed difference among cluster members of the WB and the PB methods. This deviation leads to the

following: first, in both methods, the probability that two cluster heads come into direct communication range is high which results in cluster merging. But, in the TB method, the cluster merging cannot be performed unless the difference between the average speed of the cluster heads of both clusters are within the pre-defined threshold; second, the probability that the cluster members and the cluster-head get separated soon due to high mobility; especially when the cluster is composed of few nodes.

**Number of clusters**

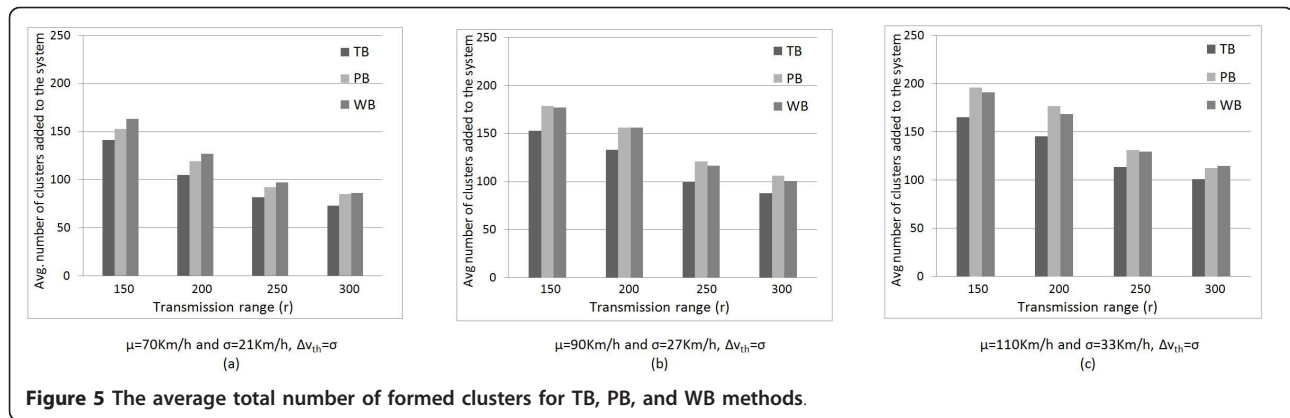
Due to high dynamics of the VANET, clusters are created (new clusters added to the system) and vanished over time, and the total number of clusters created over a period of time defines the cluster formation rate. Good clustering algorithms should be designed to reduce the rate at which clusters are created and added to the system due to the mobility of the nodes. And this can be achieved by producing relatively stable clusters and by the ability of clustering method to maintain the current cluster structure stable as much as possible. In this paper, we compare the average number of clusters added to the system, we start counting each new cluster added to the system after the algorithm is executed by all nodes and the clusters are formed (e.g., when nodes leave their current clusters due to mobility and form a new cluster, or when two neighboring clusters merge to produce a new cluster). To evaluate this metric, the total number of clusters created and added is calculated for each run, then,  $C_{avg}^{total}$ , of all methods is taken over all runs for different transmission ranges.

Figure 5a-c shows the  $C_{avg}^{total}$  added to the system over all simulation runs for different speeds and different transmission ranges. The figure shows that the  $C_{avg}^{total}$  produced by the TB method is always smaller compared to that produced by the WB and the PB methods and this number decreases as the transmission range



**Figure 4** Average cluster lifetime.





increases. This is because the TB method uses the speed difference among vehicles as a parameter to create the clusters. Thus, the clusters are more stable and have longer lifetime.

#### Overhead for clustering

All clustering algorithms incur some additional signaling overhead to form and maintain their cluster structures. The clustering overhead consists of: HELLO packets overhead, cluster setup overhead and cluster maintenance overhead.

#### Overhead due to HELLO packets

HELLO packets are broadcast by vehicles every  $T_{\text{HELLO}}$  period. These packets carry local mobility information used to compute local variability, which will be used in cluster formation and cluster-head election. Each node sends one HELLO packet every  $T_{\text{HELLO}}$  period to maintain up-to-date neighborhood information. Thus, this overhead is the same for TB, WB and PB clustering techniques.

#### Overhead due to cluster setup

According to the TB cluster formation algorithm, the COV node sends one message to initiate cluster formation process (*InitiateCluster*). After receiving this message, the node that wins the cluster-head competition broadcasts a cluster formation message (*FormCluster*) to its neighbors with its ID embedded in the message. So for the cluster formation process, two messages are sent: one by the COV and the other one by the cluster-head winner node. Each non-clustered neighbor that satisfies the speed threshold joins this cluster by sending a message. So in the TB algorithm, if the average number of nodes in a cluster is  $k_{\text{TB}}$ , then the total number of messages to setup a cluster is  $2+k_{\text{TB}}$ . For the PB algorithm, when a new node is powered up and none of its neighbors belong to other clusters, it announces itself as a cluster-head and sends a message to inform its neighbors about its new role. Neighbors that are in the registration phase (non-clustered) join this cluster by sending a join message. So in the PB

algorithm, the total number of messages to setup a cluster is  $1+k_{\text{PB}}$ , where  $k_{\text{PB}}$  is the average number of members per PB cluster. In the WB algorithm, a node that claims to be a cluster-head sends a CH-HELLO message. All non-clustered neighbors join this cluster by sending a message. So in the WB algorithm, the total number of messages to setup a cluster is  $1+k_{\text{WB}}$ , where  $k_{\text{WB}}$  is the average number of members per WB cluster. In the TB technique the average number of nodes per cluster is less than that of the other two techniques. So if a TB cluster has at least two less members than the other two types of clusters, then the cluster setup overhead per cluster is less in TB technique than in other techniques.

#### Overhead due to cluster maintenance

Cluster maintenance is done periodically by all clustering methods. The three types of events that trigger topology change in VANET can be defined as follows: a node joins the network, a node leaves the cluster, and two cluster heads come into direct communication range. If the new node, that joins the network, has non-clustered neighbors, then those nodes will form a new cluster according to the rules used by each clustering method. The overhead of cluster formation was explained earlier. However, if the new node has a neighbor that is a cluster-head, then it will try to join the cluster by sending a join message to the cluster-head, and this cluster joining overhead is same for all three methods (TB, PB and WB). When two neighboring clusters merge, the cluster-head with less number of members will lose its role and join the other cluster and become a cluster member. The losing node sends one message in one period to inform its members about its decision. If the losing node has cluster members, then the members are subject to cluster reorganization. The cluster members either join any nearby clusters or form a new cluster if they could not find a cluster to join. Overhead for joining any nearby clusters is the same for all three methods, and the overhead for cluster formation (cluster setup) is already presented



before. The upper bound on the number of messages for cluster merging is equal to the average number of members per cluster, which is  $k_{TB}$ ,  $k_{WB}$  and  $k_{PB}$  for TB, WB and PB techniques respectively.

## 6. Conclusion

VANETs are characterized by high node dynamics. Therefore, clustering methods should be designed to adapt to the VANET environment. These methods should take into account all vehicle dynamics. In this paper, we proposed a new VANET cluster formation algorithm that tends to group vehicles showing similar mobility patterns in one cluster. This algorithm takes into account the speed difference among vehicles as well as the position and the direction during the cluster formation process. After conducting a simulation experiment, we observe that our technique groups fast moving vehicles on the fast speed lanes in one cluster, while slow moving vehicles in another cluster. The simulation results show that our proposed algorithm increases the cluster lifetime and reduces vehicle transitions between clusters. The results show that our technique significantly increases the stability of the global network topology by reducing the rate at which clusters are created.

### Competing interests

The authors declare that they have no competing interests.

Received: 25 November 2010 Accepted: 16 January 2012

Published: 16 January 2012

### References

1. IEEE 802.11 TGP, Wireless Access for Vehicular Environment (WAVE)
2. NOW (Network On Wheels) Project, Available online at <http://www.network-on-wheels.de/>
3. Fleetnet Project, <http://www.et2.tu-harburg.de/fleetnet/>
4. T ElBatt, SK Goel, G Holland, H Krishnan, J Parikh, Cooperative collision warning using dedicated short range wireless communications, in *Proceedings of ACM VANET*, New York, pp. 1–9 (2006)
5. Q Xu, T Mark, J Ko, R Sengupta, Medium access control protocol design for vehicle-vehicle safety messages. *IEEE Trans Veh Technol.* **56**(2), 499–518 (2007)
6. S Dashtinezhad, T Nadeem, B Dorohonceanu, C Borcea, P Kang, L Iftode, TrafficView: a driver assistant device for traffic monitoring based on car-to-car communication, in *Proceedings of the IEEE Semiannual Vehicular Technology Conference*, Milan, Italy, (2004)
7. T Nadeem, S Dashtinezhad, C Liao, L Iftode, TrafficView: Traffic data dissemination using car-to-car communication. *ACM Mobile Comput Commun Rev (MC2R)*. **8**(3), 6–19 (2004). doi:10.1145/1031483.1031487
8. RA Santos, RM Edwards, NL Seed, Supporting inter-vehicular and vehicle-roadside communications over a cluster-based wireless ad-hoc routing algorithm, in *Proceedings of the Winter International Symposium on Information and Communication Technologies(WISICT'04)*, Trinity College, Dublin, pp. 1–6 (2004)
9. M Aoki, H Fuji, Inter-vehicle communication: Technical issues on vehicle control application. *IEEE Commun Mag.* **34**(10), 90–93 (1996). doi:10.1109/35.544327
10. H Alshaer, E Horlait, An optimized adaptive broadcast scheme for inter-vehicle communication, in *Proceedings of the 61 st IEEE Vehicular Technology Conference (VTC 2005, Spring)*, Stockholm, Sweden, (2005)
11. G Korkmaz, E Ekici, F Ozguner, An efficient fully ad-hoc multihop broadcast protocol for inter-vehicular communication systems, in *Proceedings of the IEEE ICC'06*, Istanbul, Turkey, (2006)
12. W Chen, S Cai, Ad hoc peer-to-peer network architecture for vehicle safety communications. *IEEE Commun Mag*, **4**: 100–107 (2005)
13. RA Santos, RM Edwards, NL Seed, Inter vehicular data exchange between fast moving road traffic using ad-hoc cluster based location algorithm and 802.11b direct sequence spread spectrum radio, (Post-Graduate Networking Conference, 2003)
14. G Yvonne, W Bernhard, G Hans Peter, Medium access concept for VANETs based on clustering, in *Proceedings of the 66th IEEE Vehicular Technology Conference (VTC)*, vol. 30. Baltimore, pp. 2189–2193 (2007)
15. P Fan, Improving broadcasting performance by clustering with stability for inter-vehicle communication, in *Proceedings of the 65th IEEE Vehicular Technology Conference (VTC '07, Spring)*, Dublin, Ireland, (2007)
16. P Fan, P Sistla, P Nelson, Theoretical analysis of a directional stability-based clustering algorithm for VANETs, in *Proceedings of the Fifth ACM International Workshop on Vehicular Ad Hoc Networks (VANET)*, San Francisco, CA, (2008)
17. H Su, X Zhang, Clustering-based multichannel MAC protocols for QoS provisionings over vehicular ad hoc networks. *IEEE Trans Veh Technol.* **56**(6), 3309–3323 (2007)
18. W Zhiagang, L Lichuan, Z MengChu, A Nirwan, A position-based clustering technique for ad hoc intervehicle communication. *IEEE Trans Man Cybern.* **38**(2), 201–208 (2008)
19. O Kayis, T Acarman, Clustering formation for inter-vehicle communication, in *Proceedings of the 10th IEEE Intelligent Transportation Systems Conference (ITSC'07)*, Seattle, WA, (30 September – 3 October 2007)
20. C Shea, B Hassanabadi, S Valae, Mobility-based clustering in VANETs using affinity propagation, in *Proceedings of the IEEE Globecom*, Honolulu, HI, (30 November – 4 December 2009)
21. P Basu, N Khan, T Little, A Mobility Based Metric for Clustering in Mobile Ad Hoc Networks, in *Proceedings of the 21st International Conference on Distributed Computing Systems (ICDCSW 01)*, IEEE Computer Society, Washington, DC, p. 413 (2001)
22. E de Souza, I Nikolaidis, P Gburzynski, A new aggregate local mobility (ALM) clustering algorithm for VANETs, in *Proceedings of the IEEE ICC'10*, Cape Town, South Africa, p. 10 (May 2010)
23. W Schnabel, D Lohse, Grundlagen der Straenverkehrstechnik und der Verkehrsplanung, (Bd. 1, 2. Aufl., Verlag fr Bauwesen, Berlin, 1997)
24. S Yousefi, E Altman, R El-Azouzi, M Fathy, Analytical model for connectivity ad hoc networks. *IEEE Trans Veh Technol.* **57**(6), 3341–3356 (2008)
25. M Rudack, M Meincke, M Lott, On the dynamics of ad hoc networks for inter-vehicles communications (IVC). in *Proceedings of the ICWN 40–48* (2002)
26. ASTM 2213-03, 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 (2003)
27. IEEE Standard 802.11-1999, Wireless LAN medium access control (MAC) and physical layer (PHY) specifications (1999)
28. S Krauß, Microscopic modeling of traffic flow: investigation of collision free vehicle dynamics, PhD thesis, Mathematisches Institute, Universitat zu Koln. (1998)
29. ZY Rawashdeh, SM Mahmud, Toward strongly connected clustering structure in vehicular ad hoc networks, in *Proceedings of the 2009 IEEE 70th Vehicular Technology Conference (VTC2009-Fall)*, Anchorage, AK, (20–23 September 2009)
30. R Miucic, Z Popovic, SM Mahmud, Experimental characterization of DSRC signal strength drops, in *Proceedings of the 12th International IEEE Conference on Intelligent Transportation Systems*, St. Louis, MO, (3–7 October 2009)
31. Vehicle infrastructure integration (VII) California demonstration evaluation, Final Evaluation Report, Kimley-Horn and Associates, Inc. (11 July 2006) Available online at [http://www.vicalifornia.org/publications/vii\\_demonstration\\_evaluation.pdf/](http://www.vicalifornia.org/publications/vii_demonstration_evaluation.pdf/)
32. M Chatterjee, SK Sas, D Turgut, An on-demand weighted clustering algorithm (WCA) for ad hoc networks, in *Proceedings of the 43rd IEEE Global Telecommunications Conference*, San Francisco, CA, pp. 1697–1701 (2010)

doi:10.1186/1687-1499-2012-15

Cite this article as: Rawashdeh and Mahmud: A novel algorithm to form stable clusters in vehicular ad hoc networks on highways. *EURASIP Journal on Wireless Communications and Networking* 2012 **2012**:15.