RESEARCH　　　　　　　　　　　　　　　　　　　　　　　　　　**Open Access**

Clustering and OFDMA-based MAC protocol (COMAC) for vehicular ad hoc networks

Khalid Abdel Hafeez[*], Lian Zhao, Zaiyi Liao and Bobby Ngok-Wah Ma

Abstract

The IEEE community is working on the wireless access in vehicular environments as a main technology for vehicular ad hoc networks. The medium access control (MAC) protocol of this system known as IEEE 802.11p is based on the distributed coordination function (DCF) of the IEEE 802.11 and enhanced DCF of the IEEE 802.11e that have low performance especially in high-density networks with nodes of high mobility. In this paper, we propose a novel MAC protocol where nodes dynamically organize themselves into clusters. Cluster heads are elected based on their stability on the road with minimal overhead since all clustering information is embedded in control channel's safety messages. The proposed MAC protocol is adaptable to drivers' behavior on the road and has learning mechanism for predicting the future speed and position of all cluster members using the fuzzy logic inference system. By using OFDMA, each cluster will use a set of subcarriers that are different from the neighboring clusters to eliminate the hidden terminal problem. Increasing the system reliability, reducing the time delay for vehicular safety applications and efficiently clustering vehicles in highly dynamic and dense networks in a distributed manner are the main contributions of our proposed MAC protocol.

Keywords: vehicular ad hoc network (VANET), medium access control, clustering; mobility, reliability; fuzzy logic

1. Introduction

The increase in number of vehicles on our roads and the immense number of fatal accidents they cause have driven the research and development of new-generation technologies that help drivers travel more safely. One major cause to traffic accidents is that drivers cannot consistently respond to the changing road condition appropriately. In fact, most accidents could be avoided if drivers could obtain and use relevant information of the traffic that is beyond their vision using wireless communication technology. In recognition to this problem, the IEEE community is working on the standardization of IEEE802.11p [1], which is intended to enhance the IEEE 802.11 to support vehicular ad hoc networks (VANETs) applications where reliability and low latency are crucial.

The IEEE 802.11p uses carrier sense multiple access with collision avoidance (CSMA/CA) as the basic medium access scheme in the licensed ITS 5.9 GHz (5.850-5.925 GHz) band in North America. The 75 MHz spectrum is divided into seven 10 MHz channels and a 5

MHz guard band. The control channel (CCH), channel 178, will be used for safety-related applications and system control management. The other six channels are service channels (SCH) dedicated for non-safety and commercial applications. Vehicles will alternate between the CCH channel and one or more of the SCH channels.

The standard assumes that all vehicles will be synchronized to a common time through an external system like global positioning system (GPS). Although the interval of synchronization (SI) is not specified by the standard, it is selected to be 100 ms in most safety-related applications. At the beginning of this interval, vehicles will synchronize to the control channel for a period called control channel interval CCI. The remaining time is called service channel interval SCI, where vehicles synchronize to one of the service channels, such that SI = CCI+SCI.

Vehicles will be equipped with sensors and GPS systems to collect information about their position, speed, acceleration and direction to be broadcasted to all vehicles within their range. Based on this information, drivers can better operate vehicles to avoid potential

* Correspondence: kabdelha@ryerson.ca
Electrical and Computer Engineering Department Ryerson University, Toronto, ON M5B 2K3, Canada

dangers. In this scenario, all vehicles should have fair access to the control channel such that all safety-related messages are present to all vehicles that are all made visible to every individual driver in the range.

Since most VANETs' applications are broadcasting in nature, vehicles will not send an acknowledgement (ACK) for the received broadcast messages. Therefore, the transmitter cannot detect whether a packet is received properly and hence will not resend the packet. As VANETs tend to grow in terms of number of vehicles within a certain geographical area, their applications that use broadcasting will face a challenge in managing the wireless channel capacity in terms of throughput, fairness and time delay. This is because the IEEE 802.11p uses the DCF as a MAC protocol, which is known to have a poor performance such as unbounded channel access delay and consecutive packet drops as the number of nodes increases within the communication range.

Since vehicular safety applications have strict requirements on reliability and low latency, VANETs should be self-organized and provide a distributed channel access to all nodes within the communication range. It also implies the need for ad hoc mode to support vehicle-to-vehicle (V2V) communication or intervehicle communication (IVC). In fact, the efficiency of VANETs depends on the performance and reliability of their MAC protocol, which must be decentralized to fit their ad hoc nature. The MAC protocol should cope with the fast-changing topology of VANETs and their uneven node density on the road. The vehicle density on the road varies with time and location. In some congested areas, the number of vehicles that contend for the channel is high, which results in deteriorating the DCF performance. However, in low-density areas, nodes may struggle to find a path between a source and a destination and to maintain the link between them for the whole period of communication.

To solve the aforementioned problems, we propose a novel MAC protocol called clustering and OFDMA-based MAC (COMAC) protocol where nodes dynamically organize themselves into clusters. Cluster heads are elected based on their stability on the road and with minimal overhead since clustering information is embedded in vehicles' periodic status messages. The COMAC protocol takes advantage of the OFDMA scheme and works under the IEEE 802.11p standard. We divide the control channel subcarriers into four groups. Each cluster will use a set of subcarriers that are different from the neighboring clusters to eliminate the hidden terminal problem and hence increase the system reliability and decrease the time delay for safety messages. The COMAC protocol is adaptable to drivers' behavior on the road and has a learning mechanism for

predicting the future speed and position of all cluster members using the fuzzy logic inference system (FIS). This makes the proposed protocol more efficient in maintaining the cluster topology and increases the life time of the elected cluster head and its members.

The rest of this paper is organized as follows: Section 2 presents a review of the significant contributions in the scope of VANETs MAC protocols found in the literature. The characterization of our COMAC protocol and its algorithms are introduced in Section 3. In Section 4, we analyze the proposed MAC protocol in terms of time delay, reliability, stability and network convergence. We present our simulation results in Section 5 and conclude this paper in Section 6.

2. Related work

Most of vehicular safety applications proposed in the literature rely on the IEEE 802.11p standard, which uses the DCF as its MAC protocol. The authors in [2-7] studied and evaluated the IEEE 802.11p for VANETs. They showed that this protocol has problems in predictability, fairness, low throughput and high collision rate especially in high-density networks. Due to these problems, many of the proposed solutions are based on time division multiple access (TDMA) where the channel is divided into time slots and each node is granted access during one or more of these time slots. In [8], the authors proposed a decentralized TDMA-based MAC protocol but did not specify how to synchronize the TDMA time slots among all vehicles within the range by using only one wireless channel. The authors in [9] proposed a self-organizing time division multiple access (STDMA) MAC protocol to grant channel access to all nodes within the range. In ADHOC MAC [10], the time is divided into frames, and each frame has a fixed number of slots where nodes can only reserve one or more of the free unreserved slots. However, in TDMA, strict synchronization and large overhead are needed between all nodes, and the system can only handle a limited number of vehicles within the range. This is a problem in VANETs where MAC protocol has to scale well since the number of nodes are not limited and vehicles can enter and leave the network at any time.

In [11,12], the authors proposed a space division multiple access (SDMA) scheme where the road is divided into small cells. Each cell is large enough to occupy only one vehicle. For each cell, they assigned a time slot, frequency band or a code for the vehicle in that cell to use. This scheme has poor efficiency since most of the cells are empty especially in low-density networks and suffer from the location error problem.

A few clustering-based schemes have been proposed by [13] and [14] where nodes in the network are grouped into clusters. In [13], the authors proposed a

clustering-based MAC multichannel protocol (CMCP) where each node is armed by two transceivers which they assume that they can operate simultaneously on different channels. One transceiver is used for the communication between cluster members, while the other is used to communicate with the cluster head on a different channel. Inside the cluster, the cluster head organizes the channel access between member nodes by using TDMA using one of its transceivers and different CDMA code. The other transceiver is used to communicate with the neighboring cluster heads by using the DCF of IEEE 802.11 on a different channel. This system has a very high cost and needs a very strict synchronization between all nodes in the network. Moreover, the system has a break point since all communications were done through the cluster head which uses both of its transceivers in communication with its cluster members and the neighboring cluster heads.

Since the communication requirements of VANETs' safety applications are complex and demand high throughput, reliability and bounded time delay concurrently, the design of their MAC protocol is a challenge especially in high-density scenarios where the number of nodes contending for the channel use is large. It is clear from previous studies that using TDMA or STDMA need strict synchronization and complete premapping of geographical locations to TDMA slots, but they are fair and have predictable delay. On the other hand, using CSMA scheme is less complex, supports variable packet sizes and requires no strict synchronization but has problems such as unbounded time delay and consecutive packet drops especially in high-density networks. Therefore, clustering is used to limit channel contention, provide fair channel access within the cluster, increase the network capacity by the spatial reuse of network resources and effectively control the network topology. The main challenge in clustering is the overhead introduced to elect the cluster head and maintain the membership in a highly dynamic and fast-changing topology such as in VANETs. Therefore, we propose a distributed and dynamic cluster-based MAC protocol called COMAC, which integrates OFDMA with the contention-based DCF algorithm in IEEE 802.11p. In COMAC, the network is dynamically organized into clusters where cluster memberships are changing overtime in response to vehicles mobility and density on the road. Cluster head is elected based on a stability criteria and could be taken over by another member if its stability factor has fallen below certain threshold. The proposed MAC protocol is adaptable to drivers' behavior and has a learning mechanism to predict the future speed and position of all cluster members using the fuzzy logic inference system. In COMAC, the OFDMA subcarriers of the IEEE 802.11p CCH channel are divided into four

sets, and cluster members can use only one set within their cluster. COMAC is designed to fit under the IEEE 802.11p spectrum and specifications. In our COMAC, we assume that all vehicles are moving in one direction, i.e., one-way multilane highway segment.

3. COMAC PROTOCOL

Our proposed MAC protocol aims to make a large network with highly dynamic nodes that appear smaller and more stable, to increase the system reliability and to reduce the time delay in real-time applications. The main idea of our COMAC is to partition the network into clusters of nodes that are all reachable by their cluster head. Vehicles are equipped by one transceiver that can work in omnidirectional and directional modes. Vehicles are also equipped with the global positioning system (GPS) for positioning and time synchronization purposes. Vehicles will alternate between the CCH channel and one or more of the service channels every 100 ms. While they are synchronized to the CCH channel, vehicles transmit and receive their control and safety messages in omnidirectional mode. On the other hand, they could use directional mode when they are synchronized to one of the SCH channels. We assume that all vehicles within a cluster will have the same communication range (R), i.e., they use the same transmitting power (P_t), except for the cluster head that has two levels of power: one level of P_t, which is the same as other members and dedicated to communicate with its cluster members, and a second level of power that is enough to reach a distance of $2R$ to communicate with neighboring cluster heads.

The COMAC use OFDMA where the CCH channel subcarriers are divided into four sets (c_1, c_2, c_3, c_4). The first three sets can be used by clusters where each cluster has to select different set from its neighboring clusters as shown in Figure 1. The fourth set (c_4) is temporary and can be used only by a node that cannot join a cluster or a node that is moved out from a cluster and cannot communicate any more with its former cluster head. The use of c_4 is temporary; once a node falls again within the range of a cluster head, it releases c_4 and starts to use the same set as the new cluster head. The algorithm of selecting the subcarriers set will be explained in a later subsection.

A. Clustering in COMAC

The clustering algorithm is the most important component in any clustering-based MAC protocol. The faster the nodes are clustered around their elected cluster head and the less often they reelect a new cluster head, the more the network will appear small and static. In COMAC, each vehicle has its own unique ID and collect information such as speed, acceleration and direction

Figure 1 Subcarriers assignment to clusters.
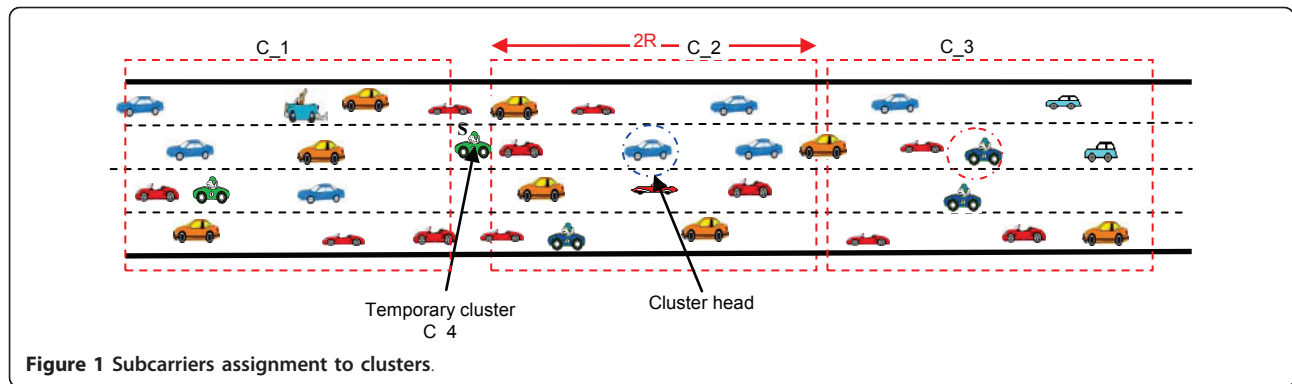
from its internal sensor network and its position from GPS system, which is also used for time synchronization. The vehicle will also calculate its weighted stabilization factor (SF_w), which is a function of the change in its relative speed and direction compared to its neighbors for the time it has been on the road. A vehicle with higher (SF_w) is more likely to be elected as a cluster head. Calculation of the parameter (SF_w) will be explained in a later subsection.

Vehicles synchronize their time with the GPS while they enter the network for the first time. At the beginning of every synchronization interval (SI), all vehicles will synchronize to the CCH channel to exchange their status messages. The status message will contain information about its type (*Type*), the vehicle's (*ID*), its (SF_w) factor, current speed (*v*), current position (Pos), acceleration (*a*) during the next period (T_f), communication range (*R*), cluster head's ID (CHID) and the backup cluster head's ID (CHBK) as shown in Figure 2. The acceleration will help to determine the future values of the vehicle's speed and position and will be determined in a later subsection. The field Type has four values: 0 is for cluster member's status message; 1 is for cluster head's first message; 2 is for cluster head's invitation message; 3 is for cluster head's last message.

The vehicle will first listen to the channel for a random length of time from [0-CCI] to check whether there are other vehicles on the network and do one of the following:

(1) If there are no other vehicles or it does not lie within the communication range of the neighboring cluster heads (lone state), it will start transmitting its status messages using the temporary subcarriers set c_4 and set the fields CHID = CHBK = 0 in its status message.

(2) If it encounters other vehicles using the same temporary set c_4 without an elected cluster head, they will start forming a temporary cluster. The vehicle with the highest SF_w will be elected as the cluster head, and if more than one vehicle have the same SF_w, they will elect the vehicle with the highest ID. The vehicle that happened to be located within the range of two or more cluster heads will select to join the cluster with the closest cluster head. The change in the cluster's status from temporary to main cluster depends on the status of the neighboring clusters and will be explained in a later subsection.

(3) If the vehicle hears other vehicles on the road whose status messages contain a cluster head ID, it will join that cluster if it is located within its cluster head's range. The vehicle will set its field CHID to the cluster head's ID and send its status message when it receives the cluster head's invitation message or the channel is being idle for time $T_w(d)$ as in Equation (4) which will be introduced in Subsection 3-D.

(4) If the vehicle moves out of its cluster head's range, it will wait for certain number of SI intervals, which is three in our protocol, before it gives up the subcarriers set that it was using in the previous cluster. The vehicle will look for a new or a temporary cluster to join as in step 1. Figure 3 depicts the finite state machine dictating the state of any COMAC node.

Type	ID	SF_w	v	Pos	a	R	CHID	CHBK

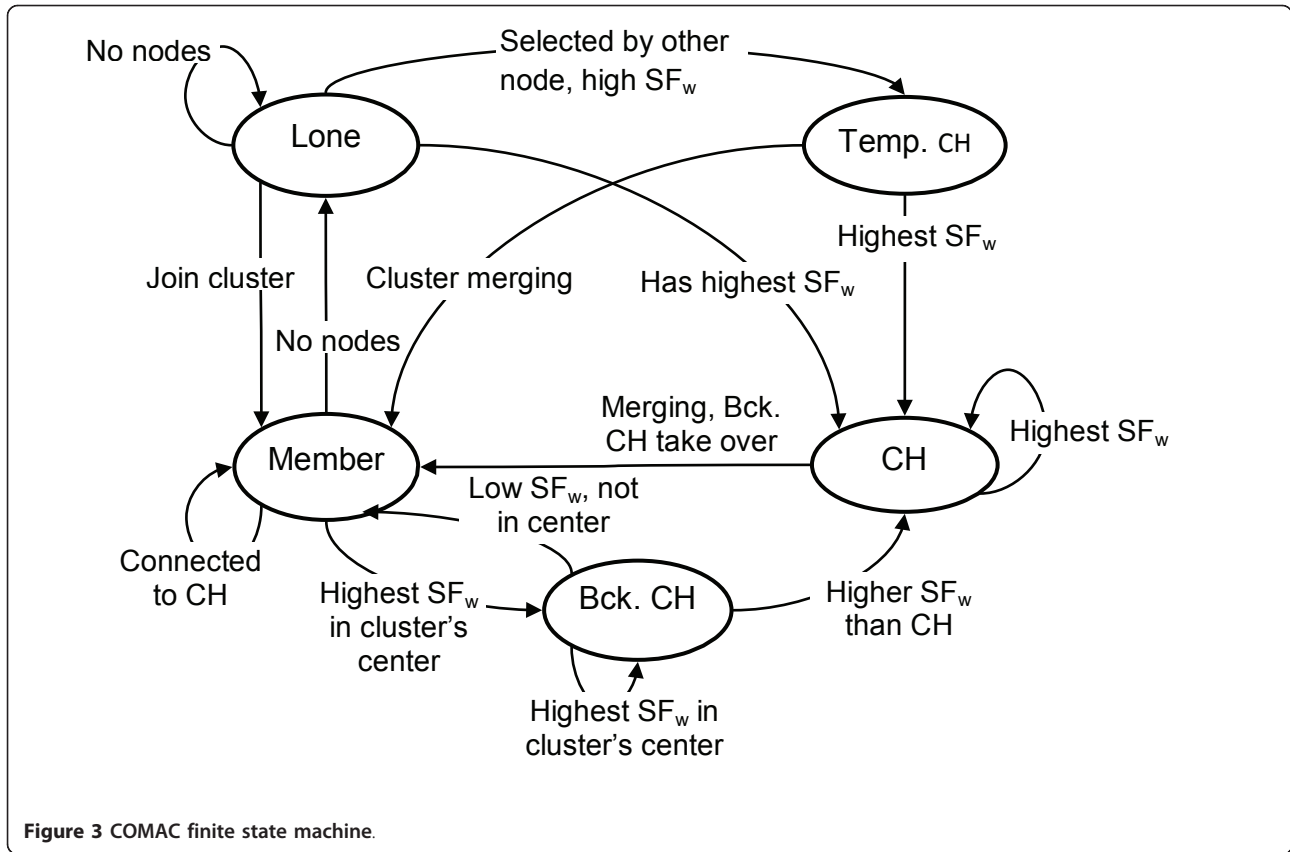**Figure 2 Status message format**.

Figure 3 COMAC finite state machine.

B. COMAC parameters

The stability and reliability of COMAC is affected by the following parameters:

(1) The stabilization factor (SF), which reflects the relative movement between adjacent vehicles. In every CCI interval, each vehicle will have information about all vehicles within its communication range and hence will calculate its average speed difference $\bar{v}_{d_j}$ from all other vehicles as:

$$\bar{v}_{d_j} = \frac{1}{n-1} \sum_{i=1}^{n-1} |v_j - v_i|, \quad j = 1, 2, \ldots, n \qquad (1)$$

where n is the total number of vehicles within jth vehicle's communication range including itself, v_j is the jth vehicle's speed in m/s. The jth Vehicle will calculate its stabilization factor (SF_j) at the end of every SI interval as:

$$SF_j = 1 - \frac{\bar{v}_{d_j}}{V_{\max}}, \quad \in [0, 1], \qquad (2)$$

where $V_{\max}$ is the maximum allowed speed on this road. If there are no other vehicles on the road, the vehicle compares its speed with $V_{\max}$ to calculate its SF factor.

(2) The weighted stabilization factor (SF_w), which is the exponential-weighted moving average of the previous values of SF factors. Each vehicle calculates its new SF_{w_i} from the new value of SF_i and the previous value of $SF_{w_{i-1}}$ as:

$$SF_{w_i} = \zeta \times SF_i + (1 - \zeta) \times SF_{w_{i-1}}, \qquad (3)$$

where $0 \le \zeta \le 1$ is the smoothing factor and chosen here to be 0.5.

(3) The vehicle's acceleration (a), which will help to predict the vehicle's speed and position in the near future (after time T_f). The decision to accelerate, to decelerate or to stay on the same speed depends on many factors such as the distance between the vehicle and its front neighbor, the relative speed between them, the road conditions and the driver's behavior.

Most of the time, the drivers' behavior and how they estimate the interdistance and other factors are subjective and not predictable. Fuzzy logic is used to deal with this uncertainty in our study. Fuzzy logic is a rule-based system that consists of IF-THEN rules that forms the key component of any fuzzy inference system (FIS) [15]. Since FIS lacks the adaptability to deal with changing external environments, we incorporate a learning technique to predict the vehicles acceleration based on the previous behavior of the driver.

The FIS system consists of a fuzzifier, rule base, reasoning mechanism and defuzzifier. The fuzzifier defines the membership functions used in the fuzzy rules. In this paper, the triangular fuzzifier is chosen to implement our FIS system. While the rule base contains a selection of the fuzzy rules, the reasoning mechanism performs the inference procedure upon those rules to derive a reasonable output. The defuzzifier is a method used to map the output fuzzy sets to a crisp output values. In this paper, we used the interdistance and the relative speed between two vehicles as the input parameters to our FIS system and the vehicle's acceleration as its output.

The membership function of the distance between a vehicle and its immediate front neighbor is μ_d and can take any of the three values: small, medium and large as shown in Figure 4. The parameter t_s is a design parameter that represents the safety following distance between two vehicles on the road, i.e., the time needed by the following vehicle with a speed of v_j to cross this interdistance.

The membership function of the relative speed between two vehicles is μ_v and can take the three values:

slow, same and fast as shown in Figure 5. The parameters α and γ are used to make the system more adaptable to the driver's behavior on the road. Initially, their values are set to $\alpha = \gamma = 1$ and will be increased or decreased by a step of ε if the driver's decision to accelerate or decelerate did not match with the predicted output values as follows: if the system predicts that the vehicle will accelerate but it did not, then $\alpha \Leftarrow (1 + \varepsilon)\alpha$; if the system predicts that the vehicle's speed will stay the same but it accelerates, then $\alpha \Leftarrow \max\{(1 - \varepsilon)\alpha, 0\}$, and if it decelerates, then $\gamma \Leftarrow \max\{(1-\varepsilon) \gamma, 0\}$ and finally if the system predicts that the vehicle will decelerate but it did not, then $\gamma \Leftarrow (1 + \varepsilon) \gamma$. By this, the values of α and γ will converge to ceratin values after a short period of time to capture the driver's behavior on the road. If the vehicle's acceleration matches with the predicted value, then keep the same values of α and γ.

The output variable, namely the predicted acceleration, is μ_{acc} and has the following fuzzy names: accelerate, stay at the same speed and decelerate. We choose the crisp outputs 2, 0 and -2 m/s^2 for the values of μ_{acc}, respectively. This is called a center-average defuzzifier, which produces a crisp output based on the weighted average of the output fuzzy sets. The output variable μ_{acc} is shown in Figure 6. Table 1 shows the fuzzy rule for the acceleration output.

C. Cluster head election
Since VANETs are highly dynamic and their network topologies change very frequently, the clustering algorithm should be distributed and operate asynchronously. Therefore, the algorithm of electing and reelecting the
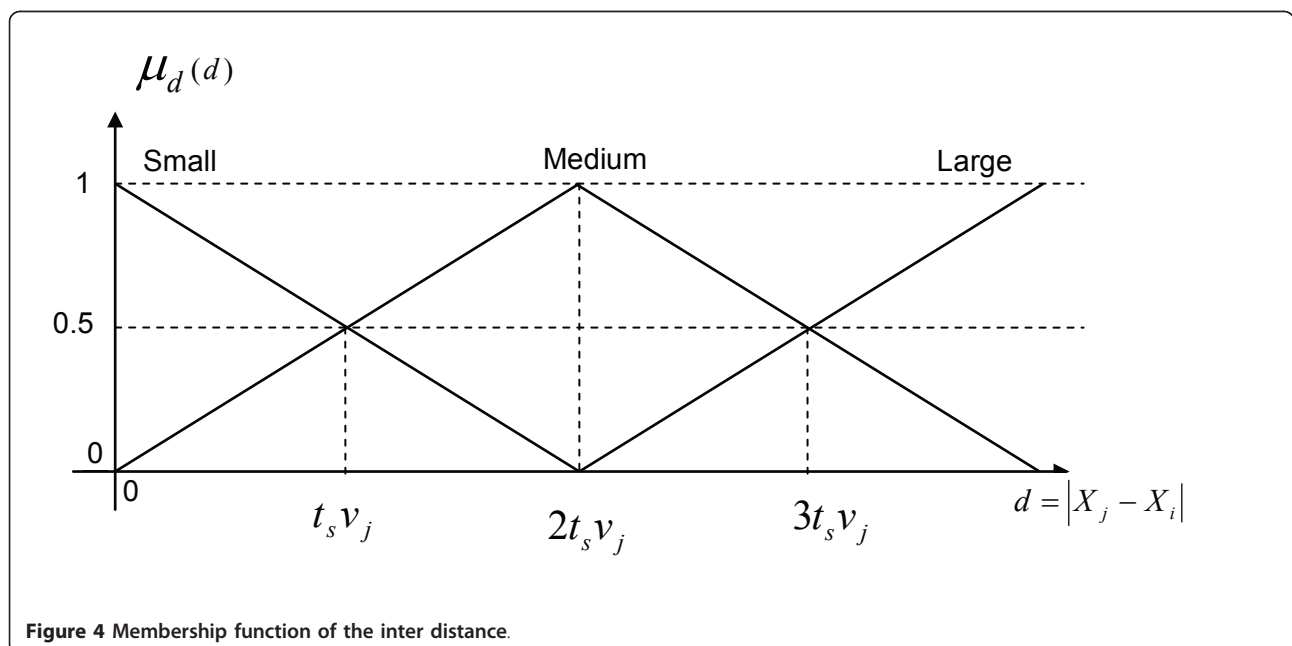

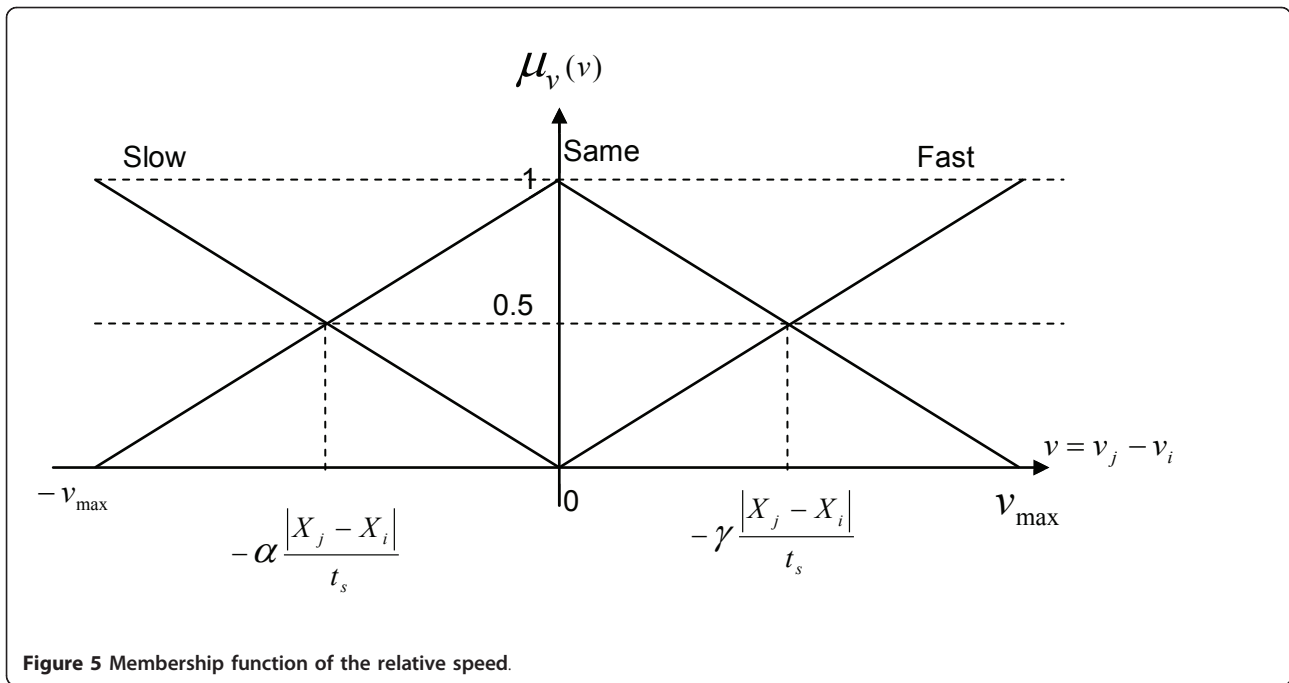**Figure 4 Membership function of the inter distance**.

Figure 5 Membership function of the relative speed.

cluster head should be fair, simple and with minimal communication and coordination among vehicles within the communication range. For clusters to look more stable compared to the highly dynamic VANET, the algorithm should not initiate cluster head reelection very frequently and nodes should join, leave and form a new cluster smoothly. Moreover, if the network initiates an election or reelection of a cluster head, the algorithm should converge to a stable clustered topology in a very short time.

In COMAC, the clustering algorithm does not require any additional messages other than the dissemination of vehicles' status messages. Therefore, when vehicles are on the road for the first time, they start sending their status messages without an elected cluster head. Once these messages are received by all nodes in the network, vehicles start calculating their SF_w factors.

If a vehicle has the highest weighted stabilization factor SF_w among all vehicles within its communication range, it will elect itself as a cluster head by setting its
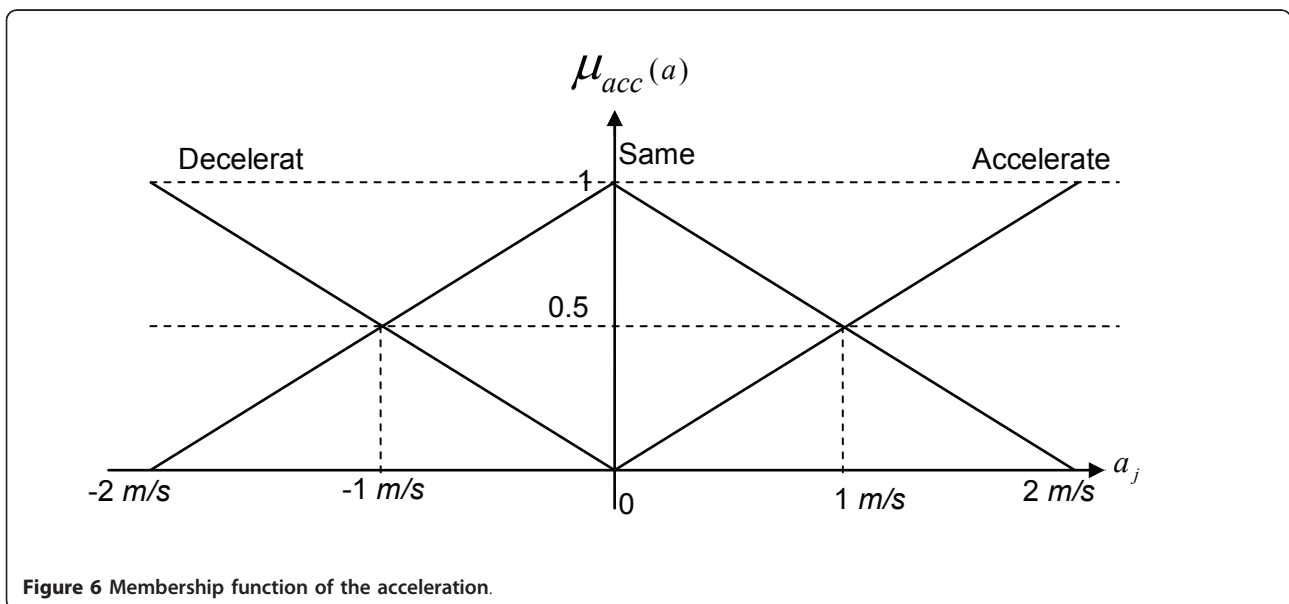

Figure 6 Membership function of the acceleration.

Table 1 The fuzzy rule of the acceleration

Rule	$\mu_d(d)$	$\mu_v(v)$	$\mu_{acc}(a)$
1	Small	Slow	Accelerate
2	Small	Same	Same speed
3	Small	Fast	Decelerate
4	Medium	Slow	Accelerate
5	Medium	Same	Same speed
6	Medium	Fast	Decelerate
7	Large	Slow	Accelerate
8	Large	Same	Same speed
9	Large	Fast	Decelerate

field CHID to its own ID. The new cluster head will start sending its status messages using one of the main subcarriers sets (c_1, c_2, c_3). All other vehicles within its range have the chance to cluster with this cluster head and use the same subcarriers set.

If there is another vehicle, within this vehicle's range, which has the highest SF_w factor, it will elect it as a temporary cluster head by setting its field BKID to the elected cluster head's ID. This temporary cluster head will check first whether it has the highest SF_w factor, if yes it will elect itself as a cluster head by setting its field CHID to its own ID, and if no, it will accept to act as a temporary cluster head and will not participate in electing a new cluster head within its range waiting either to merge with another cluster or to change its state to a main cluster.

To fasten the network convergence to a stable cluster topology, a vehicle that is not a cluster head within its own range and lies within the range of a temporary cluster head will join this cluster and will not participate in electing another temporary cluster head. A vehicle that lies within the range of two cluster heads will cluster with the closest cluster head to itself given the priority to the main cluster over the temporary cluster.

D. Cluster head's role

Once elected, the cluster head will send three extra messages: First, a consolidated message (with $Type$ = 1) will be sent at the beginning of every CCI interval. This message has information about the neighboring clusters and all current cluster members where their IDs are ordered from behind to front. Cluster members will follow this order to send their status messages within the CCI interval. At the same time, each vehicle calculates its maximum waiting time $T_w(d)$ that it should wait for its turn to access the channel based on their distance d from the elected cluster head as:

$$T_w(d) = T_A + \frac{T_A}{2}\left(1 + \frac{d}{R}\right), \qquad (4)$$

where R is the communication range used by all cluster members, $d \in [-R, R]$ is the distance from the cluster head where vehicles in front of the cluster head have positive distance and vehicles behind the cluster head have negative distance and $T_A = 6 \times 13\ \mu s$ is the arbitration interframe space (AIFS) for this type of messages as in IEEE 802.11p standard. A vehicle can send its status message when the vehicle ahead of it in the sequence finishes transmitting its status message. Otherwise, if the vehicle did not hear the message of its head neighbor, it will send its message when its $T_w(d)$ expires. After every successful transmission, each node updates its $T_w(d)$ based on the distance from the last vehicle that successfully transmits its status message. Vehicles that are in front of the cluster head will wait until their cluster head takes its turn to send its status message ($Type$ = 0) successfully. This is to eliminate the hidden terminal problem that could arise from the other side of the cluster. Second, after receiving all status messages from its cluster members, the cluster head will send a status message with $Type$ = 2, which is an invitation for new members to join the cluster and send their status messages. Third, a consolidated message with $Type$ = 3, which contains information about all of its members with enough power to reach double the communication range (R) when the channel is idle for time $(2+\psi) \times T_A$, where ψ is a random number from [0,1]. This message is intended to reach the two neighboring cluster heads.

The cluster head will also decide which subcarriers set and what communication range R that all of its members should use and synchronize it with its neighboring clusters. In the remaining time of the CCI and after sending its final message, the cluster head will accept route requests from its members if they want to communicate with other vehicles on a different channel and outside the CCI interval.

If a vehicle has an emergency message, it will contend for the channel access using the minimum contention window specified for high priority class in IEEE 802.11p, i.e., CW_{min} = 3 and waiting time $T_w(d) = 2 \times 13\ \mu s$, to send this message for several times depending on the application. Once this message is received by the cluster head, the cluster head will start transmitting this message periodically with enough power to reach double the communication range (R) and in the direction of interest, all other cluster members will defer from using the channel during this time. When the next cluster head receives this emergency message, it will broadcast it omnidirectionally with a communication range ($2R$) to reach both the next cluster and the originating cluster heads. Once the originating cluster head hears its message back from the neighboring cluster head, it will stop broadcasting it with high power while continue to

broadcast it to all of its members for several times depending on the application or until the emergency situation is cleared. The emergency message will continue to propagate in the direction of interest for a maximum number of hops depending on the application and the emergency situation.

E. Temporary cluster

Once a temporary cluster has been formed, the temporary cluster head will wait for the first chance to either merge with adjacent cluster or become a main cluster itself. If this temporary cluster head falls within half of the communication range of its adjacent cluster head, it will merge with this cluster by sending a status message that includes the new cluster head's ID using the same temporary subcarriers set. When the temporary cluster members receive this message, they will join the new cluster if they fall within the range of the new cluster head; otherwise, they will form a new temporary cluster with a new cluster head that has the highest SF_w factor among the remaining nodes that could not join the new cluster.

Since a cluster head communicates with adjacent cluster heads with double the communication range ($2R$), it knows about the subcarriers sets they use. Therefore, the temporary cluster head can change its state to a main cluster by selecting a subcarriers set that is not used by its adjacent clusters and trying its best to maintain the sequence of the subcarriers sets as c_1, c_2, c_3. The cluster head knows the subcarriers set that is used by the cluster head in front of it; therefore, it will select to use the subcarriers set that comes after it in sequence. If the front cluster uses the temporary set c_4, it will select c_1 set to start the sequence. If there is no front cluster, it will select a subcarrier set that is lower in sequence of the behind cluster. The core idea in COMAC is to let each cluster to iteratively move its subcarriers set following its immediate front cluster's set until a network convergence occur.

F. Cluster maintenance

Once the cluster head has been elected, our goal is to maintain the cluster topology as much stable as possible by not initiating the election process very frequently. Therefore, the cluster head will calculate the expected positions and speeds of all of its members after time T_f based on their advertised speeds and accelerations as follows:

$$x(T_f) = x + vT_f + \frac{1}{2}aT_f^2, \tag{5}$$

$$v(T_f) = v + aT_f. \tag{6}$$

The cluster head will remain as a cluster head if all its members are still within its range after time T_f. The cluster head will select a backup cluster head based on two criteria: first, it is the closest to the center of the cluster, and second, it has the highest SF_w factor among all vehicles around the cluster's center. If some of the cluster members will become out of the cluster head's range but still within the range of the backup cluster head, the current cluster head will hand the responsibility to the backup cluster head by setting its field CHID = CHBK. Otherwise, if some members become out of range of both the cluster head and its backup, the current cluster head will remain the cluster head in the next interval. Vehicles that became out of range will form a temporary cluster or join an adjacent cluster if they fall within its cluster head's range.

4. Analysis

The COMAC protocol is based on the weighted stability factor of vehicles on the road which measures how vehicles behave compared to the overall traffic flow. Vehicles that are well behaved are more likely to cluster with themselves around a cluster head that is moving on average with the same speed as other vehicles around it. Therefore, the network topology will look more stable where clusters are seen moving in sequence on the road instead of vehicles passing each other. This will allow achieve an acceptable levels of performance once the network converges. Vehicles will have the chance to send their status messages with less competition for accessing the channel and less vulnerable to the hidden terminal problem. In the following, we will present the performance measures of COMAC with respect to network convergence, stability, reliability, overhead and time delay.

As in [16], we built our model based on a multilane highway scenario. Since the communication range is much larger than the road's width, we simplify the network in each direction of the road as one-dimensional VANET. We assume that all status messages have the same length L bits, all vehicles have the same transmission range R meters and use the same transmission rate r_d Mbps. Vehicles arrive at the beginning of each direction of the highway segment as a Poisson process with average rate β vehicles/s. After that they follow the direction of the road with a speed uniformly distributed between V_{min} and V_{max} with means $\mu = \frac{V_{min}+V_{max}}{2}$.

From this model, we derived the distribution of vehicles that are traveling in one direction on a highway segment at the steady state as a Poisson distribution with rate $\frac{2\beta R}{\mu}$[16]. As a result, the probability of having k vehicles within a distance of $2R$ is:

$$P_{2R}(k) = \frac{\left(\frac{2\beta R}{\mu}\right)^k}{k!} e^{-\frac{2\beta R}{\mu}}. \tag{7}$$

Therefore, the interdistance x between vehicles on the road has an exponential distribution with mean $\frac{\mu}{\beta}$ as:

$$f_X(x) = \frac{\beta}{\mu} e^{-\frac{\beta}{\mu}x}. \tag{8}$$

A. Network convergence and stability

In COMAC, the cluster size is governed by the cluster head's communication range, which is a critical parameter in networks stability. Increasing the communication range results in increasing the cluster size, and hence, more vehicles will contend for using the shared channel to send their status messages. At the same time, the increase in the communication range results in more space for vehicles to move within the cluster space with less probability to cross the cluster boundary. On the other hand, decreasing the communication range results in low network stability, where vehicles are very often cross the cluster's boundary but at the same time, the number of vehicles that are competing for the channel will decrease.

To optimize the communication range and hence the cluster size is very difficult especially in a highly dynamic environment such as VANETs. In [16], the authors showed how vehicles' dynamics affect the network density and hence the reliability and throughput of VANETs' safety applications. However, in [2] and [17], the authors derived the relationship between the communication range and the network density, message sending rate, message size, data rate and channel conditions. Since each vehicle in the network has its own view of the network density and channel conditions, finding the optimal network parameters is difficult. Therefore, our main goal in COMAC is not to find the optimal cluster size but to make the network more stable.

In COMAC, we define two threshold cluster sizes (i.e., number of vehicles within the cluster head's range) $K_h = 2\lambda_h R_h$ and $K_l = 2\lambda_l R_l$, where R_h is the communication range that all vehicles will use when they enter the road, λ_h is the maximum vehicle density that corresponds to R_h and measured by vehicles per meter, R_l is the lower communication range that can be used by all vehicles which is related to a jam scenario and λ_l is the vehicles' density that triggers the change from R_l to R_h. The cluster head can sense the network density by the number of status messages that are received within the control channel interval CCI. K_h represents the maximum number of vehicles that can be accommodated within the

cluster and have the chance to send their status messages. Therefore, to prevent the frequent change in cluster size as vehicles move in and out of the cluster boundary, the cluster head will use the hysteresis mechanism as shown in Figure 7.

In low-density networks, the cluster head uses the communication range R_h because the vehicle density is below the threshold λ_h. When vehicle density reaches λ_h, the cluster head will change its communication range to R_l triggering a change in the cluster size. Vehicles that found themselves out of the cluster attempt to join another cluster or to form a new cluster, while vehicles that are still within the cluster will change their communication range accordingly. The cluster head will keep using R_l although the network density is decreasing till it reaches the threshold λ_l where it will change the communication range back to R_h triggering a new change in the cluster size. By using the hysteresis mechanism, we reduced the frequent change in cluster topology due to vehicles' high dynamics.

For vehicles that found themselves inside a new cluster, they decide to either join the new cluster or stay with their current cluster based on the distance between them and the two neighboring cluster heads. The network convergence in this case is instant unless one of the cluster heads decides to merge with a neighboring cluster leaving some members behind it. In this case, either the backup cluster head will take over or a new cluster head will be elected.

B. Time delay

In COMAC, the cluster head will broadcast first its consolidated message to all of its members indicating the start of the CCI interval. After that all cluster members including the cluster head schedule themselves for the channel access to send their status messages by first following the sequence advertised by the cluster head. If

Figure 7 The hysteresis mechanism in COMAC.
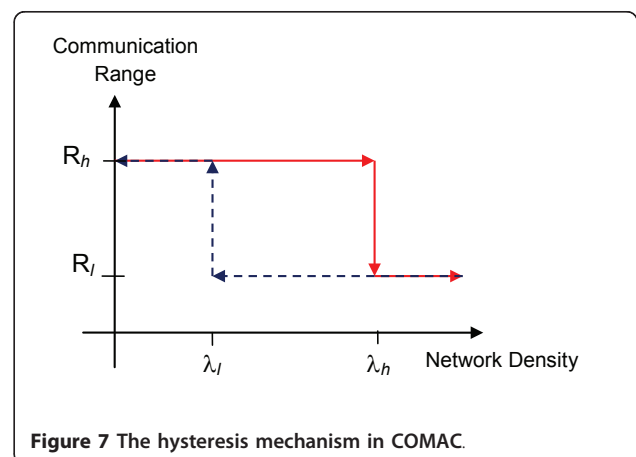
the sequence is interrupted, then vehicles can use the channel based on their calculated $T_w(d)$ as in (4). All vehicles that are in front of the cluster head will defer from using the channel until they receive the cluster head's regular status message. This is to eliminate the hidden terminal problem that could arise from the other side of the cluster. After sending its regular status message, the cluster head will wait for $(2 + \psi) \times T_A$ period after every successful transmission of a cluster member's status message before it sends its *Type* = 2 or *Type* = 3 message. By this, the cluster head gives enough time for all of its members and the recently joined members to send their status messages.

To find the upper bond for the average time till all cluster members manage to send their status messages, we assume that all members wait for time $T_w(d)$ before they send their messages. The first message that will be sent by the cluster head will take time T_{cf} as:

$$T_{cf} = T_A + \frac{L_{cf}}{r_d} + \delta, \tag{9}$$

where L_{cf} is the cluster head's first message size in bits, r_d is the data rate and δ is the propagation delay.

The first cluster member to win the channel access, assuming that it is located at distance d from the cluster head, will take T_{mf} time as:

$$T_{mf}(d) = T_w(d) + \frac{L}{r_d} + \delta, \tag{10}$$

where d in this case is a uniformly distributed random variable over the interval [-R, 0]. Therefore, the average time of T_{mf} is

$$[T_{mf}] = \int_{-R}^{0} T_{mf}(d)\frac{1}{R}d_d = \frac{5}{4}T_A + \frac{L}{r_d} + \delta. \tag{11}$$

The second vehicle to transmit its status message is the one which is the closest to the first one. Assuming that the interdistance between them is d which in this case has an exponential distribution as in (8), then its transmission time is $T_m(d) = T_w(d) + \frac{L}{r_d} + \delta$. Therefore, the average time of T_m is

$$[T_m] = \int_{0}^{R} T_m(d)\frac{\beta}{\mu}e^{-\frac{\beta}{\mu}d}d_d$$

$$= \frac{3}{2}T_A + \frac{L}{r_d} + \delta + \frac{T_A}{2R}\left[\frac{\mu}{\beta} - (R + \frac{\mu}{\beta})e^{-\frac{\beta}{\mu}R}\right]. \tag{12}$$

The cluster head will wait for time $(2 + \psi) \times T_A$ before it can send its invitation message (*Type* = 2);

therefore, the average transmitting time for this message is

$$[T_{in}] = \frac{5}{2}T_A + \frac{L}{r_d} + \delta. \tag{13}$$

The last message that the cluster head will send is a consolidated message that has information about all of its members to reach a range of 2R. Assuming the size of this message is L_{cl} bits and since the cluster head will wait for $(2 + \psi) \times T_A$ before sending this message, therefore, the average transmitting time for this last message is

$$[T_{c1}] = \frac{5}{2}T_A + \frac{L_{c1}}{r_d} + \delta. \tag{14}$$

From Equations (9), (11), (12), (13) and (14), we can find the upper bond of the total average time for all cluster members to send their status messages as:

$$T_{avg} = T_{cf} + [T_{mf}] + \left(\frac{2\beta R}{\mu} - 1\right)[T_m] + [T_{in}] + [T_{c1}] \tag{15}$$

since the average number of vehicles within a range of 2R is $\frac{2\beta R}{\mu}$ as per our model in (7).

The lower bond of the total average time for all cluster members to send their status messages is when all cluster members send their status messages following the sequence specified by the cluster head without waiting for time $T_w(d)$. Therefore, the lower bond for T_{avg} is

$$(T_{avg})_{1b} = T_{cf} + \left(\frac{2\beta R}{\mu}\right)\left[T_A + \frac{L}{r_d} + \delta\right] + [T_{in}] + [T_{c1}] \tag{16}$$

C. Reliability

We define the reliability by the probability of successfully delivering safety messages from all cluster members during the control channel interval CCI. To assure that all vehicles have the chance to send their status messages within the control channel interval CCI, we should have

$$T_{avg} \leq \rho \times \text{CCI}, \tag{17}$$

where $\rho \leq 1$ is a design parameter to spare some time from the CCI for other control messages.

From (17), we can find λ_h and R_h that we should use to trigger the change in the cluster size as discussed in Figure 7. If we assume that each cluster has the same number of vehicles K, then the size of the cluster's first message is $L_{cf} = 2K \times L$ and its last message is $L_{cl} = K \times L$. Moreover, as the communication range R increases, the term $e^{-\frac{\beta}{\mu}R}$ in (12) approaches zero. Therefore, if we

use ceratin maximum range as R_h, then we can find $\lambda_h = \frac{K}{2R_h}$ by applying (15) into (17) as:

$$\lambda_h \leq \frac{1}{2R_h} \frac{\rho \times \text{CCI} - 5.75T_A - \frac{2L}{r_d} - 3\delta}{\frac{3}{2}T_A + \frac{4L}{r_d} + \delta}. \tag{18}$$

While R_l is governed by the jam scenario where we assume that each vehicle occupy 10 m on average and the road has W number of lanes, then

$$R_l \leq \frac{10}{2W} \frac{\rho \times \text{CCI} - 5.75T_A - \frac{2L}{r_d} - 3\delta}{\frac{3}{2}T_A + \frac{4L}{r_d} + \delta}. \tag{19}$$

On the other hand, λ_l is a design parameter that can be chosen as a fraction of λ_h, such that $\lambda_l < \frac{R_l\lambda_h}{R_h}$. The smaller λ_l is selected, the less frequent the cluster size is changed. In our simulations, we set $\lambda_l = \frac{R_l\lambda_h}{R_h}$.

5. Model validation and simulation

The performance of COMAC can be assessed based on different measures, such as the stability and efficiency of the clustering algorithm, system reliability and time delay.

To test for the system stability and efficiency, we define four metrics: First, the average cluster head time ($\overline{\text{CHT}}$), which is the sum of all cluster head times divided by total number of cluster heads during the simulation period. Second, the average cluster stability factor ($\overline{\text{CSF}}$), which is the average relative speed of the cluster head compared to its members and can be calculated as

$$\overline{\text{CSF}} = \frac{1}{T_s} \sum_{t=1}^{T_s} \frac{\sum_{j=1}^{C_n(t)} \text{SF}_{w_j}}{C_n(t)}, \tag{20}$$

where $C_n(t)$ is the number of formed clusters at time t and T_s is the total number of CCI intervals during the simulation time. Third, the average cluster size ($\overline{\text{CS}}$), which is the total number of vehicles that became cluster members divided by the total number of formed clusters during the simulation time and can be calculated as

$$\overline{\text{CS}} = \frac{1}{T_s} \sum_{t=1}^{T_s} \frac{\sum_{j=1}^{C_n(t)} \text{CM}_j(t)}{C_n(t)}, \tag{21}$$

where $\text{CM}_j(t)$ is the number of cluster j's members.

Finally, the clustering management overhead, which is the total number of invitation messages sent by the cluster head compared to the total status messages sent by all cluster members within any CCI interval.

The system's reliability ($\Re$) is tested by the probability for a cluster member to send its status message during the CCI interval and can be calculated as:

$$\Re = \frac{1}{T_s} \sum_{t=1}^{T_s} \frac{\sum_{j=1}^{C_n(t)} \frac{\text{CMs}_j(t)}{\text{CM}_j(t)}}{C_n(t)}, \tag{22}$$

where $\text{CMs}_j(t)$ is the number of cluster j's members managed to send their status messages at time instance t.

The system's delay is measured by allowing one vehicle to send an emergency message and measure the time it reaches the intended distance, which is 2,000 m in our simulation, and the probability of successfully receiving this message by all vehicles within the intended distance and in the direction of interest.

The simulation is developed based on the network simulator ns-2 [18] to measure the mean cluster size, network convergence, the time delay for an emergency message to reach a certain distance and the probability of a vehicle to be within a cluster and its dwell time. ns-2 is a well-known simulator in analyzing VANETs. We used a realistic mobility model generated by MOVE [19], which is built on top of the micro-traffic simulator SUMO [20] that has the most realistic mobility traces for VANETs simulations [21].
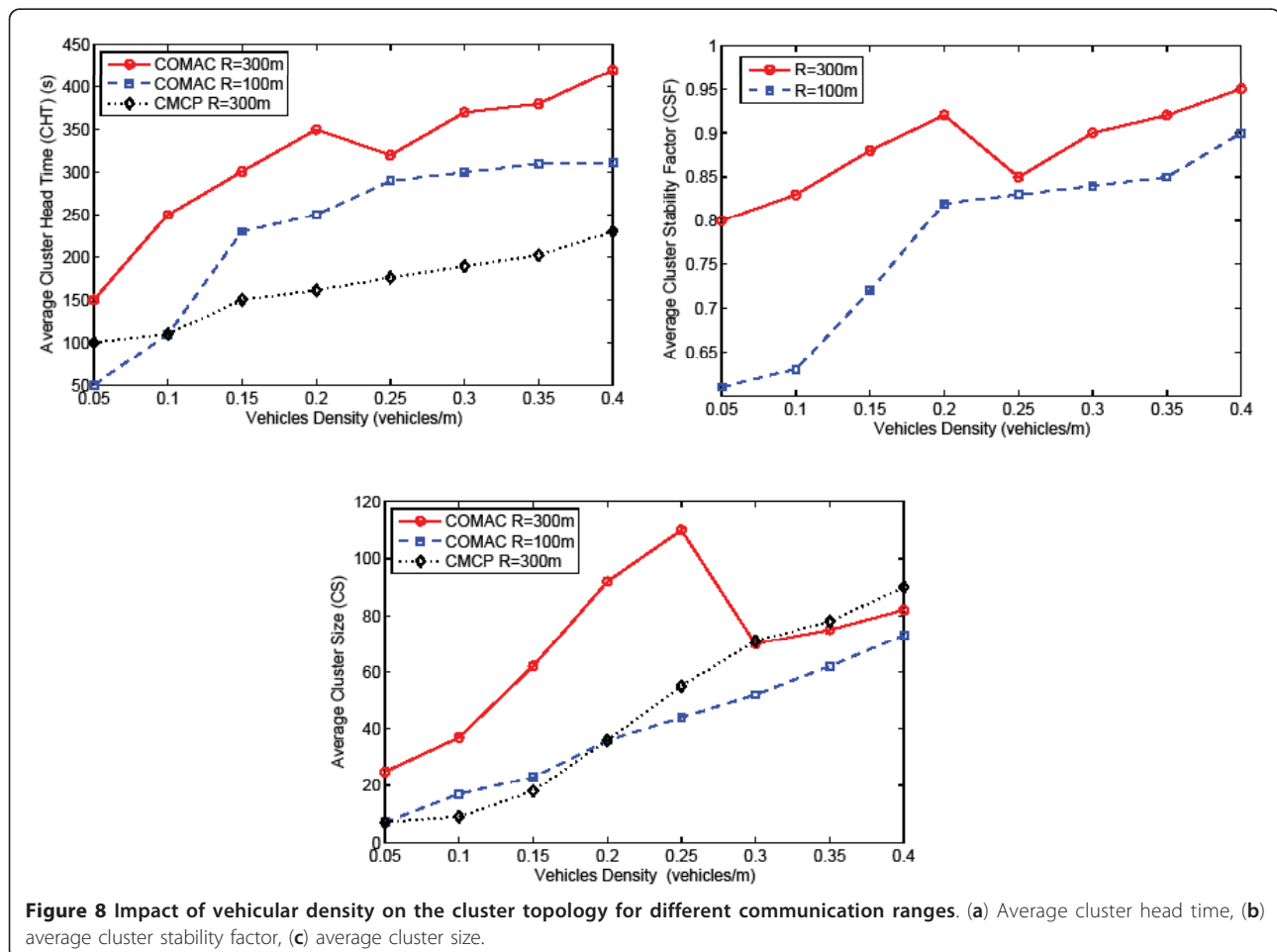
The simulation scenario is built upon one-directional 4 lanes highway segment of 8,000 m in length and looped back to form a closed rectangle. The two opposite directions of the highway are separated by more than 500 m such that the clustering could not occur across them. The vehicles' speed ranges from 80-120 km/h, which is typical for Ontario highways. We used the Nakagami-m propagation model with configuration parameters as in [22]; hence, receivers located within 100 m of the transmitter will receive the signal based on Rician distribution, and others are based on Rayleigh distribution. We assume that all vehicles are synchronized to the control channel interval through a GPS system to send their status messages. We let one vehicle to send an emergency message to be propagated to a distance of 2,000 m behind it. The safety following distance or time is selected to comply with the 3 s rule, i.e., $t_s = 3$ s and the cluster maintenance time is $T_f = 10s$. The communication range R_l is set to 200 m based on Equation (19). For R_h, we used the communication ranges 300, 200 and 100 m, which correspond to the transmission powers of 109, 48 and 12 μ W, respectively. Therefore, when the used communication range $R_h = 300$ m, we calculate $\lambda_h = 0.25$ from (18) and $\lambda_1 = \frac{R_l\lambda_h}{R_h} = 0.17$ vehicles/m. For the communication ranges of 200 and 100 m, there is no need to calculate λ_h and λ_l since these ranges are equal and less than R_l, respectively. We compare our protocol by the CMCP [13] since it is the most relevant to our work. Each simulation typically simulated 1,000 s of real time and averaged over ten times to obtain the mean value as the final performance

Table 2 Value of parameters used in simulation

Parameter	Value
Modulation and data rate	QPSK, $r_d = 6$ Mbps
Message sizes (L)	100×8 Bits
Vehicle's speed	80-120 Km/h
Communication range ($R=R_h$)	300 m ($P_t = 109$ μW)
Communication range ($R=R_l$)	200 m ($P_t = 48$ μW)
Smoothing factor (ζ)	0.5
Control channel interval (CCI)	100 ms
Cluster maintenance time (T_f)	10 s
Safety time (t_s)	3 s
ε	0.1
ρ	1
Number of lanes (W)	4
Simulation time	1,000 s
Received power threshold (R_{xTh})	3.162e-13
Noise-floor	1.26e-14
T_{tx} & T_{rx} antennas heights	1.5 m
T_{tx} & T_{rx} antennas gain $G_t = G_r$	4
Slot time (σ)	13 μs
Propagation delay (δ)	1 μs

metric. Table 2 lists the simulation parameters used unless a change is mentioned explicitly.

Figure 8 shows the impact of traffic density on the cluster topology for various communication ranges. The traffic density λ_v can be increased by either decreasing the vehicles' average speed μ or increasing the vehicles arriving rate β since $\lambda_v = \frac{\beta}{\mu}$ as per our model in (8). Figure 8a shows that the average life time of a cluster head is increasing by the increase in vehicles' density on the road. This is because the inter-distances between vehicles are decreasing, and hence, the relative speed between them is also decreasing, resulting in high stability factor for the cluster head as shown in Figure 8b. This verifies that our clustering algorithm works perfectly by allowing the cluster head to reelect itself when all or most of its members will be within its range after time T_f in the future. Figure 8c shows the increase in the average cluster size as the vehicle density increases. In COMAC, the cluster head has a rule to change the communication range when the vehicles' density reaches a threshold λ_h; hence, all of its members have the chance to send their status

Figure 8 Impact of vehicular density on the cluster topology for different communication ranges. (**a**) Average cluster head time, (**b**) average cluster stability factor, (**c**) average cluster size.

messages. This is clear on the abrupt change in the cluster size when the vehicle density reaches 0.25 vehicles/m for the used communication range R = 300 m. It is obvious that the stability of the network topology will increase by increasing the communication range as shown in Figure 8a, b and 8c. This is because vehicles will have more space to move within their cluster when the communication range is large without crossing the cluster's boundary.
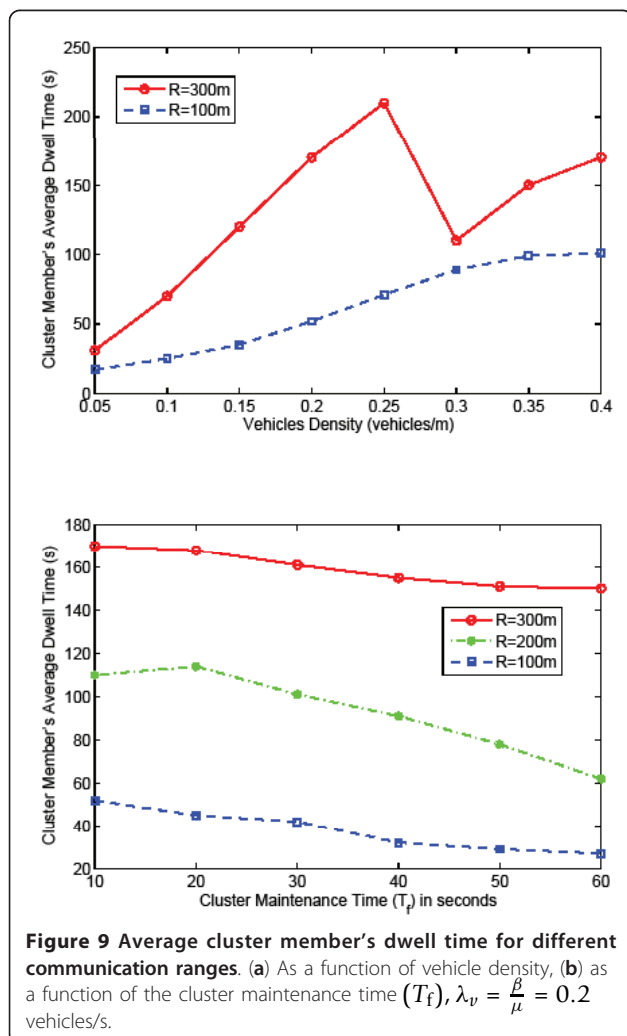
Figure 9 shows the average dwell time for a cluster member within the range of its cluster head. It is clear that the dwell time is increasing by the increase in either the vehicle density or the communication range. Increasing both the range and density will result at certain point that the cluster head will decrease the range, as explained in the hysteresis algorithm, to accommodate members not more than what the CCI interval can tolerate. The sudden decrease in the communication range results in less-stable network topology for a short period of time. This is clear from Figure 9a, which
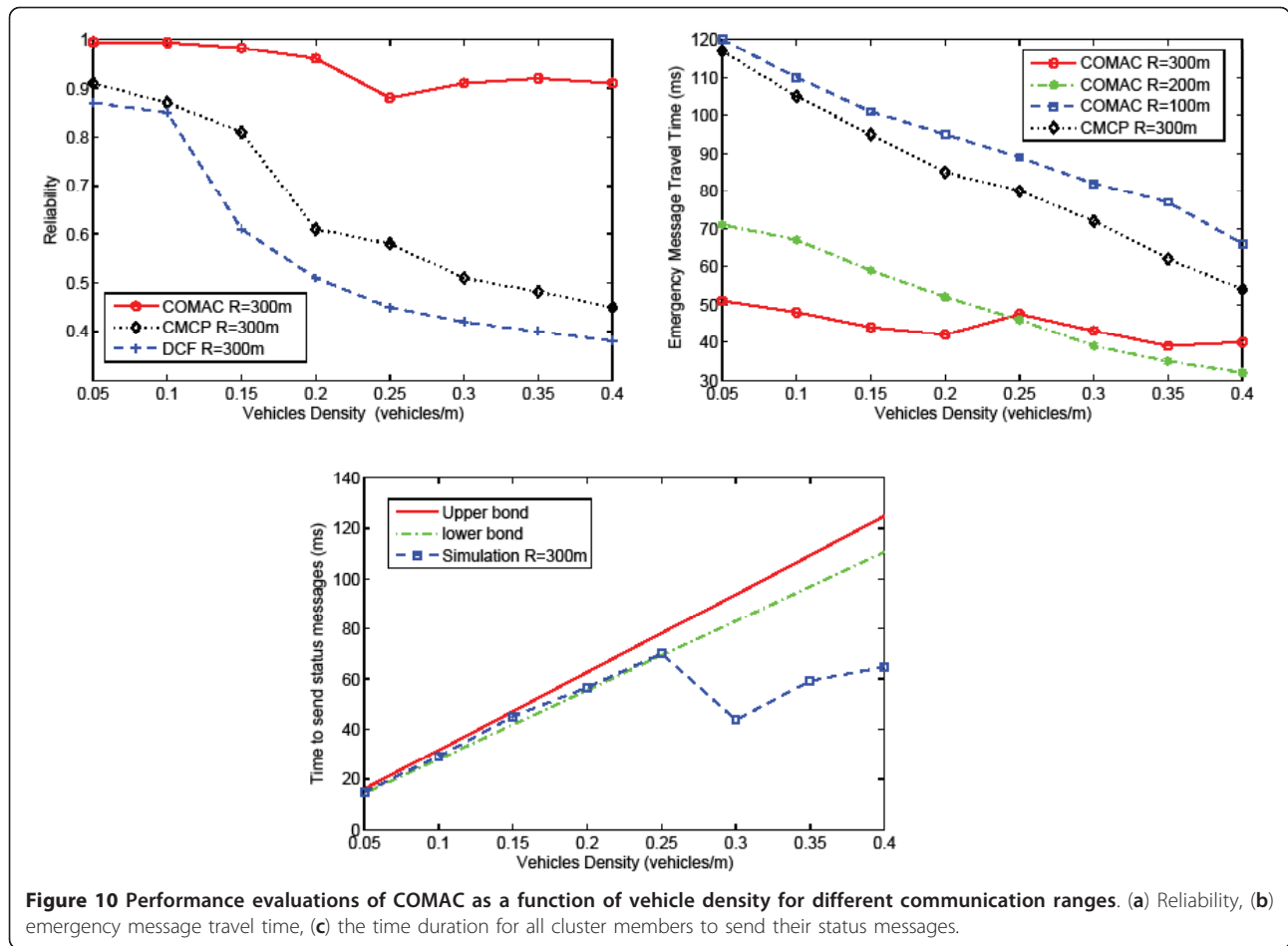
shows that the dwell time starts to increase again after the range has decreased since vehicles that found themselves outside the cluster will soon form a new cluster or join a neighbor cluster.

Figure 9b shows the average cluster member's dwell time as a function of the cluster maintenance time (T_f). As the cluster maintenance time increases, the accuracy of predicting the vehicle's future position and speed decreases. But in COMAC, this decrease is very small since our algorithm is more adaptable to the driver's reactions to the interdistance and relative speed between the two following vehicles. This proves that the learning algorithm in COMAC increases the efficiency of electing and reelecting a cluster head giving more time to cluster members to stay within their cluster's boundary. It is also clear that increasing the communication range will increase the dwell time and at the same time will decrease the effect of long maintenance time T_f. Therefore, T_f should be selected carefully based on the configured range and average vehicles' density to increase the dwell time and decrease the computational overhead in the algorithm.

Figure 10 shows the performance evaluation of our clustering algorithm as a function of vehicle density. Figure 10a shows the probability that all cluster members managed to send their status messages during the CCI interval. Since the cluster head in our clustering algorithm advertises the sequence that all of its members should follow to send their status messages, the system reliability is high especially in low-density networks. As the network density increases, the reliability decreases slightly since there is more possibility that new members will join the cluster. These new members are not included in the advertised sequence and have to compete for the channel use based on their distance from the last transmitter as in (4). This may cause collisions or disturbing the sequence, forcing the remaining members to compete for the channel use based on (4). This explains why the reliability drops to the lowest level when the cluster head changes the used communication range as the network density reaches 0.25 vehicles/m. Since the cluster head in COMAC reduces its communication range when the vehicle density reaches a certain threshold, it maintains high reliability even in a high-dense networks compared to CMCP and the DCF of IEEE 802.11p where they suffer from high collision probability and drop rate.

Figure 10b shows the time taken by an emergency message sent by a vehicle to reach a distance of 2,000 m versus the vehicle density for different communication ranges. It is obvious that as the communication range increases, the travel time decreases since the number of clusters that the message will hop through decreases. We can also see that the decrease in the vehicle density

Figure 9 Average cluster member's dwell time for different communication ranges. (**a**) As a function of vehicle density, (**b**) as a function of the cluster maintenance time (T_f), $\lambda_v = \frac{\beta}{\mu} = 0.2$ vehicles/s.

Figure 10 Performance evaluations of COMAC as a function of vehicle density for different communication ranges. (**a**) Reliability, (**b**) emergency message travel time, (**c**) the time duration for all cluster members to send their status messages.

results in increasing the emergency travel time since in low-density networks, cluster heads struggle to find a neighboring cluster head to carry the message forward.

Figure 10c shows the time needed for all cluster members to send their status messages as a function of vehicle density. It is clear that this time is less than the theoretical upper bound derived in (15). This proves that our clustering algorithm is efficient in managing the cluster size to allow all of its members to send their status messages.

Figure 11 shows the cluster management overhead when the communication range is 100 m as a function of vehicle density for both our proposed COMAC and CMCP protocols. We can see as the vehicle density increases, the overhead percentage decreases since more vehicles will manage to send their status messages. In COMAC, the overhead is much lower than that of CMCP since the cluster head in COMAC has a role of selecting a backup cluster head that will take the responsibility of the cluster if it has higher stability factor than the current cluster head. This increases the dwell time of the cluster members and the stability of the cluster topology.

From all simulations, it is obvious that our COMAC performance far exceeds that of CMCP since the proposed protocol has the ability to elect a cluster head that moves with the same speed as most of its members. It has also a maintenance algorithm to reelect and elect a backup cluster head that covers most of its members

Figure 11 Cluster management overhead versus vehicle density for range R = 100 m.
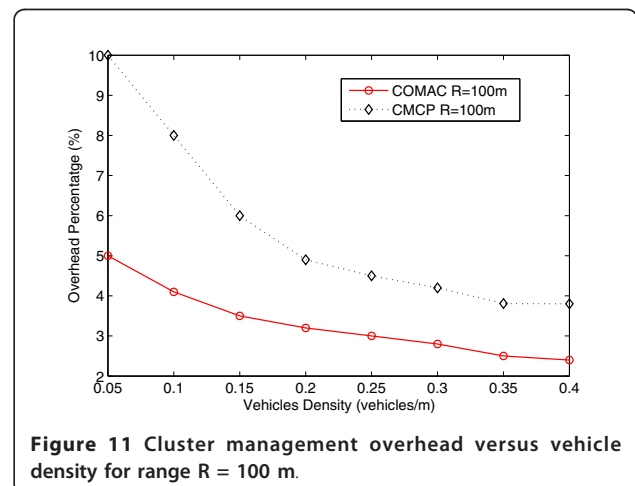
in the future. The algorithm of changing the communication range when the number of cluster members reaches a certain threshold helps to maintain a high reliability compared to CMCP and DCF especially in high-density networks.

6. Competing interests
The authors declare that they have no competing interests.

7. Conclusion
In this paper, we proposed a novel clustering-based MAC protocol for VANETs. Our COMAC protocol is a mobility-based clustering protocol where cluster heads are elected and reelected in a distributed manner according to their relative speed and distance from their cluster members. The high stability of COMAC results from its adaptability to drivers' behavior on the road and its learning process to predict the future speed and position of all cluster members using the fuzzy logic inference system. In COMAC, the cluster head can change the used communication range based on the sensed vehicle density to allow all of its members to send their status messages within the CCI interval. The clusters created by COMAC exhibit long average cluster head's life time and long average dwell time for its members. Under COMAC, safety messages are exchanged within a cluster following a sequence that is advertised by the cluster head. Therefore, the reliability of COMAC is almost the same as in TDMA schemes but without the hassle of reserving time slots and much more than fully contention-based schemes. Moreover, the cluster heads in COMAC have to select one of four subcarrier sets that is different from their cluster head neighbors to eliminate the hidden terminal problem. The simulation results show that our proposed clustering protocol can achieve a timely and reliable delivery of emergency messages to their intended recipients. They also show that COMAC is a highly stable MAC protocol for VANETs.

References
1. ASTM International, Standard specification for telecommunications and information exchange between roadside and vehicle systems–5 GHz band dedicated short range communications (DSRC) medium access control (MAC) and physical layer (PHY) specifications (2009)
2. KA Hafeez, L Zhao, Z Liao, B Ma, Performance analysis of broadcast messages in VANETs safety applications. Proc IEEE Globecommun Conf, 1–5 (2010)
3. X Ma, X Chen, Performance analysis of IEEE 802.11 broadcast scheme in ad hoc wireless LANs. IEEE Trans Veh Tech. **57**(6), 3757–3768 (2008)
4. M Torrent-Moreno, J Mittag, P Santi, H Hartenstein, Vehicle-to-vehicle communication: Fair transmit power control for safety-critical information. IEEE Trans Veh Tech. **58**(7), 3684–3703 (2009)
5. X Ma, XB Chen, Delay and broadcast reception rates of highway safety applications in vehicular ad hoc networks. Mob Netw Veh Environ, 85–90 (2007)
6. M Torrent-Moreno, S Corroy, F Schmidt-Eisenlohr, H Hartenstein, in *MSWiM 2006:Proceedings of the 9th ACM international symposium on Modeling analysis and simulation of wireless and mobile systems*, Ieee 802.11-based one-hop broadcast communications: understanding transmission success and failure under different radio propagation environments, pp. 68–77 (2006)
7. J Yin, T ElBatt, G Yeung, B Ryu, S Habermas, H Krishnan, T Talty, Performance evaluation of safety applications over DSRC vehicular ad hoc networks. ACM-Veh Ad hoc Netw. **1**, 1–9 (2004)
8. F Yu, S Biswas, A self reorganizing MAC protocol for inter-vehicle data transfer applications in vehicular ad hoc networks, in *10th International Conference on Information Technology, (ICIT 2007)*, pp. 110–115 (2007)
9. K Bilstrup, E Uhlemann, E Strom, U Bilstrup, On the ability of the 802.11p MAC method and STDMA to support real-time vehicle-to-vehicle communication. EURASIP J Wirel Commun Netw. **2009**, 1–13 (2009)
10. F Borgonovo, A Capone, M Cesana, L Fratta, ADHOC MAC, New MAC architecture for ad hoc networks providing efficient and reliable point-to-point and broadcast services. Wirel Netw. **10**, 359–366 (2004)
11. J Blum, A Eskandarian, A reliable link-layer protocol for robust and scalable intervehicle communications. IEEE Trans Intell Transp Syst. **8**(1), 4–13 (2007)
12. GM Abdalla, MA Abu-Rgheff, SM Senouci, Space-orthogonal frequency-time medium access control (SOFT MAC) for VANET. Global Information Infrastructure Symposium, 2009. GIIS '09, 1–8 (June 2009)
13. H Su, X Zhang, Clustering-based multichannel MAC protocols for QoS provisionings over vehicular ad hoc networks. IEEE Trans Vehi Tech. **56**(6), 3309–3323 (2007)
14. AB McDonald, TF Znati, A mobility-based framework for adaptive clustering in wireless ad hoc networks. IEEE J Select Areas Commun. **17**(8), 1466–1487 (1999). doi:10.1109/49.780353
15. JSR Jang, CT Sun, E Mizutani, *Neuro-fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence* (Prentice-Hall, 1997)
16. KA Hafeez, L Zhao, Z Liao, B Ma, *Proceedings of IEEE Globecommunication Conference*. Impact of mobility on VANETs safety applications, pp. 1–5 (2010)
17. KA Hafeez, L Zhao, Z Liao, B Ma, in *Proceedings of IEEE Globecommunication Conference*, A new broadcast protocol for vehicular ad hoc networks safety applications pp. 1–5 (2010)
18. Network Simulator 2 (ns2), Discrete Event Network Simulator, http://nsnam.isi.edu/nsnam/index.php/Main_Page (April 2009)
19. FK Karnadi, ZH Mo, KC Lan, in *Proceedings of IEEE Wireless Communications and Networking Conference*. Rapid generation of realistic mobility models for VANET, pp. 2506–2511 (2007)
20. D Krajzewicz, G Hertkorn, C Rssel, P Wagner, SUMO (Simulation of Urban MObility). An open-source traffic simulation, in *4th Middle East Symposium on Simulation and Modelling (MESM2002), Sharjah, United Arab Emirates*, pp. 183–187 (2002)
21. J Harri, F Filali, C Bonnet, Mobility models for vehicular ad hoc networks: a survey and taxonomy. IEEE Commun Surv Tutor. **11**(4), 19–41 (2009)
22. Q Chen, F Schmidt-Eisenlohr, D Jiang, M Torrent-Moreno, L Delgrossi, H Hartenstein, Overhaul of IEEE 802.11 modeling and simulation in ns-2, in *Proceedings of the 10th ACM Symposium on Modeling, analysis, and simulation of wireless and mobile systems*, pp. 159–168 (2007)