Evaluation of VeMAC for V2V and V2R Communications Under Unbalanced Vehicle Traffic

Hassan Aboubakr Omar*, Weihua Zhuang*, and Li Li†
*Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Ontario, Canada, N2L 3G1
†Communications Research Center, Ottawa, Ontario, Canada, K2H 8S2

***Abstract*—A vehicular ad hoc network (VANET) is an emerging technology which has a great potential of new applications in safety, traffic optimization, and entertainment. The VeMAC [1], [2] is a medium access control protocol recently proposed for VANETs, which can support efficient broadcast service necessary for high priority safety applications. The VeMAC protocol reserves disjoint sets of time slots to vehicles moving in opposite directions and to road side units (RSUs). The protocol has been evaluated for vehicle-to-vehicle (V2V) communications and under balanced vehicle traffic conditions, in which the densities of vehicles moving in opposite directions on a two-way road are approximately equal [2]. In this paper we investigate the effects of the existence of RSUs and the unbalanced vehicle traffic conditions on the VeMAC performance, via simulations in highway and city scenarios in terms of network throughput and transmission collision rate.**

I. Introduction

A Vehicular Ad Hoc Network (VANET) consists of a set of stationary units along the road, known as road side units (RSUs), and a set of vehicles equipped with on-board units for wireless communications. The main objective of VANETs is to achieve efficient vehicle-to-vehicle (V2V) and vehicle-to-RSU (V2R) communications. Based on these two kinds of communications, VANETs can support a wide variety of applications specifically targeted for vehicles [3], [4]. The main category of VANETs applications is for safety applications which aim at improving the general public safety standards and increasing the level of safety of drivers and pedestrians. Since most (if not all) of the high priority safety applications proposed for VANETs are based on a periodic broadcast of safety messages by vehicles and/or RSUs [4], and given that any failure in the delivery of safety messages can affect the safety of people on road, it is crucial that the medium access control (MAC) protocol provides an efficient one-hop broadcast service to support such safety applications. The IEEE 802.11p [5] is a current standard proposed for MAC in VANETs. The protocol does not support an efficient broadcast service, mainly because of the hidden terminal problem which results from the absence of RTS/CTS exchange for broadcast frames [6]. On the other hand, the time division multiple access (TDMA) protocols, such as VeMAC [1], [2] and ADHOC MAC [7], can support a reliable (i.e. acknowledged) broadcast service without the hidden terminal problem. Taking account of this advantage of the TDMA protocols, and given that time synchronization can be accurately achieved by using the GPS 1PPS signal, TDMA protocols are good candidates for a VANET scenario.

This work was supported by a Strategic Project grant from the Natural Science and Engineering Research Council (NSERC) of Canada.

The performance of the VeMAC protocol was previously investigated and compared with ADHOC MAC via simulations in highway and city scenarios [2]. The simulations in [2] do not include V2R communications and are limited to a balanced vehicle traffic condition, where the densities of vehicles moving in opposite directions on a two-way road are approximately equal. However, since the VeMAC protocol reserves disjoint sets of time slots to vehicles moving in opposite directions and to RSUs, the existence of RSUs and the unbalanced vehicle traffic conditions significantly affects the VeMAC performance. This paper evaluates the VeMAC protocol in comparison with ADHOC MAC for V2V and V2R communications via simulations in highway and city scenarios. As well, it investigates the effect of an unbalanced vehicle traffic condition on the performance of the VeMAC protocol in terms of network throughput and transmission collision rate.

II. System Model and VeMAC Protocol

This section summarizes the system model and the VeMAC protocol presented in [1], [2] required for the paper to be self-contained. The VANET under consideration consists of a set of RSUs, and a set of vehicles moving in opposite directions on two-way vehicle traffic roads. A vehicle is said to be moving in a left (right) direction if it is currently heading to any direction from north/south to west (east). Based on this definition, if two vehicles are moving in opposite directions on a two-way road, regardless of the orientation of the road, it is guaranteed that one vehicle is moving in a left direction while the other vehicle is moving in a right one. The VANET has one control channel, denoted by CCH. On the CCH, the time is partitioned to frames consisting of a constant number S of equal-duration time slots. Each second contains an integer (fixed) number of frames, and each frame is partitioned into three sets of time slots: $\mathcal{L}$, $\mathcal{R}$, and $\mathcal{F}$. The $\mathcal{F}$ set is reserved for RSUs, while the $\mathcal{L}$ and $\mathcal{R}$ sets are associated with vehicles moving in left and right directions respectively. Each time slot is identified by the index of this time slot within a frame, and each node (i.e. vehicle or RSU) is identified by a short identifier (ID) which is chosen by each node at random, included in the header of each packet transmitted on the CCH, and changed if the node detects that its ID is already in use by another node [7]. For a certain node x, the following two sets are defined:

- $N(x)$: the set of IDs of the one-hop neighbours of node x on the CCH, from which node x has received packets

978-1-4673-1881-5/12/$31.00 ©2012 IEEE

on the CCH in the previous S slots;

- $T(x)$: the set of time slots that node x must not use on the CCH in the next S time slots.

The set $T(x)$ is used by node x to determine which time slots it can access on the CCH without causing any hidden terminal problem. Next, how each node x constructs and updates the set $T(x)$ is discussed.

In the VeMAC protocol, each node must acquire exactly one time slot and must transmit a packet in each frame on the CCH. Once a node acquires a time slot, it keeps accessing the same slot in all subsequent frames unless a transmission collision happens. Two types of transmission collision can happen on the CCH [1]: access collision and merging collision. An access collision happens when two or more nodes within two hops of each other attempt to acquire the same available time slot. On the other hand, a merging collision happens when two or more nodes acquiring the same time slot become members of the same two-hop set (THS)[1] due to node activation or node mobility. In VANETs, merging collisions are more likely to occur among vehicles moving in opposite directions or between a vehicle and a stationary RSU since they approach each other with a much higher relative velocity as compared to vehicles moving in the same direction.

For the purpose of time slot assignment on the CCH, in the header of each transmitted packet, the transmitting node y should include the set $N(y)$ and the time slot used by each node $z \in N(y)$. Suppose node x is just powered on and needs to acquire a time slot. It starts listening to the CCH for S successive time slots (not necessarily in the same frame). At the end of the S slots, node x can determine $N(x)$ and the time slot used by each node $i \in N(x)$. In addition, since each node $i \in N(x)$ announces $N(i)$ and the time slot used by each node $j \in N(i)$, node x can determine the time slot used by each of its two-hop neighbours, $j \in N(i), j \notin N(x), \forall i \in N(x)$. Accordingly, node x sets $T(x)$ to the set of time slots used by all nodes within its two-hop neighbourhood. Then, the sets $N(x)$ and $T(x)$ are updated by node x at the end of each slot (always based on information received in the previous S slots).

Given $T(x)$, node x determines the set of accessible time slots $A(x)$ (to be discussed) and then attempts to acquire a time slot by randomly accessing any time slot in $A(x)$, say time slot k. Node x will determine whether its attempt is successful or not by observing the $S-1$ time slots following k. The attempt of node x is considered successful iff the packets received from all $i \in N(x)$ indicate that $x \in N(i)$. Otherwise, node x re-accesses one of the time slots in $A(x)$ until it successfully acquires a time slot. Once node x acquires a time slot, it keeps using the same slot in all subsequent frames unless a merging collision happens. Similar to an access collision, a merging collision is detected by node x as soon as it receives a packet from a node $i \in N(x)$ indicating that $x \notin N(i)$.

At the end of each time slot, the collision detection by a certain node x should be done <u>before</u> updating the set

[1]A two-hop set is a set of nodes in which each node can reach any other node in two hops at most.

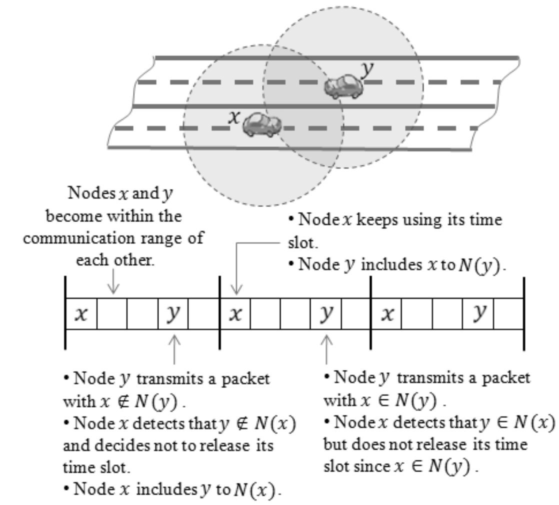


Fig. 1: The SRP condition preventing node x from unnecessarily releasing its time slot.

$N(x)$. Upon receiving a packet from a node y indicating that $x \notin N(y)$, we stress on that, node x should approve this collision detection and release its time slot iff the transmitting node $\underline{y \in N(x)}$. This condition is referred to as the slot release prevention (SRP) condition, and its objective is to prevent node x from unnecessarily releasing its time slot when it just enters the communication range of another node y. To illustrate that, consider the time slot assignment shown in Fig. 1 for the two nodes x and y. When node x enters the communication range of node y, even if no collision happens, the first packet received by node x from node y will indicate that $x \notin N(y)$. The reason is that, by the time node y transmits its packet, node y has not yet received any packet from node x to include it in $N(y)$. By applying the SRP condition, when node x receives the first packet from node y, node x determines that node $y \notin N(x)$ and does not release its time slot (recall that collision detection by node x is done before updating the set $N(x)$). After node x's transmission, the subsequent packets transmitted by node y will indicate that $x \in N(y)$ and, hence, the unnecessary release of node x's time slot is prevented. Note that, without the SRP condition, when two nodes enter the communication range of each other, one of them will eventually release its time slot even if no merging collision happens.

When a node x is attempting to acquire a time slot, a parameter called the split up parameter, denoted by τ, determines how node x accesses the time slots belonging to the $\mathcal{L}$, $\mathcal{R}$, and $\mathcal{F}$ sets. That is, consider that node x is moving in one of the right directions. Initially, node x limits the set $A(x)$ to the available time slots associated with the right directions, i.e. $A(x) = \overline{T(x)} \cap \mathcal{R}$. If after τ frames node x cannot acquire a time slot, then node x augments $A(x)$ by adding the time slots associated with the opposite direction, i.e. $A(x) = \overline{T(x)} \cap (\mathcal{R} \cup \mathcal{L})$. If, after τ more frames, node x still cannot acquire a time slot, node x will start to access any available time slot, i.e. $A(x) = \overline{T(x)}$. The same procedure applies for a vehicle moving in a left direction by replacing $\mathcal{R}$ with $\mathcal{L}$. Similarly, if node x is an RSU, for the first τ frames

TABLE I: Simulation parameters

Parameter	Highway	City
highway length	$1km$	–
# horizontal streets	–	3
# vertical streets	–	3
City street length	–	$430m$
# city blocks	–	4
City block edge length	–	$200m$
# lanes/direction	2	1
Lane width	$5m$	$5m$
Speed mean value	$100km/h$	$50km/h$
Speed standard deviation	$20km/h$	$10km/h$
Transmission range	$150m$	$150m$
# slots/frame	100	100
# slots for left directions	40	40
# slots for right directions	40	40
# slots for RSUs	20	20
Slot duration	$1ms$	$1ms$
Simulation time	2 min.	2 min.
# vehicles	25 to 265 (step = 20)	66 to 516 (step = 50)
# RSUs	55	84
Traffic load (TL)	0.24 to 0.96 (step = 0.06)	0.17 to 0.70 (step = 0.06)

$A(x) = \overline{T(x)} \cap \mathcal{F}$, and then $A(x) = \overline{T(x)}$. Note that, when $\tau = \infty$, regardless of the number of access collisions that node x has encountered to acquire a time slot, it can only access the time slots reserved for its moving direction (i.e. in the $\mathcal{R}$ set). On the other extreme, when $\tau = 0$, node x can access any available time slot on the CCH even if it does not experience any access collision.

Using the proposed scheme, a reliable broadcast service can be provided on the CCH. That is, if node x transmits a broadcast packet on time slot k, by listening to the $S-1$ time slots following k, node x can determine the set ξ of one-hop neighbors which have successfully received the packet, where $\xi = \{i \in N(x) : x \in N(i)\}$. In other words, when node i indicates that $x \in N(i)$, it is considered as an implicit acknowledgement by node i of receiving the packet broadcasted by node x.

III. Simulations

This section presents MATLAB simulations to investigate the effect of the existence of RSUs on the VeMAC protocol and to evaluate its performance under unbalanced vehicle traffic conditions using different values of the split up parameter τ.

A. Simulation Scenarios and Performance Metrics

The first scenario under consideration is a segment of a two-way vehicle traffic highway. A node can communicate with all the nodes within its communication range, i.e. no obstacles. The RSUs are placed on one side of the highway with equal distances between successive RSUs. Each vehicle moves with a constant speed drawn from a normal distribution, and the number of vehicles on the highway segment remains constant during the simulation time. When a vehicle reaches one end of the highway segment, it re-enters the segment from the other end. For this reason, to prevent the unrealistic merging collisions caused by vehicles which jump from one end to the other end, if a vehicle is located at a distance $d \leq R$ (R is the communication range) from one end of the highway segment, it can communicate with vehicles located within a distance $R-d$ from the other end of the segment. In this way, for each traffic direction, the vehicles at the end of the segment act as if they are following the vehicles at the start of the segment.

The other scenario is a city grid plan consisting of three horizontal and three vertical two-way vehicle traffic streets. All the streets have the same dimensions, and the horizontal and vertical streets are evenly spaced resulting in four identical square city blocks. The RSUs are distributed around the city blocks, and the distances between any two successive RSUs placed on the same edge of a city block are equal. The area of intersection of a horizontal street with a vertical one is referred to as a junction area. Each vehicle moves with a constant speed drawn from a normal distribution. When a vehicle reaches a junction area, it chooses one of all possible moving directions with equal probability (vehicles are not allowed to leave the simulation area during the simulation time). A node located at a junction area can communicate with nodes within its communication range located on both streets intersecting at the junction area. On the other hand, a node located at a street but not at a junction area cannot communicate with nodes located on other streets due to the existence of city blocks which obstruct the wireless signal.

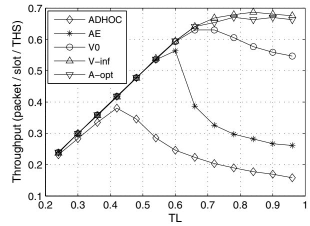
For both scenarios under consideration, all the transmitted packets are broadcast packets, the wireless channel is ideal, and the only source of packet errors is the transmission collision. Table I summarizes the simulation parameters for the highway and city scenarios. The traffic load (TL) shown in the last row of Table I is a parameter which equals $(N_v+N_r) \times \frac{2R}{L_h} \times \frac{1}{S}$ or $\frac{(N_v+N_r)}{N_s} \times \frac{2R}{L_s} \times \frac{1}{S}$ in the highway and city scenarios respectively, where N_v is the total number of vehicles, N_r is the total number of RSUs, N_s is the number of streets in the city, R is the communication range, L_h is the length of the highway segment, L_s is the length of a street in the city, and S is the number of slots per frame on the CCH. Note that, the ratio $\frac{N_v+N_r}{N_s}$ approximately equals the number of nodes in a city street, the number S represents the maximum number of time slots available for a THS, and the length $2R$ is the maximum length that a THS can occupy on the highway segment or on a city street. Consequently, the TL indicates the ratio of the number of time slots required by a THS to the total number of time slots available for a THS.

The performance metrics under consideration are the rates of access and merging collisions, defined as the average numbers of access and merging collisions respectively per slot per THS, and the throughput, defined as the average number of successful transmissions per slot per THS. A transmission by a vehicle x in a certain time slot is considered successful if and only if no other vehicles in the two-hop neighbourhood of x transmits in the same slot. Each of the metrics is obtained first using the results from the whole simulation area, and then multiplied by $\frac{2R}{L_h}$ or $\frac{1}{N_s} \times \frac{2R}{L_s}$ for the highway and city scenarios respectively. At the beginning of the simulations, the vehicles are randomly (uniformly) placed on the highway segment and on all streets of the city. The vehicles remain stationary and try to acquire a time slot by using the MAC protocol under consideration. Once no more vehicle can acquire a time slot, the vehicles begin moving and the simulation timer starts. The objective of this process is to quickly bring the system to a steady state where most of the vehicles have acquired a time slot.

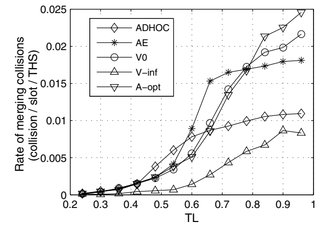
B. Simulated Protocols

Different versions of the VeMAC protocol are considered based on the value of the split up parameter τ. The VeMAC protocol with $\tau = m$ is denoted by V-m, where $m = 0, 5, 10,$ and ∞ (inf) in the simulations. The V-0 and V-inf versions of the VeMAC protocol are compared with the ADHOC MAC protocol as described in [7]. The original ADHOC MAC in [7] has two main limitations. First, due to the lack of a condition similar to the SRP condition in VeMAC, when two vehicles having acquired a time slot enter the communication range of each other, one of them releases its time slot even if no merging collision happens. Second, as mentioned in [7], a node which needs to acquire a time slot should attempt transmission in the next available time slot with probability p. For a certain time slot, the optimal probability is $p_{opt} = 1/N_c$, where N_c is the number of contending nodes attempting to acquire this time slot [7]. However, since N_c is not known to any of the contending nodes, each contending node x sets $N_c = N_{max} - N_{succ}(x)$, where N_{max} is the maximum number of nodes which can exist in a THS and $N_{succ}(x)$ is the number of nodes in the two-hop neighbourhood of node x which have successfully acquired a time slot as derived from the framing information received by node x [7]. This estimation of N_c is far from accurate. The reason is that, if a node x detects that $N_{succ}(x)$ nodes have successfully acquired a time slot, this does not mean at all that there are $N_{max} - N_{succ}(x)$ nodes which need to acquire a time slot in the two-hop neighbourhood of node x. Also, even if there are exactly $N_{max} - N_{succ}(x)$ contending nodes, they do not necessarily contend for the same time slots since each of the nodes may belong to a different set of THSs. Additionally, N_{max} is not constant since it depends on parameters such as the inter-vehicle distance and the number of lanes which considerably vary based on the networking environment.

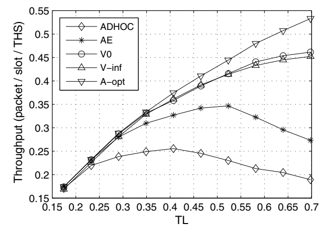
Based on the two limitations of the ADHOC MAC protocol in [7], two more versions of ADHOC MAC are considered in the simulations: the ADHOC-enhanced (AE) and the ADHOC-optimal (A-opt). The AE protocol eliminates the first limitation of ADHOC MAC by using a condition similar to the SRP condition of VeMAC. More precisely, a node x does not release its time slot based on a packet received from a node y unless node x has previously received a packet from node y, i.e. unless node y is included in the framing information [7] constructed by node x. For both AE and ADHOC MAC, the probability of accessing an available time slot by a contending node x is $p = \frac{1}{S - N_{succ}(x)}$. Note that, N_{max} is replaced by S (i.e. the maximum number of slots available for a THS) as it is not mentioned in [7] how to determine N_{max}. To evaluate the second limitation of ADHOC MAC, the A-opt protocol is implemented. The A-opt is similar to the AE protocol with the difference that, for each time slot, each contending node is aware of the number of contending nodes N_c within its two-hop neighbourhoud and sets $p = p_{opt} = \frac{1}{N_c}$. Note that, this awareness of N_c is provided by the simulator and cannot be achieved in reality. Hence, the A-opt is not a realistic protocol, but represents an upper bound on the ADHOC MAC performance.



(a) Throughput in the highway.



(b) Merging collisions in the highway.



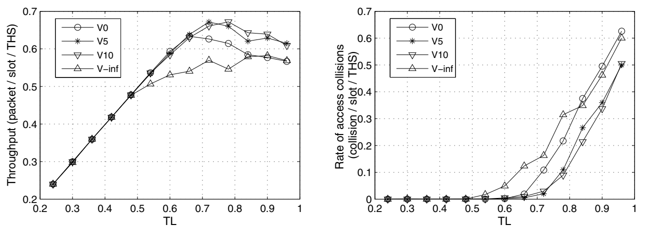
(c) Throughput in the city.

Fig. 2: Balanced vehicle traffic.

C. Simulation Results

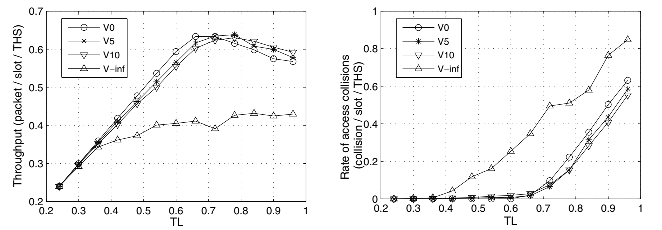
Fig. 2a shows the throughput for the highway scenario in a balanced vehicle traffic condition. It is clear that, the V-inf protocol outperforms all the other protocols including A-opt, especially for a high traffic load. The reason is that, as shown in Fig. 2b, the V-inf protocol can reduce the rate of merging collisions by strictly assigning disjoint sets of time slots to vehicles moving in opposite directions and to RSUs. This strict slot assignment eliminates any merging collision between a vehicle and an RSU or between two vehicles moving in opposite directions, which decreases the rate of merging collisions of the V-inf as compared to that of the other protocols.

The throughput in the city scenario is shown in Fig. 2c for all the protocols. Unlike the highway scenario, the throughput of the V-inf protocol is less than that of the A-opt protocol when the traffic load is high (TL > 0.35). Also, the performance difference between the V-0 and V-inf protocols is considerably reduced as compared to the highway scenario, with the V-0 protocol performing slightly better for a high traffic load. These results are due to two main factors. First, in the city scenario, all the protocols are less affected by the merging collisions due to a low moving speed of the vehicles as compared to the highway case. This low speed of vehicles limits the main advantage of the V-inf protocol in reducing the rate of merging collisions. Second, in the city scenario, the V-inf protocol suffers from the merging collisions near the junction areas due to vehicles changing their moving directions. However, in both city and highway



(a) Throughput in the highway. (b) Access collisions in the highway.

Fig. 3: Unbalanced vehicle traffic.



(a) Throughput in the highway. (b) Access collisions in the highway.

Fig. 4: Unbalanced vehicle traffic without RSUs.

scenarios, both VeMAC versions significantly outperform the AE and ADHOC protocols.

The throughputs in Figs. 2a and 2c are obtained under a balanced vehicle traffic condition. That is, at the start of the simulation, each vehicle chooses to move in one of the two opposite directions of the highway (and of each street of the city) with a probability equal to 0.5. To investigate the effect of an unbalanced vehicle traffic condition on the VeMAC performance, we consider the same highway parameters in Table I, but each vehicle initially chooses to move east or west with probabilities 0.25 and 0.75 respectively. The throughput and the rate of access collisions under this unbalanced traffic condition are shown in Figs. 3a and 3b respectively for the VeMAC protocol with different values of split up parameter τ. As shown in Fig. 3b, with a low traffic load (TL < 0.5), the rate of access collisions is negligible for all the protocols due to a low contention on the time slots. For the same reason, all the protocols provide the same throughput for a TL < 0.5, as shown in Fig. 3a. On the other hand, for a TL between 0.5 and 0.84, the V-inf protocol shows the lowest throughput, with almost the same performance as the V-0 protocol for a TL > 0.84. The low throughput and the high rate of access collision of the V-inf protocol result from the strict reservation of disjoint sets of time slots to vehicles moving in opposite directions and to RSUs. In the highway scenario under consideration, by using this strict reservation of time slots, all the vehicles moving west are contending for the time slots belonging to the $\mathcal{L}$ set, while many time slots in the $\mathcal{R}$ set remain unused due to a low number of vehicles moving in the east direction. This strict reservation of time slots becomes more harmful in the absence of RSUs. As shown in Figs. 4a and 4b, without RSUs, the V-inf protocol performs considerably worse than the other VeMAC versions starting from a traffic load as low as 0.3, due to the unnecessary reservation of time slots for the RSUs.

From the results presented in this section, it is obvious that the performance of the VeMAC protocol is highly dependent on the value of the split up parameter τ. Among the values of τ used in the simulations, there is no single τ value which provides the highest throughput in all the scenarios for all the TL values. For instance, as shown in Fig. 4a, while a TL value between 0.4 and 0.65 results in the V-0 protocol having the best performance, followed by the V-5, then the V-10 protocols, this order is totaly reversed for a TL > 0.8. Consequently, to maximize the performance of the VeMAC protocol, the τ value must be dynamically adjusted by taking into account factors such as the density of the vehicles, density of RSUs, speed of the vehicles, and ratio of the densities of vehicles moving in opposite directions. Based on these factors, how to determine the τ value should be further studied.

IV. Conclusions and Future Work

This paper investigates the effect of the existence of RSUs on the performance of the VeMAC protocol in comparison with ADHOC MAC. It is shown that, in a highway scenario, the VeMAC protocol achieves a higher throughput than the ADHOC-optimal protocol which represents an upper bound on the ADHOC MAC performance. On the other hand, in a city scenario, the throughput of VeMAC is less than that of ADHOC-optimal but considerably larger than the throughput achieved by both of ADHOC MAC and ADHOC-enhanced protocols. Also, by considering balanced and unbalanced vehicle traffic conditions, it is shown that, the throughput and the rates of transmission collisions of the VeMAC protocol are highly affected by the choice of the split up parameter τ value. Furthermore, the performance of VeMAC using the same τ value differs based on the density of RSUs, speed of the vehicles, and densities of the vehicles moving in opposite directions. Based on these factors, it is important to determine how each node dynamically updates the τ value to achieve the highest throughput. As well, a packet delay analysis of the VeMAC protocol should be conducted and comparisons with the IEEE 802.11p protocol should be made.

References

[1] H. A.Omar, W.Zhuang, and L.Li, "VeMAC: a novel multichannel MAC protocol for vehicular ad hoc networks," in *Proc. IEEE INFOCOM Workshops (MobiWorld 2011)*, Apr. 2011, pp. 413–418.

[2] H. A.Omar, W.Zhuang, and L.Li, "VeMAC: A TDMA-based MAC protocol for reliable broadcast in VANETs," *IEEE Trans. Mobile Comput.*, to appear.

[3] R.Baldessari *et al.*, "Car-2-car communication consortium manifesto," Tech. Rep. Version 1.1, Aug. 2007.

[4] "Vehicle safety communications project task 3 final report," The CAMP VSCC, Tech. Rep. DOT HS 809 859, Mar. 2005.

[5] *IEEE Std 802.11p-2010*, pp. 1–51, Jul. 15, 2010.

[6] M.Hassan, H.Vu, and T.Sakurai, "Performance analysis of the IEEE 802.11 MAC protocol for DSRC safety applications," *IEEE Trans. Veh. Technol.*, vol. 60, no. 8, pp. 3882 –3896, Oct. 2011.

[7] F.Borgonovo, A.Capone, M.Cesana, and L.Fratta, "ADHOC MAC: New MAC architecture for ad hoc networks providing efficient and reliable point-to-point and broadcast services," *Wireless Networks*, vol. 10, pp. 359–366, Jul. 2004.