Urban Multi-Hop Broadcast Protocol for Inter-Vehicle Communication Systems

Gökhan Korkmaz
korkmazg@ece.osu.edu

Eylem Ekici
ekici@ece.osu.edu

Füsun Özgüner
ozguner@ece.osu.edu

Ümit Özgüner
umit@ece.osu.edu

Department of Electrical and Computer Engineering
2015 Neil Avenue, 205 Dreese Lab.
Columbus, OH 43210-1272, US
Ph: (614) 292 3430 Fax: (614) 292 7596

ABSTRACT

Inter-Vehicle Communication Systems rely on multi-hop broadcast to disseminate information to locations beyond the transmission range of individual nodes. Message dissemination is especially difficult in urban areas crowded with tall buildings because of the line-of-sight problem. In this paper, we propose a new efficient IEEE 802.11 based multi-hop broadcast protocol (UMB) which is designed to address the broadcast storm, hidden node, and reliability problems of multi-hop broadcast in urban areas. This protocol assigns the duty of forwarding and acknowledging the broadcast packet to only one vehicle by dividing the road portion inside the transmission range into segments and choosing the vehicle in the furthest non-empty segment without apriori topology information. When there is an intersection in the path of the message dissemination, new directional broadcasts are initiated by the repeaters located at the intersections. We have shown through simulations that our protocol has a very high success rate and efficient channel utilization when compared with other flooding based protocols.

Categories and Subject Descriptors

C.2.1 [**Network Architecture and Design**]: Wireless communication; C.2.5 [**Local and Wide-Area Networks**]: Access schemes

General Terms

Design, Performance, Algorithms

Keywords

Wireless networks, multi-hop, broadcast, IEEE 802.11, IVC, vehicle, intersection

1. INTRODUCTION

Recently, Inter-Vehicle Communication Systems (IVC) have attracted considerable attention from the research community and automotive industry [1]. Many automobile manufacturers started planning to build communication devices into their vehicles for purposes of safety, comfortable driving, and entertainment. In IVC systems, broadcast is a frequently used method. Possible applications relying on broadcast include sharing emergency, traffic, weather, and road data among vehicles, and delivering advertisements and announcements. These applications generate packets of various lengths at different rates. For example, accident warnings are short packets that are generated infrequently. Another type of warning packet generated when the road is slippery because of ice or rain is also short but these packets may be sent in bursts. Finally, advertisement packets of restaurants or hotels can be broadcast in very long packets that carry pictures, directions, or even small videos.

When a message is disseminated to locations beyond the transmission range, multi-hopping is used. Unfortunately, interference, packet collisions, and hidden nodes can stop the message dissemination during multi-hop broadcast. Moreover, multi-hop broadcast can consume significant amount of wireless resources because of unnecessary retransmissions. These facts increase the importance of a MAC layer design for efficient and reliable multi-hop message dissemination. In addition, broadcast communication has another challenge in urban areas. Especially in an urban area crowded with tall buildings around intersections, it is difficult to disseminate the packets to different road segments shadowed by these buildings.

The topology and the node movement of an IVC network is constrained by roads. The resulting communication network is a special kind of Mobile Ad-Hoc Network (MANET) where the mobility rate is high but movement direction and speeds are predictable. In MANETs, flooding the network blindly is the first approach to achieve broadcasting since flooding can operate without local or total topology information. However, it has been shown in [2] that serious redundancy, contention, and collision problems occur as results of flooding. Although [2] proposes techniques to improve blind flooding, their solutions are not effective for all ranges of node densities and packet loads. Unfortunately, in IVC applications, both the node density and packet load fluctuate significantly. In [3],[4], methods to eliminate redundant packets while broadcasting is proposed using the topology information. However, in an IVC network, the large number of nodes and high mobility make such pro-active approaches impractical [5].

RTS/CTS handshake and acknowledgement mechanisms are some of the methods that make the IEEE 802.11 [6] a widely accepted wireless LAN standard for point-to-point communication. RTS/CTS mechanism decreases the effect of the hidden node problem while acknowledgement mechanism makes the protocol reliable. However, since broadcast packets have more than one destination, employing RTS/CTS and ACK packets may cause packet storms around the source. To handle this problem, some protocols use the topology information to directly select the nodes which will send CTS and ACK packets [7],[8].

In [5], IEEE 802.11 protocol is adapted for broadcasting in IVC systems by employing a distance based waiting approach before retransmissions. Although this approach distributes the highly correlated rebroadcast times, problems such as hidden nodes, collisions at high packet traffic rates, reliability, and broadcast storms still persists. Another flooding based protocol is proposed in [9] for broadcasting short packets in IVC systems. This protocol limits the channel access rate of each vehicle by defining a transmission window.

In this paper, we propose a new efficient IEEE 802.11 based Urban Multi-hop Broadcast protocol (UMB) for adhoc vehicular networks. UMB is designed to address (i) broadcast storm, (ii) hidden node, and (iii) reliability problems in multi-hop broadcast. The UMB protocol is composed of two phases, namely *directional broadcast* and *intersection broadcast*. We first introduce a new directional broadcast method where sender nodes try to select the furthest node in the broadcast direction to assign the duty of forwarding and acknowledging the packet without any apriori topology information i.e., sender selects the furthest node without knowing the ID or position of its neighbors. At the intersections, to disseminate the packets in all directions, we propose installing repeaters that forward the packet to all road segments. We showed through simulations that UMB protocol outperforms other broadcast protocols. The rest of the paper is organized as follows: In Section 2, we present the UMB protocol. In Section 3, we describe our simulation environment and discuss the results of the simulations. Finally, we conclude the paper with Section 4.

2. PROTOCOL DESCRIPTION

We assume that the vehicles of an IVC system form an ad-hoc network on a highway or in an urban area. At the intersections, there are simple repeaters which repeat the packets to the road segments incident to the intersection. We assume that since the repeater is at the intersection, it has a line-of-sight to all road segments. We also assume that each vehicle is equipped with a GPS receiver and an electronic road map. Since the vehicle mobility is high and vehicles leave and enter the network frequently, the topology of this network changes fast. Therefore, UMB protocol is designed to operate without exchanging location information among neighboring nodes.

The most important goals of our new protocol are as follows:

1. *Avoiding collisions due to hidden nodes:* In order to decrease the effect of hidden nodes, a mechanism similar to RTS/CTS handshake in point-to-point communication is employed by our new UMB protocol.

2. *Using the channel efficiently:* Forwarding duty is assigned to only the furthest vehicle in the transmission range without using the network topology information.

3. *Making the broadcast communication as reliable as possible:* To achieve the reliability goal, an ACK packet is sent by the vehicle which was selected to forward the packet.

4. *Disseminating messages in all directions at an intersection:* New directional broadcasts are initiated by the simple repeaters installed at the intersections according to the Intersection Broadcast mechanism.

2.1 Directional Broadcast

2.1.1 RTB/CTB Handshake

In order to avoid the hidden node problem while minimizing the overhead, we propose to engage in RTS/CTS handshake with only one of the recipients among sender's neighbors. If we can select the furthest away node in a linear road segment with RTS-CTS packets then other nodes in between can overhear the transmission as well and do not access the channel for a time interval specified in RTS and CTS packets. To select this vehicle, protocol divides the road portion inside the transmission range into segments. Note that these segments are created only in the direction of dissemination. If there is more than one node in the furthest non-empty segment, this segment is divided iteratively into subsegments with smaller widths. If these segment based iterations are not sufficient to pick only one node, the nodes in the last sub-segment enter to a random phase.

As a result of iteratively dividing the segments, the protocol can adapt itself to light or heavy vehicle traffic conditions. When the vehicle traffic is light, even a large subsegment width in the first iteration can be sufficient to select the furthest vehicle. For heavy vehicle traffic conditions, sub-segment width is reduced geometrically in every iteration. As an example, for a communication radius of 400 m and 10-way segmenting, the sub-segment width is reduced to 4 m in the second iteration, which is unlikely to contain more than one vehicle per lane. If the furthest vehicle cannot be selected in the second iteration, there is no need to further segment the 4 m range. Therefore, the random selection is performed starting the third iteration.

In this paper, we will refer to RTS and CTS as *Request to Broadcast (RTB) and Clear to Broadcast (CTB)*, respectively. In an RTB packet, in addition to the transmission duration, source node includes its position and intended broadcast direction. If the source wants to disseminate the message in more than one direction, a new RTB packet should be generated for each direction.

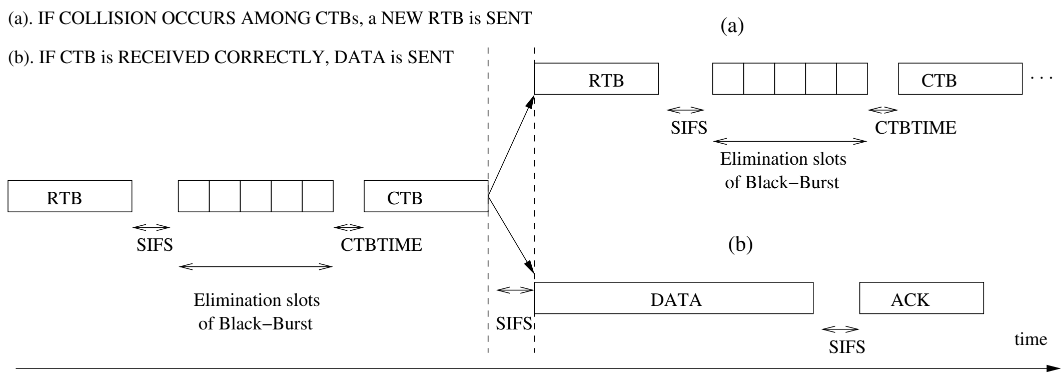


Figure 1: Sequence of packets. (a) Second RTB/CTB handshake (b) DATA/ACK.

Source vehicle obeys all IEEE 802.11 transmission rules (CSMA/CA) while attempting to send an RTB packet. When the nodes in the direction of the dissemination receive this RTB packet, they compute their distance to the source node. Based on this distance, they send an energy burst (channel jamming signal) called *black-burst*. The black-burst method was proposed in [10] and [11] to provide guaranteed access delays to rate-limited packet traffic. In these proposals, the length of the original black-burst is proportional to the time that the node has been waiting for channel access. In our directional broadcast, we use the black-burst to select the furthest node by letting receivers sending black-burst signals proportional to their distance to the source. Since the position information of all nodes are unique, using the position information to determine the length of the black-burst gives us the capability of selecting the furthest node.

The length of the black-burst signal in the first iteration is computed as follows:

$$L_1 = \lfloor \frac{\hat{d}}{Range} * N_{max} \rfloor * SlotTime, \tag{1}$$

where L_1 is the black-burst length in the first iteration, $\hat{d}$ is the distance between the source and the vehicle, $Range$ is the transmission range, N_{max} is the number of segments created, and $SlotTime$ is the length of one slot. Note that as a result of this computation, the furthest node sends the longest black-burst.

Nodes send their black-burst in the shortest possible time (SIFS) after they hear the RTB packet. At the end of the black-burst, nodes turn around and listen to the channel. If they find the channel empty, it means that their black-burst was longest and they are now responsible to reply with a CTB packet after a duration called $CTBTIME$, where $SIFS < CTBTIME < DIFS$. If they find the channel busy, it means that there are some other vehicles further away and they do not try to send CTB packet.

When there are more than one vehicle in the furthest non-empty segment, they all find the channel empty after sending their black-bursts and continue to send CTB packets. However, since all vehicles start sending the CTB packets at the same time, their CTB packets will collide. When the source node detects a transmission but cannot decode the CTB packet, it detects the collision and repeats the RTB packet after SIFS time as shown in Figure 1(a). This time, only the nodes which have sent CTB packets join the collision resolution. In order to pick only one node, the furthest non-empty segment is divided into N_{max} sub-segments. This process continues iteratively until a successful CTB packet is received by the source or D_{max} iterations are completed. The length of the black-burst for the i^{th} iteration (L_i) is computed as follows:

$$\begin{aligned} L_i &= \lfloor \frac{\hat{d} - Llongest_{i-1} * W_{i-1}}{W_{i-1}} * N_{max} \rfloor * SlotTime \\ & i = 2, 3, ..., D_{max} \\ W_i &= \frac{Range}{N_{max}^i}, \end{aligned} \tag{2}$$

where $Llongest_i$ and W_i are the longest black-burst and the segment width in the i^{th} iteration, respectively.

Note that in an RTB packet, source only indicates that there has been a collision: It is the receiver nodes' responsibility to choose the segment to be split. Only nodes who have sent the longest black-burst in the previous $(i-1)^{th}$ iteration can join to the current (i^{th}) iteration. As a result, $Llongest_{i-1}$ is the black-burst length of these nodes in the previous iteration and $Llongest_{i-1} + 1$ is the segment to be split.

If the segment based black-burst cannot resolve the collision after the D_{max}^{th} iteration, the vehicles that have sent the CTB response in the last iteration enter the random collision resolution phase. In this phase, vehicles choose random black-burst lengths from $[0, N_{max} - 1]$ slots. When there is a collision, nodes whose CTBs have collided will choose another random number. If the source cannot get a successful CTB after Ran_{max} random iterations, it waits a random amount of time and tries the segment based collision resolution from the beginning. Starting the node selection process from the beginning can happen at most RET_{max} number of times. The segment based iterations decrease the segment to a very short strip after D_{max} iterations. As a result, only a small number of nodes will be left at the beginning of the random phase and this will increase the success probability of this phase.

Detecting an empty channel after sending the RTB packet, the source node assumes that nobody has received its RTB packet. In this case, source node goes back to the first segment based iteration after a random amount of time. Details of this backoff procedure are the same as those of the IEEE 802.11 standard when CTS is not received.

2.1.2 Transmission of DATA and ACK

After receiving a successful CTB, the source node sends its broadcast packet as shown in Figure 1(b). In this broadcast packet, the source node includes ID of the node which has successfully sent the CTB. We will refer to this node as the *corresponding node of the source*. This node is now responsible for forwarding the broadcast packet and sending an ACK to the source. This ACK packet ensures the reliability of packet dissemination in the desired direction. Although all other nodes between the source and the ACK sender receive the broadcast packet, they do not rebroadcast or acknowledge it. If the ACK packet is not received by the source before the ACK timeout, the source goes back to the first segment based iteration after a random amount of time. Details of this backoff procedure are the same as those of the IEEE 802.11 standard when ACK is not received. Note that there is a maximum number of times (RET_{max}) source node can go back to the first iteration.

2.2 Intersection Broadcast

When there is an intersection in the path of the packet dissemination, new directional broadcasts should be initiated to all road directions at the intersection. Since there is a repeater at the intersection, it is the best candidate to initiate the directional broadcasts. This is because, among other nodes, repeaters have the best line-of-sight to the other road segments, especially when there are tall buildings around the intersection.

2.2.1 Finding the repeater and branching

When a node is selected to forward a packet and it is outside the transmission range of a repeater, it continues with the directional broadcast protocol as described in Section 2.1. On the other hand, if the node is inside the transmission range of a repeater, the node sends the packet to the repeater using the point-to-point IEEE 802.11 protocol. Note that each node knows the locations of itself, intersections, and repeaters with the help of the GPS and digital road map. According to our protocol, a node sends RTS to the repeater and only the repeater replies with the CTS packet if the channel is empty. Upon receiving the CTS packet from the repeater, the node sends the DATA packet and the transmission ends when it receives an ACK packet from the repeater. After receiving this broadcast packet, the repeater initiates a directional broadcast in all road directions other than the direction where it received the packet from.

An example of intersection handling is illustrated in Figure 2. In this figure, vehicle A uses the directional broadcast to reach B. Note that A is out of the transmission range of the repeater C. On the other hand vehicle B is in the transmission range of repeater C; therefore vehicle C uses IEEE 802.11 protocol to communicate with repeater C. Once repeater C receives the message, it initiates directional broadcasts to the north and south directions. Since the repeater D is in the transmission range of repeater C, it also sends the packet to repeater D using IEEE 802.11 protocol.

2.2.2 Loops

A packet can be delivered from one intersection to the other if there are enough cars in the road segment joining these two intersections. When there is a gap between vehicles whose length is larger than the transmission range,

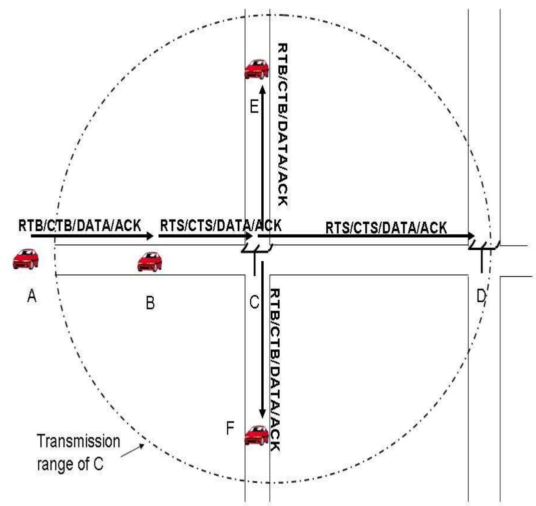


Figure 2: UMB prtocol

the two intersections are disconnected. If a path can be found starting from an intersection and ending in the same intersection using connected intersections, it is possible to have packet loops. When there is a packet loop, packets traverse the same road segments multiple times and waste bandwidth.

UMB protocol handles the looping problem with caching mechanisms. In the first approach, all cars in the network record the packet IDs when they hear packets. However, this can be costly in terms of memory usage. In the second approach, only the repeaters at the intersections record the packet IDs and they do not forward the packet if they have already received it. According to this approach, since the packet dissemination can be stopped only at the intersections, the packet may traverse a road segment twice as can be seen in Figure 3. In this figure, the car on the road segment DC initiates a broadcast. The packet is disseminated on two paths, namely *PATH 1* and *PATH 2*. Although both paths are ended successfully and a packet loop is avoided, the road segment AB is traversed twice. Either of the caching approaches can be implemented in order to avoid loops as a part of the UMB protocol; however there is a trade of between memory and bandwidth usage.

2.2.3 Optimization for long DATA packets

When a repeater receives a packet, it forwards it in all road directions, except the road direction from which it received the packet. Since our directional broadcast protocol is employed while forwarding the packets, the RTB/CTB/DATA/ACK handshake is repeated several times in intersection regions. As a result, the same information is potentially received by nearby nodes multiple times. Especially for long data packets, these repetitions waste significant amount of bandwidth. Moreover, keeping the channel busy around the repeater will degrade the overall performance of the network since packets from all directions will wait for the repeater to be idle.

In order to increase the efficiency of the protocol, repeaters do not repeat the information in the DATA packet if their

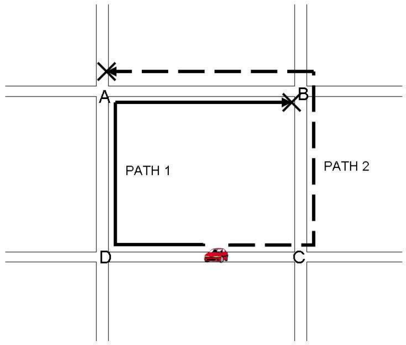


Figure 3: Using ID caches only in repeaters

corresponding node has already received this message. Corresponding node is the node that has successfully send the CTB packet to the repeater. In its CTB packet, corresponding node sets a bit if it has already overheard the packet before. Note that, as a result of this optimization, we have decreased the length of the DATA packet, however the repeater still needs to send a short DATA packet to assign the duty of forwarding to the corresponding node.

3. PERFORMANCE EVALUATION

3.1 Simulator

In order to evaluate the performance of the system, we have developed the Wireless Simulator (WS), which is based on an event driven simulation library CSIM [12]. WS models the MAC layer and the physical layer of the wireless network. The vehicle movement and the road structure is simulated by a separate simulator written in MATLAB.

3.2 Protocols

In addition to UMB, we have simulated two more MAC layer protocols using WS. In this paper, we will refer to these protocols as *802.11-distance* and *802.11-random*. They are flooding based modifications of IEEE 802.11 standard which route packets without the network topology information or any neighborhood knowledge. They try to avoid collisions among rebroadcast packets by forcing vehicles to wait before forwarding the packet. According to these protocols, every node must rebroadcast every distinct packet they receive once.

The first protocol, 802.11-distance, employs the idea proposed in [5], where the waiting time of the vehicles is inversely proportional to their distance from the source. The waiting time WT is computed as follows:

$$WT = (-\lfloor \frac{\hat{d}}{Range} * maxSlot \rfloor + maxSlot) * SlotTime, \quad (3)$$

where $maxSlot$ is the maximum possible number of slots a node waits before forwarding the packet. This waiting time aims the furthest node to broadcasts the packet first. As in IEEE 802.11 standard, nodes decrease their waiting time counters when they find the channel empty and freeze them when the channel is busy. System proposed in [5] computes the waiting time continuously, however in 802.11-distance implementation, waiting times are discrete since all waiting times are computed as multiples of $SlotTime$ in IEEE 802.11 standard.

In the second protocol, 802.11-random, when a node receives a broadcast packet, it will wait for a random duration (WT) before forwarding the packet.

$$WT = nSlots * SlotTime, \quad (4)$$

where $nSlot$ is random number between [0,$maxSlot$].

Finally, the UMB protocol is simulated with the following parameters: RET_{max} =15, N_{max} =10, D_{max} =2, Ran_{max}=3.

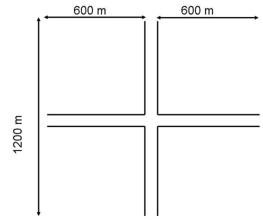


Figure 8: Road Structure I: one intersection, 1200 m x 1200 m

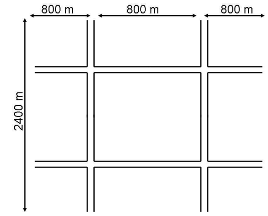


Figure 9: Road Structure II: four intersections, 2400 m x 2400m

Table 1: Parameters of the simulator

description	value
transmission range	400 m
data rate	1 Mbps
frame body	2312 bytes
base protocol	802.11b
$maxSlot$	32
simulation time ($simtime$)	60 s
simulation repetitions	30

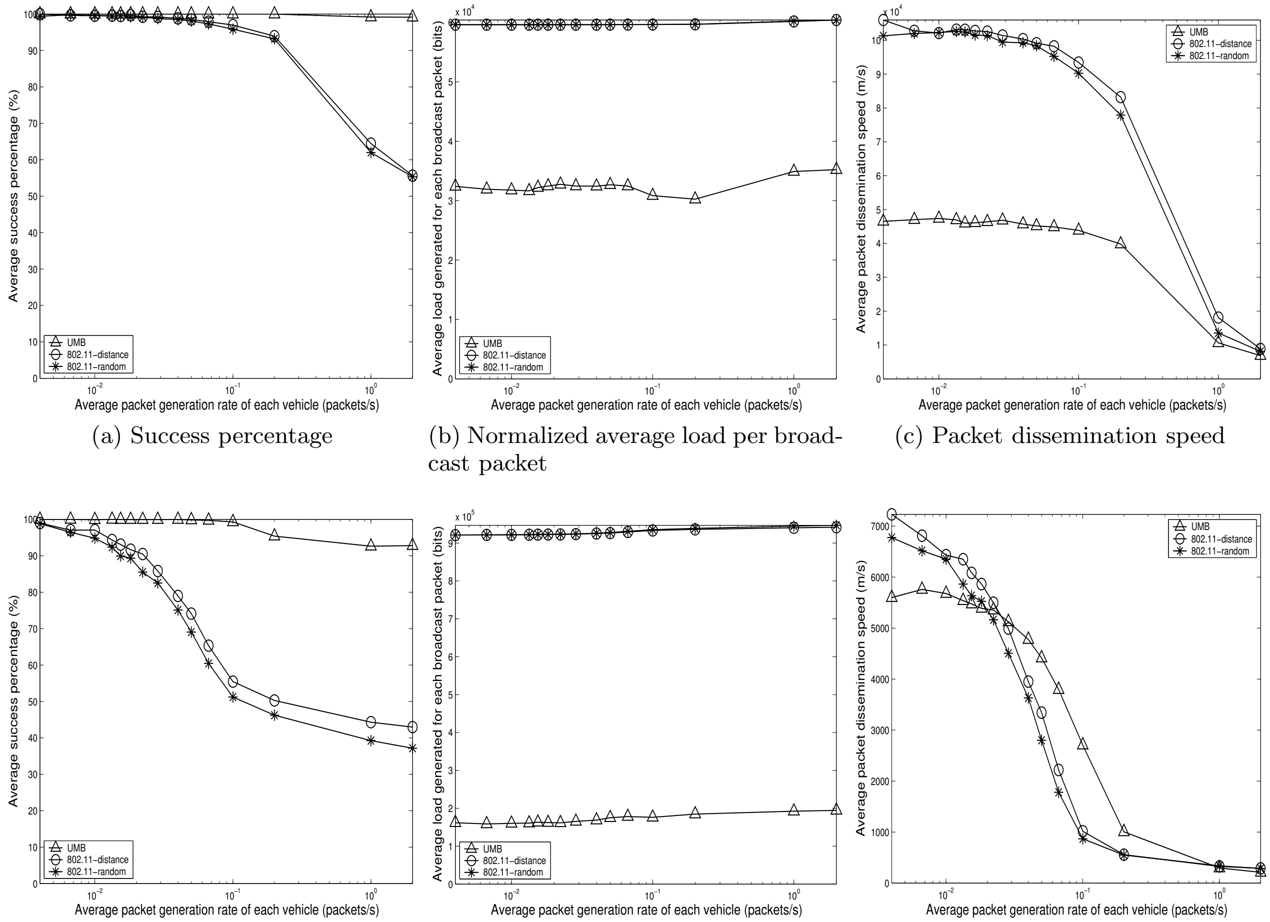
3.3 Common Simulation Parameters

Two types of road structures are implemented in our simulator. The simple road structure (Figure 8) includes one intersection with 600 m road segments. In addition to this simple structure, we have also created a road structure (Figure 9) with 4 intersections which can cause packet loops as discussed in Section 2.2.2. In these road structures, each road segment contains two lanes, one for each direction of traffic flow. The vehicles are randomly placed on road segments with exponentially distributed interspaces. For the sake of simplicity, lane changes, turns and overtaking is not modeled for vehicle movement. Each vehicle is assigned a speed from a Gaussian distribution with mean 40 km/h and standard deviation 5 km/h at the beginning of simulation and this speed remains constant during the simulation.

The common parameters of the simulator are summarized in Table 1. Simulator uses the 802.11b as the MAC layer. Detailed information about these parameters can be found in the IEEE 802.11b standard document [13].

3.4 Performance Metrics

Three metrics have been defined to compare the performance of UMB protocol with 802.11-distance and 802.11-random:



(a) Success percentage (b) Normalized average load per broadcast packet (c) Packet dissemination speed

(d) Success percentage (e) Normalized average load per broadcast packet packet (f) Packet dissemination speed

Figure 4: One intersection. Average vehicle density= 10 veh/km per lane. In (a), (b), (c) payload is 100 bytes and in (d), (e), (f) the payload is 2312 bytes

1. *Success Percentage*: Success Percentage of a packet is the ratio of the cars that receive the broadcast packet to the total number of cars in the simulation. When the average success percentage is lower than 100%, it means that the broadcast packets were not received by all vehicles.

2. *Packet Dissemination Speed (m/s)*: Speed of a packet at a point is computed by dividing the *distance travelled* by the packet to the *delay*. In the context of this paper, *delay* refers to the time elapsed between the instant the packet enters the source queue and the reception time of the packet by another node.

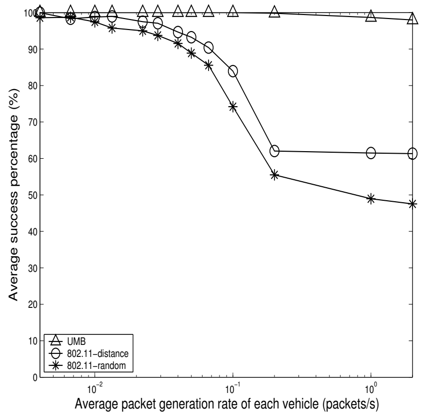
3. *Load Generated per Broadcast Packet* is the total number of bits transmitted to disseminate a packet to the whole network. In order to compute the average load, we divide the total number of bits sent by the total number of broadcast packets generated during simulation. This metric gives the total traffic generated by one broadcast packet in the network. Note that small values correspond to efficient usage of the channel.

When a packet is lost, it can reach only some part of the network and it generates a smaller load compared to a packet that reaches all nodes. For fair comparison, we divide the load generated by the $SuccessPercentage$ and define a normalized metric for *the average load generated per broadcast packet*. We have observed that this normalized metric is approximately constant for all packet generation rates.

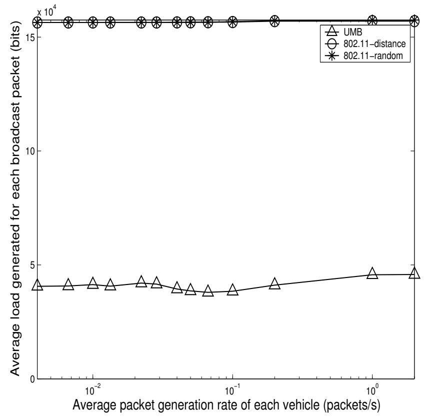
3.5 Results

3.5.1 *One intersection, average vehicle density= 10 veh/km per lane and total number of vehicles=61*

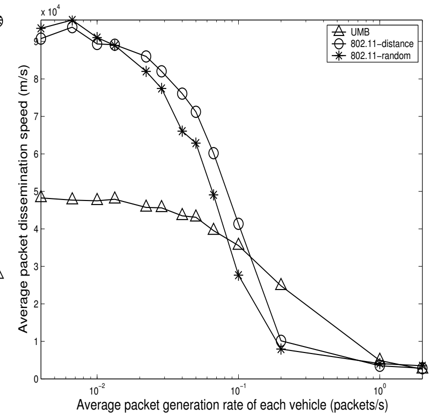
In this scenario, a simple map with one intersection is simulated with an average vehicle density of 10 veh/km per lane.



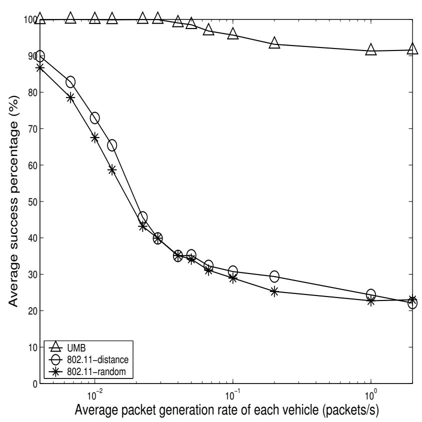
(a) Success percentage



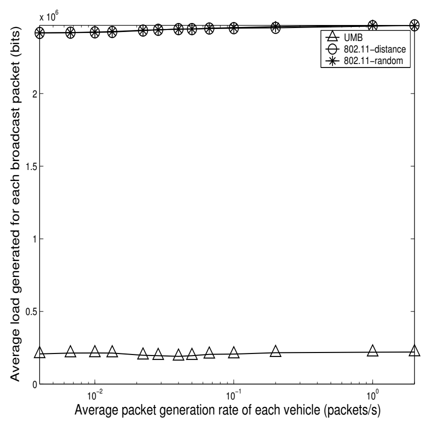
(b) Normalized average load per broadcast packet



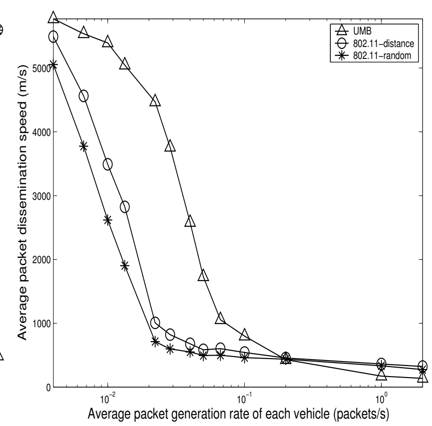
(c) Packet dissemination speed



(d) Success percentage



(e) Normalized average load per broadcast packet



(f) Packet dissemination speed

Figure 5: One intersection. Average vehicle density= 33.3 veh/km per lane and payload=100 bytes. In (a), (b), (c) payload is 100 bytes and in (d), (e), (f) the payload is 2312 bytes

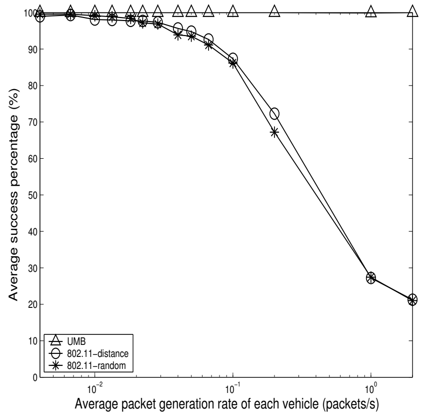
Figure 4(a) and Figure 4(d) depict the average success percentage when a payload length of 100 bytes and 2312 bytes are used respectively. In both figures, we can see that UMB protocol achieves approximately 100% success rate when the packet generation rate is low. When the packet generation rate is increased, UMB starts loosing some packets in the scenario with a long payload, on the other hand it is affected slightly (≈ %1) in the scenario with a short payload. 802.11-distance and 802.11-random protocols perform poorly because of packet collisions due to hidden nodes and the lack of the acknowledgment mechanism.

Figures 4(b) and 4(e) show the normalized average load generated per broadcast packet. In both figures, we can observe that UMB protocol generates less load while disseminating the packet to the whole network. As the packet generation rate increases, the packets of 802.11-random and 802.11-distance protocols start to collide and their success percentage decreases. Since some of the packets are lost, the load generated per packet becomes lower. However when we normalize the average load by dividing it by the success percentage, we have observed that this normalized values are almost constant at all rates. The length of the handshake packets (RTB,CTB,ACK) becomes negligible when the length of the data packet is long. In this case, UMB protocol performs approximately 5 times better than the other protocols as can be seen in Figure 4(e). This ratio decreases when the length of RTB,CTB, and ACK packets are comparable to DATA packet length and the number of cars in the transmission range is small (Figure 4(b)).

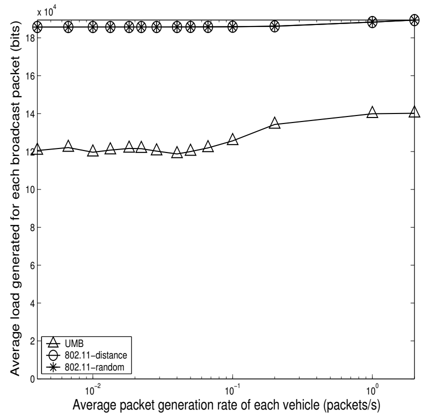
In Figures 4(c) and 4(f), it can be observed that the packet dissemination speed of all three protocols decrease when load is increased. Since the overhead of the handshake mechanism is comparable to the DATA packet in Figure 4(c), flooding based protocols, especially 802.11-distance protocol is faster than UMB, whereas the speed of all protocols are comparable when DATA length is large (Figure 4(f)).

3.5.2 One intersection, average vehicle density= 33.3 veh/km per lane, total number of vehicles=160

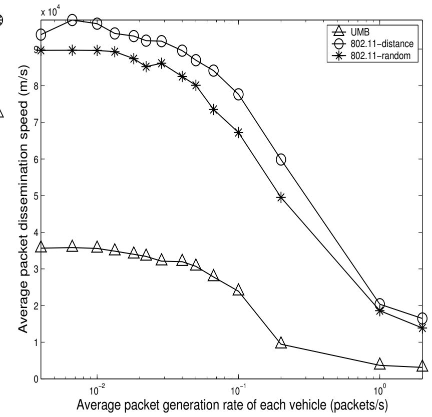
In this scenario, the same map with one intersection is used, however the vehicle traffic density is increased to 33.3



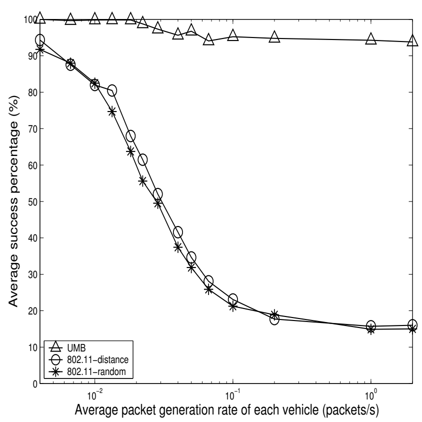
(a) Success percentage



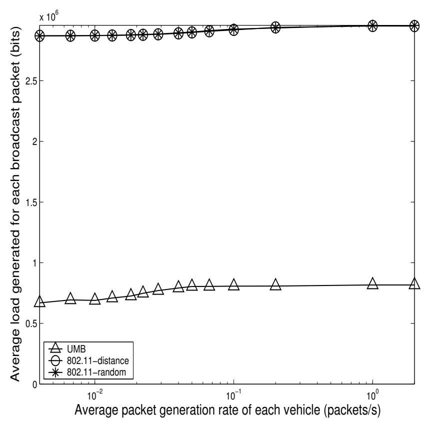
(b) Normalized average load per broadcast packet



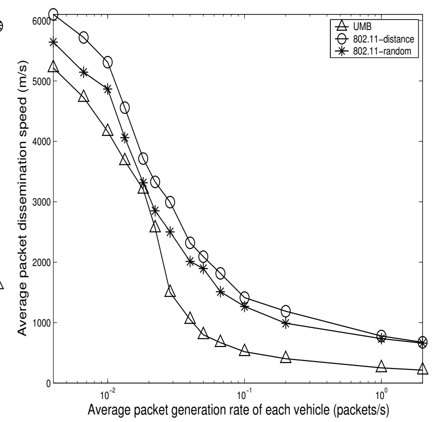
(c) Packet dissemination speed



(d) Success percentage



(e) Normalized average load per broadcast packet



(f) Packet dissemination speed

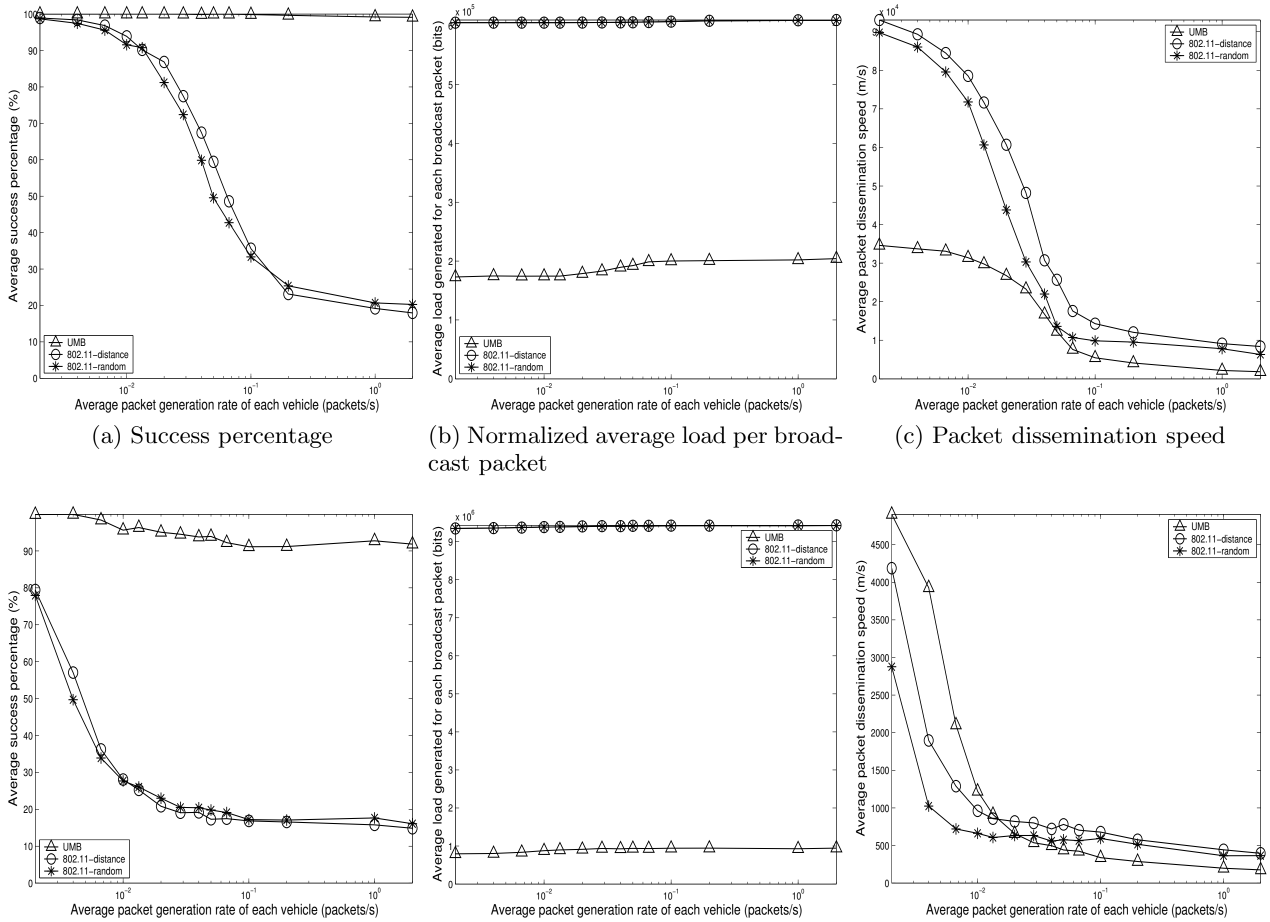
Figure 6: Fours intersection. Average vehicle density= 10 veh/km per lane and payload=100 bytes. In (a), (b), (c) payload is 100 bytes and in (d), (e), (f) the payload is 2312 bytes

veh/km per lane. Figure 5(a) and Figure 5(d) depict the average success percentage when a payload length of 100 bytes and 2312 bytes are used respectively. The increase in the number of vehicles becomes more effective when the payload is long. We can observe that the decrease in the success percentage of the 802.11-distance and 802.11-random protocols happens at a lower packet generation rate in Figure 5(d). This is because of the increase in the unnecessary rebroadcasts due to the higher number of vehicles. When the number of vehicles is increased, overall packet generation rate of the system also increases. This high packet rate also decreases the performance of UMB when especially long DATA packets are used as can be seen in Figure 6(d). However the success rate of UMB is higher than other flooding protocols at all packet generation rates.

Figures 5(b) and 5(e) show the normalized average load generated per broadcast packet. When the number of vehicles in the transmission range increases, the load generated by the flooding based protocols also increases, however the load generated by the UMB stays approximately the same. This is because, UMB protocol assigns the duty of forwarding the broadcast packet to only one vehicle in the transmission range while flooding based protocols assigns this duty to every vehicle. When we compare the results of current scenario with the results of section 3.5.1, we see that for both data lengths, the normalized load generated by the UMB protocol stays almost the same while the normalized load generated by the flooding based protocols increases approximately 2.6 times. This increase is equal to the increase in the total number of vehicles in the simulated network which increased from 61 to 160.

As can be seen in Figures 5(c) and 5(f), the packet dissemination speed of all three protocols decrease when the packet generation rate is increased. In Figure 5(c), UMB protocol performs worse than the other protocols when the packet generation rate is low. On the other hand, as illustrated in Figure 5(f), when we increase the length of the DATA packet, UMB protocol superior the winner in terms of speed over the other two protocols.

3.5.3 Four intersections, average vehicle density= 10 veh/km per lane, total number of vehicles=190



(a) Success percentage (b) Normalized average load per broadcast packet (c) Packet dissemination speed

(d) Success percentage (e) Normalized average load per broadcast packet (f) Packet dissemination speed

Figure 7: Fours intersection. Average vehicle density= 33.3 veh/km per lane, In (a), (b), (c) payload is 100 bytes and in (d), (e), (f) the payload is 2312 bytes

In this scenario, we have increased the number of intersections and formed a vehicle traffic which is able to create packet loops as described in Section 2.2.2. In these simulations, we have employed caching mechanism in repeaters to avoid packet loops.

Since we have a larger map in this scenario, the total number of vehicles in the network is higher. This increases the normalized average load generated per broadcast packet (Figure 6(b) and Figure 6(e)) when compared with the one intersection scenario discussed in Section 3.5.1. This increase in load affects the flooding based protocols negatively. As a result, as illustrated in Figure 6(a) and 6(d), the success rate of the flooding based protocols becomes worse than their success rates in one intersection scenario although the vehicle density remains the same.

As can be seen in Figures 6(c) and 6(f), just like in the scenario with one intersection with 10 veh/km, when the DATA length is short, flooding based protocols are faster than the UMB but when we increase the length of the DATA packet, the speed difference between UMB and flooding based protocols decreases.

3.5.4 Four intersections, average vehicle density= 33.3 veh/km per lane, total number of vehicles=619

As a result of the increase in the vehicle density, we have a 619 cars in our network. Since each car can initiate a broadcast packet, the overall packet generation rate of the system also increases. Coonsequently, as can be seen in Figure 7(a) and Figure 7(d), the success rate of all protocols, especially the flooding based protocols, decrease significantly. As in one intersection case of Section 3.5.2, the ratio of average load generated by the flooding protocols to that of the UMB protocol increases in both Figure 7(a) and Figure 7(e) when compared with the low vehicle density scenario. This increase in the ratio shows that in terms of load generated per broadcast packet, UMB protocol is not affected from increasing the vehicle density as much as flooding based protocols. Figure 7(c) and Figure 7(f) shows the packet dissemination speed for short and long DATA packets. Figure 7(c) depicts that when the DATA packet length is short, 802.11-distance and 802.11-random disseminate the packets faster than UMB protocol. Increasing the length of DATA packet increases the speed of UMB protocol relative to other protocols. In Figure, 7(f), UMB becomes the fastest one at low packet generation rates and its speed is close to others at high packet generation rates.

4. CONCLUSIONS

In this paper, we have presented a new efficient multi-hop broadcast protocol UMB for inter-vehicle communication in urban areas. Since this new protocol obeys 802.11 rules, it can coexist with other 802.11 modems which do not use this broadcast protocol. We have shown through simulations that our UMB protocol has a very high success percentage even at high packet loads and vehicle traffic densities. Moreover, since the forwarding duty is assigned to only one vehicle in the dissemination direction, it utilizes the channel very efficiently. In our future work, we plan to improve the UMB protocol to handle intersections without any repeaters.

5. ACKNOWLEDGMENTS

The authors would like to thank K. Tokuda, S. Nakabayashi and T. Shimizu of OKI Electric Industry Co. for their contributions.

6. REFERENCES

[1] H. Hartenstein, B. Bochow, A. Ebner, M. Lott, M. Radimirsch, and D. Vollmer, "Position-Aware Ad Hoc Wireless Networks for Inter-vehicle Communications: The Fleenet Project," in *ACM symposium on Mobile Ad Hoc Networking and Computing, MobiHOC*, Oct 2001.

[2] Y.-C. Tseng, S.-Y. Ni, Y.-S. Chen, and J.-P. Sheu, "The broadcast storm problem in a mobile ad hoc network," *Wireless Networks*, vol. 8, pp. 153–167, March-May 2002.

[3] W. Lou and J. Wu, "On reducing broadcast redundancy in ad hoc wireless networks," *IEEE Transactions on Mobile Computing*, vol. 1, pp. 111–122, April-June 2002.

[4] I. Stojmenovic, S. Seddigh, and J. Zunic, "Dominating sets and neighbor elimination based broadcasting algorithms in wireless networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 13, pp. 14–25, Jan. 2002.

[5] L. Briesemeister and G. Hommel, "Role-based multicast in highly mobile but sparsely connected ad hoc networks," in *IEEE/ACM Workshop on Mobile Ad Hoc Networking and Computing (MobiHOC)*, (BOSTON), pp. 45–50, Aug. 2000.

[6] The Institute of Electrical and Electronics Enginneers (IEEE), "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications." http://standards.ieee.org, ANSI/IEEE Std.802.11, 1999. (a.k.a. ISO/IEC 8802-11:1999(E)).

[7] K. Tang and M. Gerla, "MAC reliable broadcast in ad-hoc networks," in *MILCOM 2001*, vol. 2, pp. 28–31, Oct 2001.

[8] M.-T. Sun, L. Huang, A. Arora, and T.-H. Lai, "Reliable MAC layer multicast in IEEE 802.11 wireless networks," in *Proceedings of the International Conference on Parallel Processing*, pp. 527–536, 2002.

[9] K. Tokuda, M. Akiyama, and H. Fujii, "DOLPHIN for inter-vehicle communications system," in *Proceedings of the IEEE Intelligent Vehicle Symposium*, pp. 504–509, 2000.

[10] J. L. Sobrinho and A. S. Krisnakumar, "Distributed multiple access procedures to provide voice communications over IEEE 802.11 wireless networks," in *Global Telecommunications Conference*, vol. 3, pp. 1689–1694, 1996.

[11] J. L. Sobrinho and A. S. Krisnakumar, "Quality-of-service in ad-hoc carrier sense multiple access wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 17, pp. 1353–1368, August 1999.

[12] H. Schwetman, *CSIM User's Guide Rev. 1*. Microelectronics and Computer Technology Corporation Technical Report Number ACT-126-90, Rev. 1, 1991.

[13] LAN/MAN Standards Committee of the IEEE Computer Society, "Part11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Higher-Speed Physical Layer Extension in the 2.4GHz Band." IEEE Std 802.11b-1999, 1999.