Impacts of Radio Access Protocols on Cooperative Vehicle Collision Avoidance in Urban Traffic Intersections

Fan Yu and Subir Biswas
Electrical and Computer Engineering Department,
Michigan State University, East Lansing, MI USA
{yufan,sbiswas}@egr.msu.edu

Abstract—**This paper presents the design of a Cooperative Collision Avoidance application, and evaluates its performance with DSRC-recommended 802.11 Medium Access Control (MAC) and with a novel Vehicular Self-Organizing MAC (*VeSOMAC*) protocol. *VeSOMAC* is designed as a fully distributed TDMA protocol that relies on an in-band control exchange technique for autonomous TDMA slot allocation among vehicle-mounted wireless communication modules. A hybrid traffic and wireless network simulator has been developed for evaluating both network level and application level performance in the presence of different wireless access protocols. Detailed network and vehicular traffic simulation models have also been developed for evaluating a Cooperative Collision Control (CCA) application, operating in urban traffic intersection scenarios. Simulation results demonstrate that unlike the 802.11 style contention based protocols, a schedule based protocol such as *VeSOMAC* can offer better vehicle safety performance through smaller and bounded packet latency in vehicular ad hoc networks.**

***Index Terms*—Inter-vehicle Networks, MAC, Self-configuration, Intelligent Transportation System, DSRC**

I. INTRODUCTION

A. *Background and Motivations*

The Intelligent Transportation System (ITS) [1] architecture is currently being developed for enhancing vehicle safety using vehicle-to-vehicle (V2V) and vehicle-to-roadside (V2R) communications. Dedicated Short Range Communication (DSRC) [2], an emerging communication standard for ITS, was developed for 75 Mhz spectrum at the 5.9 Ghz band. Although IEEE 802.11p is recommended as the Medium Access Control (MAC) for DSRC, the protocol is known to suffers from unbounded delivery latency [3-5] at higher traffic loads primarily due to its underlying random access nature. It has been demonstrated [6] that the latency issue can be severe in the presence of vehicle crowding and broadcast storms during road emergency events. In this paper we propose a MAC protocol that can avoid such large delays by avoiding non-deterministic medium access.

B. *Related Work*

The vehicular MAC protocols in the literature are categorized as *contention-based* and *schedule-based*. The contention based approaches are generally not sensitive to underlying mobility and topology changes. As a result, vehicle movements do not usually impose any reconfiguration overhead due to the network topology changes. However, for all protocols in this category, the unbounded delay due to underlying random access is a serious issue. Although in some variations of CSMA/CA and 802.11 [7, 8] the issue is somewhat mitigated, the fundamental reasons for unbounded delay still remains.

These contention based approaches, however, are completely agnostic about the underlying mobility and topology changes. As a result, unlike the schedule-based protocols as explained later, the vehicle movements do not impose any MAC reconfiguration overhead due to the topology changes. This is a major advantage of the contention based approaches in vehicular networks, in which the rate of topology changes can be very high.

For the schedule-based TDMA protocol in [9], the slots are self-selected by nodes in a distributed manner. While providing bounded latency, the slot reallocation due to topology changes in this protocol may often incur a large convergence delay caused due to collision resolutions during the slot reallocation process itself. The protocol in [6] proposes a token ring based MAC protocol with its maximum delivery delay bounded by the round-trip token time. Delay for this protocol can be very large for large rings formed during vehicle crowding.

In the protocol LCA [10], TDMA slots are allocated based on a vehicle's instantaneous geographical location, which is pre-mapped to a TDMA slot. For this protocol, there is no reconfiguration latency due to topology changes. However, the system requires complete pre-mapping of geographical locations to TDMA slots, which may not be practical for transportation systems with large geographical coverage.

From the existing literature we conclude that schedule based protocols are desirable for their bounded delay, which is a critical requirement for ITS safety applications. However, the researchable question that still remains: *how to cope with the frequent topology changes in a vehicular network by fast TDMA reconfiguration.*

C. *Contributions*

The primary contributions of this paper are as follows. First, a novel distributed TDMA based medium access control protocol for wireless vehicular networking is developed. Second, based on DSRC and the emerging Intelligent Transportation System (ITS) use cases, an Urban Intersection Crash Warning (*UICW*) application is constructed. Finally, the *UICW* application is evaluated and analyzed using both 802.11, the DSRC-recommended MAC protocol, and the proposed TDMA protocol.

D. *Proposed VeSOMAC Protocol*

A distinctive feature of Vehicular Self-Organizing MAC (*VeSOMAC*) is its distributed design that does not rely on roadside infrastructure or virtual schedulers such as leader vehicles. This allocation autonomy, coupled with a novel bitmap based in-band signaling mechanism, allows *VeSOMAC* to perform fast slot reconfiguration after vehicle topology changes in urban traffic situations. Fast slot reconfiguration translates into low convergence latency which is missing in the existing deterministic protocols such as in [6] and [9].

II. VESOMAC PROTOCOL COMPONENTS

A. Frame and Slot Structure

The transmission slots (and packets) in *VeSOMAC* are of constant duration τ, and a frame is of duration T_{frame}, which defines the minimum periodicity of transmission from any vehicle. Therefore, the allocated rate to a vehicle is $\lambda_{alloc} = 1/T_{frame}$ packets per sec. In *VeSOMAC*, since a bitmap in the packet header is used for exchanging slot timing information, it is mandatory for each vehicle to send a packet every frame, even if no application data is available. *VeSOMAC* can operate in both synchronous and asynchronous modes. In the synchronous mode, all vehicles need to be time synchronized, and they share the same frame boundaries. In the asynchronous mode, vehicles can maintain their own frame boundaries.

B. Timing Constraint

MAC slot allocations needs to satisfy the following constraint: *No two one-hop or two-hop neighbors' slots can overlap. Overlaps between one-hop and two-hop neighbors cause direct and hidden collisions respectively.*

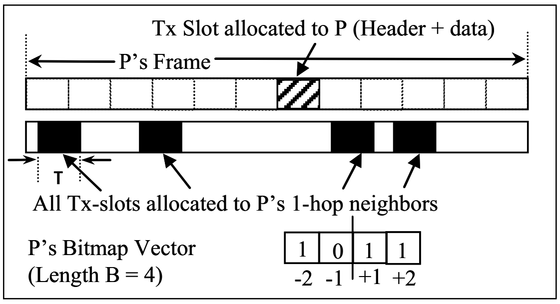


Fig. 1: Example in-band bitmap for asynchronous *VeSOMAC*

C. In-band Header Bitmap

Information about allocated slots is exchanged among the vehicle onboard units using a Bitmap Vector in each packet header. The concept is explained in Fig. 1. The top segment illustrates a vehicle *P's* allocated Tx-slot within its own TDMA frame. The middle row depicts the Tx-slots occupied by all of *P's* one-hop neighbors. Although these neighbors' slots are shown with respect to *P's* frame, each neighbor maintains its own asynchronous frame. The bottom row in Fig. 1 shows the bitmap vector that vehicle *P* inserts in each of its transmitted data packet headers. Middle of the bitmap represents *P's* own slot time. The bitmap vector here is 4-bit long and each bit represents the occupancy status of two slots around *P's* own Tx-slot. For example, the '1' in "+1" location indicates that two slots immediately following *P's* slot are already fully or partially occupied. Similarly, a '0' in the "-1" location indicates that vehicle *P* perceives both the slots before its own slot to be free. The bitmap vector length is a design parameter whose maximum value is the frame slot count. In Fig. 1, the frame size is 12, whereas the bitmap length is 4, which can convey the occupancy information about only 8 slots. With a bitmap size *4*, *P* is unable to represent the occupancy information about one of its neighbors' slots - the one in extreme left. To avoid this, the bitmap size should be the same as the frame size.

Using this header bitmap, a vehicle continuously informs its 1-hop neighbors about the slots occupied by its 1-hop neighbors. By listening to the 1-hop neighbors' transmissions and their bitmaps, a vehicle can detect the slot locations of its 1-hop and 2-hop neighbors. This information can then be used by the vehicle for choosing a collision-free Tx-slot that complies with the *timing constraint* as stated above. Since all timing information is relative, this approach allows *VeSOMAC* to be implemented with or without time synchronization.

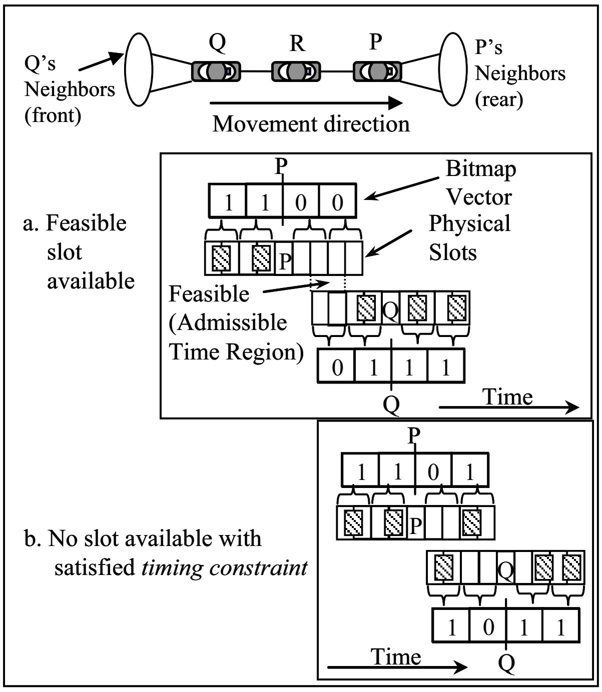


Fig. 2: Slot feasibility scenarios in asynchronous *VeSOMAC*

D. Transmission Slot Feasibility

A feasible transmission slot for a vehicle is one that satisfies the *timing constraints*. A feasible time region for a vehicle is defined by the region that is represented by shared '0's in the bitmaps transmitted by all its neighbor vehicles. A slot chosen from this feasible region is guaranteed to satisfy the *timing constraint.*

Consider the example in Fig. 2. A new vehicle *R* joins in between two unconnected vehicles *P* and *Q*. Bitmaps (with length 4) from *P* and *Q*, as received by the new vehicle *R*, are shown in Fig. 2:a. The shared '0's in the bitmaps of *P* and *Q* indicate a feasible time region for vehicle *R*. Since a shared '0' indicates that the corresponding feasible region is not used by any of *P's* and *Q's* 1-hop neighbors, a slot chosen in that region is guaranteed to be hidden collision free from all of *R's* 2-hop neighbors. And, since the slot within the feasible region is within the bitmap of all *R's* 1-hop neighbors (*P* and *Q*), it is guaranteed to be not used by any of those 1-hop neighbors. However, for the allocation in Fig. 2:b, since there are no shared ‘0’s, no region is feasible for *R*. If *R* chooses a slot from the time region indicated by a ‘0’ in *P’s* bitmap, then it would collide with a 1-hop neighbor of *Q*. Therefore, because of the violated *timing constraint,* a hidden collision cannot be avoided.

III. VESOMAC PROTOCOL OVERVIEW

A newly joined vehicle attempts to choose a collision free slot right after the slot of the vehicle immediately ahead. Upon choosing a slot, the vehicle starts transmitting data periodically once per frame. This may force vehicles in the neighborhood to an unstable allocation state. But then a distributed and iterative slot movement is used by all the neighborhood vehicles, including the new vehicle, to incrementally attain stable allocations. During these iterations, each vehicle attempts to place its slot behind the slot of its immediate front neighbor.

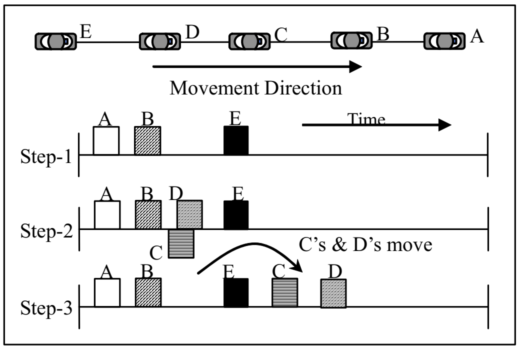


Fig. 3: Iterative slot movements for allocation convergence

Consider the topology in Fig. 3, in which the allocation step-1 depicts Tx-slots chosen by vehicles *A, B, and E* before vehicle *C* and *D* enter the network. With the allocation in step-1, vehicles *A*, *B*, and *E* each has a collision free slot that satisfies the *timing constraint*. Upon entering the network, *C* and *D* learns about the slot locations of their 1-hop neighbors *B* and *E* from their periodic transmissions, and the 2-hop neighbors (*A*) through the bitmaps in *B’s* and *E’s* packet headers. Then in Step-2, *C* and *D* attempt to independently select *timing constraint* compliant collision free slots. But in step-2 in this example, nodes *C* and *D* happen to choose overlapping slots which make them unstable. Using the collision resolution mechanism, as described in the following section, both *C* and *D* move their slots as shown in step-3. All vehicles at step-3 become stable. At this stage, if another vehicle joins causing a collision, all the affected vehicles will again move their slots iteratively to reach a mutual steady state.

A. Collision Detection and Resolution

Packet collisions are detected using implicit acknowledgements through the bitmaps. From the bitmaps transmitted by all its neighbors, a vehicle can infer if all those neighbors have received its own transmission. If not, the vehicle concludes that its transmission was missed due to a collision. If the situation persists for a preset number of frames, a collision is declared. In Fig. 2, if vehicles *P* and *Q* choose overlapping slots, a hidden collision will take place at *R*. Since *R* is not able to listen to *P’s* and *Q’s* transmissions due to the collision, it will indicate those two overlapping slots to be empty (‘0’) in its own bitmap. Upon receiving *R’s* bitmap, *P* looks for its own slot location in that bitmap to see if *P’s* transmission was successfully heard by *R*. A ‘0’ corresponding to *P’s* Tx-slot will indicate that there was a collision. If the situation persists for a preset number of frames then *P* will move its Tx-slot iteratively for resolving the collision. *Q* will also behave similarly.

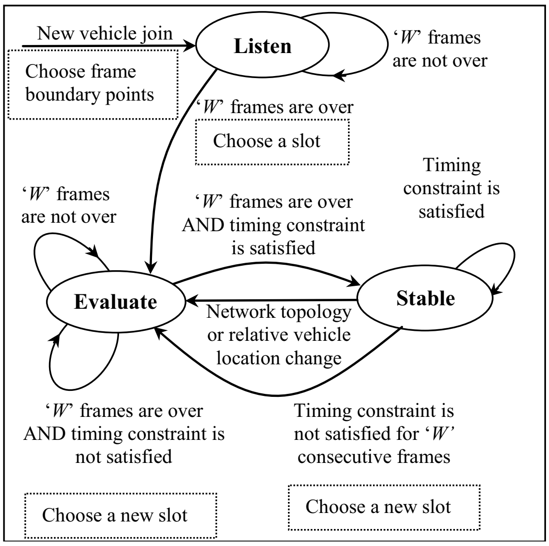


Fig. 4: State machine for the VeSOMAC protocol logic

B. Protocol State Machine

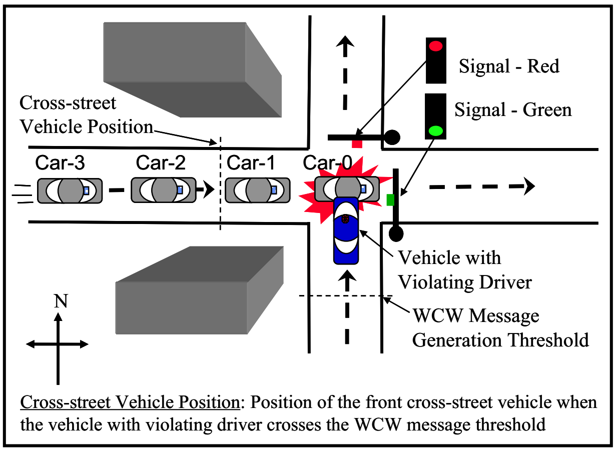
In *VeSOMAC* state-machine (see Fig. 4), the *Stable* state for a vehicle indicates the allocation steady state, and *Listen* and *Evaluate* are transient states. After a slot is chosen through the *Listen* state, a vehicle spends a preset (*W*) number of slots in the *Evaluate* state before getting into the *Stable* state. Any subsequent perturbations will force the vehicle to switch from the *Stable* to the *Evaluate* state. After a new slot is chosen, a vehicle enters in the *Evaluate* state. Subsequently, the slot is evaluated for *W* number of frames to ensure that its allocation became stable. When all the vehicles in a neighborhood reach the *Stable* state, the protocol converges.

C. Model for Frame Size Dimensioning

For packet duration of τ seconds, the channel capacity is $1/\tau$ packets per second per vehicle (ppsv). If *M* is the maximum number of combined 1-hop and 2-hop neighbors, then the wireless bandwidth in a neighborhood is shared by (*M+1*) vehicles. Therefore, the maximum data rate that can be allocated to each vehicle is given by: $\lambda_{\max} = 1/\tau(M+1)$ …. (1). Let the actual allocated data rate be λ_{alloc} ppsv ($\max[\lambda_{alloc}] = \lambda_{\max}$) and the corresponding frame duration be T_{frame} seconds. With one slot per vehicle per frame allocation, $T_{frame} = 1/\lambda_{alloc}$ and since $T_{frame} = F \times \tau$, one can write: $F = 1/(\tau \times \lambda_{alloc})$ …. (2). Considering $\lambda_{alloc} \le \lambda_{\max}$, from Eqns. 1 and 2 it can be written as $F \ge M+1$ …. (3). This equation represents the bound imposed by the timing constraint.

For the asynchronous *VeSOMAC*, according to the bitmap constraint, the bitmap from a vehicle is required to represent the slots of all its N 1-hop neighbors, and since each neighbor's slot can occupy at most two bits in the bitmap, $B \ge 2N$. Also, since each bit may correspond to at most two slot locations, one can write $F \ge 2B$. Combing these two, it can be written: $F \ge 4N$ …. (4). From Eqns. 3 and 4, the lower bound of frame size for asynchronous *VeSOMAC* is: $F \ge \max(M+1, 4N)$ …. (5). For the synchronous *VeSOMAC*, while the timing constraint poses the same lower-bound for F described by Eqn. 3, the bitmap constraint requires the conditions $B \ge N$ and $F \ge B$ to be satisfied. Therefore, the lower bound is: $F \ge \max(M+1, N)$ …. (6).

The frame size F should be chosen between the lower bound, computed through Eqns. 5 or 6, and an upper bound decided by the tolerable MAC delay which is $F.\tau/2$ in the average case and $F\tau$ in the worst case.



Cross-street Vehicle Position: Position of the front cross-street vehicle when the vehicle with violating driver crosses the WCW message threshold

Fig. 5: Urban Intersection Crash Warning (*UICW*) [2]

IV. URBAN INTERSECTION CRASH AVOIDANCE

Based on the DSRC-recommended use cases [11, 12], an Urban Intersection Crash Warning (*UICW*) ITS application is constructed and evaluated using both 802.11 and the proposed *VeSOMAC* protocol. V2V communication was leveraged for reducing vehicle crashes caused by traffic violating drivers in a one-way traffic intersection as shown in Fig. 5.

The example *UICW* execution in Fig. 5 depicts a situation when the South-to-North traffic light is red and the East-to-West light is green. Vehicle crashes occur when a violating driver on the South-to-North street runs the red light and collides with the cross-street vehicles. With *UICW* [2, 11] turned on, the DSRC onboard unit in the violating driver's vehicle first detects the situation from the vehicle's speed and location with respect to the intersection. If such a situation is detected when the vehicle reaches a threshold distance from the intersection, it starts broadcasting periodic Wireless Collision Warning (*WCW*) packets (e.g. once every 100ms [11, 13]). Upon receiving a *WCW* packet for the first time, each vehicle on the cross-street starts decelerating (e.g. at the rate of 4 m/s^2 [2, 14, 15]), after a driver's reaction time, in order to avoid any impending crash due to the event. Also, it rebroadcasts the packet when received for the first time. Fast *WCW* message delivery from the violating driver's vehicle across the cross-street vehicles is essential to reduce the number of vehicles involved in a chain crash in this situation.

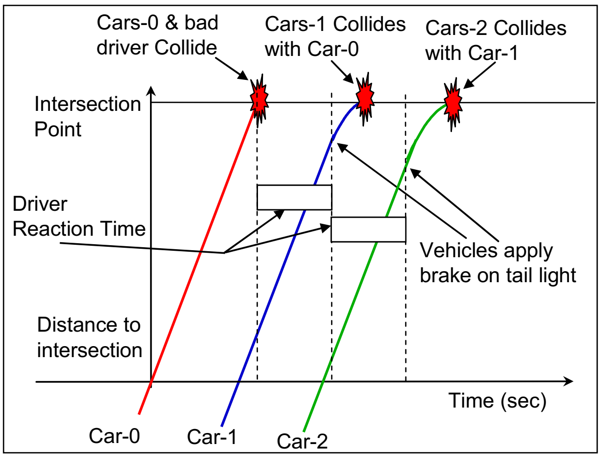


Fig. 6: Dynamics of a cross-street chain crashes without *UICW*

A. *UICW Operational Details*

Fig. 6 illustrates the dynamics of a chain crash after the front car (Car-0) on the West-to-East street collides with the vehicle with violating driver on the South-to-North street. For the sake of clarity, the dynamics of only three vehicles are presented in Fig. 6. The y-axis in Fig. 6 represents the vehicles' positions in terms of the distance from the street intersection point, as a function of time. As shown in the figure, the driver in Car-1 starts decelerating when he or she sees the tail brake light of Car-0, and the driver in Car-2 and Car-3 do so when they see the brake lights of the vehicles ahead. Note that a vehicle starts decelerating after a driver's reaction delay following when the vehicle ahead applies its brake. In Fig. 6, with a finite driver's reaction time, Car-0 gets hit by Car-1 at the intersection point. Subsequently, Car-1 is hit by Car-2, and Car-2 is hit by Car-3. This example shows when the drivers react solely on the visual information (tail brake light), how all the vehicles on the West-to-East street can end up in a chain crash.

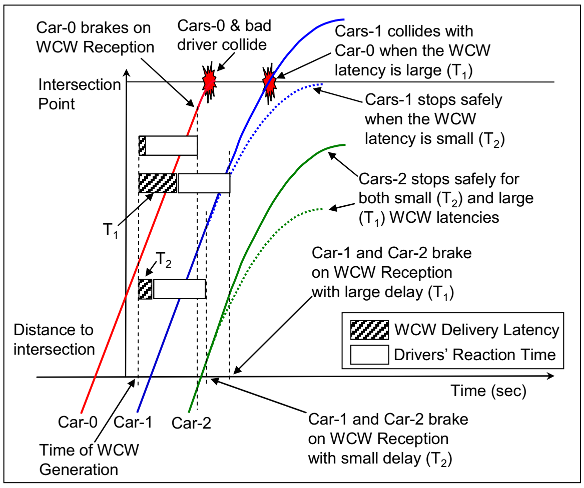


Fig. 7: Leveraging *UICW* for reducing intersection vehicle crashes

For the same scenario, the usefulness of the *UICW* application is illustrated in Fig. 7. With UICW turned on, all West-to-East vehicles apply their respective brakes and start decelerating after a combined delay of Wireless

Collision Warning (*WCW*) message delivery and a driver's reaction time following the generation of the *WCW* message by the vehicle of the violating driver. In the depicted example in Fig. 7, and for the given *WCW* delivery latency and driver's reaction time, although the driver in Car-0 is able to apply its brake, due to insufficiently available stop distance the vehicle is not able to avoid a crash with the vehicle in the South-to-North direction.

For Car-1 and Car-2, two scenarios are explored: with a large WCW message delivery delay T_1, and a small delivery delay T_2. For the same driver's reaction time, while the smaller WCW latency can save Car-1, the larger latency cannot. For Car-2, however, because of its sufficiently available stop distance, a crash can be avoided even with the large WCW delivery latency. The scenarios in Figs. 6 and 7 illustrate that: a) the *UICW* application can indeed reduce vehicle crashes when the vehicles in the West-to-East street rely on wireless warning messages, and b) low message delivery latency is a key to the overall success of this ITS application. This also reinforces the need for a MAC layer protocol with low and deterministic delivery latency. The core logic for *WCW* message generation and interpretation are shown as the pseudo-code in Fig. 8.

B. Multi-slot Message Broadcast with TDMA MAC

With TDMA, when the vehicle with violating driver in generates its *WCW* message, there is a possibility that this message will collide due to a MAC slot overlapping between the generating vehicle and at least another vehicle within its wireless range on the West-to-East street. Such packet collisions can occur before a TDMA slot reallocation can take place as a response to the network topology change caused by the vehicular movements. From an *UIWC* crash avoidance standpoint, these packet collisions can prove fatal and need to be avoided.

A multi-slot MAC broadcast mechanism has been introduced in which the *WCW* generating vehicle sends the message on multiple TDMA slots in a frame, in addition to on its own allocated slot. This way, if the generating vehicle's slot does collide with that of a cross-street vehicle, the redundancy in the multi-slot MAC broadcast will improve the chance of a packet collision free transmission of the *WCW* message. The redundant slots are chosen randomly within the TDMA frame during the successive *WCW* message transmission to avoid any persistent packet collisions, thus further improving the chance of successful delivery of a *WCW* message. In a *UICW* scenario with vehicle density D (average number of 1-hop radio neighbor of a vehicle) and frame size F (number of TDMA slots per frame), if the multi-slot MAC broadcast redundancy is n (the number of TDMA slots used for a multi-slot MAC broadcast), then the probability of a collision free WCW packet transmission can be written as: $1-(D/F)^n$. Note that this multi-slot MAC broadcast is needed only for the *UICW* style safety application, and may not be enabled for data intensive non-safety applications.

```
/*WCW Generation Logic at the vehicle w/ violating driver*/
do{
     keep checking the vehicle speed,
    distance to the intersection,
    and traffic signal status;
    if (this vehicle will be unable to stop before intersection
        && the traffic signal is currently red
        && distance to intersection <= threshold distance){
        /* the vehicle is expected to run a red light */
        Start originating WCW msg. periodically;
    }
}
```

```
/*  Msg Interpretation Logic at the West-to-East vehicles */
/*  A WCW Message arrives at a West-to-East vehicle*/
check the message originator and sequence number;
if (new msg. originator){
    store new msg. originator id and seq. #;
    rebroadcast the msg.;
}
else{
    if (seq. # is new){  // newly updated msg.
        store new seq. #;
        rebroadcast the msg.;
    }
    else{  // received old msg.
        drop the msg.;
    }
}
```

Fig. 8: Pseudo-code for *WCW* message generation and interpretation

V. PERFORMANCE EVALUATION

A hybrid simulator had been developed for joint evaluation of wireless network protocols and ITS applications within NS2 network simulator [16]. A vehicle traffic module that can interact with ITS applications, driver behavior logic, and the wireless network has been added within NS2. Fig. 9 depicts the architectural components of the developed Vehicular NeTwork Simulator (*VeNTSim*) system which is designed to be open for incorporating the evolving DSRC and other radio technologies, ITS applications [2], and their required network protocols. *VeNTSim*, is designed with open APIs for incorporating both DSRC and non-DSRC radio technologies. The goal is to architecturally evaluate and characterize the impacts of various wireless technologies on heterogeneous ITS applications proposed by the research community and various standardization consortiums.

The networking functions in *VeNTSim* has been developed on top of NS2 network simulator [16] by adding a vehicle mobility module that can react to the received wireless messages according to the modeled vehicle following logic with various drivers' reaction models. An ITS application modeling module, capable of simulating a series of ITS applications such as cooperative collision control, cooperative cruise control and emergency vehicle preemption [2] has been also added. The synchronous version of *VeSOMAC* has been implemented at the NS2 MAC layer, so that they can be compared with the 802.11 protocol running in the same radio environment. 802.11 is chosen for comparison because it is the current DSRC-recommended protocol.

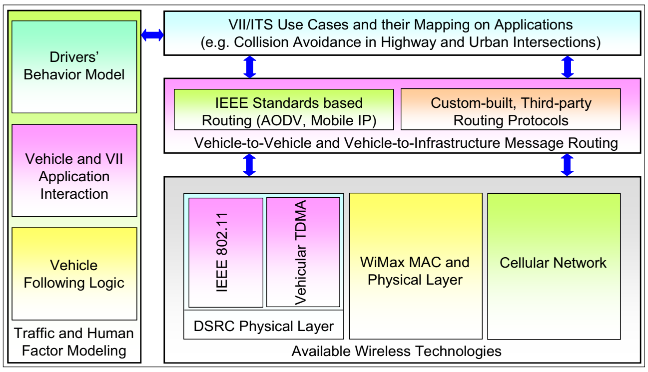


Fig. 9: *VeNTSim*: ITS application and network evaluation tool

A. *Experimental Parameters*

The *UICW* application was simulated using *VeNTSim* in the presence of background traffic generated by non-safety ITS applications. Due to their non-deterministic message recipients, all *UICW* traffic is forwarded using MAC layer broadcasts and multi-hop broadcast forwarding [11]. The non-safety background traffic is unicast forwarded both at the MAC and the routing layers. Each presented data point corresponds to the average from 500 independent simulation runs. The vehicle following logic in *UICW* comprises of the intersection traffic rules, and the drivers' behavior is modeled in terms of the reaction time with different ranges and distributions. The baseline simulation parameters are summarized in Table 1.

TABLE1: BASELINE EXPERIMENTAL PARAMETERS

Vehicle and Scenario Related	
Vehicle-count in West-to-East Direction	10 to 25 vehicles
Vehicle Speed	45 mph (20 m/sec)
Inter-vehicle Spacing	9m to 15m ≡ [0.45 sec to 0.75 sec]
Vehicle Length	4 m
Emergency Deceleration	8 m/s^2
Regular Deceleration	4 m/s^2
Drivers' Reaction Time	Fast [0.5 sec to 1.0 sec]
	Slow [0.75 sec to 1.5 sec]
WCW Message Generation Threshold Distance	50 m
Cross Street Vehicle Position	20 m to 50 m
Network Related	
Channel	DSRC 5.9 GHz band, 24Mbps
Radio Model	Two ray ground
Radio Range	100m
MAC Protocols	IEEE 802.11 and Worst case *Synchronous VeSOMAC*
WCW Packet Size	300 bytes (0.1 ms)
WCW Message Period	100 ms
VeSOMAC Frame Size	100 packets (10 ms)
State Evaluation Time W (frames)	3
Channel Packet Error Rate	5%

B. *Vehicle Crash Performance*

The percentages of 26 vehicles (25 on the West-to-East and one on the South-to-North with violating driver) that crash during the simulated *UICW* incident are reported in Fig. 10. When the *UICW* application is turned off, the drivers respond only to visual information, and a large number of West-to-East street vehicles crash. It was observed that the front vehicle on the cross-street first collides with the violating driver's vehicle, and then the vehicles behind engage in a chain collision. With no *UICW*, at 45 mph, even with a large vehicle spacing of 15 meters, almost 75% of the cross-street vehicles crash due to the red light running of the violating driver.

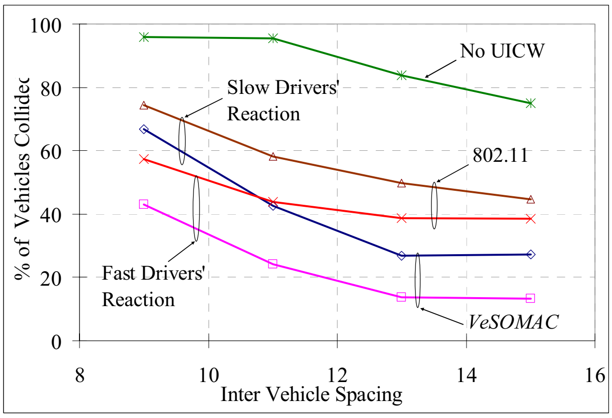


Fig. 10: Vehicle crash performance with 802.11 and *VeSOMAC*

C. *Impacts of MAC Protocols*

When the *UICW* application is turned on, with *VeSOMAC* as the MAC protocol, it was possible to bring the crashes down to nearly 13% (3 vehicles), which is with 15m spacing and fast drivers' reactions (0.5-1 second). With 802.11, the vehicle crash probabilities are observed to be significantly higher; that is 39% for 15m spacing and fast drivers' reactions. As expected, fewer vehicles crash with increasing vehicle spacing. This is because with larger inter-vehicle space, a vehicle gets a longer time cushion for safely stopping before crashing into the vehicle in front. Also, a fast drivers' reaction time helps to prevent the collision as shown in Fig. 10. The reason that *VeSOMAC* has much smaller collision number can be explained from Figs. 11 and 12 as follows.

The cross-platoon [1] *WCW* delivery latency for an example run of *UICW* with *VeSOMAC* is presented in the top graph of Fig. 11. This latency is defined by the duration between when the violating driver's vehicle generates the first *WCW* message after crossing the threshold point (see Fig. 5) and when it is delivered to a vehicle. Relative stop distances between consecutive vehicles are reported in the middle. With a vehicle length of 4m, any relative distance of 4m or less corresponds to a crash. For vehicles avoiding a crash, the relative distance thus indicates the margin of safety provided by the *UICW* application. The bottom graph reports the severity of crashes in terms of the relative speed between two crashing vehicles. Relative speeds greater than zero indicates a crash and its severity.

Similar results for an example *UICW* run with 802.11 MAC are reported in Fig. 12. For *VeSOMAC*, since there are no packet collisions and the cross-platoon latencies are very small (up to only 43 ms compared to seconds in 802.11), the crashes involve only the front of the cross-street vehicles. For 802.11, due to packet collisions the *WCW* latency increases significantly towards the rear of the cross street vehicles. This increase in latency causes a cluster of vehicles to crash due to insufficient reaction time. This explains the chain crashes at the middle of the platoon starting from vehicle 7 in Fig. 11. Crash performance from these specific *UICW* runs with *VeSOMAC* and 802.11 are consistent with the average

---

[1] This point onwards, the string of vehicles on the West-to-East street will be referred to as a platoon.

crash results from 500 different experiments presented in Fig. 10.

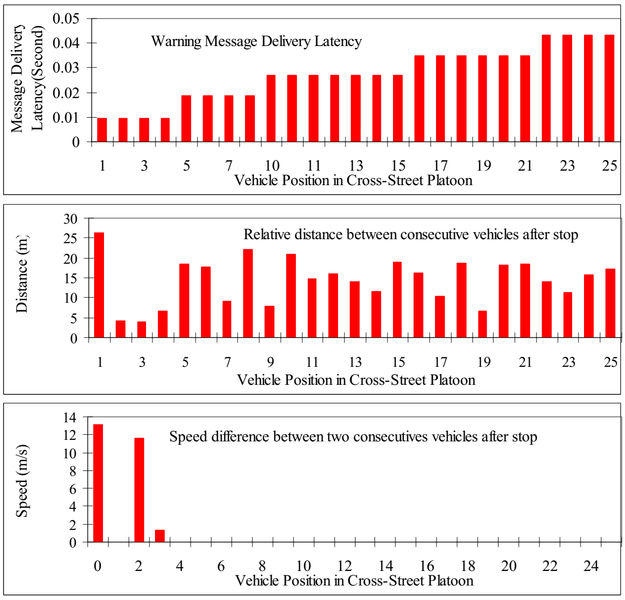


Fig. 11: Latency and crash statistics for *UICW* with *VeSOMAC*

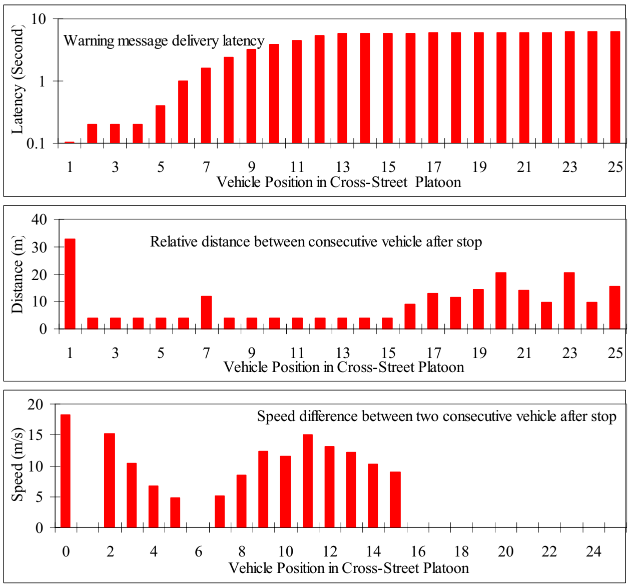


Fig. 12: Latency and crash statistics for UICW with 802.11

Based on these results we conclude that for *UICW* application, the schedule based *VeSOMAC* protocol offer significantly better vehicle crash performance compared to the DSRC-recommended contention based 802.11 protocols.

Packet drop statistics across the cross-street vehicles is presented in Fig. 13. Due to collisions, 802.11 is susceptible to frequent packet drops. For instance, with a background non-safety data rate of 40 packets per second per vehicle (ppsv), on an average the *UICW* application with 802.11 would loose the first WCW message by the time it reaches the 4th cross-street vehicle. Meaning, if the message was not periodically broadcast by the violating driver's vehicle, the vehicles beyond the 4th vehicle on the West-to-East street would not have received the message, thus suffering from the possibility of chain crashes. Similarly, the 2nd *WCW* message gets lost by the time it reaches the 5th vehicle. However, because of zero collisions, *VeSOMAC* can deliver the very first *WCW* message to all cross-street vehicles. These drop results reinforce the baseline *UICW* crash performance findings reported in Fig. 10.

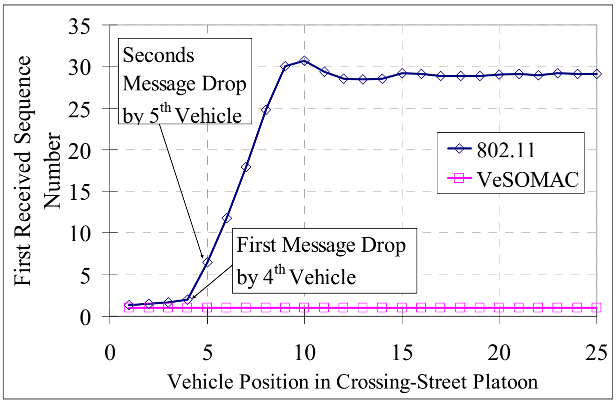


Fig. 13: *WCW* message drop at different platoon locations

D. *Impacts of Vehicle Count and Speed*

Crash performance with varying vehicle count and speed is reported in Fig. 14. As expected, with higher vehicle speeds, more vehicles crash. This is because for a given vehicle spacing, at higher speeds, the vehicles get lower stop distances to avoid a crash.

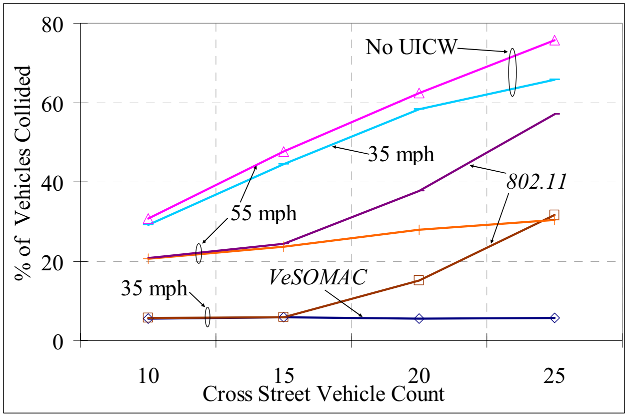


Fig. 14: Crash performance with varying speed and vehicle count

Unlike the crash results in Fig. 10 (for 25 vehicles), with fewer cross-street vehicles 802.11 performs as well as *VeSOMAC*. This is because the delay and drops for 802.11 is small (see Fig. 11 and 12) and comparable to those of *VeSOMAC* for the front cross-street vehicles. With larger number of vehicles, however, the latency is larger – which explains the escalated crashes for 802.11.

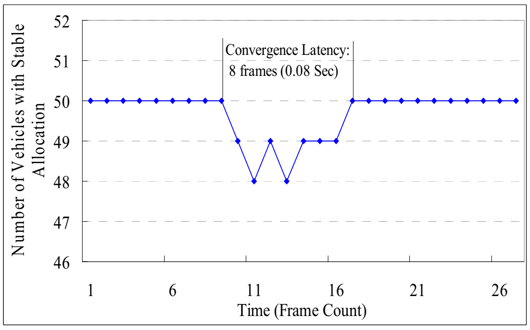


Fig. 15: VeSOMAC convergence after a topology change

E. *VeSOMAC Protocol Convergence*

During a topology change, the convergence latency for *VeSOMAC* is defined as the time interval from when at least one vehicle becomes unstable to when all the involved vehicles become *Stable* (see Fig. 4). The scenario shown in Fig. 15 corresponds to an 8-frame-long (0.08s) convergence process following a MAC instability triggered by a vehicle passing 23 vehicles in front.

Fig. 16 reports *VeSOMAC's* convergence performance when a vehicle passes varying number of vehicle ahead. As expected, the convergence latency increases with longer passing events because more vehicles' slots are prone to be violated in these cases. For all the experimented scenarios, the post-passing allocations have always converged within 88 ms.

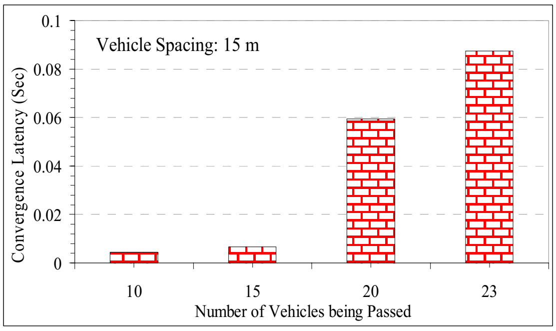


Fig. 16: Convergence latency for intra-platoon vehicle passing

VI. SUMMARY AND ONGOING WORK

This paper has developed a novel Vehicular Self-Organizing MAC (*VeSOMAC*) protocol which relies on an in-band control exchange technique for autonomous TDMA slot allocation among vehicle-mounted wireless communication modules. The paper also developed an Urban Intersection Crash Warning (*UICW*) application that leverages inter-vehicular wireless networking using traditional wireless MAC and the proposed *VeSOMAC* protocol. Finally, the impacts of 802.11 and *VeSOMAC* have been evaluated for the *UICW* application using a hybrid vehicle traffic and wireless network simulator. Simulation results demonstrate that unlike the 802.11 style contention based protocols, *VeSOMAC's* TDMA mechanism can offer better vehicle safety through smaller latency and packet drops. It was also shown that during topology changes, *VeSOMAC* can reallocate TDMA slots with a fast protocol convergence mechanism.

Ongoing work includes application of *VeSOMAC* for non-safety scenarios including inter-vehicle data streaming, and internet service provisioning to moving vehicles on freeways and urban traffic scenarios.

REFERENCES

[1] "Intelligent transportation systems," United States Department of Transportation, www.its.dot.gov/index.htm.

[2] L. Armstrong, "Dedicated Short Range Communications (DSRC)," www.leearmstrong.com/DSRC/DSRCHomeset.htm.

[3] Y. Wang and B. Bensaou, "Achieving Fairness in IEEE 802.11 DFWMAC with Variable Packet Size," in IEEE Global Telecommunication Conference, 2001.

[4] Y. Liu, F. Dion, and S. Biswas, "Dedicated Short-range Wireless Communications for Intelligent Transportation System Applications: State of the Art," ITS and vehicle-highway automation, vol. 1910, 2005.

[5] T. Pagtzis, P. Kirstein, and S. Hailers, "Operational and Fairness Issues with Connection-Less Traffic Over IEEE802.11b," in IEEE International Conference on Communication (ICC), 2001.

[6] M. Ergen, D. Lee, R. Sengupta, and P. Varaiya, "WTRP-Wireless Token Ring Protocol," IEEE Transaction on Vehicular Technology, vol. 53, 2004.

[7] N. H. Vaidya, "Mobile Ad-hoc Networks: Routing, MAC and Transport Issues," http://www.crhc.uiuc.edu/~nhv.

[8] R. M. Yadumurthy, A. Chimalakonda, m. Sandashivaiah, and R. Makanaboyina, "Reliable MAC Broadcast in Directional and Omni-directional Transmissions for Vehicular Ad Hoc Networks," the 2nd ACM International workshop on Vehicular ad hoc networks, 2005.

[9] T. Liu, J. A. Sylvester, and A. Polydoros, "Performance Evaluation of R-ALOHA in Distributed Packet Radio Networks with Hard Real-time Communication," in IEEE Vehicle Technology Conference 1995.

[10] S. Katragadda, G. M. C.N.S., R. R. M.S., K. S. M, and R. Sachin, "A Decentralized Location-based Channel Access Protocol for Inter-vehicle Communication," in Vehicular Technology Conference (VTC 2003-Spring), 2003.

[11] H. Krishnan and C. Kellum, "Use of Communication in Vehicle Safety Application," Internal Report of General Motors Company 2002.

[12] F. Yu and S. Biswas, "Impacts of Radio Access Protocols on Cooperative Collision Avoidance in Urban Traffic Intersection Scenarios," in IEEE 66th Vehicular Technology Conference (VTC 2007), Baltimore MD, USA, September-October, 2007.

[13] Q. Xu, T. Mak, J. Ko, and R. Sengupta, "Vehicle-to-Vehicle Safety Messaging in DSRC," in the 1st ACM International Workshop on Vehicular ad hoc networks, October 01-01, 2004.

[14] S. Biswas, R. Tatchikou, and F. Dion, "Vehicle-to-Vehicle Wireless Communication Protocols for Enhancing Highway Traffic Safety," in IEEE Communication Magazine, 2006.

[15] "Michigan Auto Accident News," www.michiganautolaw.com/blog/michiganautoaccidents/index.php.

[16] "The network simulator: NS-2," www.isi.edu/nsnam/ns.

Fan Yu received the BS degree in automation control and the MS degree in pattern recognition and intelligent control from the Huazhong University of Science and Technology, Wuhan, Hubei, China, in 2001 and 2004, respectively. He is currently working toward the PhD degree in Electrical and Computer Engineering at Michigan State University. His research interests include wireless sensor networks, mobile ad hoc networks, vehicular networks, and embedded systems. He is a student member of the IEEE.

Subir Biswas received the PhD degree from the Univ. of Cambridge. He held various research positions in the NEC Research Institute, Princeton, and AT&T Laboratories, Cambridge. He is an Associate Professor and the director of the Networked Embedded and Wireless Systems Laboratory, ECE Department, Michigan State University. His research interests include wireless data networking, low-power network protocols, and application-specific sensor networks. Subir has published over 70 peer-reviewed articles in the area of network protocols and is a co-holder of three US patents. He is a senior member of IEEE and a Fellow of the Cambridge Philosophical Society.