

An Inter-Vehicular Communication Architecture for Safety and Entertainment

C. E. Palazzi, *Member, IEEE*, M. Roccetti, *Member, IEEE*, S. Ferretti, *Member, IEEE*

***Abstract*— Inter-vehicle communication (IVC) is emerging in research prominence for the interest that is generating in all major car manufacturers and for the benefits that its inception will produce. The specific features of IVC will allow the deployment of a wide set of possible applications which span from road safety to entertainment. Even if, on the one hand, these applications share the common need for fast multi-hop message propagation, on the other hand, they possess distinct characteristics in terms of generated network traffic. The current research state of the art only proposes solutions specifically designed for a single application (or class) that are not directly extendable to a general IVC context. Instead, we claim that a *privileged* architecture exists, which is able to support the whole spectrum of application classes. To this aim, we propose a novel IVC architecture that adapts its functionalities to efficiently serve applications in quickly propagating their messages over a vehicular network. We conducted an extensive set of experiments that demonstrate the efficacy of our approach. As representative case studies, we considered two application classes that for their network traffic characteristics are at the opposite boundaries of the application spectrum: safety and entertainment.**

Manuscript received July 22, 2007. Partial financial support for this work is provided by: the Italian MIUR under the initiatives MOMA and DAMASCO.

Claudio E. Palazzi is with the Dipartimento di Matematica Pura e Applicata, Università degli Studi di Padova, Via Trieste 63, 35131 Padova, Italy (phone: +39 049 827 1426; fax: +39 049 827 1499; e-mail: cpalazzi@math.unipd.it).

Marco Roccetti, and Stefano Ferretti are with the Dipartimento di Scienze dell'Informazione, Università di Bologna, Mura Anteo Zamboni 7, 40127 Bologna, Italy (phone: +39-051-2094503; fax: +39-051-2094510; e-mail: {roccetti, sferrett}@cs.unibo.it). Marco Roccetti is the corresponding author.

***Index Terms*—Inter-Vehicular Communication, Multi-hop Broadcasting, Networked Interactive Entertainment, Road Vehicle Safety.**

I. INTRODUCTION

WHETHER you love them or hate them, vehicles represent such a fundamental component in our society that cities like Los Angeles look more like being developed having in mind a car-population, rather than a human-population. Now, their influence in our life is going to become even more pervasive thanks to the forthcoming inter-vehicle communication (IVC) capabilities [1].

Through these new communication capabilities, vehicles will provide several new services to their passengers while driving them around. A limited but representative list of new services that will be made available by the DSRC/IEEE 802.11p technology includes road vehicle safety, road navigation support, location-related commercials, and networked interactive entertainment.

A feature typically shared by these services is that of having application messages transmitted through a multi-hop, ad-hoc IVC among a group of vehicles (namely a *car platoon*) covering an area of few kilometers [2]-[7]. The problem in this context is that communications require very tight message delivery time to be effective: typically, under the threshold of few hundreds of milliseconds [7], [8].

Scientific literature reports that the propagation speed of messages decreases with the increase of vehicles that attempt to forward them over the multi-hop path toward destination and, in general, with the network traffic [7], [9]-[11]. Several approaches have been proposed, e.g., transmission rate adaptation, topology awareness-based optimization, intelligent election of message forwarders. Unfortunately, each of these schemes is affected by at least one of the following problems: redundant multi-hop transmissions, unrealistic assumptions about the vehicular environment, and generation of an elevate number of control messages. This causes an inefficient utilization of the (limited) available resources, thus negatively affecting the final performance of the system.

Furthermore, this situation is exacerbated by the fact that different applications have peculiar

characteristics, such as the generated network traffic, that existing solutions address in a specific way without a general vision. Instead, the efficiency of an IVC architecture should be independent from the application in use.

To this aim, we propose a novel IVC architecture intended to permit fast multi-hop broadcast of messages generated by a generic application. Our solution is able to adapt its functionalities to efficiently serve a wide variety of possible IVC applications, ranging from those generating very few and sporadic messages to those generating a continuous and intense transmission flow.

In essence, our solution works in a distributed way among vehicles in a car platoon and its main features are as follows.

- i) An **efficient priority scheme** to choose the next-hop forwarder of a broadcast message based on the distance from the previous sender and on the expected transmission range. This way, redundant transmissions (and message propagation delays) are reduced.
- ii) A **dynamic estimation of the transmission range** that is used to correctly compute the aforementioned priority. This represents a fundamental feature in a highly dynamic scenario such as a vehicular network.
- iii) Only a **minimal overhead** caused by very few data that have to be exchanged among vehicles to appropriately feed the transmission range estimator. When possible, data for the transmission range estimator are directly embedded into regular application messages, without resorting to control messages (*hello messages*) at all.
- iv) A **general solution framework** that enables quick propagation of messages whatever is their generation rate and type, thus efficiently serving the whole spectrum of IVC applications.

In conclusion, the main contribution of this work is twofold. First, responding to a well known problem in multihop message broadcasting over wireless networks, we propose a lightweight transmission range estimator that is able to provide a crucial information to select the best forwarder so as to increase the

message delivery speed. Second, we have assembled our estimator with a vehicular network architecture so as to bring our novel technical solution into a practical context where different real applications may be run.

The rest of the paper is organized as follows. In Section II we specify the scenario that we are considering, whereas in Section III we review related work. Our architectural solution is explained in Section IV. The experimental assessment and results are presented in Section V and Section VI, respectively. Finally, Section VII concludes this paper.

II. CONSIDERED SCENARIO

We consider a group of cars that drive at various speeds on a multi-lane, strip-shaped portion of road, e.g., a highway. Surroundings are represented by buildings, hills, trees, and several other possible sources of interference. IVC is thus exposed to frequent and ample variations in terms of transmission range and available bandwidth.

Vehicles are assumed to be endowed with on board systems for communication, self-localization, monitoring, and entertainment. This is a realistic assumption since:

- i) the ongoing development of the DSRC/IEEE 802.11p technology promises to provide vehicles with communication capabilities [1];
- ii) the popularity of GPS-based devices is increasing as they are becoming less and less expensive;
- iii) cars are more and more endowed with all sort of sensors to detect failures in the car but also dangers from outside the car (e.g., radars and cameras);
- iv) about a million of new cars and SUVs are sold every year in US with pre-installed TV/DVD systems [12].

This technology can be put to good use in a multitude of ways when placed on board, enabling a wide range of IVC applications and services.

In this paper we analyze the performance improvement achieved by our architecture when supporting

two of these application classes that, for their network traffic characteristics, are at the opposite boundaries of the IVC application spectrum: road vehicle safety and networked interactive entertainment. As can be seen in Table I, these two application classes share the same requirements for quick and reliable message delivery to all engaged vehicles; yet, other features may differ greatly: road vehicle safety applications tend to generate a little burst of traffic and for a limited time [5], [9]; whereas networked interactive entertainment generates a continuous flow of transmitted messages [8], [13]. As said, an IVC architecture should be effective whatever application were in use.

We consider two practical case studies, one from each of these two application classes: accident alert communication and online gaming, respectively. We choose this two cases as they represent two IVC “killer” applications that will have a great market penetration [1], [14].

TABLE I
APPLICATION PROPERTIES

Feature	Road Vehicle Safety	Networked Interactive Entertainment
<i>msg sources</i>	one - several cars	many cars
<i>msg payload</i>	20 - 200 bytes	40 - 1000 bytes
<i>msg rate</i>	~1 per day – 3 per second	1 – 10 per second
<i>session length</i>	few seconds	several minutes
<i>main metric</i>	msg delivery time	msg delivery time
<i>second metric</i>	reliable delivery	reliable delivery

A. Problem Statement

One of the most prominent issues in vehicular networks regards the quick, epidemic, and scalable delivery of data among all participants that share the same application. As IVC is wireless and hence shared in nature, it would be highly inefficient to utilize a scheme that, for every game event, generated as many unicast, multi-hop transmissions as the number of players. Rather, the fastest and less resource-consuming way to perform this operation is represented by multi-hop broadcast of messages over the car platoon.

Yet, if no intelligence is applied to the multi-hop broadcasting scheme, any node in the network would simply relay every received message. The consequent explosion in terms of transmitted messages would lead to high congestion, collisions, delays, and even to transmission paralysis of the vehicular network. In some sense, this phenomenon can be seen as a particular instance of what is generally known in MANET's literature as the *broadcast storm problem* [16].

Therefore, typical IVC applications require the presence of a solution able to efficiently propagate messages over a car platoon. Such solution has to guarantee a limited utilization of network resources and a quick message delivery, whatever kind of application were in use.

As a final remark, we point out that for many applications message delivery reliability can be important too. Unfortunately, the need for conciseness imposes us to limit the scope of this paper to a delimited problem and its solution. We have hence preferred to focus mostly on the speed of message propagation rather than on its reliability. Yet some preliminary evaluation of the reliability issue has been reported in Section VI-B and Section VI-E. To further explore this issue, a number of feasible solutions could be integrated in future with our system, e.g., [3], [11], [16], [17].

III. IVC MULTI-HOP BROADCAST: BACKGROUND

It is widely accepted in literature that a fast IVC multi-hop broadcast passes through having as few redundant transmissions as possible [7], [9], [10]. Indeed, a slow broadcast delivery can be due to a non-

optimal number of hops experienced by a message to cover the whole vehicular network and, more in general, by an excessive number of simultaneous forwarders. Now, we review prominent approaches proposed in literature which address this issue

Transmission rate adaptation. To reduce the delivery delay due to an excessive number of transmissions, several works (e.g., [11], [18]) propose a backoff mechanism that reduces the frequency of message transmissions or retransmissions when congestion is the cause for propagation delays. Moreover, message forwarding can be completely stopped by a car as soon a following vehicle starts to forward the message too [7]. Unfortunately, all these schemes neglect to consider a crucial factor: the number of hops a broadcasted message traverses before reaching its area-of-interest.

Topology awareness-based optimization. Minimizing the number of hops a message has to traverse to reach its car platoon may be obtained through building a Minimum Connected Dominating Set (MCDS) from a graph whose nodes represent vehicles in the platoon and an edge connects one node to another if the second is in the transmission range of the first one [19]. However, the implementation of this algorithm for IVC leads to great practical difficulties as it would require a complete and continuously updated knowledge of the network topology, obtainable with at least $O(n \log n)$ control messages, when considering n cars [20]. Moreover, the high mobility of vehicular networks may render obsolete the computed MCDS even before its completion.

Intelligent election of message forwarders. To minimize the number of hops that a message experiences during its propagation over the vehicular network, [9] and [21] assign different contention windows to each car receiving a message; their respective contention windows are inversely proportional to the distance from the previous sender. Each of these cars randomly selects a waiting time within its contention window before forwarding the message (if nobody else already did). Yet, these schemes are affected by the assumption that there is a unique, constant, and well-known transmission range for all cars in every moment. This is obviously not realistic [22].

IV. THE IVC ARCHITECTURE

From the analysis of related works, it is clear how an efficient multi-hop broadcasting mechanism passes through having the farthest vehicle in the sender's transmission range becoming the next forwarder. However, to implement this scheme, the following has to be true.

Assumption 1. *Each receiving vehicle is aware of its relative position within the sender's transmission range.*

Unfortunately, this assumption is not true at all with many current systems. To this aim, we designed an IVC architecture that satisfies this assumption and enables quick delivering of multi-hop broadcast messages among users in a car platoon that are sharing a given application (whatever the application and its traffic features are). We named this architecture *Privileged IVC Architecture (PIVCA)*.

As shown by Fig. 1, the main novelty of the proposed architecture is the inclusion of an IVC Session Layer between the Application Layer and lower layers. This new layer is responsible for optimizing the broadcast of each application message over the car platoon and is composed of three main components: the *Hello Message Generator*, the *Time Handler*, and the *Parameters Handler*, which, in turn, includes *PIVCA*'s core: the *Transmission Range Estimator*.

In the following, first, we present the global functioning of the IVC Session Layer, then, we explain the meaning and use of parameters employed by our solution and, finally, we discuss the algorithm that determines the message forwarder.

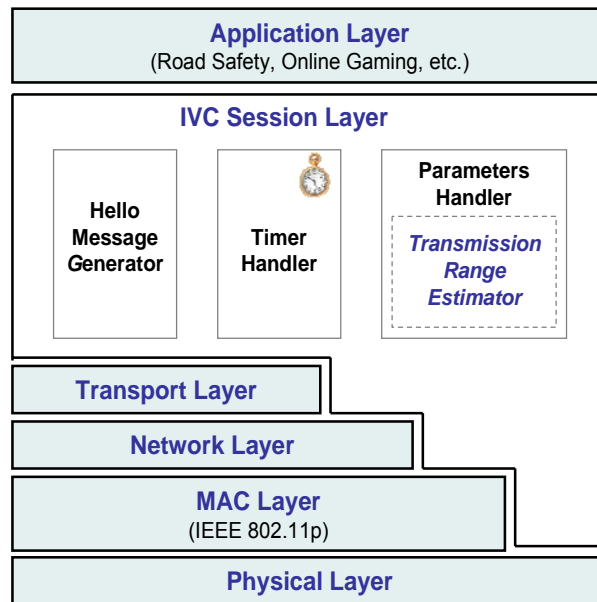


Fig. 1. The *Privileged IVC Architecture (PIVCA)*.

A. Global Functioning of the IVC Session Layer

Each message generated by the Application Layer of a node has to pass through the IVC Session Layer before being transmitted. Here, the **Parameters Handler** encapsulates the message in a new one that includes some parameters in its header. These are then extracted and utilized by the Parameters Handler of the receiving nodes to update their own parameters. Among these parameters, there is also an estimation of the transmission range as computed by the **Transmission Range Estimator**. The transmission range estimation is an original contribution of this work and represents the most important parameter sent along with the application message. It allows receiving vehicles to learn how far from the sender a given message will still be intelligible and which is their relative position within this distance. This way, Assumption 1 becomes true and vehicles can determine their own priority in becoming the next forwarder (i.e., the farthest vehicle in the sender's transmission range will be privileged in becoming the next forwarder).

To compute a correct transmission range estimation, the IVC Session Layer needs information, i.e., parameter values, from other vehicles with a certain regularity. This does not mean that messages (and

updated parameters) have to be transmitted/received very frequently as done by other schemes [19], [20]. Rather, empirical evidence shows that even hearing just one message every 100 ms is enough for *PIVCA* to provide a useful estimation of the transmission range. However, certain applications such as, for instance, alert message propagation for traffic safety generate messages only once in a while (minutes). These applications would benefit from a readily available transmission range estimation too as, depending on the network performance, more lives will be saved [8]. Therefore, if no message is exchanged in the network for a certain time, the **Hello Message Generator** takes care of generating a hello message just for the sake of broadcasting parameters and keeping the transmission range estimators of vehicles around running.

Clearly, measuring how much time has elapsed since last message transmitted or received requires a means to keep track of the time and, in case, trigger the generation of a hello message. This is the task of the **Time Handler**, which is also responsible for setting timers in order to implement the distributed mechanism for prioritizing farther vehicles in becoming the next forwarder of a given application message: the farther the vehicle, the smaller the contention window (and hence the time waited before forwarding the message).

For the sake of clarity, the basic functioning of our architectural solution can be summarized as follows.

- i) To achieve a fast multi-hop broadcast of an application message over a car platoon, *PIVCA* tries to reduce as much as possible the total amount of transmissions for its propagation; this is achieved by prioritizing farthest vehicles in the message's transmission range in becoming the next forwarder.
- ii) To implement this priority system, every sender includes in the sent message its own current transmission range estimation. Receiving vehicles can thus be aware of their position within the sender's transmission range and compute a waiting time which is inversely proportional to the distance from the sender. The vehicle with the smallest waiting time will be the next forwarder.
- iii) Each vehicle continuously computes and updates its transmission range estimation by exploiting data (e.g., position, hearing capability) coming from other vehicles around. To limit the amount of network

traffic, when possible these data are included in regular application messages, otherwise, few special control (hello) messages are specifically generated to transmit them.

iv) Timers are utilized to:

- determine when enough time has passed and it is hence opportune to transmit a hello message to feed the transmission range estimators of vehicles around;
- determine when waiting time is passed which was associated with the priority of a vehicle in becoming the next forwarder of an application message, and it is time for the considered vehicle to take charge of forwarding it (if no one else was faster).

B. Employed Parameters

More in detail, in any application or hello message generated by a vehicle the Parameters Handler inserts:

- sender's position, driving direction, and identifier;
- message's direction of propagation and identifier;
- sender's *backward maximum distance* (BMD) parameter;
- sender's *frontward maximum distance* (FMD) parameter;
- sender's *backward maximum range* (BMR) estimation;
- sender's *frontward maximum range* (FMR) estimation.

Parameters BMD and FMD represent the maximum distance from which another vehicle, backward or frontward respectively, has been heard by the considered one.

The Parameters Handlers of receiving vehicles exploit the received information about position, BMD, and FMD to compute their BMR and FMR values. BMR and FMR represent how far a transmission is expected to go before the signal becomes too weak to be intelligible and specifically:

- BMR is obtained by considering only messages coming from following vehicles; its value is computed as the maximum among i) the longest distance from vehicles that generated these

messages and ii) the highest among their FMD values included in these messages.

- FMR utilizes only messages sent by preceding vehicles; its value corresponds to the maximum among i) the longest distance from which a message has been received from a preceding vehicle and ii) the highest BMD advertised within these messages.

Computing correct values of BMR and FMR, and broadcasting them with any application message is one of the unique features of our solution and results of paramount importance as it allows other vehicles around to determine their relative position within the sender's transmission range.

Considering for simplicity only the case where cars are all driving in the same direction and application messages are only sent backward, then we have the following purposes for parameters included by *PIVCA* in each message.

Messages received from the front. These messages allow the receiver to compute FMD; its value will then be declared by the receiver in its future messages to claim: *“This FMD value is the farthest distance from which I have been able to hear another car in front of me”*.

Messages received from the back. As messages include the sender's FMD and position, messages received from the back provide the receiver with information about the hearing capabilities of following cars. This is exactly what the receiver needs to know in order to compute its BMR, which will then be sent along with its eventual future messages as it were saying: *“This BMR value is the maximum backward distance at which some car is able to hear me”*.

C. Forwarder Selection Algorithm

The rationale of our solution is to prioritize farther vehicles in forwarding received messages. Vehicles' priorities to forward a message are determined by assigning different waiting times from the reception of the message to the time at which they will try to forward it. This waiting time is computed based on a *contention window*, as inspired by classical backoff mechanisms in IEEE 802.11 MAC protocols.

At each hop, the self-elected forwarder updates BMR and FMR fields of the message with its computed

values so as to make available proper parameters for the next portion of road.

The contention window (CW) factually utilized by each vehicle is measured in slots and varied between a minimum ($CWMin$) and a maximum ($CWMax$), depending on the distance from the sending/forwarding vehicle ($Dist$) and on the advertised estimated transmission range. The transmission range corresponds to BMR if the message is directed backward, or to FMR if the message is directed forward. The two cases, indicated by CW_B and CW_F are shown in (1) and (2), respectively.

$$CW_B = \left\lceil \left(\frac{BMR - Dist}{BMR} \times (CWMax - CWMin) \right) + CWMin \right\rceil \quad (1)$$

$$CW_F = \left\lceil \left(\frac{FMR - Dist}{FMR} \times (CWMax - CWMin) \right) + CWMin \right\rceil \quad (2)$$

Upon receiving the message from the front, the considered car extracts BMR and utilizes (1) to determine its CW_B and then computes a random waiting time based on it. If, while waiting, the same message is heard again, coming from behind, then the message has already propagated over the considered car that can hence stop trying to forward it: somebody else already did it. Conversely, if the same message is heard from forward, then this means that a preceding car has already forwarded it; the application message forwarding procedure has hence to be restarted with the new parameters included in the message by the last forwarder.

If the waiting time expires without having heard any other car forwarding the same message then the considered car includes in the message its own current parameters, including BMR, and broadcasts it (obviously, the same, even if specular, procedure applies to the FMR parameter).

V. EXPERIMENTAL ASSESSMENT

We carried out an extensive experimental study to test our architectural solution and compare it with other possible schemes inspired by scientific literature. The main tool utilized for our experiments is the well known NS-2 simulator (version ns-2.29) [23]; we modified the MAC layer's parameters to make it behave similar to the IEEE 802.11p, and added software to compare *PVCA* with other solutions. The baseline simulative configuration is summarized in Table II and our NS-2 modules and scripts are available online [24].

TABLE II
BASELINE SIMULATIVE CONFIGURATION

Feature	Value
<i>Simulations run for each configuration</i>	40
<i>Wireless model</i>	Two-ray ground reflection
<i>Type of road</i>	Highway with multiple lanes
<i>Vehicles' speed</i>	70-140 Km/h
<i>CWMin - CWMax</i>	32 slots – 1024 slots
<i>Idle time before Hello Msg generation</i>	100 ms
<i>Application Msg size</i>	200 B
<i>Hello Msg size</i>	50 B

More in detail, the number of slots for $CWMin$ and $CWMax$ was inspired by the standard IEEE 802.11 protocol. The idle time threshold after which a hello message is generated by the Hello Message Generator is set equal to 100 ms. This means that about 10 hello messages with a payload of 50 B are generated every second, within a transmission range area, (only) if no application transmission happens.

We have compared our \mathcal{PIVCA} against other three possible solutions. Two of them are inspired by the solution presented in [9]. This scheme is similar to \mathcal{PIVCA} in that it attempts to have the farthest node in the transmission range as the next-hop forwarder. However, it differs from \mathcal{PIVCA} because it simply assumes to have the transmission range parameter constantly equal to a known predetermined value, rather than being able to dynamically compute it according to the factual channel condition. Compared to our \mathcal{PIVCA} shown in Fig. 1, it hence lacks the Parameters Handler (with the Transmission Range Estimator). We name this architectural solution *Fixed300* if it utilizes 300 m as the transmission range parameter, and *Fixed1000* if it employs 1000 m.

Needless to say, *Fixed300* and *Fixed1000* perform ideally when the factual transmission range is 300 m and 1000 m, respectively. In any other situation, the utilization of a wrong parameter can result in performance degradation. The choice of focusing on 300 m and 1000 m of transmission range comes straightforward from the IEEE 802.11p draft, which indicates these two values as the boundaries for a highway scenario [1].

We also evaluated *Random*, a solution that does not employ any distance prioritization, to represent an architecture employing a classic multi-hop broadcasting scheme. Simply stated, every car computes a random waiting time within the contention window before forwarding the message. The adopted contention window is initially set to $CWMin$ and follows a general backoff mechanism by which its value doubles every time a transmission attempt results in a collision and decreases linearly with every successful transmission.

With the accident alert communication application, we let the simulation run for 2 minutes, after which

the first vehicle in the car platoon generated an alert message. Instead, with online game application, the number of players periodically generating game events varied from 2 to 50 [13], [25].

VI. *P*IVCA'S PERFORMANCE

To compare the various schemes, we analyze their ability in quickly delivering messages to all interested vehicles: accident alert messages in Section A (with crash performance evaluation in Section B) and online game events in Section C. The same experiments are utilized also in Section D to check how our system scales with increased transmission ranges. For completeness, we dedicate Section E to investigate reliability: the percentage of messages that are correctly received by all participants. Finally, in Section F, we analyze the influence of the time slot duration on *P*IVCA's performance.

To measure the performance, we consider only messages belonging to the worst case: those sent by the car leading the car platoon to cover the whole vehicular network.

A. *Accident Alert Communication: Delivery Time with Short Transmission Range*

We start our evaluation focusing on the accident alert communication application. As mentioned, this application generates a message only in case of abnormal behavior of some vehicle. Its low message transmission rate makes hence necessary to transmit (few) hello messages to get information about vehicles' transmission ranges. Obviously, the overhead due to hello messages employed by our architecture is very limited, i.e., less than 1 Kb/s within a transmission range area.

Next charts present outcomes related to the transmission of an alert message that has to be propagated from a certain vehicle to all following vehicles in a 4 Km range. We consider 4 Km as there is no point in transmitting an instantaneous alarm farther: even a car driving at 120 Km/h would employ 2 minutes to reach the 4 Km-away critical area. However, nothing impedes to set a larger (or smaller) area-of-interest if deemed more appropriate. The number of vehicles per Km of (multi-lane) road varies from 50 to 250

representing different density levels; the factual range is 300 m.

The number of hops an alert message requires in average to be propagated over the whole 4 Km of area-of-interest is reported in Fig. 2. *PIVCA* requires in average as few hops to propagate the message as the ideal scheme for this scenario (i.e., Fixed300). Instead, Fixed1000 and Random require many more hops as the forwarders' prioritization mechanism of the former fails due to a wrong value of the transmission range, while the latter does not use any prioritization at all.

Another interesting outcome of Fig. 2 is represented by the fact that the number of hops for each scheme does not vary when augmenting the car density. This happens thanks to the fact that the election of the next forwarder for the compared schemes is either based on vehicles' positions within the sender's transmission range or chosen randomly, which are both two properties that do not depend on vehicle density.

However, with a higher number of vehicles it is more likely that two or more vehicles simultaneously try to forward the alert message, thus resulting in collisions, retransmissions, and waste of time. In point of this fact, Fig. 3 shows the total delay required to propagate an alert message over the whole car platoon: all compared schemes are negatively affected by very high vehicle densities; yet the ideal scheme (Fixed300) and our *PIVCA* outperform the other two.

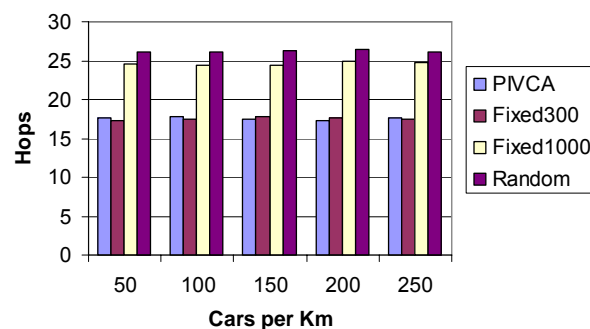


Fig. 2. Alert message; avg number of hops to cover a 4 Km long vehicular network; 200 μ s slot, 300 m of transmission range.

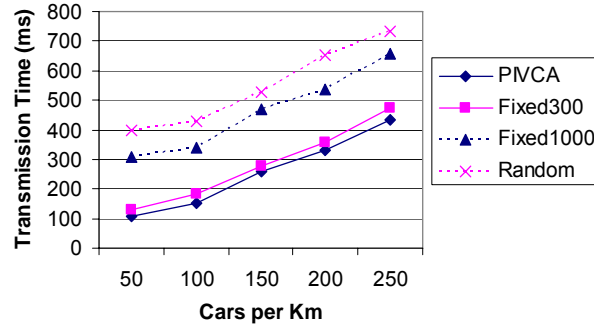


Fig. 3. Alert message; avg transmission time to cover a 4 Km long vehicular network; 200 μ s slot, 300 m of transmission range.

Moreover, even if Fig. 2 shows that \mathcal{PIVCA} and Fixed300 have similar outcomes, Fig. 3 reports that the former is always slightly better than the latter when considering the final transmission time. This is due to the fact that Fig. 3 includes also other factors such as the number of time slots within the employed contention window that the forwarder waits before transmitting the message. Indeed, even if we have set the transmission range to be 300 m, yet, the wireless model realistically generates interferences as it would happen in real life; hence, the factual transmission range oscillates around 300 m. Conversely from Fixed300, \mathcal{PIVCA} dynamically adapts to the changing transmission range, employing (slightly) more appropriate contention windows that reduce the final transmission time.

B. Accident Alert Communication: Crash Performance

For the sake of completeness, we have also performed a crash performance evaluation similar to what done in [7] and [17]. In essence, we have mathematically modeled a car platoon of 30 cars traveling at 32 m/s (115 Km/h), with an inter-vehicular distance of 28.8 m (the distance traveled in 0.9 s with a 32 m/s speed). At a certain point, the vehicle in front of the platoon has an accident by which its speed is quickly reduced with a deceleration of 8 m/s². By braking, following cars decelerate at 4 m/s². If a car hits another one in front of it, then both their speeds decelerate at 8 m/s² until stopping. The reaction time of drivers is assumed to be uniformly distributed in the range between 0.75 s and 1.5 s. Drivers are assumed to be able

to see two cars in front of them; they are also assumed to react by braking, after the reaction time, if their car receives an alert message or if one of the two vehicles in front of them brakes. The position of each car is checked every 0.125 ms and an accident is assumed to happen if the distance between two vehicles becomes smaller than 4 m (the car's length).

Four scenarios are compared: i) no alert messages are implemented, only reaction times; ii) alert messages are implemented through our system and all vehicles are able to receive them; iii) alert messages are implemented and delivered through our system but due to bad connectivity or low market penetration of the technology only one vehicle out of three (randomly chosen and uniformly distributed over the car platoon) is able to receive the message; iv) alert messages are implemented through our system but their delivery delay is 2.5 times what suggested by NS-2.

The performances achieved in the four scenarios are reported in Fig. 4, Fig. 5, Fig. 6, and Fig. 7, respectively. The charts show the relative speed between two consecutive cars when they stop or hit each other: an accident occurred when the value is higher than zero and to higher values correspond worst accidents.

As it is evident, our system effectively reduces the number of vehicles involved in a chain accident. The charts also show a tradeoff between the vehicular network coverage achieved by the alert communication and its effectiveness; for instance, even if vehicle 5 is able to brake in time, it still results involved in the accident as vehicle 6 (that did not receive any alert message) crashes into it. To this aim, solutions able to ensure a wider coverage (e.g., transmission power management [17]) may be explored in combination with our system. Still, even with just one vehicle out of three receiving the alert message, the performance results significantly better than with no alert message at all. This outcome is not surprising: even few randomly distributed cars receiving the alert message and braking in advance help following cars in anticipating their braking time (they brake as the vehicle in front of them, that received a timely alert message, is braking).

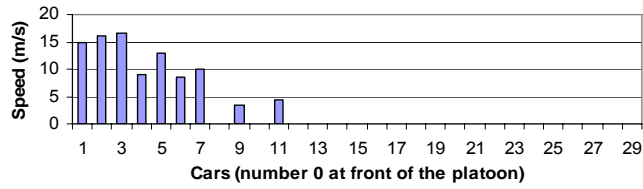


Fig. 4. Speed difference between two consecutive cars after stop; regular system with no accident alert communication.

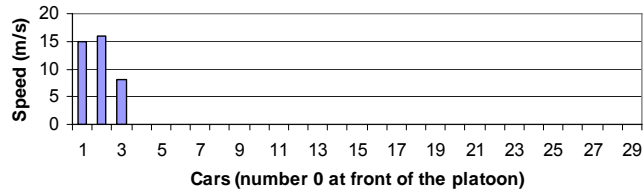


Fig. 5. Speed difference between two consecutive cars after stop; all vehicles receive the accident alert communication.

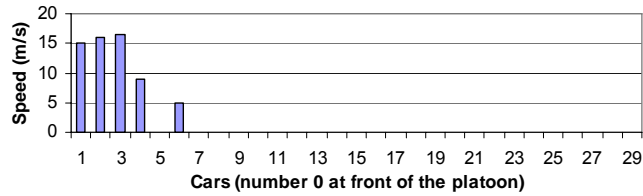


Fig. 6. Speed difference between two consecutive cars after stop; only one vehicle out of three receives the accident alert communication.

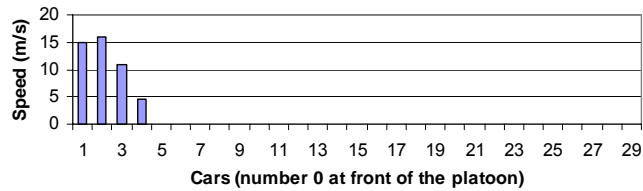


Fig. 7. Speed difference between two consecutive cars after stop; all vehicles receive the accident alert communication; scenario considering an augmented propagation time for the alert message (2.5 times the NS-2's one).

Finally, since simulations are generally an imperfect model of the real world [23], [26], we have also tested our solution in the crash performance test considering the pessimistic case where the factual transmission time for an alert message were 2.5 times what NS-2 simulations suggested. Results are reported in Fig. 7; the comparison with Fig. 5 shows that in this case there is one more car involved in the accident, even if at a low speed, and that vehicle 3 hits the preceding car at a slightly higher speed. This demonstrates that even at these conditions our solution would be able to save lives if compared to the case with no accident alert communication.

C. Online Games: Delivery Time with Short Transmission Range

We consider now the online gaming application and compare \mathcal{PIVCA} Fixed300, Fixed1000, and Random on a 8 Km long vehicular network, 200 μ s of slot duration, 50 simultaneous players, and 300 m of transmission range. We have investigated performances when considering different generation rates for game events at each player. Specifically, game events were generated at each vehicle every 100 ms, 300 ms, or 500 ms.

Outcomes are presented in Fig. 8 and Fig. 9. The first property that emerges is that \mathcal{PIVCA} always obtains better results than the other three schemes, even better than Fixed300 that is supposed to be the ideal scheme in a scenario with 300 m of transmission range. This result is due to the ability of \mathcal{PIVCA} in adapting to the slight variations of the transmission range generated by the wireless model, as already discussed for Fig. 3.

The second evident property is represented by the fact that all four schemes seem almost independent from the employed message sending rate. This indicates that the considered experiment configuration did not saturate the network; only with the highest message sending rate (100 ms of inter-departure time) we can observe a little degradation of the performance, due to the increased congestion in the network.

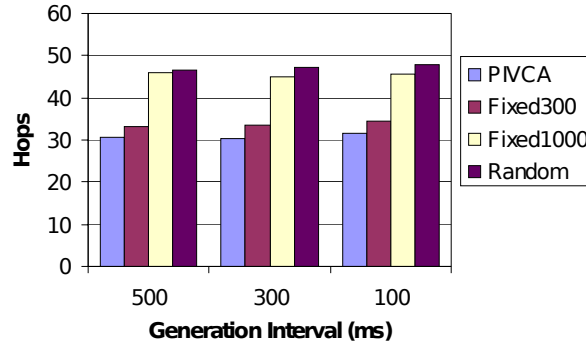


Fig. 8. Game event; avg number of hops to cover a 8 Km long vehicular network; 200 μ s slot, 50 players, 300 m of transmission range.

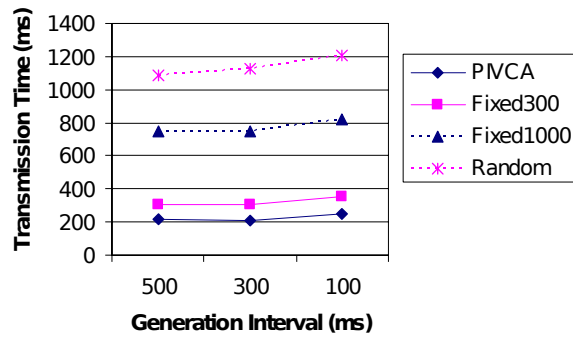


Fig. 9. Game event; avg transmission time to cover a 8 Km long vehicular network; 200 μ s slot, 50 players, 300 m of transmission range.

D. Delivery Time with Long Transmission Range

To complete our evaluation of *PIVCA*'s performance, we have considered a scenario with 1000 m of transmission range. Compared to the 300 m case, we expect to witness a higher transmission interference as a single transmission range area hosts more communicating vehicles. This is especially true for the online game application since its high message generation rate and multitude of sources. Therefore, we present here outcomes only for this application, which is more stressful for the system than accident alert communication.

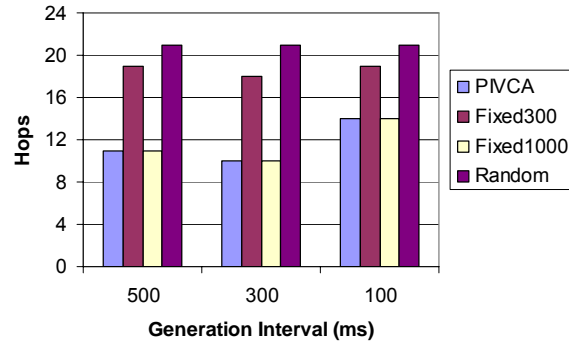


Fig. 10. Game event; avg number of hops to cover a 8 Km long vehicular network; 200 μ s slot, 50 players, 1000 m of transmission range.

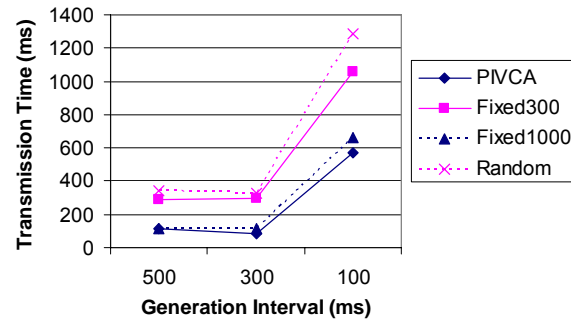


Fig. 11. Game event; avg transmission time to cover a 8 Km long vehicular network; 200 μ s slot, 50 players, 1000 m of transmission range.

The number of hops required to cover the car platoon in this scenario is shown in Fig. 10; as expected, Fixed1000 outperforms Fixed300. Without any predetermined knowledge *PIVCA* succeeds in properly estimating the transmission range for each vehicle and performs as Fixed1000.

Analyzing the final transmission time required to propagate a message over the whole car platoon, we notice that if the game application generates a message on each vehicle every 100 ms the total transmission time is considerably higher than the other two considered cases (see Fig. 11). This is due to an excessive increment of the traffic on the wireless channel: many players are in range of each other when considering

a transmission range of 1000 m. This causes collisions and time consuming retransmissions as confirmed also by results shown in the next subsection. However, even in this case *PIVCA* and the ideal scheme (here, Fixed1000) outperforms the other two.

E. Delivery Reliability

While it is not our aim to propose solutions for reliability, we completed our study with an evaluation of *PIVCA*'s ability in delivering messages to all vehicles engaged in the same application. We present outcomes for the game application as its intense network traffic represents a tougher challenge for reliable transmissions than propagating a single alert message.

Outcomes for the case with 300 m and 1000 m of factual transmission range are reported in Fig. 12 and Fig. 13, respectively. All the various schemes present similar performances. In particular, high reliability is guaranteed until considering the case where each player generates a game event every 100 ms. In this case, the percentage of messages received by all vehicles drops to about 90 % with a transmission range of 300 m and to less than 65 % with a transmission range of 1000 m.

The second case is clearly more critical as in a 1000 m transmission range area there are roughly three times the number of players (and transmitted messages) present in a 300 m one. As partially anticipated at the end of the previous subsection, the combination of a long transmission range with an intense message generation rate is cause of collisions, retransmissions, delays, and even lack of delivery reliability.

In case of intense congestion there is not much that can be done but reducing the group of simultaneous players. Instead, with mild congestion, significant improvements can be achieved also by resorting to aggregation of messages, or to exploiting game semantics to determine which game events supersede others and transmit only the former ones [25].

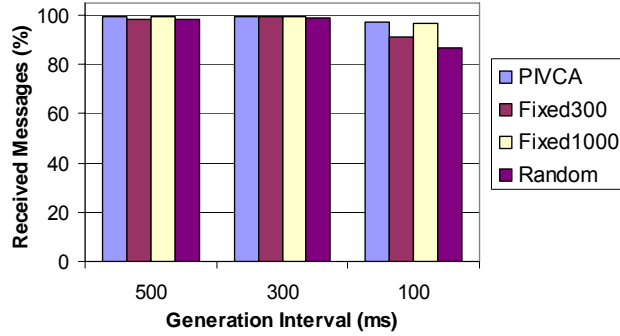


Fig. 12. Percentage of game events successfully received by all 50 players; 8 Km long vehicular network, 200 μ s slot, 300 m of transmission range.

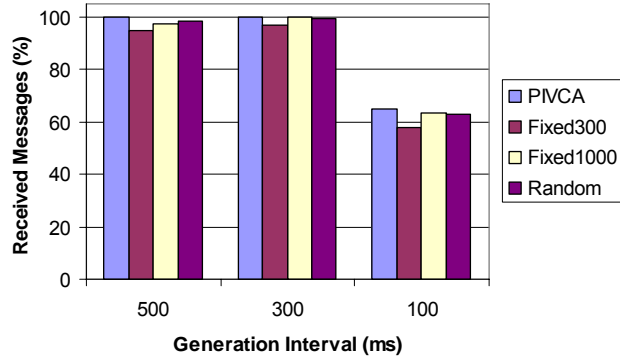


Fig. 13. Percentage of game events successfully received by all 50 players; 8 Km long vehicular network, 200 μ s slot, 1000 m of transmission range.

F. Influence of the Time Slot Duration on PIVCA

when employing different time slot sizes for the contention window. We considered a scenario with 300 m of factual transmission range and 50 players. Outcomes shown in Fig. 14 represent the instantaneous variation of the delivery time of each single game event to the farthest player in the car platoon. Clearly, an increase of the total delivery time corresponds to a longer time slot duration. When considering fast paced games (or accident alert communications), only small delivery delays are acceptable. Therefore, time slots larger than 200 μ s would not be a feasible configuration.

However, too small time slot values could result in elevated collisions, interference, and lack of reliability in delivering each message to all other vehicles. In point of this fact, Fig. 15 shows the percentage of game events that were successfully delivered to all players. As it is evident, when employing time slots of 9 μ s, the increased delivery speed seen in Fig. 14 is paid with a noticeable reduction of the reliability (see values for one message generated every 100 ms from every player).

In summary, 200 μ s is the only value among the tested ones that provided both fast and reliable message delivery.

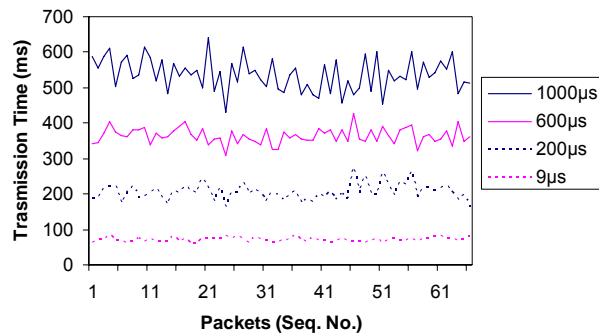


Fig. 14. Transmission time trend of game events over a 8 Km long vehicular network; PIVCA, 50 players, 300 ms of message generation interval for each player, 300 m of transmission range.

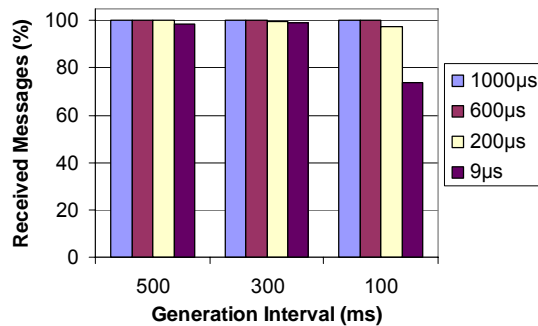


Fig. 15. Percentage of game events successfully received by all 50 players; PIVCA, 8 Km long vehicular network, 300 m of transmission range.

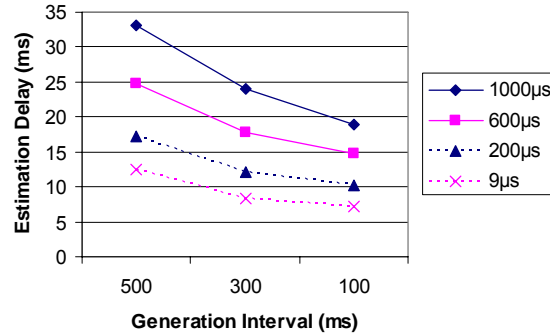


Fig. 16. Avg time required by a vehicle to estimate the transmission range with less than 5 % of error; online game session, 50 simultaneous players, 8 Km long vehicular network, 300 m of transmission range.

As expected, with higher message generation rates, *PIVCA* needs less time to compute the correct estimation. This is a logic consequence of the fact that the estimation is based on data about vehicles' positions and "hearing" distances which are exchanged through application/hello messages. Therefore, the more the messages, the more the information received and the quicker the correct estimation can be built. However, all the evaluated configurations show little delays, 20 ms at most; this proves that our architectural solution is able to adapt extremely rapidly, which is a crucial feature in a highly dynamic scenario such as a vehicular network.

We evaluate in Fig. 16 the speed of *PIVCA*'s Transmission Range Estimator in dynamically computing the transmission range of each vehicle with at most 5 % of overestimation or underestimation of its real value.

VII. CONCLUSION AND FUTURE WORK

IVC represents the forthcoming frontier in mobile communication. A particularly interesting case study is represented by distributed interactive applications run by vehicles spread in a few kilometers of range. This class of applications is featured with the requirement of fast message delivery, through multi-hop

broadcast, over the vehicular network. Nonetheless, other features such as the message generation rate may sensibly vary when considering different specific applications.

To support a broad range of IVC applications, we have designed a software architecture, named *JIVCA*, that permits fast multi-hop propagation of messages to all engaged vehicles. This result is achieved via smart solutions aimed at reducing the number of transmissions required to propagate the message. Our architecture is able to adapt its functionalities to the network traffic generated by different applications. Advantages in employing *JIVCA* have been demonstrated through an extensive set of experiments.

This work can be extended in several directions. First, we have just finalized a prototype of our solution implemented on PDAs and are now starting a real testbed evaluation campaign. At the same time, we are also enhancing the utilized algorithms to be proficient even in a grid based scenario (i.e., streets of a town). We also plan to combine *JIVCA* with some of the solutions mentioned in Section II-A which are aimed at increasing the reliability of IVC. Finally, to ensure the widest possible compatibility, *JIVCA* has been designed to operate at higher network layers; however, studies show that there is an optimal choice of transmission range for the latency when considering 802.11 based MAC protocols [17], [27]. It would be interesting to combine the two approaches to improve even further the performance of the system.

ACKNOWLEDGMENT

We are sincerely grateful toward the anonymous referees, whose insightful suggestions significantly helped us in improving the final version of this manuscript.

REFERENCES

- [1] *Dedicated Short Range Communications (DSRC) Home*. [Online]. Available: <http://www.learmstrong.com/dsrc/dsrchomeset.htm>

- [2] M. Guo, M. H. Ammar, E. W. Zegura, "V3: a vehicle-to-vehicle live video streaming architecture", in *Proc. of 3rd IEEE International Conference on Pervasive Computing and Communications*, Kauai, HI, Mar. 2005.
- [3] N. Balan, J. Guo, "Increasing broadcast reliability in vehicular ad hoc networks," in *Proc. of the 3rd ACM International Workshop on Vehicular Ad Hoc Networks (VANET 2006), ACM MobiCom 2006*, Los Angeles, CA, Sep. 2006.
- [4] C. E. Palazzi, M. Roccetti, S. Ferretti, G. Pau, M. Gerla, "Online games on wheels: fast game event delivery in vehicular ad-hoc networks", in *Proc. of 3rd IEEE International Workshop on Vehicle-to-Vehicle Communications 2007 (V2VCOM 2007) - IEEE IVS 2007, Istanbul, Turkey*, Jun. 2007.
- [5] M. Aoki, H. Fujii, "Inter-vehicle communication: technical issues on vehicle control application", *IEEE Communications Magazine*, vol. 34, no. 10, pp: 90-93, Oct. 1996
- [6] S. Tsugawa, "Inter-vehicle communications and their applications to intelligent vehicles: an overview", in *Proc. of IEEE Intelligent Vehicle Symposium 2002*. Versailles, France, Jun. 2002 .
- [7] S. Biswas, R. Tatchikou, F. Dion, "Vehicle-to-vehicle wireless communication protocols for enhancing highway traffic safety", *IEEE Communication Magazine*, vol 44, no. 1, pp: 74-82, Jan .2006.
- [8] L. Pantel, L. C. Wolf, "On the impact of delay on real-time multiplayer games", in *Proc of NOSSDAV'02*, Miami, FL, May 2002.
- [9] E. Fasolo, R. Furiato, A. Zanella, "Smart broadcast algorithm for inter-vehicular communication", in *Proc. of Wireless Personal Multimedia Communication (WPMC'05)*, Aalborg, DK, Sep. 2005.
- [10]G. Korkmax, E. Ekici, F. Ozguner, U. Ozguner, "Urban multi-hop broadcast protocol for inter-vehicle communication Systems", in *Proc. of 1st ACM Workshop on Vehicular Ad-hoc Networks (VANET'04)*, Philadelphia, PA, Oct. 2004.
- [11]X. Yang, J. Liu, F. Zhao, N. Vaidya, "A vehicle-to-vehicle communication protocol for cooperative collision warning", in *Proc. of 1st Annual International Conf. on Mobile and Ubiquitous Systems:*

Networking and Services (MobiQuitous'04), Boston, MA, Aug. 2004.

[12] *Consumer Electronics - IT Facts* [Online]. Available: <http://www.itfacts.biz/index.php?id=P1861>

[13] J. Färber, “Network game traffic modelling” in *Proc. of NetGames 2002*, Braunschweig, Germany, Apr. 2002.

[14] *MMOGCHART* [Online]. Available: <http://www.mmogchart.com/>

[15] M. B. Yassein, O. Khaoua, S. Papanastasiou, “Performance evaluation of Flooding in MANETs in the Presence of Multi-Broadcast Traffic”, in *Proc. of 11th International Conference on Parallel and Distributed Systems - Workshops (ICPADS)*, Fukuoka, Japan, Jul. 2005.

[16] Q. Xu, T. Mak, J. Ko, R. Sengupta, “Medium access control protocol design for vehicle–vehicle safety messages”, *IEEE Transactions on Vehicular Technology*, vol. 56, no. 2, pp. 499-518, Mar. 2007.

[17] F. Ye, M. Adams, S. Roy, “V2V wireless communication protocol for rear-end collision avoidance on highways”, in *Proc. of Vehi-Mobi Workshop*, IEEE ICC, Beijing, China, May 2008.

[18] L. Wischhof, A. Ebner, H. Rohling, “Information dissemination in self-organizing intervehicle networks”, *IEEE Transactions on Intelligent Transportation Systems*, vol. 6, no. 1, pp. 90-101, Mar. 2005.

[19] A. Zanella, G. Pierobon, S. Merlin, “On the limiting performance of broadcast algorithms over unidimensional ad-hoc radio networks”, in *Proc. of WPMC04*, Abano Terme, Italy, Sep. 2004.

[20] P. J. Wan, K. Alzoubi, O. Frieder, “Distributed construction of connected dominating set in wireless ad hoc networks”, in *Proc. of IEEE INFOCOM 2002*, Jun. 2002.

[21] J. J. Blum, A. Eskandarian, “A reliable link-layer protocol for robust and scalable intervehicle communications”, *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 1, pp. 4-13, Mar. 2007.

[22] C. E. Palazzi, S. Ferretti, M. Roccetti, G. Pau, M. Gerla, “How do you quickly choreograph inter-vehicular communications? A fast vehicle-to-vehicle multi-hop broadcast algorithm, explained”, in

Proc. of IEEE CCNC/NIME 2007, Las Vegas, NV, Jan. 2007.

- [23] Q. Chen, D. Jiang, V. Taliwal, L. Delgrossi, “IEEE 802.11 based vehicular communication simulation design for NS-2”, in *Proc. of the 3rd ACM International Workshop on Vehicular Ad Hoc Networks (VANET 2006), ACM MobiCom 2006*, Los Angeles, CA, Sep. 2006.
- [24] *Fast Broadcast – NS-2* [Online]. Available: <http://www.math.unipd.it/~cpalazzi/fastbroadcast.html>
- [25] C. E. Palazzi, S. Ferretti, S. Cacciaguerra, M. Rocchetti, “Interactivity-loss avoidance in event delivery synchronization for mirrored game architectures”, *IEEE Transactions on Multimedia, IEEE Signal Processing Society*, vol. 8, no. 4, pp. 874-879, Aug. 2006.
- [26] Q. Chen, F. Schmidt-Eisenlohr, D. Jiang, M. Torrent-Moreno, L. Delgrossi, H. Hartenstein, “Overhaul of IEEE 802.11 modeling and simulation in ns-2”, in *Proc. of the 10th ACM Symposium on Modeling, analysis, and simulation of wireless and mobile systems (MSWiM '07)*, Chania, Crete Island, Oct. 2007.
- [27] S. Roy, C-L Huang, “Performance analysis of IEEE 802.11 MAC based collision avoidance in vehicular platoon”, in *Proc. of the 1st International Workshop on Vehicle to Vehicle Communication (V2VCOM 2005), MobiQuitous 2005*, San Diego, CA, Jul 2005.