Communities on the Road: Fast Triggering of Interactive Multimedia Services

Claudio E. Palazzi

Dipartimento di Matematica Pura e Applicata, Università degli Studi di Padova, Via Trieste, 63 - 35131 Padova, ITALY

cpalazzi@math.unipd.it

Stefano Ferretti

Dipartimento di Scienze dell'Informazione, Università di Bologna, Via Mura Anteo Zamboni, 7 - 40127 Bologna, ITALY

sferrett@cs.unibo.it

Marco Roccetti

Dipartimento di Scienze dell'Informazione, Università di Bologna, Via Mura Anteo Zamboni, 7 - 40127 Bologna, ITALY

roccetti@cs.unibo.it

Abstract Social network communities are involving millions of users, representing one of the main reasons why people log on the Internet from their home computers. Part of this success is certainly due to the possibility for end users to reverse the traditional publisher/consumer model, achieving control over service consumption, and gaining the opportunity to produce multimedia contents instantaneously available worldwide. Social network communities are not destined to be confined in traditional wired networks. Indeed, mobile users could greatly benefit from applications that combine social networks and location-based multimedia services. It is hence of particular interest to consider the next frontier in wireless networking, i.e., vehicular networks, and imagine how community-based services could be provided in this highly variable context, enabling the sprouting of communities on the road. To this aim, we address here one of the specific challenges in this scenario, i.e., the fast delivery of service triggering messages generated by a user to a certain area where another user can provide the requested multimedia service (e.g., live video streaming, traffic updates, friends finder, status messages of social network applications). We discuss the state-of-art for this technical challenge and compare it against our solution, which is able to dynamically adapt to different transmission conditions as those featuring a vehicular network. In essence, the main innovation of our contribution amounts to a transmission range estimator that enables vehicles to know their current transmission range, independently from

changes in the vehicular network topology, and use it to maximize the efficiency in transmitting service-triggering messages.

Keywords: Live Video Triggering, Social Network Communities, V2V Communication, V2V Multimedia Applications, Vehicular Communities.

1. Introduction

Social network communities have revolutionized the way we meet, communicate, work, and present ourselves. Several motivations are at the basis of this global success and the main one is the possibility offered to users to revert the traditional publisher/consumer model, enabling the control over the timing, channel, and format by which multimedia contents are consumed, or even created [1]. Indeed, we can identify two important properties in the services offered by social network communities to their participants: first, the instantaneous interaction with like-minded individuals over a world scale; second, the enrichment of people's communication capabilities via self-generated profiles and multimedia contents [2].

These two properties, interactivity and multimediality, feature the genre of social network service that results both more appealing to end-users and more challenging under a research point of view. This is especially true if we consider the combination among social network services and mobile users. Indeed, mobile users are willing to engage both traditional online communities and new ones that will be fostered by location-based services, specifically leveraging on the ubiquity of mobile devices [3]-[6].

The whole scenario will include the possibility for users on the road to discover, retrieve, request, and manage location-based services and multimedia contents in an interactive way. However, this requires a fast service-triggering mechanism that is both crucial and challenging, especially when considering highly mobile networks such as vehicular ones [7], [8]. The term “triggering message” is employed here with the broad meaning of a message containing some new event related to the multimedia application being supported. Such triggering messages may serve to control the streaming of a multimedia flow (e.g., start, stop, pause, change codec), but they may also embody instant messages or status messages typical of social applications (think at status messages in Facebook, for example). In this scenario, to effectively provide interactive multimedia services to community members, it is important to guarantee that triggering messages are responsively delivered through the vehicular network (VANET). Indeed, these messages are responsible for the prompt management of all community-based services requested by participants.

The transmission of such triggering messages may not represent a tough issue in traditional networks, as it is possible to simply rely on the wired Internet to transmit messages and on some servers to provide the requested service (e.g., collecting and distributing multimedia contents). A triggering message has typically a small size and does not necessarily embody itself a multimedia content. Hence, in a classic wired-network scenario, its transmission may be quickly accomplished without introducing network congestions. Instead, the situation radically changes when trying to provide interactive multimedia services to social network communities on the road [9]. In this case, we have to deal with the fact that vehicles are highly mobile and that they could be only seldom covered by access points along the road, thus needing to rely on ad-hoc multi-hop networking to still be engaged in social community networking. Moreover, vehicular networks are highly variable: i) the numbers of communicating nodes may greatly vary in a very short period of time and ii) vehicles quickly pass by very different surroundings thus experiencing wide variations in terms of transmission range and available bandwidth. Since messages are transmitted in an ad-hoc manner in the wireless network, even if small-sized, triggering messages may create collisions and transmission errors that may cause a severe degradation of communication performances. Moreover, when such messages are employed to control some multimedia flow (e.g., start, stop, pause), their delivery also influences the transmission of multimedia contents, which may further affect the overall multimedia distribution service, if not adequately controlled. Despite the adverse networking conditions featuring vehicular networks with respect to traditional wired one, the stringent requirement for a responsive management of multimedia services available to the community remains unaltered.

To this aim, previous scientific works have demonstrated that a fast delivery of a triggering message in an ad-hoc connectivity context as a vehicular network can be achieved if the message is propagated over the network with long hops; in essence: having the farthest receiving node in the sender/forwarder’s transmission range becoming the next forwarder of that message [10], [11]. Yet, all these works overlooked at the high variability of a vehicular network and only considered simulative cases where the transmission range is constant and known a priori by all nodes in the network.

This is a clear oversimplification and motivates us in writing this article. In fact, we discuss here how an efficient transmission range estimator can actually be built to work in vehicular networks to support efficient forwarding of triggering messages for the handling of social and multimedia applications. The main innovation of our transmission range estimator is that it provides a fundamental parameter for efficient forwarding of a message without the need for complete topology knowledge; this permits to efficiently operate even with a continuously changing topology (vehicles surpassing each other or changing direction), without the need to resort to intensive transmission of control messages (i.e., *presence messages*) to update all vehicles’ positions.

As a result, members of communities on the road become able to quickly send triggering messages for their favorite community-based multimedia services; in other words, even while driving, they will be able to discover, retrieve, request, and manage location-based services and multimedia contents in an interactive way.

The rest of the paper is organized as follows. In Section 2, we further clarify the considered scenario and the involved research issues by presenting a practical case study of service for communities on the road. Then, in Section 3, we survey the state-of-art of solutions aimed at minimizing the number of hops/transmissions to propagate a message over a vehicular network and point out their shortcomings. Being the transmission range of vehicles a critical information for improving the performance of message forwarding mechanisms, we devote Section 4 to the discussion of an effective distributed algorithm that allows each vehicle to estimate its own transmission range. Then, Section 5 and Section 6 report on experimental results that demonstrate the performance improvement achieved through the proposed algorithm. This way, we show how a challenging problem in providing control for interactive multimedia services to vehicular communities can actually be solved. Finally, we conclude this paper in Section 7.

2. A Representative Example: Triggering A Live Video Streaming Among Vehicles

We consider a group of vehicles endowed with communication capabilities that form a community able to generate and share multimedia documents in real-time. The real-time feature in this context is particularly interesting both for the involved research challenges and for the appealing services that could be deployed and that are specifically beneficial for a community on the road [12].

Just to make a timely example, think of *approximeeting*, which refers to the creation of arrangements among people or parties that are initially loose and progressively refined through a series of forewarnings about time and location, while participants actually get closer to the time and location of the final meeting [13]. This concept can be easily generalized; instead of determining the final time and location for a meeting, people may be interested in getting any sort of service available thanks to the proximity of a provider, that could also be another user who is generating a specific multimedia content in real-time. Several examples can be imagined; just to mention a few of them: i) a driver could be interested in knowing the traffic situation a few miles ahead to decide the best route, thus sending a request for traffic updates to other drivers [10], [11]; ii) a driver could be interested in receiving information on free parking places [25]; iii) first aid vehicles could watch the scene of an accident even before arriving on site by demanding a live video-stream to people with camera-endowed cellphones passing by the accident area [15]; iv) tourists could take advantage of pictures/videos generated by other tourists to decide whether to stop in a certain place; v) a person may receive live updates about friends or people with same interests located in the same area where she/he is traveling. In this context, a crucial technical problem is how to trigger in a fast way the chosen service among those that can be provided by other community users [14], [15]. Indeed, such a delivery scheme over VANETs can be thought as an important substrate to ensure viable multimedia / social car-to-car services.

For the sake of clearness and without loss of generality, let us consider a specific instance of a general community-based service: live video streaming generated by a camera on the curb or on a vehicle and triggered by a user driving another vehicle. Simply, a vehicle’s driver or passenger wants to receive a live video stream from a certain location, which represents the *area-of-interest*; thereby, a video triggering message has to be transmitted from the requesting car to a static or mobile camera in the area-of-interest, even through multi-hopping, in the least possible time. This request will be then picked up by a member of the community that is located in (or is traveling through) the area-of-interest and that is endowed with a camera.

We do not forget that in the live video streaming context classic problems for researchers and practitioners are related to the actual transmission of quality video (and audio), even with challenging conditions of congestion and mobility. Yet, we are here interested in another key problem: the minimization of the time required to factually put on the air and then manipulate a video stream through vehicle-to-vehicle (V2V) communication and, thereby, the minimization of the time required to propagate a video triggering message within a vehicular network. Indeed, minimizing the time needed to propagate the triggering message for the live video streaming results crucial for several reasons: i) it allows to quickly activate a video stream that could be critical for safety operations; ii) it is of support for the generation of a smooth video streaming even while switching between several sources (e.g., camera-endowed vehicles leaving and entering the area-of-interest); iii) it enables efficient remote control, both manual and automatic, of the chosen community-centric multimedia service; iv) it liberates network resources more quickly thus avoiding congestion, collisions, and other problems that would affect the performance even of other applications run in the same vehicular community.

To this aim, scientific literature reports that the main problem impeding a fast control message delivery in V2V communications is represented by a non-optimal (i.e., too high) number of hops/transmissions experienced by the message to cover its path [10], [11], [16]. Indeed, depending on the vehicular density, each hop/transmission corresponds to a time wastage from few tens to hundreds of ms [7]. Therefore, a multi-hop transmission of a message may take seconds to reach its area-of-interest. If we consider that a car driving at 130 Km/h on a highway covers ~36 m every second, then it is evident how a delayed message may trigger the camera (or any other requested/available device) on a vehicle when it is too late. Furthermore, negative effects of a delayed delivery of control messages become even worse when considering applications that are automatically, rather than manually, controlled and/or that may involve human lives [8], [17].

3. Fast V2V Multi-Hop Transmissions: State-Of-Art

Sending a triggering message for a live video stream or any other kind of application running in a vehicular community on the road basically consists in the fast delivery, even through multi-hops, of a broadcast message from one vehicle to others belonging to the same community and traveling in a certain area-of-interest [14]. Such problem lies in the broader research area of appropriately routing of messages in ad-hoc networks; this general topic presents several (and often antithetic) issues to cope with such as, just to mention a few, route discovery, maintenance and recovery, hop minimization, maximization of the duration of the route due to the node mobility [24]. Here, we will focus only on the need for minimizing the number of transmission hops required to cover a given car platoon, reducing the number of forward messages and consequently, fasten the data delivery.

To guarantee the message delivery, geographical coordinates and *store-carry-and-forward* techniques can be utilized [18]. The most relevant solutions proposed by researchers follow the greedy principle, also exploited by geo-routing schemes, by which the message delivery will be faster if performed through longer leaps that bring the message closer to its target: hops should hence be as long/few as possible.

To tackle this problem, a theoretically optimal algorithm has been proposed that propagates messages to cars making use of the notion of *Minimum Connected Dominating Set* (MCDS), which is a subset of nodes in a graph (in this case, the vehicular network), with minimal cardinality and direct connection to any node in the graph via at least one of its elements [19]. For instance, given the positions and the transmission ranges of vehicles shown in Fig. 1, a possible MCDS that minimizes the propagation of a message from car A to car K is represented by {A, C, F, G}. However, the computation of the MCDS is in general an NP-complete problem, and requires a continuously updated knowledge of the whole network topology [20]. It goes without saying that this is not a scalable solution, especially in a highly crowded and mobile environment such as a vehicular network.

From a practical standpoint, [21] proposes a backoff mechanism that reduces the frequency of message retransmissions when congestion is causing collisions and [8] suggests that, when a car has received a message from a following vehicle, it should refrain from forwarding it as the reception of this message is a clear confirmation that subsequent cars have already received it. Unfortunately, both these two schemes do not consider a very important factor in determining the final propagation delay of a message: the number of hops a message traverses along its path.

In [7], the minimization of the number of hops is achieved by individuating the farthest car within the source's transmission range, which has to forward the message. To this aim, each car that has received the message emits a jamming signal with a duration that is directly proportional to the distance between the considered car and the message's source. The car with the longest jamming signal is clearly the farthest car from the source and should become the next forwarder. Even if this guarantees a minimum number of propagation hops, the time wasted to determine the next forwarder through jamming signals could make this scheme not suitable for a real-time scenario.

Finally, [10] and [16] propose solutions that try to reduce the number of propagation hops through a distributed contention process based on the position of vehicles and on geometric heuristics that minimize the remaining distance to destination (*position-aware greedy forwarding*). Our solution can be thought as an evolution of that proposed in those papers. Thus, we now briefly sketch its functioning, for a better understanding of the overall approach.

In substance, the philosophy of the position-aware greedy forwarding scheme is that of considering the nodes that have received a message and electing as message forwarder the geographically closest one to the destination. In a car-platoon scenario, this can be translated into guaranteeing that the farther the distance of a vehicle from the sender of the message, the higher the probability that the vehicle will be selected as the next forwarder. The approach assumes that all cars involved in the protocol are equipped with a positioning system, e.g., GPS, to perfectly determine the geographical location of the car. (Such assumption is however quite reasonable when one realizes that cars participating in the message propagation must have installed a wireless device, which is currently more uncommon to see in a car, with respect to a GPS.) When a message is broadcast in the VANET, the sender's position is included in the message. Based on this information, each car autonomously computes its own contention window with an inverse proportion of the distance from the sender. Roughly, a contention window is the size of a time interval which is usually utilized to generate a random point in time, corresponding to a timeout. Here, the contention window is determined through the ratio among the vehicle's distance from the sender and the sender's transmission range (see next sections for further details on the formula utilized by our solution to set a node's contention window). Then cars generate a random timeout, comprised within this contention window. The idea is that the smaller the contention window, the smaller the generated random time interval. Consequently, farthest cars will be privileged in becoming the next forwarder of the message, whereas cars between the sender and the next forwarder will not transmit the message as it would be redundant.

This solution represents the state-of-art in this field; unfortunately, the effectiveness of formulas employed by [10] and [16] is based on a parameter that is not known by cars: their factual transmission range. Indeed, in a highly mobile environment such as a vehicular network, the transmission range varies very frequently, and the performance of this solution will be negatively affected by the employment of a wrong (maybe randomly chosen) parameter.

Having a correct and updated information about vehicles' transmission range is hence of primary importance to enable the fast propagation of a triggering message. Therefore, we dedicate the next section to describe in detail a simple, yet effective, way to provide vehicles with this valuable information so as to render position-aware greedy forwarding schemes fully effective and employable with success to quickly transmit triggering messages for community-based multimedia services [15].

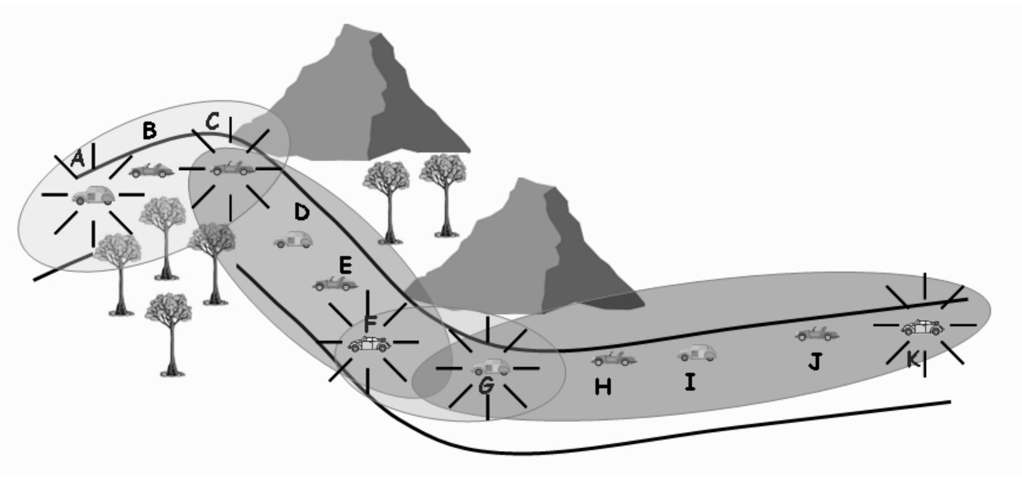

Fig. 1. Vehicular community scenario; circled areas represent the backward transmission range of the leftmost vehicle in that area.

4. Implementing An Efficient Transmission Range Estimator For A Vehicular Community

The idea at the base of the transmission range estimator discussed here is to have very few presence messages exchanged among vehicles in order to gain knowledge about the hearing capabilities of other vehicles around, i.e., the transmission range. Yet, the mechanism does neither require to achieve a perfect knowledge of the network topology, nor to necessarily have all vehicles generating these presence messages. Then, when a triggering message is sent or forwarded, the sender/forwarder includes its transmission range estimation in the triggering message so as to have all receiving vehicles aware of this value and efficiently apply the adopted position-aware greedy forwarding scheme to determine the next forwarder.

More in detail, on each car is installed a GPS that provides accurate information about time and position while power and computational resources are supposed largely adequate for our application's requirements. Time is divided into rounds and, at a certain random time chosen within each round, each vehicle tries to send a presence message. The first presence message sent in a certain round stops the sending procedure of other vehicles hearing that message. Therefore, in a certain area as large as the transmission range, only one presence message is sent per each round.

In each presence message, the sender includes i) its own position, ii) its *backward maximum distance* (BMD), and iii) its *frontward maximum distance* (FMD). Parameters BMD and FMD represent the maximum distance from which another vehicle, backward or frontward respectively, has been heard by the considered one. Clearly, data stored to determine these parameters are periodically refreshed.

Vehicles exploit presence messages to compute their estimated transmission ranges, both backward (*backward maximum range*, BMR) and frontward (*frontward maximum range*, FMR). To this aim, both the highest distance from which another vehicle has been heard sending a presence message and the highest maximum distance advertised within heard presence messages are employed. Specifically, BMR is obtained considering only presence messages coming from following vehicles and is computed as the highest among: i) all distances from vehicles that have generated these presence messages and ii) their included FMD values. Instead, FMR utilizes only presence messages sent by preceding vehicles and is equal to the highest value among: i) all distances from vehicles that have sent the considered presence messages and ii) their advertised BMDs.

Summarizing the functioning of the estimator with the help of Fig. 2, each car can be both a sender and a receiver of both presence and triggering messages. Considering for simplicity only the case where triggering messages are sent backward (the frontward case is just specular), then the various message typologies have the following purposes:

1. **Presence messages received from the front** allow the receiver to compute the FMD; this value will then be declared by the receiver in its presence messages as it were saying: *"This FMD value is the farthest distance from which I have been able to hear another car in front of me"*. As an example, in Fig. 2, upon the reception of a presence message from A (message 1), C updates its current FMD.
2. **Presence messages received from backward** include sender's FMD and position, thus providing the receiver with information about the hearing capabilities of following vehicles (see Fig. 2, message 2).
3. **Triggering messages** (transmitted backward) include the sender's BMR as it were saying: *"This BMR value is the maximum backward distance at which some car would be able to hear me"*, (message 3 in Fig. 2). The BMR value could be computed by the triggering message sender thanks to received information discussed at point 2.

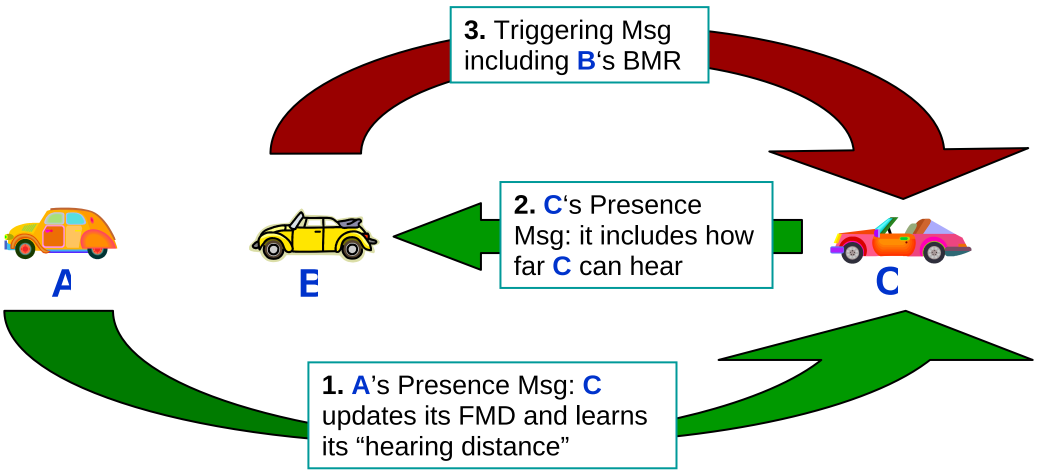

Fig. 2. Functioning of the (backward) transmission range estimator.

With the help of Fig. 3 and Fig. 4 we explain our scheme's behavior during the procedures for sending and receiving presence messages, respectively. Focusing on the presence message sending procedure (Fig. 3), in every turn, each car determines a random waiting time (lines 1 and 2) after which, if neither other transmission is heard nor collision happened (line 3), proceeds with transmitting a presence packet that includes all the estimated parameters related to the transmission range (lines 4 and 5).

```
0 for each turn
1    sending_time := random(turn_size);
2    wait(sending_time);
3    if ¬(heard_presence_msg() ∨ heard_collision())
4        presence_msg.parameters := fill_parameters();
5        transmit(presence_msg);
6    endif
7 endfor
```

Fig. 3. Presence message sending procedure.

The presence message receiving procedure is depicted in Fig. 4. In particular, a car receiving a presence message determines its own position (line 1), extracts from the message the sender's position (lines 2), and determines the distance between itself and the sender (line 3). Then, to estimate BMR and FMR, the vehicle exploits an heuristics that updates BMD or FMD by resorting to the following equation (see also lines 5, 8 in the code in Fig. 4):

$$xMD = \max(xMD_{current}, d) \qquad (1)$$

where xMD represents FMD or BMD, depending on whether the message arrives from frontward or backward, $xMD_{current}$ is the current value to be updated, and d is the distance of the vehicle that broadcast the message. In simple words, each time a message is received from a vehicle farther than others previously heard, this new information is stored through this parameter.

Needless to say, since cars are moving, transmission conditions dynamically change. Thus, old stored values are meaningless after the vehicle has covered a given distance. With this in view, the estimation of $xMD_{current}$ expires after a tuned timeout, to be promptly updated based on more recent messages.

Based on the obtained FMD and BMD, at each vehicle, FMR and BMR can be computed. In particular, BMR is heuristically calculated by considering messages coming from vehicles behind the considered one; this value is computed as the largest among all received FMDs and all distances from vehicles that generated them. Similar, specular considerations can be made for FMR. In other words, the two transmission range estimations are updated as follows (see also lines 6, 9 of the code in Fig. 4):

$$xMR = \max(xMR_{current}, d, msg.\bar{x}MD) \qquad (2)$$

where, xMR represents FMR or BMR, $xMR_{current}$ is the current value to be updated, d is the distance of the vehicle that broadcast the message, and $msg.\bar{x}MD$ is the data contained in the received message, related to the maximum hearing distance, i.e., FMD is considered if the message is received from backward, BMD is considered in the opposite case.

```
 0  // a message is received
 1  mp := my_position();
 2  sp := msg.sender_position;
 3  d := distance(mp, sp);
 4  if (received_from_front(msg))
 5     FMD := max(current_FMD, d);
 6     FMR := max(current_FMR, d, msg.BMD);
 7  else // received from back
 8     BMD := max(current_BMD, d);
 9     BMR := max(current_BMR, d, msg.FMD);
10  endif
```

Fig. 4. Parameters' updating procedure.

As to the temporal duration of a round, we claim that its size could be in the order of magnitude of a second. Indeed, it is likely that transmission ranges vary depending on the movements of the car and the obstacles they encounter during such movement. Cars move at a speed of the order of few m/s. Empirically, it is reasonable to assume that presence messages (which serve to understand if the transmission range changes) are sent during 1 s long rounds. As demonstrated in our simulations, these heuristics provide vehicles with accurate estimations of transmission ranges.

When a triggering message is broadcast along the car platoon, the transmission range estimation (BMR or FMR depending on the direction towards the destination) is used by vehicles on the message's path to determine which one among them will have to take upon itself the task of becoming the next forwarder. Since our aim is that of minimizing the number of hops to reduce the propagation delay, we want the farthest vehicle within the sender's transmission range to perform this task. Therefore, the longer the distance of a receiving vehicle from the sender, the higher the priority of that vehicle in becoming the next forwarder.

In particular, 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 randomly computed based on a contention window value, as inspired by classical backoff mechanisms in IEEE 802.11 MAC protocols [23].

At each hop, the self-elected forwarder updates BMR and FMR fields of the message with its computed values so as to utilize proper parameters for the successive portion of road.

Thus, vehicles compute their contention windows by simply plugging the estimated transmission range parameter xMR (BMR if backward, FMR if frontward) into the geometric heuristic employed by the adopted position-aware greedy forwarding mechanism. Inspired by [10], in our system the contention window CW of each vehicle can vary between a minimum number of time slots (*CWMin*) and a maximum one (*CWMax*), depending on the distance from the sending/forwarding vehicle (*Dist*) and on the advertised xMR, as summarized by (3).

$$CW = \left\lfloor \left(\frac{xMR - Dist}{xMR} \times (CWMax - CWMin) \right) + CWMin \right\rfloor \tag{3}$$

Using (3), the farthest vehicle in the sender's transmission range is privileged in becoming the new forwarder. Indeed, the nearer the vehicle to the sending car, the larger the contention window; larger contention windows make more likely that a larger timeout value will be chosen and, hence, that somebody else will be faster in forwarding the game event. To better understand how *CW* determines the priority of a vehicle in becoming the next forwarder, we provide in Fig. 5 the pseudocode for the triggering message forwarding procedure.

Upon receiving the message, the considered car determines the direction of propagation of the message (line 1), extracts from the message the parameters it needs to plug them into (3) and determine its *CW* (line 2); then, it computes a random waiting time based on it (lines 3-4). If, while waiting, the same message is heard again, coming from the opposite direction where it came (line 5), then the message has already propagated over the considered car that can hence stop trying to forward it: somebody else already did it (line 6). Conversely, if the same message is heard from the opposite direction of propagation, i.e., the same direction where the message came (line 7), 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 (line 8).

If the waiting time expires without having heard any other car forwarding the same message then the considered car broadcasts it (line 11) including the estimated transmission range (line 10). Obviously, if the broadcast fail, a backoff mechanism is utilized to compute the next transmission time.

```
 0 // a triggering message is received
 1 dir = determine_direction_of_propagation();
 2 CW:= utilize_eq_3();
 3 rcw := random(CW);
 4 wait(rcw);
 5 if (forward_heard_from(direction))
 6    exit();
 7 else //
 8    restart_forwarding_procedure();
 9 else
10    triggering_msg.parameters := parameters;
11    transmit(triggering_msg);
12 endif
```

Fig. 5. Application message forwarding procedure.

As a final observation, it is important to point out that not only is the overhead caused by the aforementioned (few) presence messages significantly smaller if compared to other schemes using a similar approach, but they also have to be employed only when dealing with applications that generate sporadic messages (e.g., service triggering). Otherwise, the required parameters (i.e., FMD, BMD, BMR, FMR, etc.) can be included in regularly exchanged application messages, thus eliminating the need for presence messages at all.

5. Simulation Assessment

In this section we report on simulation results that shed light on the effectiveness of employing an efficient transmission range estimator, (i.e., our estimator, as discussed in the previous section).

To assess our scheme, we have employed a discrete event based simulator that we built. It is worth noting that modeling time evolving through discrete steps is quite reasonable in this scenario, since wireless transmission protocols usually employ the notion of time slots to perform transmissions [23]. The simulator allows to model a strip-shaped road with several cars traveling at different speeds. We put the focus on a car platoon in linear strip since this is a classic experimental scenario employed to assess whether a broadcasting procedure effectively distributes messages in a VANET [7], [8], [10]. Of course, based on such a configuration, all complications arising in more complex traffic scenarios, e.g., urban crossroads, very steep mountain roads, are neglected. Nevertheless, the simulation is of help to understand the proper functioning of the proposed approach in classic traffic configurations.

Our simulator does not perfectly mimic all the features of the IEEE 802.11 wireless protocol. This is simply due to the fact that IEEE 802.11p standard (the standard for VANETs) is still in a draft status while the paper is written. For the same reason, we claim that the use of wireless network models available in well-known simulation tools, e.g., NS-2, would introduce inaccuracies and flaws that would make useless the addiction of simulative details of the MAC layer protocol. In any case, the simulator is perfectly able to emulate transmission collisions, the calculation of congestion windows, and traditional backoff schemes (employed also for the schemes contrasted against our approach).

As several wireless models in network simulation tools do (e.g., NS-2), even if the average transmission range is fixed in our simulator we modeled this parameter to randomly vary in time and in different portions of the strip, so as to mimic interferences as it would happen in real life. These interferences make the factual transmission range oscillate around 300 m. The variation was however less than the 10% of the transmission range.

During different simulations, we varied the dimension of the road from 4 to 20 Km. Vehicles were set to move along with speeds uniformly distributed in the range 72-144 Km/h, passing by very different surroundings (e.g., buildings, hills, curves). Cars were initially placed at random points along the road. In particular, considering a freeway with multiple lanes, several vehicle densities are generated: from 60 to 240 cars per Km. V2V multi-hop communication is assumed to be supported by DSRC/802.11p technology, which has been declared able to guarantee a maximum range of 1000 m under optimal conditions, or a smaller range at very high speeds (around 300 m for a car travelling at 200 Km/h) [22]. For the sake of conciseness, results are presented only for the representative case of 300 m; indeed, outcomes for the other values of the varying transmission range are coherent with those presented here.

Based on our simulator, cars were allowed to randomly generate novel triggering messages, by modeling them as Poisson processes. Upon reception of a triggering message, cars are engaged in the broadcasting procedure. Collisions in the transmissions are simulated when two (or more) cars in overlapping transmission ranges try to simultaneously broadcast a message (i.e., during the same time slot).

Inspired by values used by the IEEE 802.11 protocol, *CWMin* and *CWMax* in (3) are set equal to 32 and 1024 slots, respectively. Finally, a positioning system (e.g., a GPS device) providing accurate information about position and trajectory is assumed to be present on-board.

The mechanism endowed with the transmission range generator explained in this article, namely *Fast Triggering* (*FT* in the following charts), is compared against other schemes that do not make use of dynamically estimated transmission range. They are as follow. Similar to *FT*, *Static* corresponds to the solution proposed in [10] and assigns different forwarding priorities through backoff delays that depend on the node's distance from the source. Basically, it utilizes (3), yet using a predetermined maximum transmission range value, regardless of the actual one.

Two instances of the *Static* scheme are employed: the *Static* scheme utilizing fixed BMD, FMD values of 300 m (*Static300*) and the *Static* scheme utilizing fixed BMD, FMD values of 1000 m (*Static1000*). Clearly, with 300 m of actual transmission range, *Static300* corresponds to the ideal solution.

We also evaluated schemes that do not employ any distance prioritization; simply, every car computes a random waiting time within its contention window before forwarding the message (if no one else already did). We name this class of schemes *Random*. More in detail, *RandomLow* and *RandomHigh* utilize fixed contention windows equal to *CWMin* and *CWMax*, respectively. Finally, *RandomInc* employs a traditional backoff scheme where the contention window, comprised between *CWMin* and *CWMax* is doubled every time there is a collision, thus adapting to the congestion on the channel.

A final note: values in the charts correspond to average results after 40 simulations and are presented with their 95 % confidence intervals.

6. Measured Performances

In order to compare the aforementioned schemes, parameters that have a direct delay impact have been considered; in particular: the total number of hops required to propagate the triggering message to the area-of-interest, the total number of messages sent, and the total number of transmission collisions.

In particular, Fig. 6 shows the average number of hops that a triggering message experiences to cover each Km, along with related confidence intervals. As it is evident, *FT* achieves results that are comparable with those achieved by the ideal scheme where the actual transmission range is known *a priori*, i.e., *Static300*. Moreover, as the location of the next forwarder depends on the employment of priorities and not on the vehicle density, it is not surprising that the number of hops per Km is not influenced by the traffic conditions for any of the compared schemes.

To understand the causes of results reported in Fig. 6, we have to analyze the total number of messages transmitted per Km. To this aim, Fig. 7 shows that *FT* and the ideal scheme *Static300* perform better than the others. This happens for two main reasons. First, if using large contention windows for all vehicles, as done by *Static1000* and *RandomHigh*, farthest vehicles' likelihood in becoming the next forwarder is sensibly reduced. Consequently, the average hop length for these schemes results shorter, thus augmenting the number of message transmissions required to cover the considered road portion. Second, when small contention windows are employed by all vehicles, as done by *RandomLow* and *RandomInc*, the number of message collisions increases, thus sometimes requiring retransmissions to propagate the message.

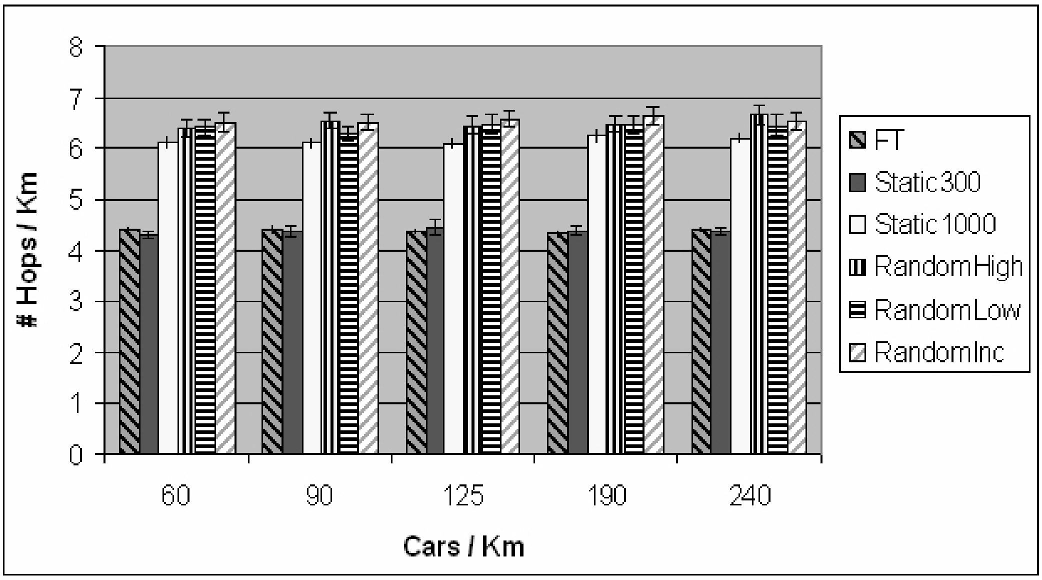

Fig. 6. Average number of hops per Km (and related 95 % confidence interval).

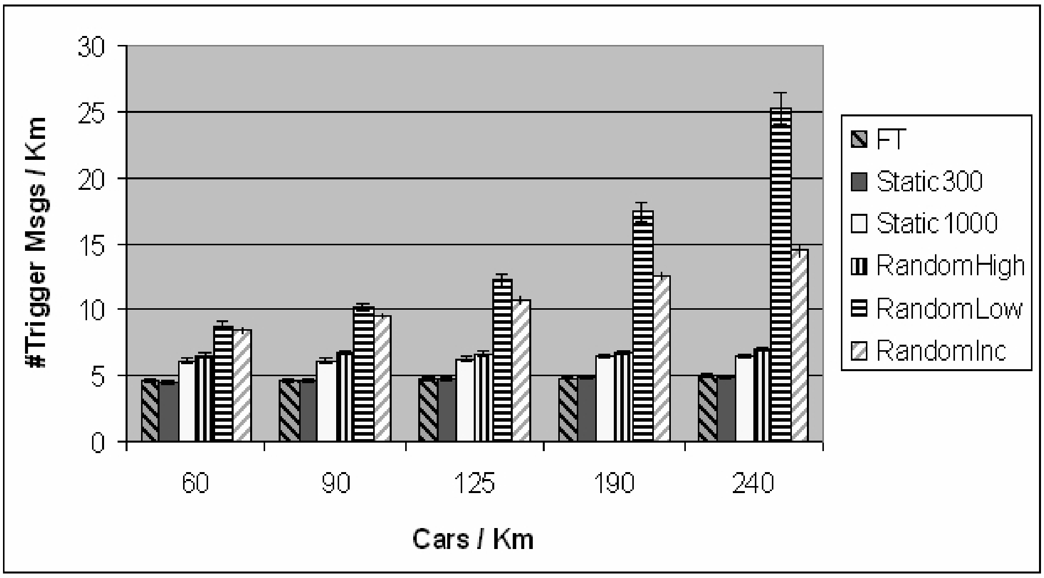

Fig. 7. Average number of messages sent per Km (and related 95 % confidence interval).

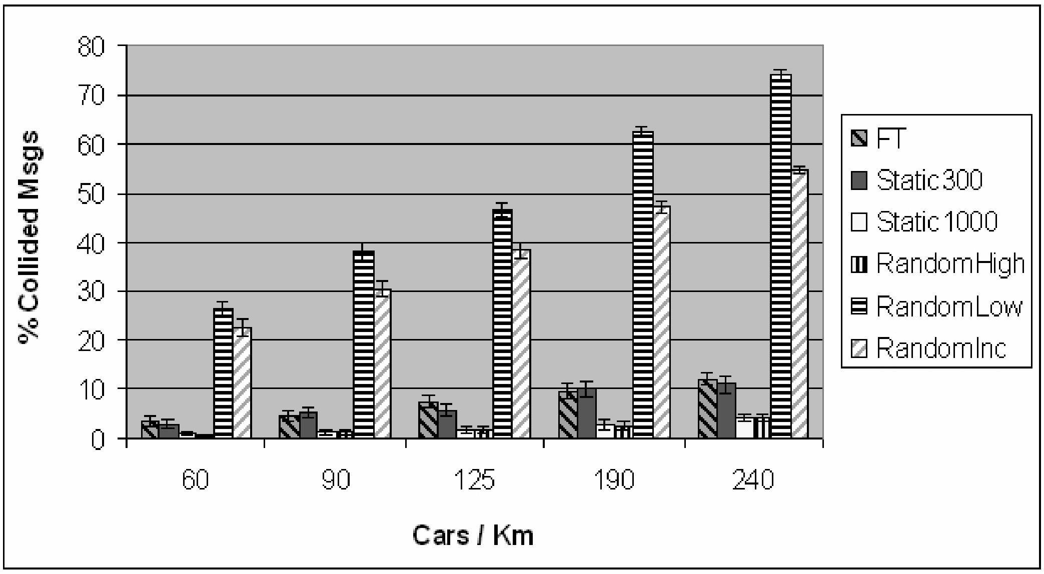

Fig. 8. Average percentage of message collisions per Km (and related 95 % confidence interval).

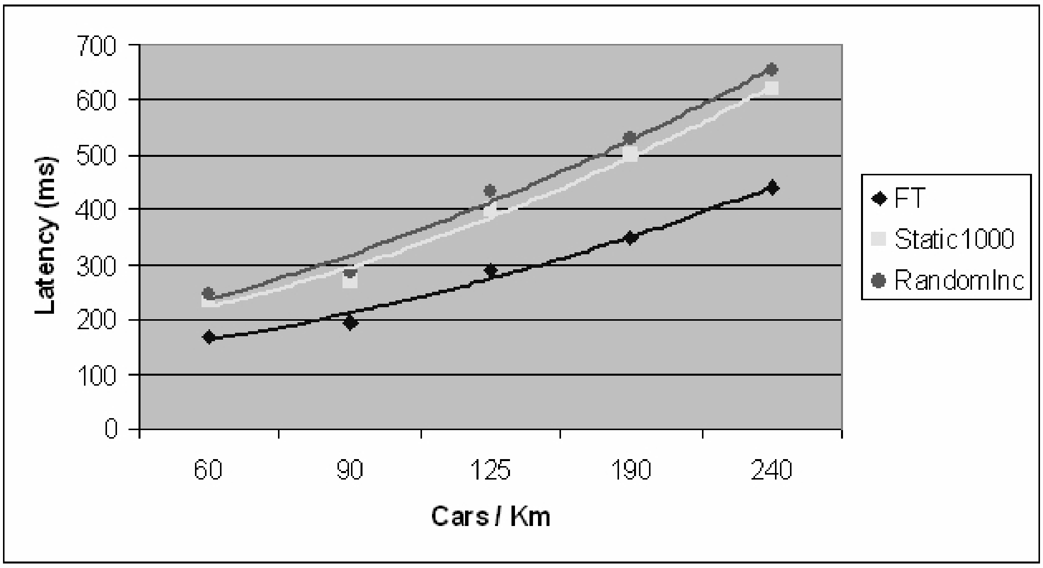

Fig. 9. Triggering message delivery latency per Km, with second order trend lines.

The latter problem is confirmed by Fig. 8, which shows the percentage of collisions observed during the forwarding activity of triggering messages. The highest number of collisions is experienced by *RandomLow* and *RandomInc*. This is due to the fact that the contention window utilized by these schemes is undersized for a scenario with a high node density such as V2V communications.

Lower percentages of collided messages have been measured for schemes that exploit large contention windows, i.e., *Static1000* and *RandomHigh*.

Unfortunately, this obvious result is coupled with a well known negative property by which larger contention windows statistically generate longer waiting times before a message is sent [23]. Therefore, even if *Static1000* and *RandomHigh* do not waste much time in retransmissions, they still waste time by waiting, in average, a higher number of time slots before each message is sent.

A tangible measure of the delivery time impact of the various solution typologies is reported in Fig. 9. Values and trend lines were computed utilizing the hops-per-Km ratio shown in Fig. 6 and per-hop latencies measured in [7]. In particular, Fig. 9 reports only results for the three different schemes *FT*, *Static1000* and *RandomInc*. As expected, *Static1000* and *RandomInc* have delivery latencies sensibly higher than *FT*, with a time raise that goes from 37 % to 49 %. Even if the trend shows an increase of the transmission latency with the vehicle density for all schemes, with *FT* this phenomenon results more graceful.

Remarkably, for all considered settings, *FT* is able to dynamically adapt to the real transmission conditions thanks to its transmission range estimator, thus avoiding potential propagation delay sources (i.e., excessive number of hops, transmitted messages, and collisions).

Finally, we assessed how schemes behave when a number of vehicles generate multiple triggering messages. Specifically, we have investigated performances when 50 vehicles were set to generate triggering messages. Different per-vehicle average generation rates were evaluated, i.e., 100 ms, 300 ms, or 500 ms. Outcomes are presented in Fig. 10. From the graph, it emerges that our scheme always achieves better results than the other three schemes, even better than *Static300* that is supposed to embody the ideal scheme in a scenario with 300 m of transmission range. This result is due to the ability of our approach in adapting to the slight variations of the transmission range generated by the realistic wireless model. In any case, all schemes seem to not be influenced by an increment of the message generation rate, thus confirming the viability of triggering message distribution for the support of social and multimedia applications over VANETs.

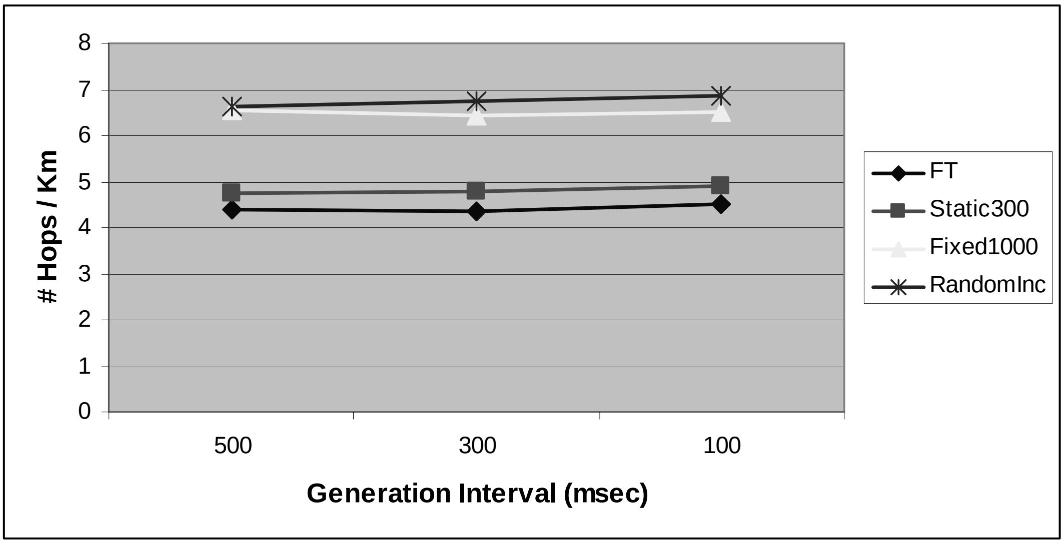

Fig. 10. Average number of hops per km; 50 generating cars.

7. Conclusion

Social network communities, especially those related to user generated multimedia contents, have quickly become one of the most popular services enjoyed by users through their home or office connections. Following this trend, it easy to notice their potential on wireless, mobile networks; indeed, social network communities are going to be even more appealing if offered to mobile users thanks to their combination with location-based (or proximity-based) services.

In this paper, we considered the problem of providing a fast triggering system for activating and managing general multimedia services (e.g., video-streaming, approximeeting) offered to vehicular communities on the road. Addressing this problem, we surveyed the state-of-art and highlighted the need for an efficient transmission range estimator. Then, we reported on how to implement a simple, yet effective, transmission range estimator to respond to this need. Comparisons between a position-aware greedy forwarding mechanism employing this transmission range estimator (namely, Fast Triggering) and state-of-art solutions demonstrate that the former achieves results comparable with those of an ideal scheme. Even better, as prominent advantage of Fast Triggering, this result is achieved without requiring perfect knowledge of the network topology, but just employing the described transmission range estimator.

As future work, the concept behind Fast Triggering can be extended in several research directions. For instance, it could be used to support also the transmission of the actual multimedia stream (and not just to trigger it), or coupled with routing protocols in ad-hoc and sensor networks to reduce nodes' energy consumption.

References

1. J. Preece, *Online Communities – Designing Usability, Supporting Sociability*, Chichester, UK, John Wiley and Sons, 2000.
2. M. Cha, H. Kwak, P. Rodriguez, Y.-Y. Ahn, S. Moon, "I Tube, You Tube, Everybody Tubes: Analyzing the World's Largest User Generated Content Video System", *Proc. of Internet Measurement Conference (IMC'07)*, San Diego, CA, USA, Oct 2007, 1-14.
3. J. M. Leimeister, M. Daum, H. Krcmar, "Towards M-Communities: - The Case of COSMOS Healthcare" , *Proc. of the 36th Annual Hawaii International Conference on System Sciences (HICSS'03)*, Big Island, HI, USA, Jan 2003, pp 8.
4. Global Information, Inc., *Mobile Social Networking and User Generated Content Market Insight 2008*, Published by Visiongain, Feb 2008, Web site: http://www.infoedge.com/product_type.asp?product=VG-4002, last visited Nov 2008.
5. A. Burak, T. Sharon, "Usage Patterns of FriendZone – Mobile Location-Based Community Services", *Proc. of 3rd International Conference on Mobile and Ubiquitous Multimedia (MUM 2004)*, College Park, MD, USA, Oct 2004, 93-100.
6. M. Heitmann, C. Prykop, P. Aschmoneit, "Using Means-end Chains to Build Mobile Brand Communities", *Proc. of the 37th Annual Hawaii International Conference on System Sciences (HICSS'04) Track 7 - Volume 7*, Big Island, HI, USA, Jan 2004, HICSS. IEEE Computer Society, Washington, DC, 70196.3.
7. J. Yin, T. Elbatt, G. Yeung, B. Ryu, S. Habermas, H. Krishnan, T. Talty, "Performance Evaluation of Safety Applications over DSRC Vehicular Ad Hoc Networks", *Proc. of the 1st ACM international Workshop on Vehicular Ad Hoc Networks (VANET '04)*, Philadelphia, PA, USA, Oct 2004, ACM, New York, NY, 1-9.

8. S. Biswas, R. Tatchikou, F. Dion, “Vehicle-to-Vehicle Wireless Communication Protocols for Enhancing Highway Traffic Safety”, *IEEE Communication Magazine,* 44(1):74-82, Jan 2006.
9. G. Kortuem, J. Schneider, D. Preuitt, T. G. C. Thompson, S. Fickas, Z. Segall, “When Peer-to-Peer comes Face-to-Face: Collaborative Peer-to-Peer Computing in Mobile Ad hoc Networks”, *Proc. of 1st IEEE International Conference on Peer-to-Peer Computing (P2P2001),* Linköpings Universitet, Sweden, IEEE Computer Society, Washington, DC, 75.
10. E. Fasolo, R. Furiato, A. Zanella, “Smart Broadcast Algorithm for Inter-vehicular Communication”, *Proc. of WPMC’05*, Aalborg, Denmark, Sep 2005.
11. G. Korkmax, E. Ekici, F. Ozguner, U. Ozguner, “Urban Multi-hop Broadcast Protocol for Inter-vehicle Communication Systems”, *Proc. of ACM VANET’04,* Philadelphia, PA, Oct 2004, 75-86.
12. J. Rybicki, B. Scheuermann, W. Kiess, C. Lochert, P. Fallahi, M. Mauve, “Challenge: Peers on Wheels - A Road to New Traffic Information Systems”, *Proc of the 13th Annual International Conference on Mobile Computing and Networking (MobiCom 2007),* Montreal, Quebec, Canada, Sep 2007, 215-221.
13. C. Satchell, S. Singh, “The Mobile Phone as a Globalising Artefact”, *Proc. of the 11th International Conference on Human-Computer Interaction (HCI 2005)*, Las Vegas, NV, USA, Jul 2005.
14. M. Guo, M.H. Ammar, E.W. Zegura, “V3: a Vehicle-to-Vehicle Live Video Streaming Architecture”, *Proc. of IEEE PerCom 2005*, Kauai, HI, Mar 2005, 171-180.
15. M. Roccetti, M. Gerla, C.E. Palazzi, S. Ferretti, G. Pau, “First Responders' Crystal Ball: How to Scry the Emergency from a Remote Vehicle”, *Proc. of IEEE NetCri 07 – IEEE IPCCC 2007,* New Orleans, LA, Apr 2007, 556-561.
16. H. Füßler, J. Widmer, M. Käsemann, M. Mauve, H. Hartenstein, “Contention-Based Forwarding for Mobile Ad-Hoc Networks”, *Elsevier Ad Hoc Networks,* 1(4):351-369, Nov 2003.
17. B.Gallagher, H. Akalsuka, H. Suzuki, “Wireless Communications for Vehicle Safety: Radio Link Performance and Wireless Connectivity Methods”, *IEEE Vehicular Technology Magazine,* 1(4):4-24, Dec 2006.

18. W. Zhao, M. Ammar, E. Zegura, “A Message Ferrying Approach for Data Delivery in Sparse Mobile Ad Hoc Networks”, *Proc. of ACM MobiHoc’04*, Roppongi, Japan, May 2004, 187-198.
19. A. Zanella, G. Pierobon, S. Merlin, “On the Limiting Performance of Broadcast Algorithms over Unidimensional Ad-hoc Radio Networks”, *Proc. of WPMC’04*, Abano Terme, Italy, Sep 2004, 165-169.
20. P.J. Wan, K. Alzoubi, O. Frieder, “Distributed Construction of Connected Dominating Set in Wireless Ad hoc Networks”, *Proc. of IEEE INFOCOM 2002*, New York, NY, Jun 2002, 141-149.
21. X. Yang, J. Liu, F. Zhao, N. Vaidya, “A Vehicle-to-Vehicle Communication Protocol for Cooperative Collision Warning”, *Proc. of the First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, 2004. MOBIQUITOUS 2004*, Boston, MA, USA, Aug 2004, 114-123.
22. ASTM E2213-03, “Standard Specification for Telecommunications and Information Exchange Between Road-side and Vehicle Systems – 5.9GHz Band Dedicated Short-Range Communications (DSRC) Medium Access Control (MAC) and Physical Layer (PHY) Specifications”, ASTM International, Jul 2003.
23. IEEE 802.11g, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, Amendment 4: Further Higher Data Rate Extension in the 2.4GHz Band.
24. G. Lim, K. Shin, S. Lee, H. Yoon, J. S. Ma, “Link Stability and Route Lifetime in Ad-hoc Wireless Networks”, *Proc. of IEEE Int. Conf. on Parallel Processing Workshops (ICPPW ’02)*, Vancouver, BC, Canada, Aug 2002, 116-123.
25. M. Caliskan, D. Graupner, M. Mauve, “Decentralized Discovery of Free Parking Places”, *Proc. of the 3rd ACM International Workshop on Vehicular Ad Hoc Networks (VANET’06)*, Los Angeles, CA, USA, Sep 2006, 30-39.