



HAL
open science

Adaptive Contention Window Scheme to Improve Multihop Broadcast in VANETs

Adel Berradj, Zoubir Mammeri

► **To cite this version:**

Adel Berradj, Zoubir Mammeri. Adaptive Contention Window Scheme to Improve Multihop Broadcast in VANETs. IEEE 81st Vehicular Technology Conference (VTC2015-Spring), IEEE Vehicular Technology Society, May 2015, Glasgow, Scotland, United Kingdom. pp.1–6, 10.1109/VTC-Spring.2015.7145667 . hal-03193106

HAL Id: hal-03193106

<https://hal.science/hal-03193106>

Submitted on 13 Apr 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Adaptive Contention Window Scheme to Improve Multi-hop Broadcast in VANETs

Adel Berradj, Zoubir Mammeri

IRIT - Paul Sabatier University

Toulouse, France

Email: {adel.berradj, zoubir.mammeri}@irit.fr

Abstract—Most VANET safety applications use multi-hop broadcast communications to disseminate safety information as far as needed in the zone of relevance. Recently, many multi-hop broadcast forwarder selection schemes have been proposed. Their aim is to reduce the number of forwarders by selecting a subset of nodes in the zone of relevance that forward the messages to cover the entire zone as fast as possible. In this paper, we propose a density-distance based multi-hop broadcast scheme to address the problems of (i) redundant broadcasts particularly in dense network and (ii) the high latency in sparse networks. We present a solution where both the distance and the network density are considered in calculating the waiting time before deciding whether to rebroadcast received messages or not. Simulation results show that our scheme achieves better performance in terms of number of rebroadcasts and dissemination delay compared to existing solutions.

Keywords—VANETs; safety applications; multi-hop broadcast; network density; node distance.

I. INTRODUCTION

Road safety applications are the main target of VANETs, though other applications such as Infotainment and Road traffic oriented may be provided to road users [1]. Road traffic-oriented applications can help to improve road safety by providing drivers with information on the conditions of road traffic. This information allows the driver to avoid congested roads and make traffic flow fluid. Infotainment applications like internet access or peer-to-peer applications are dedicated to passenger comfort, and they can bring added value, which will help increase VANET market penetration rate.

Unlike the conventional Internet access where messages are typically exchanged on a unicast basis, dissemination of information in VANET safety applications requires broadcast because the information (in particular alert messages) is of interest to all the vehicles in the zone of relevance. In addition, to reach all the vehicles in the zone of relevance, some of them being out of the transmission range of the message source, multi-hop broadcast is required to deliver the messages. A perfect broadcast mechanism is the one which enables delivery of the data to all interested nodes on time and with optimal utilization of resources. Unfortunately, wireless communication is generally unreliable and poses many challenges to overcome such as the problem of hidden stations, packets collision, channel fading, and obstacles.

In this paper we focus on the multi-hop broadcast in VANETs for safety applications. Our aim is to propose a protocol for the dissemination of the alert messages in highway context. Our protocol is a distance-based scheme in which

nodes wait for a random time before rebroadcast the message they receive. This time is drawn in a specified interval (i.e. a contention window). The boundaries of this time interval are inversely proportional to the distance from the last forwarder. Unlike other proposed schemes, our broadcast protocol includes network density in calculating the time interval from which the random waiting time is drawn. In fact, when the network density is high, our broadcast protocol results in wider waiting time interval, which reduces the probability that multiple nodes located at the same distance from the last forwarder, draw out the same value of the timer. When the network is sparse, our broadcast reduces the time that each node waits before deciding whether to forward the received message or not. Therefore the dissemination delay is reduced. We propose two functions to calculate the time interval boundaries with which we expect to: (i) reduce the number messages forwarders particularly when the network density is high and (ii) reduce the message dissemination delay particularly in sparse networks.

The rest of the paper is organized as follows. In section II we present a brief overview on the related work about alert messages dissemination schemes. Motivation is given in section III. Then, our proposed broadcast protocol is described in section IV. Section V evaluates the performance of our protocol. Finally, we conclude this work in section VI.

II. RELATED WORK

To reduce the number of hops and to decrease the delivery delay, most of multi-hop broadcast forwarder selection mechanisms are based on the distance between the sender and the receiver. Their principle is to select nodes allowing fast progress in the direction of dissemination to forward the messages. The farthest nodes to the last forwarder will have the lowest waiting time. The node that succeeds to forward the message avoids to the remaining nodes forwarding the same message. It is worth noticing that, some work has focused on adapting the transmission power or the transmission rate, depending on the network conditions (such as network density, channel condition, etc.). Other mechanisms use a hierarchical structure of the network (i.e. clustering). Due to the space limits, we focus only on contention-based forwarder selection mechanisms. Some of existing protocols based on this mechanism require one hop information and others use additional mechanisms to ensure reliability and efficiency of dissemination.

In UMB [2] a handshake mechanism RTB/CTB similar to IEEE 802.11 CTS/RTS is proposed. The authors aim to ensure

reliability and to reduce the effect of the hidden terminal problem. In their protocol, the transmission range in the direction of dissemination is divided into multiple segments. A jamming signal proportional to the distance from the last forwarder is used to select the next forwarder. Smart Broadcast [3] (SB) also relies on the RTB/CTB hand shake mechanism. However, it uses an implicit instead of explicit acknowledgment as used in UMB and it does not use any collision resolution scheme as in UMB. In FR-EDM [4], the total number of segments is not fixed as in UMB and in SB, rather it depends on the block size, which is defined according to the network density. To ensure reliability, the message is repeatedly broadcasted with a calculated cycle. In Bi-Zone [5], the transmission range of the last forwarder is divided into two adjacent areas using a distance threshold. Nodes whose distance to the last forwarder is greater than the threshold have smaller waiting times than those whose distance is less than the threshold. Consequently, the contention window depends on the distance to the sender and the fixed threshold.

Others protocols use a broadcast probability to reduce the number of forwarders. In these protocols, each node rebroadcasts the message according to some probability. The lower the broadcast probability is, the lower the cost (i.e. bandwidth consumption) of data dissemination and the lower the probability of delivery of messages to all recipients are. The higher the broadcast probability is, the higher the dissemination cost and the higher message delivery probability are. Consequently, there is a tradeoff between the dissemination cost and delivery probability. Existing broadcast protocols proposed different ways to derive the broadcast probability: a fixed value during all network lifetime, a value depending on channel condition [6], [7], a value depending on the number of messages received within some time interval [8], a value depending on the number of one-hop neighbors [9], [10], the distance between the sender and receiver of the message [8], [11]. Each broadcast probability calculation model has its pros (simplicity and scalability) and cons (overhead due to the collection of data used in probability calculation). In existing protocols [8], [12], probability-based mechanism is combined with other mechanisms (e.g. distance or density) to improve broadcast performance. In our broadcast protocol we adopt the distance-density-based mechanism, which is based on a novel calculation scheme of the time interval from which the nodes take their waiting time before rebroadcast.

III. MOTIVATION

Unlike other VANET applications, in safety applications the data handled is very critical and may have impacts on human life. This makes delay and reliability the main critical requirements that must be taken into account in the development and evaluation of the broadcast protocols targeting these applications. However, the satisfaction of these two requirements is very challenging and there is a tradeoff between dissemination reliability and delay.

Most of schemes presented in section II aim at reducing the dissemination delay by selecting as next forwarder the node allowing the maximum progress in the direction of dissemination. They differ mainly in the function that calculates the time interval (i.e. the contention window) from which the waiting time before rebroadcast is drawn out. Nevertheless,

most of them [2], [5], [8], [13] still suffer message collisions in dense networks and high latency in sparse networks. Their limitation is that they do not consider the network density when calculating the time interval. In fact, in [5] for example, before rebroadcasts the received message, each node draw out a timer in the interval $[T_{lower}, T_{upper}]$. The value of T_{lower} is set to 0 for the far nodes where the distance to the last forwarder is greater than a predefined distance threshold, and their T_{upper} is calculated as follows:

$$T_{upper} = T_{max} \times \left(1 - \frac{Dist}{R}\right) \quad (1)$$

where R is the transmission range, T_{max} is the maximum waiting time and $Dist$ is the distance from the last forwarder. For closer nodes where the distance from the last forwarder is less than the distance threshold, T_{upper} is fixed to the maximum waiting time (T_{max}) and their T_{lower} is calculated as follows:

$$T_{lower} = T_{max} \times \left(1 - \frac{Threshold}{R}\right) \quad (2)$$

Let us consider the case where several nodes are located at the border of the transmission range (e.g. highway with multiple lanes in the same direction). In this case: $Dist/R = 1$, then $T_{upper} = 0$. Consequently, all these nodes have the same value of T_{upper} , which is 0 and they rebroadcast the message at the same time without waiting. This leads to a higher message redundancy, channel load, channel contention, and waste of the bandwidth, which is particularly damaging in dense networks. With (1), even if some nodes are not exactly at the same distance from the last forwarder, their time intervals are almost similar. In this case, the time interval of each node widely overlaps with the ones of its neighbors and the non-overlapping portion is small. This increases the probability that several nodes draw out the same waiting delay value, which is very likely when the network density is high. Although [13] defines another forwarder selection mechanism with a different function, the same problem may occur. In fact, in [13] the waiting time is computed based on a contention window (CW) that varies between a minimum (CW_{min}) and maximum CW_{max} , depending on the distance from the last forwarder and on the advertised estimated transmission range, as shown in (3):

$$CW = \left\lceil \left(\frac{EstTrans}{Dist} \times (CW_{max} - CW_{min}) \right) + CW_{min} \right\rceil \quad (3)$$

where $EstTrans$ the advertised estimated transmission range and $Dist$ is the distance to the last forwarder. Moreover, these mechanisms favor nodes which are at the border of the transmission range. It is well known that the reception probability is low in such a zone [14]. If the selected nodes do not receive the broadcast message correctly or may not exist as in the case of sparse networks or in an area with high signal fading and shadowing, nodes near to the forwarder are penalized and wait unnecessarily.

IV. PROPOSED BROADCAST PROTOCOL

In this section, we introduce our broadcast protocol that aims at overcoming the problems described in Section III. To reduce the number of forwarders and the delivery delay, our broadcast protocol considers both the distance and network density in calculating the waiting time before deciding whether

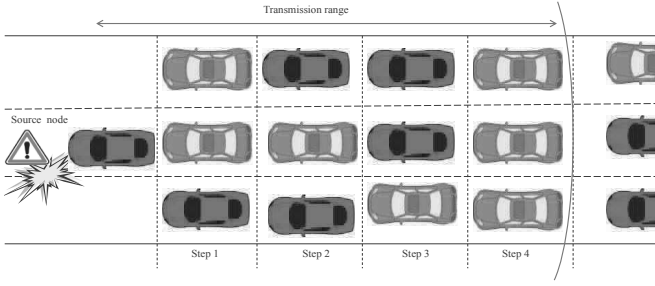


Fig. 1. An example illustrating our broadcast protocol

to rebroadcast or not. Therefore, our protocol relies on two functions that calculate the values of the boundaries of the time interval $[T_{lower}, T_{upper}]$. In the following, we describe our broadcast protocol for alert message dissemination in highway context.

A. Assumption

We consider a group of vehicles on a highway. We assume that all vehicles are equipped with a GPS receiver; therefore, each vehicle is able to get its own position. All vehicles are equipped with WAVE/DSRC device enabling them to communicate, either directly or through multi-hop communication. Each node periodically broadcasts a *Cooperative Awareness Message* (CAM) that includes its position, direction, speed, etc. [15]. The CAMs are used by most of safety applications and they are considered as background traffic. In our case CAMs are not necessary exchanged at a high frequency; they are used by nodes to have an instantaneous estimation of the number of their neighbors. Each node maintains a table in which it stores information about nodes within its transmission range. This information is used by nodes to estimate their local density. We define the local density as: $Density = N/N_{max}$, where N is the number of one-hop neighbors and N_{max} is the maximum of one-hop neighbors in a jammed traffic scenario. When an accident occurs or a dangerous situation is detected, a *Decentralized Environment Notification Message* (DENM) [16] is generated by the concerned vehicle. The DENM must be forwarded until it reaches all vehicles in the specified zone of relevance (i.e. danger zone). This message includes the event location, the direction of dissemination, the size of the zone of relevance, etc.

B. Mechanism description

Our broadcast protocol relies on the distance criterion. It tries to select the farthest node from the last forwarder to rebroadcast the message. As shown in Fig. 1, the transmission range of the sender/last forwarder in the direction of dissemination is divided into multiple steps of a fixed width. Each step is assigned a time interval from which the nodes located in this step will draw their waiting times before deciding whether to rebroadcast or not. The boundaries T_{upper} and T_{lower} of the time interval are given in (4) and (5), respectively. The detailed description of the functions will be discussed later. When a vehicle detects a dangerous situation, it broadcasts a DENM to the nodes in the zone of relevance. Each receiver of the broadcast message, checks whether it is a new message, and if it is in the direction of dissemination of the message. This

is based on the information included in the DENM. If both conditions are fulfilled, the receiver triggers its timer. When the timer expires, the node adds its coordinates in the message and rebroadcasts it. If a node receives the same message during its contention period, it exits the contention phase and it discards the message. It is noteworthy that the receiver of the alert message uses its current GPS coordinates and the coordinates of the source node (i.e. the initiator of the DENM) included in the DENM to determine whether it is located in the zone of relevance or not.

$$T_{upper} = \max \left\{ T_{max} \times (1 - \Delta_{dist}) \times Density, T_{min} + CW_{length} \right\}, \quad (4)$$

$$T_{lower} = T_{upper} - CW_{length} \quad (5)$$

where T_{min} and T_{max} are the minimum and the maximum waiting time, respectively; $Dist$ is the distance from the last forwarder, $Density$ is the local density perceived by the node under consideration, R is the transmission range; CW_{length} and Δ_{dist} are given in (6) and (8) respectively.

$$CW_{length} = (T_{max} - T_{min}) \times Density / StepNumber \quad (6)$$

$$StepNumber = \frac{R}{StepLength} \quad (7)$$

$$\Delta_{dist} = \left\lfloor \frac{Dist}{StepLength} \right\rfloor \times \frac{1}{StepNumber} \quad (8)$$

C. Detailed Description of the functions

The basic idea is to have time intervals where the two boundaries slide according to the distance of the node from the last forwarder and according to the density of the network. Indeed, we are aiming at expanding the width of the time interval when the network density is high to reduce the probability that two or more nodes pick out the same value of the timer. We try to find a compromise between delivery delay and message redundancy.

1) T_{lower} : Unlike other schemes where the value of T_{lower} is fixed, in our protocol the value of T_{lower} depends on the distance of the node from the last forwarder and also on the network density. When the network density is high, nodes close to the last forwarder are penalized because their T_{lower} does not start from zero preventing them to start rebroadcast. This reduces the overlapping of the time intervals of the inner nodes with those located at the limit of the communication range. In the case of low density, the value of T_{lower} decreases. This reduces the waiting time because at the same time the value of T_{upper} and the interval width are lowered. In other words, when the density is low, our protocol avoids unnecessary blocking of nodes close to the sender to forward while waiting for hypothetical farthest nodes to rebroadcast.

2) T_{upper} : T_{upper} is based on the distance to the last forwarder and the network density. The function of T_{upper} allows to extend the interval as a function of the network density. When the network density is high, unlike existing mechanisms, the nodes located at the border of the communication range have not T_{upper} equal/close to zero. In this case, our protocol results in a wider time interval, which reduces the probability that two or more nodes located at the same distance to the sender pick out the same timer value. Consequently, the number of forwarders and redundant messages are reduced, particularly in the case of dense networks. In (8), we take the integer part of the division of the distance by the number of steps ($Dist/StepLength$) to consider that the nodes close to each other as belonging to the same step. We fixed $StepLength$ to 5 m, which corresponds to the average vehicle length. The step size can be adjusted to meet the needs of specific application requirements. In some work like in [4], the step size is adjusted according to the density of the network. It is reduced when the network is dense and expanded when the density is low. In our broadcast protocol, it is the time interval which is adjusted according to the density of the network.

3) *Parameter CW_{length}* : It is the time interval width to pick out random waiting time. It depends on both the network density and the number of steps created. When the network is sparse, CW_{length} decreases to tighten the interval and therefore the waiting time is reduced. When the network is dense, CW_{length} is increased and the interval is expanded to reduce the probability that two or more nodes pick out the same value of the timer. It should be noticed that (4) and (5) result in a time interval of a fixed width for a given network density.

4) T_{max} and T_{min} : The values of T_{max} and T_{min} are fixed according to the requirements of the considered application and T_{min} may be set to any value selected according to specific requirements in the interval $[0, T_{max}]$.

V. PERFORMANCE ANALYSIS

In this section we evaluate the performance of our protocol using ns-2.34 [17] simulator. We compare its performance with Random Distance-based scheme (RDS), which is a similar scheme that does not include the density in calculating the time interval. RDS was inspired by the mechanisms presented in [5], [13], where the function that calculate T_{upper} is the one given in (1) and the value of T_{lower} is set to 0.

A. Simulation Parameters

In our evaluation, we modified MAC layer parameters available in ns-2 to make it similar to IEEE 802.11p, and we deactivated the ACK/RTS/CTS mechanisms. Mobility traces are created by a traffic simulator SUMO simulator [18]. We simulate a scenario of 4 km section of highway with 3 lanes in each direction. The length of the relevance area is 2 km and the vehicles density varies from 20 vehicles/km to 120 vehicles/km. We use six mobility traces by varying the number of vehicles and the maximum speed of vehicles to cover different speed scenarios on highways including high speed, medium speed and very low speed. The scenarios with 20 vehicles/km and 40 vehicles/km have speed ranging in [100km/h, 120km/h], the scenarios of with 60 vehicles/km

Parameter	Value
Simulation area	2 km*3lanes
Zone of Relevance	2 km
Transmission range	300m
DENM size	256 bytes
CAM size	100 bytes
T_{min}	0
T_{max}	1024
Slot time	0.5ms
Propagation model	Nakagami
Simulation time	200s
Data rate	12Mbps

TABLE I. SIMULATION PARAMETERS

and 80 vehicles/km have a speed ranging in [80 km/h, 100 km/h], the scenarios with 100 and 120 vehicles/km have a speed ranging in [20 km/h, 40 km/h]. CAMs are sent with a frequency of 10 Hz. Inspired by the standards IEEE 802.11 we set T_{upper} to 1024 slots. Clearly, the duration of the time slot impacts the performance of protocols and the delivery time increases with the increase in the duration of the time slot. However if the value of the slot is too small, the number of message collisions increases. We have tested several values of time slots (0.2 ms, 0.4 ms, 0.5 ms, 0.6 ms, and 0.8ms) and the provided results show that the value of 0.5 ms provides a better performance in term of both message delivery delay and message redundancy. Therefore, due to the space limits we report only the results with 0.5 ms time slot value. Table I summarizes the simulation parameter setting. The performance metrics to evaluate our protocol are: (i) Number of rebroadcasts which is the average number of times the message has been rebroadcast in the zone of relevance, (ii) Dissemination delay defined as the average delay time in which the message reach the farthest nodes in the zone of relevance (2 km), and (iii) Dissemination reliability, in which we measure the proportion of vehicles in the zone of relevance that receive the DENM messages.

B. Simulation results

1) *Average number of rebroadcasts*: Fig. 2 shows the result of the average number of rebroadcasts in the zone of relevance (2 km). As expected, the number of rebroadcasts in RDS increases with the increase in the network density. In fact, when the density of the network increases the distance between nodes decreases and more nodes are close each other. As we explained in section IV, in this case, it is more likely that two or more nodes with the same timer value rebroadcast at the same time. This results in an increased number of message collisions and therefore increased message rebroadcasts. As we can see in Fig. 2, our proposed protocol outperforms RDS. The number of rebroadcasts is considerably reduced particularly at 100 vehicles/km and 120 vehicles/km where the number of rebroadcasts is reduced by 50% and 57% respectively.

2) *Average delivery delay*: As we can see on the Fig. 3, the average delay of RDS starts with a high value (336 ms at 20 vehicles/km) and it decreases up to 133 ms at 60 vehicles/km. Indeed, when the network is sparse the probability of finding a forwarder at the limit of the communication range (with a timer

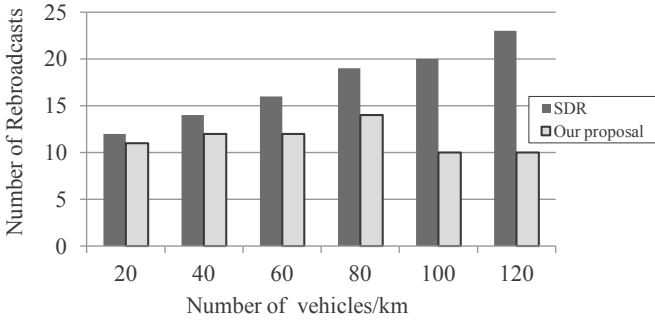


Fig. 2. Average number of rebroadcasts in 2 km vs. number of vehicles/km

value equal or near to zero) is low. So, the inner nodes close to the last forwarder have more chance to countdown to zero and rebroadcast, which increases the average dissemination delay. The decrease in the dissemination delay is due to fact that the network becomes more connected, which increases the probability of finding a forwarder at the limit of the communication range.

As we can see on Fig. 4 representing the average number of hops for a message to reach a distance of 2 km. Our proposal requires a few hops for the message to be propagated over the zone of relevance. RDS requires much more hops particularly at 100 vehicles/km and 120 vehicles/km. Indeed, in RDS as the timer is fully random, the selected node is not necessarily the farthest one. In addition, when numerous nodes located at the limit of the transmission range fail to forward (due to the messages collision), it is more likely that another inner node countdown to zero and forwards, which increases the number of hops. In our protocol the number of hops does not increase when the density of the network increases and remains at seven hops. In fact, with a transmission range of 300 m and a zone of relevance of 2 km, the lowest number of hops is seven, which corresponds to our results (cf. Fig. 4). The number of hops increases slightly (eight hops) in the case of 20 and 40 vehicles/km. This is due to the fact that the network density is low and the selected forwarders may be not at the limit of the communication range.

Fig. 3 also shows that our proposal achieves a better dissemination delay under different vehicle density values. Indeed, unlike RDS, in our proposal T_{upper} is lowered when the network density is low, to reduce the dissemination delay when the network is sparse. As we explained in section IV, the higher the density, the greater function (4) extends the time interval T_{upper} side to reduce the probability that two nodes draw the same timer value. Moreover, at a network density of 120 vehicles/km, the average delay of our proposal does not exceed 120 ms, which acceptable for safety applications.

3) *Analysis of dissemination reliability*: In the previous scenarios, the delivery ratio, which measures the proportion of vehicles in the zone of relevance that receive the DENM messages, is 100% for both protocols except when the number of nodes is 20 vehicles/km, the reception ratio is 97%. In fact, in this case the network is sparsely connected and holes may exist in the topology. As the distribution of nodes is not uniform, when the message is broadcast, no node may exist to rebroadcast it; therefore, it dies before to cover the entire dissemination distance. To explore this issue and to

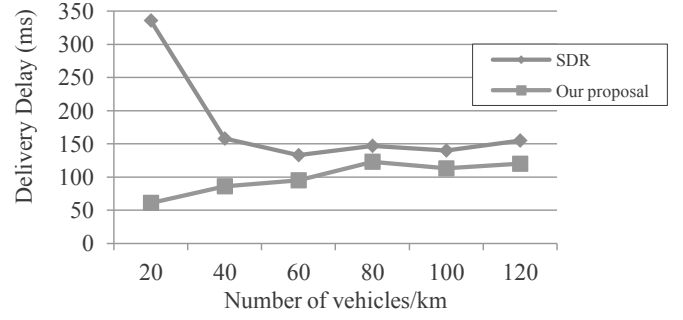


Fig. 3. Average message reception delay at 2 km vs. number of vehicles/km

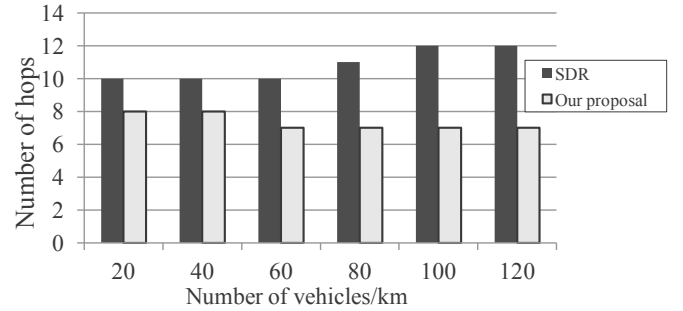


Fig. 4. Average number of hops vs. number of vehicles/km

force the simulation generating the cases where messages are not forwarded, therefore lost, we include randomness in the decision of the selected forwarder. To force the occurrence of holes, i.e. no connectivity, we simulate the same scenario under a restriction, i.e. selected forwarders rebroadcast with a probability P . If a node does not broadcast, it means that the network is disconnected and the message broadcast is not received by any node in the direction of dissemination. We vary this rebroadcast probability to measure its effect on the reception ratio of our proposed broadcast protocol and SDR. We use four mobility scenarios: 20, 40, 60, and 80 vehicles/km to covers scenarios of low and medium network density. Fig. 5 and Fig. 6 show the delivery ratio of the two protocols when varying the probability and the network density. As expected, both protocols are negatively affected by very low probability of connectivity. However, SDR is slightly better than our broadcast protocol when the network connectivity is very low (0.2). Indeed, as in the scenario with 20 vehicles/km the redundancy generated by SDR is low (cf. Fig. 2), the difference between the reception ratio of the two protocols is not significant; however, in the other scenarios, SDR generates more redundancy than our broadcast protocol which allows to provide a better reception ratio. In fact, in our broadcast protocol, the forwarder rebroadcasts the message only once and in the case of network disconnection, the message dies before covering the entire dissemination distance. Some solutions have been proposed in the literature to consider this issue [19]. For example, the implicit acknowledgment where the node rebroadcasts the message periodically until it hears the same message broadcast by another selected relay, may be included in our protocol. We will address this issue in our future work.

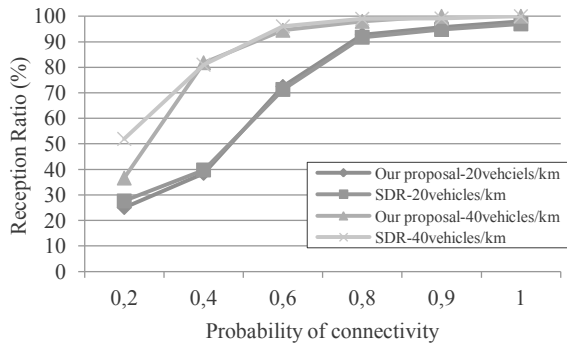


Fig. 5. Reception Ratio vs. Probability of connectivity

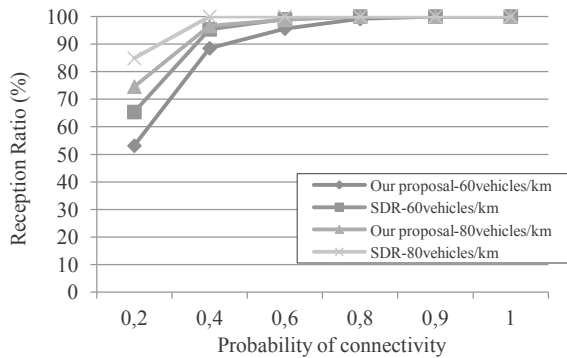


Fig. 6. Reception Ratio vs. Probability of connectivity

VI. CONCLUSION

In this paper, we propose a protocol for alert message dissemination on highways. Recent proposed protocols have a limitation with regard to dissemination delay in sparse networks and message rebroadcasts in dense networks. Our broadcast protocol tries to overcome these problems by considering the network density when calculating the waiting time. Our simulation analysis shows that, our protocol results in improvement up to 57% in message rebroadcast when the network density is high, and 82% in dissemination delay when the network is sparse.

REFERENCES

- [1] P. Papadimitratos, A. De La Fortelle, K. Evensen, R. Brignolo, and S. Cosenza, "Vehicular communication systems: Enabling technologies, applications, and future outlook on intelligent transportation," *Comm. Mag.*, vol. 47, no. 11, pp. 84–95, Nov. 2009.
- [2] G. Korkmaz, E. Ekici, F. Özgüner, and U. Özgüner, "Urban multi-hop broadcast protocol for inter-vehicle communication systems," in *Proceedings of the 1st ACM International Workshop on Vehicular Ad Hoc Networks*, New York, NY, USA, Sep. 2004, pp. 76–85.
- [3] E. Fasolo, A. Zanella, and M. Zorzi, "An effective broadcast scheme for alert message propagation in vehicular ad hoc networks," in *Proceedings of the IEEE International Conference on Communications*, Istanbul, Turkey, Jun. 2006, pp. 3960–3965.
- [4] J. F. Lee, C. S. Wang, and M. C. Chuang, "Fast and reliable emergency message dissemination mechanism in vehicular ad hoc networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference*, Sydney, Australia, 2010, pp. 1–6.
- [5] F. Hrizi, C. Bonnet, J. Harri, and F. Filali, "Adapting contention-based forwarding to urban vehicular topologies for traffic safety applications," *Annals of Telecommunications*, vol. 68, no. 5-6, pp. 267–285, Jun. 2013.

- [6] H. Jiang, H. Guo, and L. Chen, "Reliable and efficient alert message routing in vanet," in *Proceedings of the 28th International Conference on Distributed Computing Systems*, Beijing, China, Jun. 2008, pp. 186–191.
- [7] M. Slavik and I. Mahgoub, "Statistical broadcast protocol design for unreliable channels in wireless ad-hoc networks," in *Proceedings of the IEEE Global Telecommunications Conference*, Miami, FL, USA, Dec. 2010, pp. 1–5.
- [8] K. A. Hafeez, L. Zhao, Z. Liao, and B. Ngok Wah Ma, "A new broadcast protocol for vehicular ad hoc networks safety applications," in *Proceedings of the IEEE Global Telecommunications Conference*, Miami, FL, USA, Dec. 2010, pp. 1–5.
- [9] B. Bako, E. Schoch, F. Kargl, and M. Weber, "Optimized position based gossiping in vanets," in *Proceedings of the 68th IEEE Vehicular Technology Conference, VTC*, Calgary, Canada, Sep. 2008, pp. 1–5.
- [10] M. Slavik and I. Mahgoub, "Stochastic broadcast for vanet," in *Proceedings of the 7th IEEE Conference on Consumer Communications and Networking Conference*, Las Vegas, Nevada, USA, Jan. 2010, pp. 205–209.
- [11] O. K. Tonguz, N. Wisitpongphan, F. Bai, P. Mudalige, and V. Sadekar, "Broadcasting in vanet," in *Proceedings of Mobile Networking for Vehicular Environments*, Anchorage, AK, May 2007, pp. 7–12.
- [12] K. Ibrahim and C. Michele, "p-ivg: Probabilistic inter-vehicle geocast for dense vehicular networks," in *Proceedings of the IEEE 69th Vehicular Technology Conference*, Barcelona, Spain, Apr. 2009, pp. 1–5.
- [13] C. E. Palazzi, M. Rocchetti, and S. Ferretti, "An intervehicular communication architecture for safety and entertainment," *IEEE Trans. Intell. Transport. Sys.*, vol. 11, no. 1, pp. 90–99, Mar. 2010.
- [14] F. Schmidt-Eisenlohr, M. Torrent-Moreno, J. Mittag, and H. Hartenstein, "Simulation platform for inter-vehicle communications and analysis of periodic information exchange," in *Proceedings of the Fourth Annual conference on Wireless on Demand Network Systems and Services*, Obergurgl, Tyrol, Austria, Jan. 2007, pp. 50–58.
- [15] *Intelligent Transport Systems (ITS); Vehicular Communications; Basic set of Applications; Part 2: Specification of Cooperative Awareness Basic service, version 1.3.0*, ETSI TS EN 302 637-2, 2013.
- [16] *Intelligent Transport Systems (ITS); Vehicular Communications; Basic set of Applications; Part 3: Specification of Decentralized Environmental Notification Basic Service, version 1.2.0*, ETSI TS EN 302 637-3, 2013.
- [17] Network simulator—ns (version 2). [Online]. Available: <http://isi.edu/nsnam/ns/>
- [18] Sumo project. [Online]. Available: <http://sourceforge.net/projects/sumo>
- [19] A. Berradj and Z. Mammeri, "Multi-hop broadcasting in vanet for safety applications: Review and classification of protocols," *International Journal of Business Data Communications and Networking*, vol. 9, no. 4, pp. 86–104, Oct. 2013.