American Journal of Electrical and Computer Engineering
2020; 4(2): 16-34
http://www.sciencepublishinggroup.com/j/ajece
doi: 10.11648/j.ajece.20200402.11
ISSN: 2640-0480 (Print); ISSN: 2640-0502 (Online)

TDSM: Trusted Dissemination Safety Messages in a Multi-hop V2V Communication

Hanaa Sami Basheer[1, *], Carole Bassil[2], Bilal Chebaro[3]

[1]Institute of Laser for Postgraduate Studies, Photonics Unit, University of Baghdad, Baghdad, Iraq

[2]Faculty of Science II, Lebanese University, Pierre Gemayel Campus, Jdeidet, Lebanon

[3]Faculty of Science I, Lebanese University, Al-Hariri Campus, Beirut, Lebanon

Email address:
hana@ilps.uobaghdad.edu.iq (H. S. Basheer)

[*]Corresponding author

To cite this article:
Hanaa Sami Basheer, Carole Bassil, Bilal Chebaro. TDSM: Trusted Dissemination Safety Messages in a Multi-hop V2V Communication. *American Journal of Electrical and Computer Engineering*. Vol. 4, No. 2, 2020, pp. 16-34. doi: 10.11648/j.ajece.20200402.11

Received: July 1, 2020; **Accepted**: July 17, 2020; **Published**: August 13, 2020

Abstract: Broadcasting is a transmission of the same message to multiple recipients to form the communication process between wireless vehicular networks. Yet, many challenges still need to be addressed to ensure proper broadcasting. These include resolving the different security problems that affect the vehicular network efficiency while exchanging messages, and achieving low dissemination overhead and minimum broadcast delay while maintaining high delivery ratio. In this paper, we present a novel model for trusting the safety message before disseminating it, through multi-hop V2V communication. We also, ensured message high delivery rate with minimum time delay. To this end, we recommend the idea of dividing the digital roadmap of the interested area into small fixed size segments. To transmit the packets between vehicles, we depend on pairs of concatenated information composed of the beacon message and the safety message, where the sending time is arranged according to the road density. The model relies on the idea of using a forwarder to rebroadcast the safety message between segments. Choosing the best forwarder is thus based on a calculated weight value for links between the vehicles. Our contribution is achieved by adding two decentralized data trust stages: to entrust the safety message information in one-hop, and before disseminating it farther through multi-hop. The simulation results using NS2 and SUMO showed the effectiveness of the model. The two stages of the trust method are also verified.

Keywords: V2V, Message Dissemination, Data Trusting

1. Introduction

The rapid increase in transportation systems and the quick rhythm of life have contributed to the increasing trend of vehicular accidents. This trend has in turn forced the vehicles' industries to enhance the available vehicular safety tools, and to add more traffic prediction tools to alert drivers. Vehicular Ad-hoc Networks (VANET) have been developed to improve traffic safety, by supplementing different safety services to the nodes with high mobility within the network, aside to several other non-safety services that can be available when connected to the infrastructure. In VANET, the Wireless Access in Vehicular Environment (WAVE), coupled with the Dedicated Short-Range Communication (DSRC) standard, is adopted to be the de facto standard for vehicular communications. The communication between vehicles themselves, or between vehicles and the infrastructure through roadside unit (RSU) devices, can be arranged with devices such as onboard unit (OBU), sensors, antennas, and geographic position system (GPS). Several categories of applications are presented in vehicular networks, and the most comprehensive classification divides the applications into safety services, traffic management, and comfort applications [1]. VANET is mainly targeted at the fast delivery of safety messages to the intended destination and is mostly considered than the other services. Tony et al. stated that the main aim of VANET is the provision of sufficient real-time quality of services (QoS) for safety applications, while simultaneously supporting non-safety applications [2]. We are mostly concerned on how to deliver the warning messages between vehicles quickly and correctly.

Due to the dynamic vehicular networks topology, VANET

still faces many challenges. To deal with these challenges, the Immers, L. H. et al, have suggested classifying them based on the kind of application [3]. Meanwhile, for the safety applications, several factors should be ensured by the networks,. These include: using an efficient broadcast method, keeping services available, dividing the area into clusters to ensure nodes connection, and ensuring suitable call admission control to prioritize the safety messages.

Vehicular networks are exposed by different kinds of attacks, but the most challenging one is the multi-layer attack, that causes denial of services (DoS) as mentioned in [4].

In a one-hop-communication, message dissemination via link layer is initiated by flooding packets to all the nodes in the radio range of a sender. Message dissemination can also be done through a multi-hop communication, by rebroadcasting a message to reach a wider area. Transmission service in the IEEE 802.11p is unreliable, as the message is broadcasted to all the nodes in an area, with no message delivery assurance [1]. To address this problem, an idea was presented to divide a given area by number of hops (geo-casting) to bound the flooded messages, depending on the intent of the message and the scheme used [5]. For instance, the use of peer-to-peer message distributing scheme is important when monitoring an urban scenario [6], while a multi-casting scheme is ideal for message dissemination to all the neighboring nodes, even at a low node density, as an arbitrary group of nodes can be used to form a layer overlay [7].

The dissemination mechanisms are now implemented on the application layer, so that the interaction with the routing layer is limited. Message dissemination can be enhanced based on the needed requirements, which vary across the applications. Over the years, several broadcast dissemination protocols have emerged [1], such as the counter-based schemes, location-based schemes, and distance-based schemes. More schemes have also been developed, such as the farthest node scheme, probability-based scheme, cluster-based scheme [1, 8], fastest node-based scheme, carry-store-forward mechanism, push-based and pull-based mechanism [9], and time slotted multi-hop transmission protocol [10, 11]. However, while reviewing these dissemination techniques used to assure V2V communications, it is not possible to do a quantitative comparison among them, as each method evaluates its results based only on the metrics presented. Instead, we provide a comprehensive feedback to the entire system in our survey [12].

This research aims to develop a dissemination model for the safety messages delivery, which addresses the issue of message collision and the broadcast storm problem. The model thus considers the message transmission time to single nodes, and the quantity of relay nodes used. Acknowledgment (ACK) is used at the application layer to ensure warning messages propagation, since DSRC does not provide ready to send and clear to send mechanisms [1].

Recently, several considerations are overlooked by the security system in VANET, such as data consistency verification, authentication, availability, privacy, non-reputation, and real-time constraints [13]. Virtually almost all authentication and security schemes rely on some data generated from the vehicle and a center node for controlling. However, the security of VANET entities is paramount yet not enough, especially when a trusted vehicle begins to act maliciously. Therefore, there is a need to trust packet's information in order to prevent the possible manipulation of message information. Trusting the message data was first considered by Raya et al. (2007), where a data trust, rather than entities trust, was established for ephemeral ad hoc networks, in order to prevent the manipulation of information by authorized nodes [14]. The presented model is concerned in V2V communication, without the support of the infrastructure which needs a decentralized trust scheme. Thus, we started by proposing hypotheses that help in building data trusted schemes [15].

This study mainly aims to present an efficient broadcast scheme for the dissemination of trusted emergency information, through multi-hops V2V communication. We focus on the preparation of all nodes in the network during a normal situation. This allows the node to immediately select the forwarder node, that transmits the suitable warning message to be rebroadcasted farther by the forwarder. Selecting the best forwarder depends on calculated weight links value, where the sender node will choose the node with the highest weight value to be its forwarder. Then, it will wait for the ACK from the forwarder, to ensure message delivery and minimize system congestion. If the forwarder does not reply with ACK during some time, then the sender starts sending to the next best forwarder until receiving the ACK. In the proposed scheme, the forwarding node is responsible for rebroadcasting the emergency message to the next segment, immediately after trusting its data. An efficient broadcast is possible only by avoiding network problems such as a broadcast storm. Thus, in our TDSM model, we focus on entrusting the information in public warning messages during transmission, through multi-hop V2V communication. The use of encryption methods, and the privacy of the participating nodes can be with benefit, but this comes with no advantage if an authorized node behaves maliciously, by transmitting either faulty or altered message information. Hence, we adopt the idea of trusting the data of the transmitting message, to ensure the accuracy of its information during transmission. Our trusting method is of two stages: the first is through one-hop, while the second starts before rebroadcasting the message to the next hop.

This paper provides three main contributions. First, it introduces a new broadcast approach, which depends on five main steps: (1) using a pair of information packets containing the beacon and the message information <B, M>, for minimizing the number of packets sent during an emergency; (2) dividing the digital roadmap into small segments of fixed sizes; (3) arranging the nodes in an ordered manner, depending on the weight of the connected links, which is estimated at every time interval; (4) using a directional broadcast method, which implies a consideration of backward message transmission; (5) selecting the forwarder node with respect to the highest link weight value. The

forwarder node is responsible for rebroadcasting the message to the next segment, and replying with the ACK to the sender. If no ACK is received during a calculated time interval, the sender node will choose the next best forwarder, until receiving the ACK to ensure message delivery. Second, the study suggests a scheme for queueing the incoming alert messages with respect to some parameters, to start sending the message with the highest priority first. Third, based on probability, the study adds two decentralized algorithms for trusting the information of the message. The first data trust scheme is to trust/un-trust all the messages transmitted on the same segment of the road. The other trust scheme starts when the forwarder node has rebroadcasted the message to the nodes at the next road segment; it is meant to trust the forwarded message before its farther disseminated.

The proposed TDSM model was analyzed and compared against flooding broadcast. The TDSM performance was evaluated through simulation studies, which were benchmarked against a Bi-direction stable communication (BDSC) [16]. This in turn employs three-terms criteria: packets drop during active dissemination, safety messages overhead, and the rate of safety messages reachability. The evaluation studies demonstrated the benefit of depending on the link weight values for forwarder node selection.

The rest of this paper is structured as follows: a summary of the related works is reviewed in section II. Then, in section III, a discussion of the motivation, assumptions, and problem statement, together with the proposed scheme is provided. A performance evaluation for our work is illustrated in section IV. Finally, in section V, the work is concluded.

2. Related Work

In this section, we provide an overview of the related works, by considering data dissemination in V2V communication. An Efficient dissemination method depends on four aspects; selecting a set of rely nodes to rebroadcast the safety messages farther, avoiding VANET problems such as broadcast storm and hidden node problems**,** ensuring message delivery by using ACK reply, and guaranteeing the accuracy of message information.

The basic protocol dissemination concept does not support a retransmission or acknowledgment mechanism, yet both are important to achieve a high delivery rate and to reduce packet drop [4]. Proceeding in the retransmission of a message to all the nodes in the area may cause packet overhead, due to the higher rate of message transmission in the area. Thus, carefully selecting the forwarder nodes marks an efficient way to avoid message overhead [1]. Moreover, the rate of message redundancy must be at a suitable level, to keep the balance between avoiding broadcast storm and increasing message reachability [1]. Another problem in vehicular communication is the hidden node, It occurs when two messages sent from different vehicles collide when received by the same vehicle simultaneously [1]. Many proposals have been presented to address the issue of message collision, but two practical protocols are outstanding: the ready-to-broadcast/clear-to-broadcast (RTB/CTB), and the cluster-based routing protocols (CR). Regrettably, the RTB/CTB can lead to message congestion; the CR protocol addressed this by depending on the estimation of the travel speed and time. However, this protocol requires a frequent cluster leader node change, which is not ideal for use with dense traffic. Virtually, all the cluster-based approaches are ideal for routing and traffic monitoring only [17]. In 2004, a suggestion for the enhancement of the RTB/CTB was made, based on the selection of one of the recipients to do the handshaking. So, upon a successful implementation of this proposal, the system can prevent the transmission of any unnecessary message [18]. However, this can result in a situation of how to select the best recipient from the numerous nodes, which can wait for its clearness acknowledgment. In 2007, this situation was addressed by the proposal of an efficient directional broadcast (EDB) method, concluding that the only farthest receiver can forward packets in an opposite direction and reply with acknowledging. With the EDB, the recipient is meant to wait for a period before disseminating the message farther. The waiting period is determined based on the distance between nodes, where the farthest recipient is with shortest waiting time. When the sender receives the ACK, it ought to stop rebroadcasting the packet [19]. During the years many dissemination methods were suggested for V2V communications, with each having different goals.

A bi-direction stable communication protocol (BDSC) approach was proposed in 2014, for the selection of a set of qualified relay nodes from all nodes, after quantitatively estimating the quality of the links between the source and the potential relay nodes. The quality of the links is checked before selecting the relay nodes. This is done for two purposes: the prediction of the periodic link quality, and the link selection operation, which must be carried out in emergency situations. To achieve a quantitative representation of the link qualities, each node is meant to locally run the link quality estimation operation, where its cycle ends within T_{BDSC} time duration. The T_{BDSC} is individually predefined for each neighboring node, so, upon the expiration of the T_{BDSC}, there will be a replacement of the previous value of the quantitative representation with the new value, before updating the quality of the database at each node. The link selection operation is dependent on the feedback from the link quality estimation process. With this approach, the goal of reducing the alert message redundant rate is achieved, through the reduction of the number of relay nodes, as well as backward message forwarding; but the issue of messages collision persists [20].

Another way to reduce the number of relay nodes is the urban multi-hop broadcast protocol (UMBP), presented in 2015. The model was proposed for the dissemination of emergency messages via multi-hop V2V platforms. In this model, there are several assumptions for the consideration of the urban road layout, as well as for the selection of the position of the forwarding nodes. At the first hop, the bi-directional broadcast uses the forwarding node selection in each direction, before using directional broadcast in the next

hop, based on the RTS/CTS/DATA/ACK handshakes. When an emergency message is created by the sender, it must sense the Broadcast Inter-Frame Space (BIFS) idle channel before accessing the medium, whose length must satisfy the condition of equation (1):

$$T_{SIFS} < T_{BIFS} < T_{DIFS} \ldots \quad (1)$$

Where, SIFS=Short Inter-Frame Space interval, and DIFS=Distributed Inter-Frame Space interval. The model focuses on lowering delays in the transmission of emergency messages, reducing message redundancy, and preventing message collision [21].

A Simple and Efficient Adaptive Data (SEAD) dissemination scheme was proposed in 2016, to ensure a fast and reliable delivery of packets, irrespective of limited bandwidth for real-time applications. The system was proposed to ensure a reduction in the packet drop ratio, reduce end-to-end delay, and increase the rate of packet delivery. It relies on the density of vehicles and the direction of a message broadcast to calculate the rebroadcasting probability of the packet, in order to prevent broadcast storm. Probability in SEAD is inversely related to the rate of redundancy (r) calculated from equation (2). The redundancy rate (r) is updated upon the receipt of a message with time. If a message is received from the front side of a vehicle, it is processed and rebroadcasted farther. But, if the message comes from the rear, it will be taken as ACK. The waiting time W_t for a message is calculated based on the distance parameter as in equation (3); so, if (r) did not increase after the expiration of W_t, then, the message broadcast with probability P, adjusted according to vehicles' density; otherwise, the rebroadcast decision is canceled [22].

$$r = \frac{total - recieved - messages(original + dupicate)}{total - new - message(original)} \quad (2)$$

$$W_t = \left[N_t * \left(1 - \frac{\min(D_{ij}, R)}{R} \right) \right] * \delta \quad (3)$$

Where N_t=a fixed number of segments, D_{ij}=distance between the transmitter i and the receiver j, R=average transition rate, and TM=hop delay (comprising of the medium access delay and propagation delay). It must be more than 1.

A moving zone-based routing protocol (MoZo) model was proposed in 2017. The MoZo uses connected vehicles to generate a dynamic moving zone, which facilitates data transmission. A vehicle is selected from each zone to serve as a lead vehicle, responsible for the management of the other vehicles and messages. The roads are first converted into a graph, where they act as the edges, while the intersections act as the vertexes. Each road segment is denoted as r (start point "st", endpoint "ed"); so, if the direction of a vehicle TMis toward the ed, then TM=1, otherwise TM=-1The lead vehicle knows the distance of each vehicle (l_u) from the starting point when they exchange the hello packets, as well as their speed (v) at time (t_u). This assists the lead vehicle to estimate the position of each vehicle using equation (4) at the timestamp (t).

$$l(t) = l_u + \delta \bullet v \bullet (t - t_u) \quad (4)$$

The model deploys 2 simple data structures to disseminate the message properly and to maintain zones. These structures include the combination of the location and velocity tree (CLV-tree), and the leaving event queue, which stores the estimated time stamps in an increasing trend. The queue is updated as soon as a new vehicle enters the zone, or when it sends a message to the lead vehicle. This model ensures a high rate of message delivery and minimizes communication overload. However, the idea is dependent on the information of the members and on the perceived future direction. This is considered a weakness due to the vehicles' privacy, despite its dependence on fewer vehicles' location information [23].

3. Trusted Dissemination Safety Messages (TDSM)

a) General considerations

While developing our solution, the main concern was how to disseminate safety messages between moving vehicles, if they were out of the infrastructure coverage. In fact, when a vehicle senses an abnormal situation on the road, it is highly important to notify other vehicles about this situation, to be aware of it on time and take appropriate actions. To increase the reliability of the safety message, we add a decentralized trusted dissemination scheme. Our framework tests the proposed scheme on an area in between two RSUs, where this area is divided into fixed length segments. The main problems that the TDSM model tackles are: the hidden node problem, the broadcast storm, and the reliability of the message while the rebroadcasting it, through trusting its information.

In order to achieve our goals in such a context, we identify four main needs, which are:

1) Dividing the road into fixed small segments: the segmentation helps to avoid the difference in vehicles radio ranges, and to reduce message collision because of the hidden node problem, by accomplishing a full node connections at each segment.
2) Assigning a forwarder node within each segment: the choosing node ensures message delivery by sending an ACK reply to the source node, only to minimize the number of transmitted messages. The forwarder is also responsible of trusting the information of the safety message before rebroadcasting it to the next segment.
3) Queuing the receiving messages by the nodes based on message's priority: since any node can process one message at a time, priority attributes are added to the receiving message before queuing, so the recipient node will process the message with the highest priority first.
4) Data trusting of a message: even with the use of the security methods based on IEEE 1609.2 standard, there will still be a need to trust safety messages, in order to avoid authorized vehicles from acting maliciously. A

trust scheme with two methods is used in TDSM.

b) Model assumptions

The assumptions presented in this section are used to achieve the requirements that were defined in the previous section to build TDSM. We consider the following four assumptions:

1) The logical segmentation of the road, supported by the digital GPS of the area map, installed in the on-board vehicle device (OBU).
2) The assignment of a forwarder node for a vehicle v_i within the segment, where an equation is used to calculate the link weight value for every connected vehicle $v_{j\ (j\neq i)}$ placed in behind v_i. The weight value is calculated at each vehicle in the network depending on three different parameters; the distance from the behind node v_j, the behind node v_j speed, and the number of connected nodes to v_j. Each node saved a table of nodes IDs associated with the calculated weight value, which are sorted in an ascending order accordingly. When a node senses an abnormal event, a node from the table with the highest weight value is assigned immediately as the best forwarder.
3) The possibility of creation of different safety warning messages by different vehicles in an abnormal situation. A pre-defined priority value should be added in a new field within every warning message, to help the recipient nodes in enqueuing the incoming message, based on its priority. The sending node stops its broadcast once it receives an acknowledgment from the best forwarder within the segment.
4) The data trust in the transmission of the safety message. Our decentralized trust scheme is divided into two methods; the first method is for a single hop, and the second method involves trusting the safety message by the forwarder before rebroadcasting to the next segment.

c) Problem statement

Normally, the dissemination of warning messages is initiated when a vehicle predicts an abnormal situation. It immediately generates a relative event message, to be propagated to its neighboring vehicles at the same segment. In our work, we rely on a directional broadcasting (i.e. considering the backward transmission only), though the source node will choose from its saved table a vehicle to be the forwarder. The forwarder is responsible for replying with an acknowledgment to the source node, as well as for rebroadcasting the warning message farther. Depending on one forwarder for each source is not ideal, due to the hidden node problem, possible connection gap, and the prior assumption that every node have the ability to arrange all the 'behind nodes' in the stored table, in an decreasing order, depending on the link weight values. This arrangement helps the source node in choosing the next best forwarder on need.

To overcome the limitation of transmitting the safety messages to the neighboring nodes in the same segment only, we give the forwarder the ability to connect to the nodes placed at the first few meters from the Next Hop Segment (NHF). This helps the forwarder to rebroadcast the warning messages farther to the next segment. The transmitting procedure will continue through multi-hops, in order to serve vehicles in a wide area with the warning information. It can also be saved in the cloud database when an infrastructure is available. Moreover, a priority value is added to each message type, to reduce the time delay and to increase the reachability rate of the warning messages. Then, the nodes can start to first send the most urgent message, with the highest priority value.

We adapted the idea of using a set of valid pseudonym identifiers (VIDs) from [14] to ensure the protection of a vehicle's privacy. Yet, this does not stop an authorized vehicle from behaving abnormally without being noticed, due to its ability to change identity with time. The connected vehicles must be protected against insider nodes that have malicious activities, as well as from the outside attackers. Warning messages contain public information, so the use of a trusted scheme is necessary to secure message information. For that reason, we added a decentralized data trust method to the safety message, prior to its dissemination in one segment, and another trust method before rebroadcasting it farther. The data trusting scheme helps the vehicles in deciding whether to rebroadcast the message farther to the next segment or to discard it and notify it as a misbehavior authority.

d) Model definition

TDSM considers the propagation of a warning message between vehicles, even when no infrastructure is available. The distance (Dv) in the area of figure 1 is without internet coverage, where all vehicles are out of the radio range of the existing RSUs. We take a highway as our work platform, with three one-direction lanes. The digital map is logically divided into small fixed size segments, where we use 350m for each segment. This segment size was chosen to neglect the differences in vehicles radio range. This helped in getting almost a fall connection between the vehicles at the same segment, and forming a 1-hop network as shown in figure 2. Each segment has a unique identifier SID, where every vehicle belongs to one segment only, according to its GPS position. In a normal situation, the messages are frequently exchanged between connected vehicles in the same segment at every time interval, carrying their status to update each vehicle table with the behind vehicles data.

In TDSM model, whenever a vehicle joins a segment then it will start exchanging hello packet with the neighboring nodes. In addition, it starts calculating the weight of every connected link with the vehicles in behind, and saves them in ascending order. This arrangement helps the node to face a sudden abnormal event by selecting the best forwarder immediately.

The Next Hop Forwarder (NHF) zone is supposed to be with a maximum length of 50m. We assume that the nodes belonging to the NHF zone of any segment have the ability to receive a safety message, which is rebroadcasted by a forwarder from the front segment. These nodes placed at the NHF zone deploy TDSM to continue with the dissemination procedure, until reaching the targeted destination.

e) TDSM message structure

In recent delay-based approaches, the Distributed Optimized Time (DOT) relies on beaconing to provide neighboring data. DOT assumes a maximum beacon size of 324 bytes, to be transmitted correctly in density traffic, where an increase in beacon size causes messages overhead [24]. Worth to note, the beacon size does not have a predefined standard value, where this value can be assumed depending on the protocol requirement.

Standard beacon header field is as shown in Figure 3, where it is consisted of the fields of common header only [25].

It is a concern that warning messages propagate with a lesser time delay, in order to warn drivers on time, while maintaining suitable redundancy rate to avoid message collision.

In an abnormal situation, the created warning messages are transmitted between neighboring vehicles, with the vehicle status beacons. Thus, the large amount of messages can cause a broadcast storm problem.

Therefore, our suggestion is to create packets consisting of pair of information; the beacon information (*B*) and the data of the warning message (*M*). A packet is thus formed as (<*B, M*>). In normal situation (*M*), the packet will remain empty, until an abnormal event is sensed and a safety message is created to fill this part.

Our suggestion for the beacon field (*B*) is the vehicle operational status, as well as network topology as shown in Figure 4. The warning message field (*M*) is to include the message details, beside the forwarder identity *(F_ID*) of the node that must reply with *ACK,* as shown in Figure 5. This will be discussed in details in the TDSM principle section.

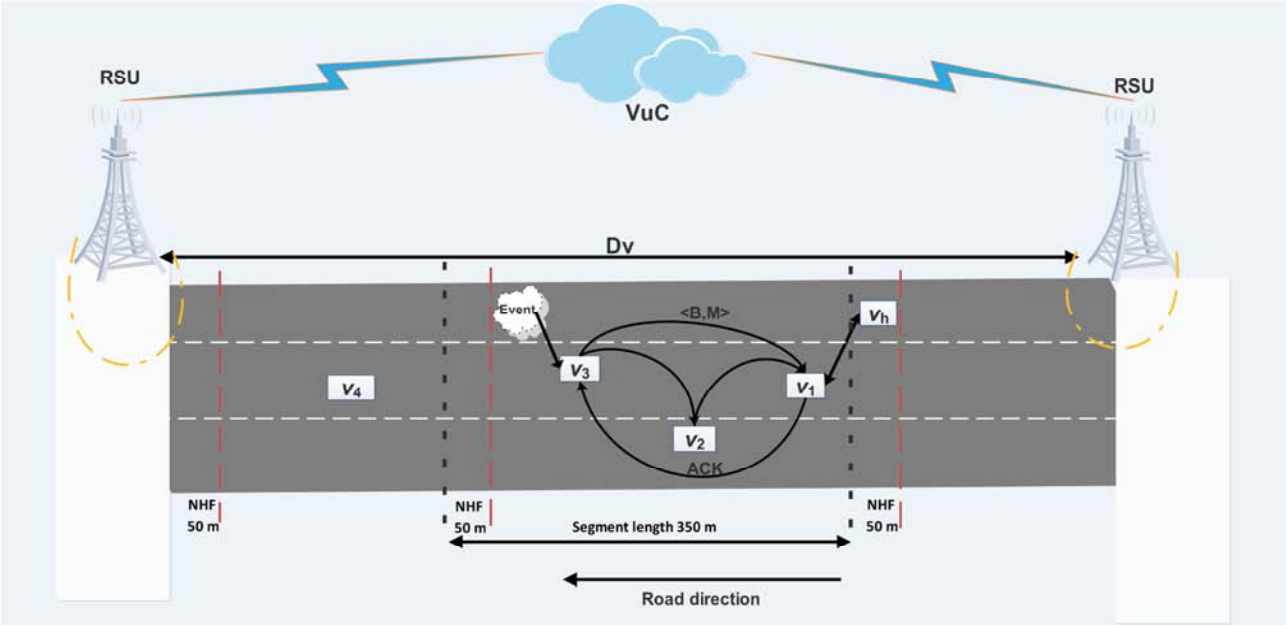

Figure 1. *Depicting road distance without infrastructure within a coverage area (Dv).*

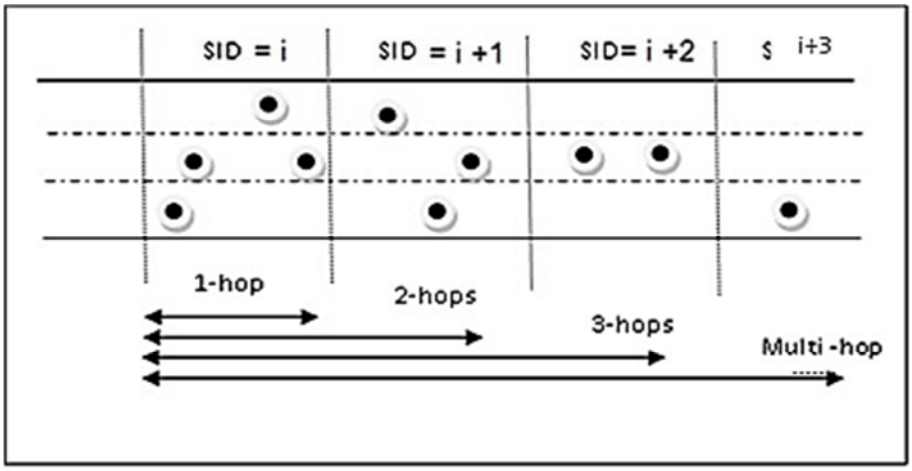

Figure 2. *The distribution of vehicles on each segment as one hop.*

Field #	Field name	Octet/bit position		Type	Unit	Description
		First	Last			
1	Common header	Octet 0	Octet 35	Common header	n/a	Common header

Figure 3. *Information of the beacon packet header field.*

<(ID, time, position, direction, velocity, counter, SID), *M*>

Figure 4. *Beacon fields (B), paired with empty message (M).*

The beacon fields are illustrated as below:

1) ID: it refers to the node's unique identity. We find that the suitable way to protect the vehicle's privacy is by using a set of valid pseudonym identifiers (*VID*s) as mentioned in [26]. These can be updated each time the vehicle undergoes periodic maintenance.
2) Time: this field indicates the starting time of the hello packet <*B, M*>, and is updated at each sending time.
3) Position: this field indicates the vehicle's position at the sending time based on its (GPS).
4) Direction: it refers to the message's transmit direction, which can help a vehicle in storing the behind vehicles information.
5) Velocity: it refers to the vehicle's speed.
6) Counter: this field stores the number of the connected neighboring nodes at the sending time.
7) SID: it refers to the road's segment unique identity that the node belongs currently to.

Warning info.				
Tag	Time stamp	F_ID	Code	Priority

Figure 5. *Message M fields.*

a) The Proposed TDSM Schema

In this section, we discuss the TDSM scheme in two stages; (i) Oh-TDSM to handle the dissemination of the alert message in the same segment; and (ii) Mh-TDSM to handle the rebroadcast of the alert message to the next segment. These two stages continue running until reaching the target destination, through multi-hop communication. The desired target is to reach one of the existing RSUs, in order to update the traffic cloud database with the situation.

There are 4 main phases in the Oh-TDSM stage. Phase 1 is the preparation step, which is initiated in each node soon as the node joins the network. Thus, the node is ready to react immediately when there is a need. This phase is processed at every time interval during normal road situations. Phase 2 starts when a node senses an abnormal event. This phase involves the creation and propagation of a suitable warning message to the neighboring nodes in the same road segment. Phases 3 and 4 depict how we improve the forwarder processing, by placing priority on each message. This helps to enqueue the incoming messages and to trust their data, before rebroadcasting it farther using a decentralized data trust scheme.

TDSM principles:

In a normal situation, the neighboring nodes in the same road segment exchange their status by sending and receiving <*B, M*> messages every interval of time, where *M* stays empty. Each vehicle placed in front of others, creates an ordered table containing all the connected vehicles' ID in behind. These are organized according to their calculated link weights, starting from the vehicle with the highest link weight value.

Every node positioned in front of others is responsible to do weight (*W*) calculation for the backward connecting link. This calculation is based on three parameters that affect node's action, which are: the distance from the behind node (*d*), the behind node speed (*v*), and the number of nodes connected to it (*c*). We assign for each parameter a score based on its current value as shown in table 1, where the scores' values are suggested during implementation.

Table 1. *chosen score for the three parameters.*

Distance (meters)	S_d	No. of connected nodes	S_c	Velocity (km/hour)	S_v
$200 < d$	10	$50 \leq c$	10	$90 \leq v$	5
$125 \leq d < 200$	7	$25 \leq c < 50$	7	$60 \leq v < 90$	8
$50 \leq d < 125$	3	$c < 25$	2	$v < 60$	2
$d < 50$	1	$c=1$	1	$v=0$	1

Equation (5) represents the final weight value W_i for the connected links to the node v_i at time T_B.

$$W_i = S_{id} + S_{ic} + S_{iv} \quad (5)$$

We take into consideration that the far node location (*d*) is more appropriate for disseminating the messages, as well as a node with a high speed (*v*) and high number of connected nodes (*c*). This is in light of our concern to rebroadcast the

message to a wide area quickly, and to avoid chain accident, through delivering the warning to a large number of vehicles in the segment to take the right action.

For more explanation, the node placed in front such as (*V'*) in Figure 6 will save an ordered table of the behind connected neighboring nodes, placed at the same segment (SID). The node with the highest link weight value is considered as the best forwarder (e.g. *F1* in Figure 6). Vehicle (*V'*) is connected to vehicle (*X*), but (*V'*) does not calculate or store the link weight between them, since they are from different segments. Node (*F1*) is the only node allowed to exchange warning message with node (*X*), (where (*F1*) is with tag=1 to indicate that it is a forwarder, and (*X*) is placed at NHF zone). In this way, we manage to achieve an arrangement for all neighboring nodes, in order to be ready to use the best forwarder for immediate rebroadcasting, whenever an abnormal event occurs.

In an abnormal situation, nodes that sense the event start deploying TDSM through the immediate creation of the warning message. The nodes initiate their sending time T_B and set their tags to be equal to 1, in order to indicate that they are the source of the warning message. If any of the source nodes receive the same warning message from nodes placed in front of them, it will set its tags to zero again, and act as a relay node only. Each source node sets the *F_ID* field with the node ID of the highest weight value, and computes an acknowledgment time (T_{ACK}), to bind the waiting time for an acknowledgment (*ACK*) from the chosen forwarder. If the time T_{ACK} expires and no *ACK* is received yet, then the source node retransmits the warning message and repeats all the procedure again, using the next best forwarder chosen from the stored table. When the forwarder receives the message and reply with an *AC*K, it starts entrusting the message data before rebroadcasting it farther. More details will be shown regarding this in the following subsections.

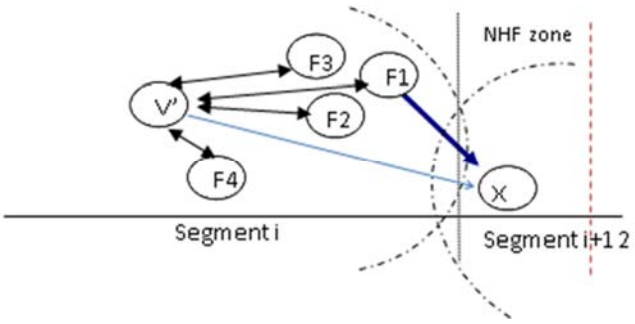

Figure 6. *Front node V calculates the link weight for all the rear nodes on the same segment.*

Oh-TDSM version 1

This version presents the preparation step in each node. It starts at the moment the node joins the vehicular wireless network by exchanging <*B, M*> messages. This message exchange is done at every time interval T_B, to keep the data in the neighboring vehicles updated during normal road situation, where *M* remains empty.

Every node positioned in front of others is responsible to do the weight calculation (*W*) for the backward connecting link, based on equation (5). Every foreground node, in turn, assigns a fixed time interval T_B after sending each <B, M> packet, where T_B ranges between 1sec and 5sec, given that the average link lifetime between two vehicles is a few seconds to 10 s [1]. Exchanging a packet between two nodes can be simply explained by the sketch in Figure 7, where the front vehicle (S') transmits its hello packet in all directions (step 1). However, the only reply that should be considered is the one coming from nodes positioned behind, because we are considering the backward mode (step 2).

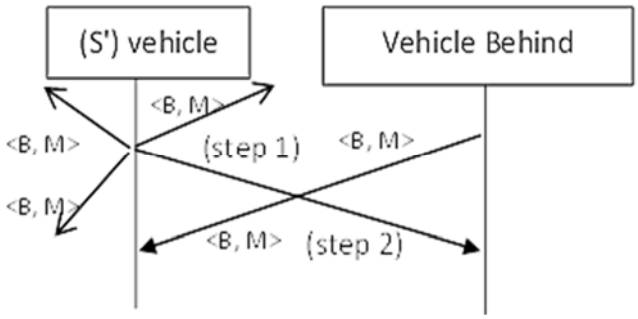

Figure 7. *Packet exchange between neighboring nodes every T_B.*

As we mentioned before, this process is repeated at every T_B, to update the stored nodes status table in each node, because vehicles' status are rapidly changed due to their movement.

Oh-TDSM version 2

When an abnormal situation appears, such as a chain accident, the nearest nodes that sense this event become the source nodes. They create the warning message (*M*), where source-tag within the warning message will be set to 1**.** These nodes start adding message code with its priority value as illustrated in table 2, which we adopt a few types from our previous study [12]. The alert message fields, shown in Figure 5, are filled with the proper information, where the Time stamp refers to the sending time and do not change. Each source node then assigns a chosen forwarder (*F_ID*) from the saved table. If any source node receives a warning message from a node placed in front, it will reset its source-tag field to zero, and stop acting as a source node during the second retransmit round. This adjustment helps in reducing *ACK* replays for the whole network, by reducing the source nodes number. The recipient nodes arrange the incoming messages in a queue, and the forwarder nodes reply with *ACK* to its source node.

Table 2. *Ordered Message Priority.*

Code	Priority	Message type
001	1	Cooperative Collision Warning
002	2	Electronic Emergency brake light
003	3	Post-crash notification
004	4	Slow vehicle advisor
005	5	Cooperative violation warning

The warning message validation (T) at the recipient nodes can be checked in comparison with the received time T_R at the recipient node, as in Figure 8, where the Time stamp is

one of the message fields.

The clocks are supposed to be synchronized and the considered time is the Universal Coordinated Time (UCT) to validate the message validation test. Note that T_p is the propagation time.

```
Case received_message:
        Check T = Time stamp + T_P
         If T_R and T are in the adequate time
        range
          then <B,M> is valid;
                  else
                   ignore message <B,M>;
Break;
```

Figure 8. *Message validation.*

Moreover, the recipient nodes continue checking the *source_tag* field and *SID* field for every incoming message. If the *source_tag* is equal to 1 and *SID* is equal to the recipient node *SID*, then the message is generated by the source node that sensed the event within their segment. The message, thus, has precedence in the queuing and is to be sent before any other message that has the same priority value.

When the forwarder receives a message *M*, it replies with an ACK immediately. If the *ACK* is received from the forwarder *F_ID* within T_{ACK} time duration, the source node will stop sending (*M*). Otherwise, upon the expiration of T_{ACK} with no *ACK* receipt, the source node uses the next best forwarder ID and replaces it in the *F_ID* field. It also initiates T_B, so that the source node starts a new round of transmission, by retransmitting the message again. This process is repeated until the source node receives an *ACK* from the forwarder to ensure message delivery.

The source node computes the period (T_{ACK}) waiting for the *ACK* using equation (6), when sending the first alert message simultaneously.

$$T_{ACK} = \left(1 + \frac{k}{k_{\max}}\right) \times 2 \times \frac{d}{v} \tag{6}$$

Where *k* represents the number of vehicles connected to the forwarder, *d* is the distance from the source node to the forwarder node, *v* is the propagation speed, and $K_{\max}$ is the maximum vehicle number any segment can have.

When the source node stops sending the warning messages, then, *M* will be empty in <*B, M*> again. At that point, the forwarder starts enqueueing the incoming messages, and begins the trust scheme to check the message data before rebroadcasting it to the next hop.

Oh-TDSM version 3

In an abnormal situation, the nodes generate multiple messages to create an alert to the event. In our model, we focus on the safety messages' delivery time delay and the messages collision. These problems are minimized using two methods; (i) checking some fields from <*B, M*> to neglect duplicate copies of the same message, and (ii) queuing the incoming messages to be organized based on priority proving, since a node can deal with only one message at a time. Queuing upon priority helps place the message with the highest priority to be sent first.

We classify the warning messages according to their priority values as per Table 2. The fields of <*B, M*> that are used for checking are *Time stamp*, *SID*, *source_tag*, *message serial number*, *code*, and *priority*. The recipient nodes check all of the below factors: the message validation as shown before (i.e. Figure 8), the segment (*SID*) of the sending node, and whether the message is coming from the source node directly (*source_tag*=1), in order to queue the incoming messages according to their priority.

In this part of our algorithm, we have analyzed four possible cases of the received messages coming from different nodes, assuming that all are valid nodes. The messages are checked based on *SID*, *source_tag*, and *priority* fields as per the following:

1) The messages coming from different segments are discarded.
2) When receiving different copies of the same message (same serial number), the receiver keeps the priority to the one with *source_tag* field equal to 1.
3) When receiving different messages (having a different serial number), but having the same priority values (having the same message type), the recipient node deals with the message that comes from the closest node.
4) When receiving different message types with different priority values from nodes with *source_tag* field equal to 1, the receiving node processes the message that has the highest priority value.
5) Note that in version 3, we define a new data structure, which is a list containing the information related to every message already processed to avoid re-queuing them.

In order to take in account, the mentioned problem, we consider 2 cooperation processes running in each node:

1) The first one is responsible of receiving and queuing safety messages;
2) The second one is responsible of processing the received safety messages;

Oh-TDSM version 4

There is still a weak point to be addressed, that occurs when an authorized participant behaves as a malicious node and attacks the warning messages. This has led to the extension of our work to incorporate a trusted data scheme, aiming to entrust the data by every node before disseminating the message. The trust scheme begins soon as a new valid safety message is received; it is based on the Bayesian network (BN) and we named it Bayesian Trust Scheme (BTS). Our main idea of BTS, presented in a previous work [27], is based on using local observation of different variables to the extent of the probability of an abnormal situation. This probability is used to decide whether to trust the received warning message and thus to rebroadcast it farther using our dissemination technique, or to discard it and consequently notify the misbehavior authority. We have used

four local variables and got a high priority of trusting the message data in the cases where 2-3 variables out of 4 variables are true. To note, depending on a high number of local variables allows having highly reliable decisions of trusting message information.

Mh -TDSM

In order to transmit the warning message to the next segment (next hop), and for more data reliability, we enhance our TDSM at the level of the forwarder by adding another trusting scheme. This is based on the endorsement of the incoming message, by the vehicles that send the same message. We name our data trust method the "Endorsing Trust Scheme" (ETS).

When a forwarder receives a message, the trusting scheme ETS starts to trust its data before rebroadcasting it farther. To explain how ETS is implemented, we present it as in the case of the forwarder (F_ID) (e.g., v_1 in Figure 1), who will reply to the source node (e.g., v_3 in Figure 1) immediately with ACK, and then starts ETS. We define a counter (C) referring to the number of directly connected neighboring vehicles in every node. Another counter (E) is defined to count the number of nodes sending the same warning message. To note, the node retransmitting the same warning message is counted once, so the forwarder increases its counter (E) only once for the same node. If (E) is less than the actual connected neighboring vehicles number (C), then, the node chooses a random number (N) which is bounded by $\frac{C}{2} < N < C$ to start making a decision according to three possible cases:

1) If E ≥ N, then the message information is promising and can be trusted to be retransmitted or rebroadcasted farther to the next hop.
2) If $\frac{N}{2} < E < N$, then the node must decide based on a binomial distribution, as in equation 7 [28], either to trust the message or discard it and notify the misbehavior authority.

$$P(E) = \sum_{E=\frac{N}{2}+1}^{N-1} \binom{N}{E} \times P^E \times (1-P)^{(N-E)} \quad (7)$$

3) If, $E = \frac{N}{2}$ then the node does not trust the incoming warning message but rather discards it and notifies the misbehavior authority.

Soon as the forwarder trusts the alert message M with ETS and BTS, it immediately changes its source_tag field to 1. Consequently, it acts as a source node and rebroadcasts the message farther to the next segment. The nodes placed in the NHF zone of the next segment must trust this incoming message before approval to transmit it farther.

As mentioned before, we give the nodes placed at the first few meters (NHF) zone the ability to exchange warning messages with nodes from the front segment, whose source_tag equals to 1. To further elaborate on this suggestion, the node v_i in Figure 9 is placed in the NHF zone, where it exchanges information with the nodes from the segment ahead. If the packets have a warning message, then *vi* immediately starts BTS to take a decision, either to trust the message and retransmit it farther using our dissemination technique or to neglect it.

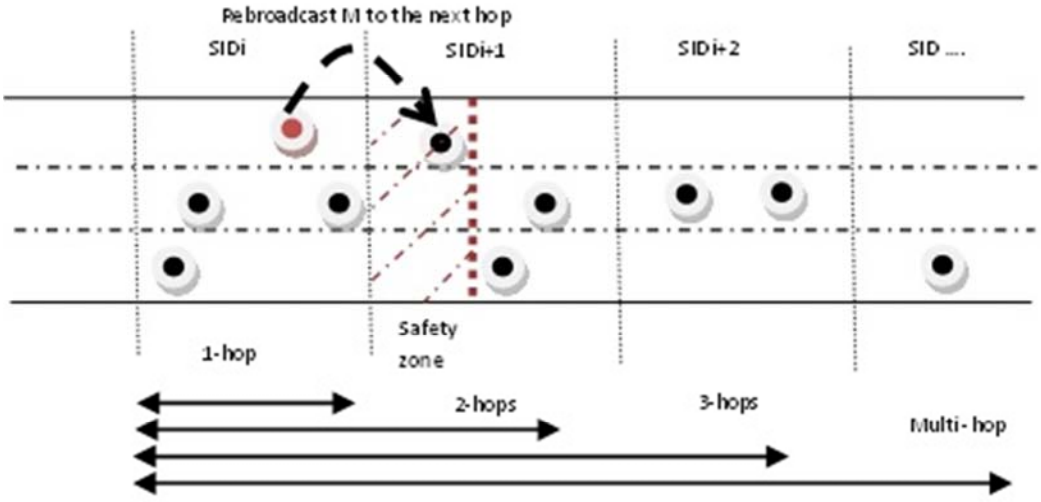

Figure 9. *The forwarder rebroadcast the warning message to the next hop (i.e. segment).*

4. Simulation Results and Performance Evaluation

The performance of the TDSM has been evaluated on an NS2 simulator [29]. This simulator can effectively address the intended simulation requirements in this study, as it supports different road traffic densities where the efficient of the TDSM can be evaluated.

C++ is employed for the implementation of the detailed protocol during this implementation. The efficiency of the runtime is important as it contributes to the building of a suitable scenario for the proposed TDSM. Furthermore, the SUMO traffic simulator was used to generate the highway traffic flow [30]. These software programs are operated in an Ubuntu 12.4 operating system. The efficiency of the proposed TDSM method has been benchmarked against the Bi-Direction Stable

Communication (BDSC) method, in terms of the usage of weighted links between vehicles. BDSC depends on a set of qualified relay nodes selected from the pool of nodes, based on the quantitative determination of the link qualities between the source and potential relay nodes. Link quality estimation operation for each node is run locally and its cycle ends within the T_{BDSC} duration [20].

The next subsection describes the simulation scenario deployed in this study. In addition, the performance metrics are defined and the simulation results are analyzed afterwards.

4.1. Simulation Scenario

Our simulation scenario is built as a 3-lane highway of 1500m length. We consider a road with length of 1.5 km that is without infrastructure coverage, as the area D_v that shown in Figure 1. During the test phase, we have monitored the transmitted beacons and coded the road segments. Later, an abnormal situation has been introduced to evaluate the dissemination of the warning message and the priority-based message queuing. To build an adequate scenario, the physical and MAC layers were set via IEEE 802.11p implementation. The simulation parameters are presented in Table 3, which shows that we used a GPSR routing protocol [31, 32].

We have selected different road densities for each segment (of 350m length) to represent different road traffic conditions. The traffic density is measured in Vehicle/m. Densities of 50, 60 and 70 vehicles are classified as sparse traffic scenario, while densities above 70 to 90 vehicles are classified as medium scenario. A scenario of 100 vehicles represents a high traffic density.

Table 3. *Simulation setting.*

Simulator	Network Simulator (NS) 2.35
Number of vehicles	50, 60, 70, 80, 90, 100
Area	1500m x 1500m
Communication range	Vehicles-350m
Segment size	350m
Number of lanes	3 (unidirectional)
Mac type	IEEE 802.11p
Simulation time	150 seconds

The prototype of the proposed simulation approach was designed based on the following assumptions:

1) Assuming a virtual accident spot point.

2) Message broadcasting depending on one priority value, and different priority values using priority queue, which are maintained in all the nodes.

3) Dealing with the messages that are transmitted backwards from the same roadside only.

4.2. Performance Criteria of TDSM

Our main objective is to disseminate a trusted warning message between road segments, until it reaches its destination, which is the RSU, to update the cloud of road traffic database. This objective is based on calculating the weight link values to choose the best forwarder, using ACK to ensure delivery, before rebroadcasting the warning message farther. Road segmentation, directional broadcasting, and changing dissemination time, are all factors used to preserve a reasonable data redundancy rate, what would result in a high delivery ratio and a low packets drop. We will describe in the coming subsections the performance criteria of message dissemination, in the context of our simulation. We will also evaluate TDSM's performance by comparing it with an existing method (BDSC).

4.2.1. Data Message Redundancy Rate (RR)

Redundancy rate is the counted number of the same data message transmitted by a vehicle during a time interval. A low redundancy rate can cause message delivery failure, while a high redundancy rate can cause a broadcast storm problem. In the TDSM, we depend on the retransmitting mechanism to ensure message dissemination. The source node continues to retransmit the message backward until it receives ACK from the forwarder. Although a retransmission process increases the redundancy rate of transmitted messages, we have limited this procedure with time interval (T_{ACK}). Figure 10 shows a case of transmitting safety messages in one segment of 350 meters with different densities. When there are only 50 vehicles, the source node broadcasts the warning message with RR=2 while waiting for ACK (T_{ACK}). When the number of vehicles increases to more than 90 vehicles, the RR also increases. Moreover, if the source node needs to execute a second transmission cycle and there is another source node placed behind, then, this node stops acting as a source node and changes to be a relay node.

Figure 11 represents a TDSM scenario where a source node is disseminating a warning message through a wide area that is divided into segments. A point is taken on the Figure at the time the forwarder replies with ACK, in order to indicate the end of the source resending the warning message at that specific segment. It is noteworthy that even when the number of vehicles keeps increasing within one segment, the RR for the warning message is still within a reasonable value, in reference to flooding broadcast, as shown in Figure 12. This is due to targeting the rear nodes in the same segment, and to stopping the transmission when the source node receives ACK or when the T_{ACK} has expired.

Figure 12 shows that the increase of packets number within one segment is limited, due to the limitation of the resending time. If the T_{ACK} has expired, then, a new resending round starts, after initiating the sending time T_B within the same segment.

4.2.2. Sending Time T_B

The sending time of a beacon T_B is fixed but can be changed according to the distance between the source node and the nodes behind, to avoid re-sending at the same time when the network density is high. Figure 13 shows how the setting of T_B can be changed with respect to the distance from the neighboring vehicles, as in equation 8. In this equation, T_D is a constant factor that corresponds to the time interval considered by GPSR routing protocol, added to the equation that is equal to 0.1 sec. An average distance from the neighbors (*Avg_dis_neighbors*) divided by the segment length (*Seg_length*) is computed and used.

$$T_B = T_B + \left(T_D \times \frac{Avg\_dis\_nieghbors}{Seg\_length} \right) \quad (8)$$

4.2.3. Safety Message Packets Drop (PD)

Packet drop refers to the total number of packets that are either dropped or non-delivered during the transmission of safety messages. Normally, data dissemination methods must achieve almost 100% data delivery efficiency. The rate of *(PD)* is inversely proportional to the delivery data rate, although rebroadcasting the received messages has a different time delay. This is due to scheduling the message at the recipient node, which depends on the number of data packets received. In a VANET, a node may be surrounded by up to 100 neighbors, where such a situation may cause network congestion and heavy collision during packet transmission. TDSM overcomes the problem of safety message drop by using the backward direction transmission technique, the road segmentation basis, and the dependency on the vehicles average distance to calculate a different resending time T_B,. This is a clear advantage over the BDSC approach, where the rate of dropped packets increases in line with the number of vehicles. The behavior of the two approaches is illustrated in Figure 14, where TDSM simulation results show no packet drop over the whole network, irrespective of the traffic density, since it uses the backward retransmission technique. On the contrary, the BDSC shows a high rate of packets drop in dense traffic scenario.

In TDSM, we avoid safety messages, what leads to an increase in the number of packets over the whole network as the number of vehicles increases, and in turn causes network overhead. The increase in the number of receiving packets increases the time delay, as in the message queue each node can deal with only one message at a time.

4.2.4. Data Message Overhead (MO)

Message overhead is the ratio of the total number of packets generated to deliver a safety message (P_{delv}) to the total number of vehicles intended to be reached (V_{behind}). The safety message is intended to be transmitted to the vehicles behind the source vehicle as assumed in our approach.

$$MO = \frac{P_{delv}}{V_{behind}} \quad (9)$$

The simulation shows promising message dissemination over the network, yet it has a negative impact on the system overhead.

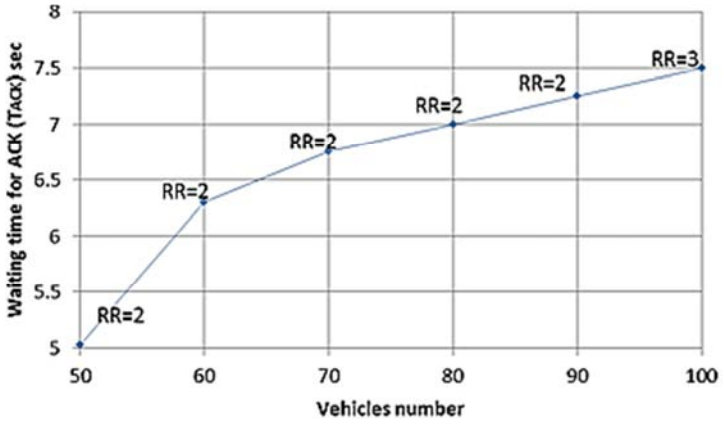

Figure 10. *Relation between (RR) and road density.*

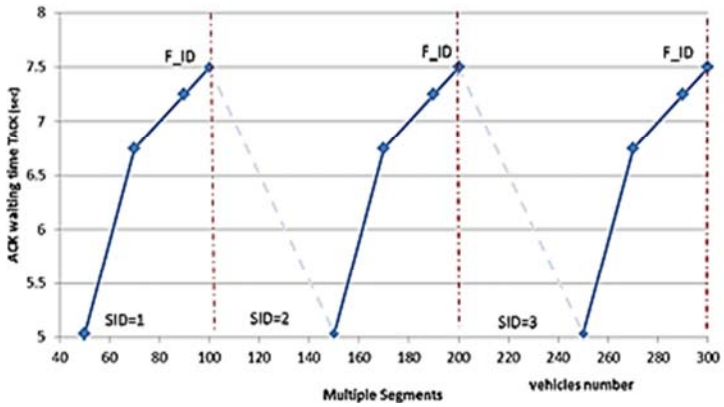

Figure 11. *Safety message redundancy rate in multiple segments.*

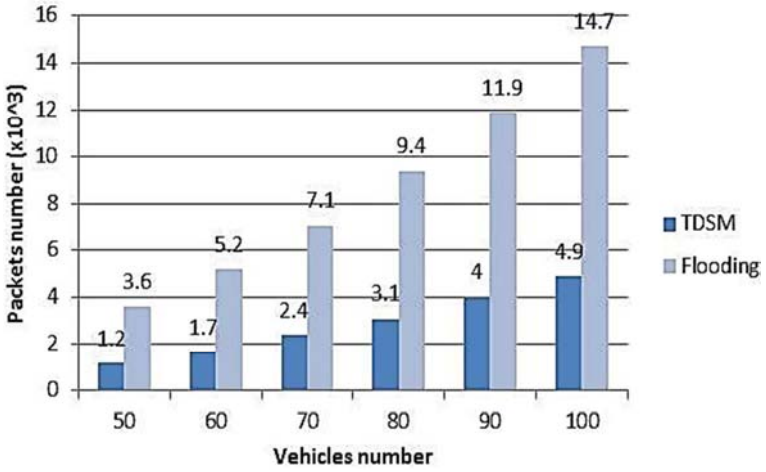

Figure 12. *Transmitting packets number.*

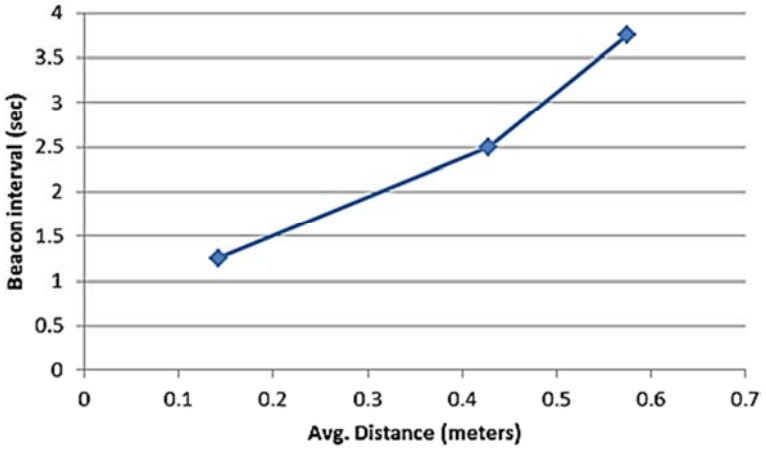

Figure 13. *Setting beacon period time according to distance.*

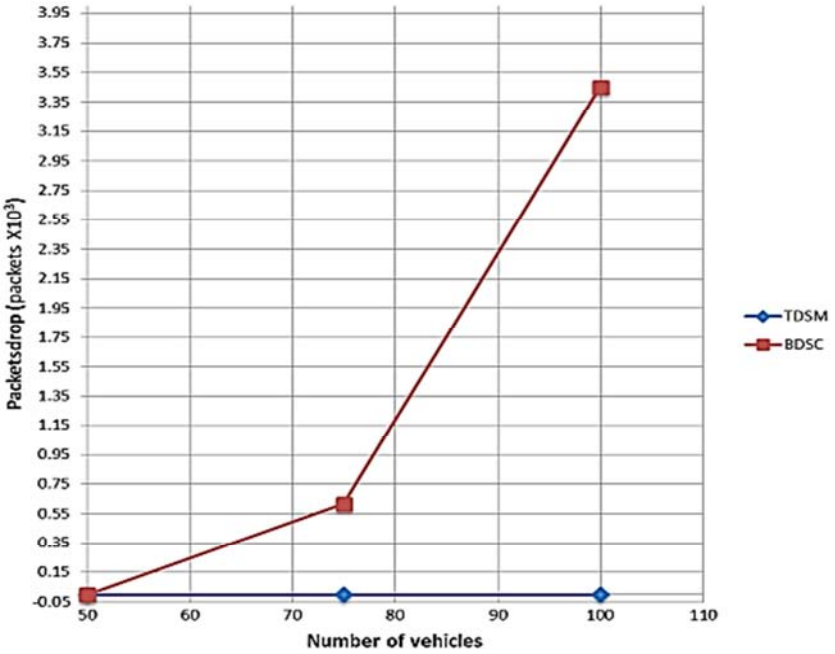

Figure 14. *Packets dropping in network.*

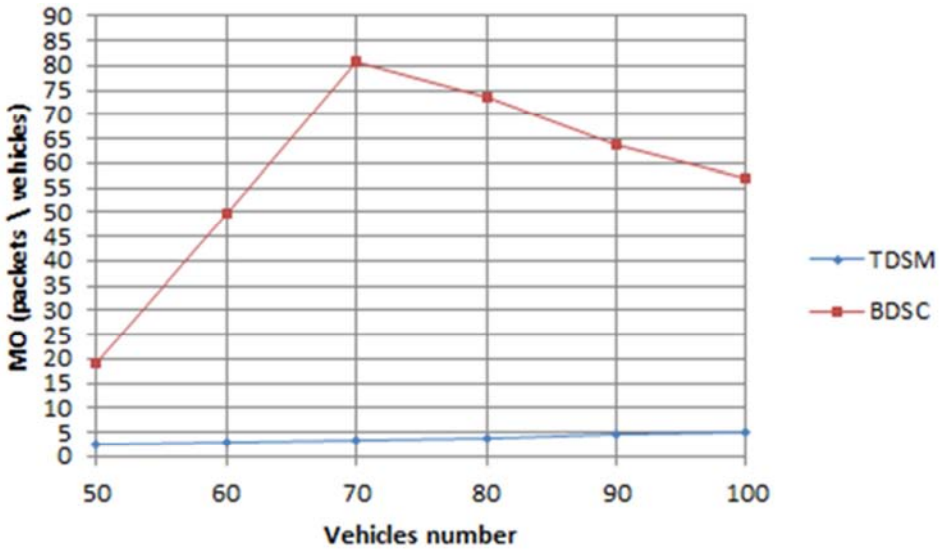

Figure 15. *Safety message network overhead.*

Given that the TDSM has achieved a zero packet drop, it forces a high number of messages to be transmitted, with the increase in the number of vehicles. Therefore, the message overhead problem may emerge in a dense traffic scenario.

An increase in message overhead is considered a normal scenario, knowing that the issue has been addressed, through dividing the road into segments, and using a directional broadcasting approach. In Figure 15, there is an increase in the message overhead of the BDSC approach in line with the increase in the number of vehicles, to some point where it starts to reduce with increases in the traffic density. The BDSC shows a decrease in the message overhead when the number of vehicles exceeds 70. This is a dangerous indicator as the packets may begin to collide and may not be transmitted.

4.2.5. Message Delivery Ratio (DR)

Message delivery ratio is the ratio of the number of data packets received at the destination to the number of data packets sent by the sources.

$$DR = \frac{P_{reciev}}{P_{sent}} \tag{10}$$

A major aim of the TDSM is to achieve a high ratio of delivery for the warning message in V2V multi-hop data dissemination, and to lower the waiting time of the emergency messages at the recipient nodes. Consequently, TDSM overcomes packets dropping by using a retransmission method, yet it dictates an increase in the number of sending packets. This has a significant effect on the rebroadcasting time, as the incoming messages are queued at the recipient node (as the node deals with one message at a time). Since no packets drop, which achieve a very high warning message reachability in TDSM, therefore the delivery of the packets to the forwarder is ensured. However, when the whole network (one segment or more) is with high traffic density (more than 70 vehicles in each and every segment), the simulation result has shown an increase in the message scheduling delay. Hence, the safety messages wait longer in the queue before being rebroadcasted farther.

To overcome the time increase in message scheduling delay for the safety messages, particularly at the recipient nodes in high-density network, we added a field of priority value to the safety message format. This helps in scheduling the messages according to their priority value, in order to reduce the emergency message scheduling time delay (warning message with the highest priority is enqueued at the beginning to be sent first, as will be shown in the next section). The higher the message delivery rate, the higher the system reliability. Figure 16 shows that the safety message delivery in TDSM is 100% when the vehicle number is less than 70. But when the number of vehicles exceeds 70, the delivery rate is reduced. This is due to the high road density that affects the delay message scheduling at the recipient nodes. In other words, the message delivery rate decreases whenever the number of vehicles increases. In the existing BDSC approach, the delivery rate decreases when the traffic density is low or medium, due to high collision and overhead. The figure shows that BDSC has achieved a better message delivery rate at a high traffic density than in the case of 70 vehicles. This in fact results of the involvement of fewer packets, as the rest of the packets were lost to collision. The message delivery rate in the TDSM is much better compared to that of BDSC, due to the priority value of each message type, that improves the enqueuing scheme. Note that even when the delivery rate with TDSM decreases, we could compensate IEEE 802.11 p lack of reliability by guaranteeing a fast and efficient delivery, with zero safety messages drop.

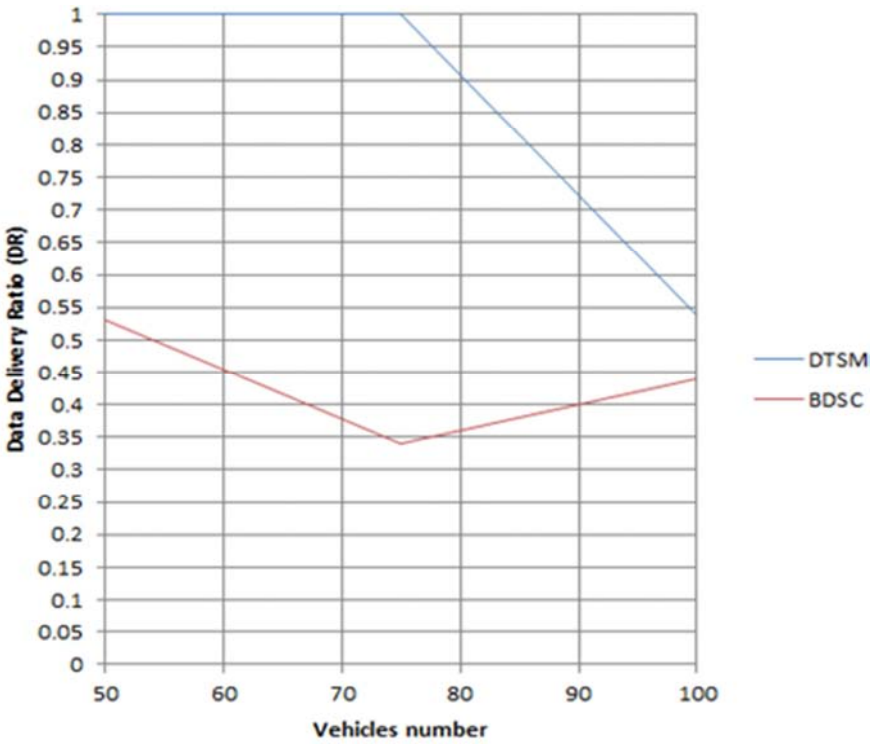

Figure 16. *Safety message delivery.*

4.2.6. Message Queuing Scheme

In a normal situation, the incoming transmitted messages at the recipient nodes are enqueued according to their delivery time, where messages that come first are first rebroadcasted. A data queue is ensured in all the connecting nodes, as each node can only handle one packet at a time. Thus, if a node receives a high number of messages, it schedules them for rebroadcasting; what causes an increase in the schedule delay.

Regardless of the number of packets dropping, the delay of delivering packets in dense networks increases due to the increasing waiting time at every recipient node queue. TDSM aims to reduce the packet delivery time delay for urgent messages in the queue, using a special arrangement, based on message priority. This helps in processing the messages with high priority first.

In TDSM, the warning message is enqueued according to the priority of its type, after checking its validation, the SID, and the source-tag field, as illustrated in Oh-TDSM algorithm phase 3. The simulation scenario that considers injecting an accident at some point is shown in Figure 17, where the transmission of safety messages with high priority is initiated. This is well explained as follows:

Assume an accident occurs at some point in segment SID=2, the nearest node ID=23 acts as a source node, and creates a safety message M, adding priority P1=1 (as for cooperative collision warning message). The source node starts propagating M to all the neighboring nodes with IDs=8, 7, 21, and 22. Note that only the nodes with IDs=21 and 22 are behind, and the node with ID=22 has the highest weight link value. This means that only nodes with ID 21 and 22 will be aware of M, and that node 22 is responsible for replying with ACK to the node with ID=23. If another message is created by node ID=7 with P2=2 (as for transit vehicle warning signal message), then, nodes with ID 21 and 22 will receive two packets from the same segment, but with different priorities. The recipient nodes will store the two messages in a queue and start processing the queuing scheme to schedule the messages with P1, then, P2 according to the highest priority.

Figure 18 shows TDSM delay for message scheduling, where the messages with high priority (1) are scheduled with less delay compared to low priority messages (3).

4.3. Trust Schemes Simulation Results

We have adapted our previous work on the data trusting scheme [27] to be used in one segment. So, when any node receives a safety message, it starts its own BTS. If the node trusts the incoming message, then, it retransmits it farther. The second decentralized data trust scheme, will be activated after the forwarder replies with an ACK and trusts the incoming message, using ETS, before rebroadcasting farther.

The nodes at the NHF zone allow message exchange with the forwarder, which has a tag equal to 1 (tag=1), where the ETS is processed.

Earlier in this study, we have used a simple model for simple mathematical representation, with an emphasis on the Greenshield model of traffic flow theory [33] to test ETS for trusting the incoming messages to the forwarder node in one segment, before disseminating them farther. Figure 19 provides more explanation by showing a toy depiction of our mathematical representation. Here, we assume that vehicle ID=V1 is positioned at the map point 10 when it senses an abnormal situation. Then, V1 immediately initializes T_B,

generates a warning message, and starts its backward propagation to all its nine neighboring nodes. Suppose that Table 4 indicates the table of information stored in V1, then, vehicle ID=V9 is the chosen forwarder according to the computed weight value. When the forwarder V9 receives a message from V1, it immediately replies with ACK and starts ETS. With the ETS result, the forwarder V9 decides whether to rebroadcast the message farther or not. Previously, in the Mh-TDSM algorithm, we have shown that when the number of messages received by the forwarder is coming from a number of vehicles (E), and (E) is less than the actual number of connected neighboring vehicles (C) to the same node. The forwarder will choose a random number (N) bounded by $\frac{C}{2} < N < C$.

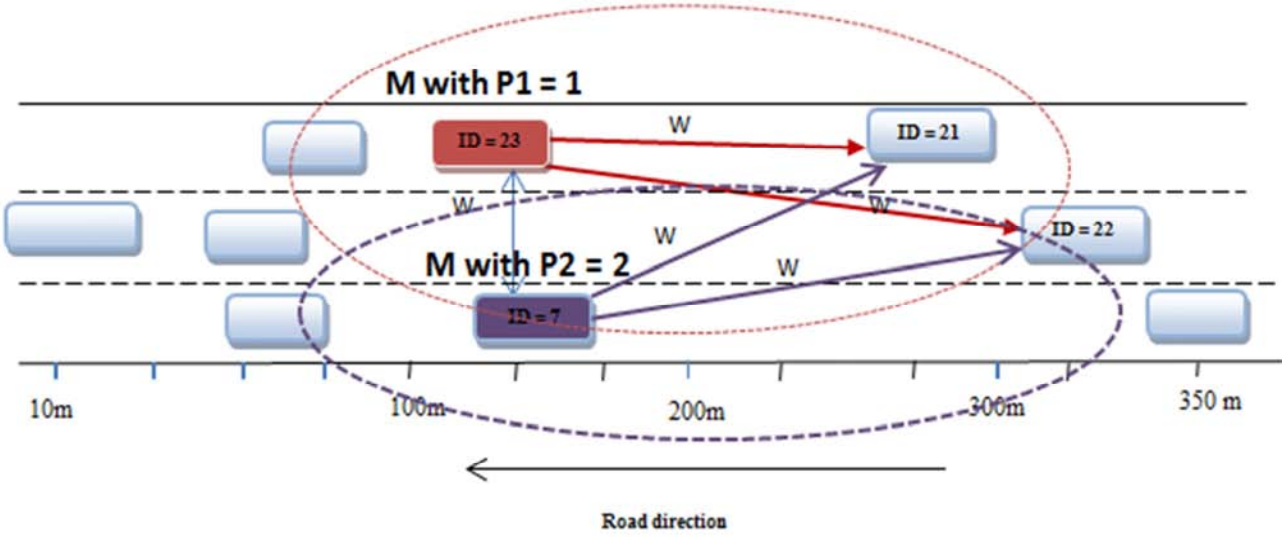

Figure 17. *Receiving different messages with different priorities.*

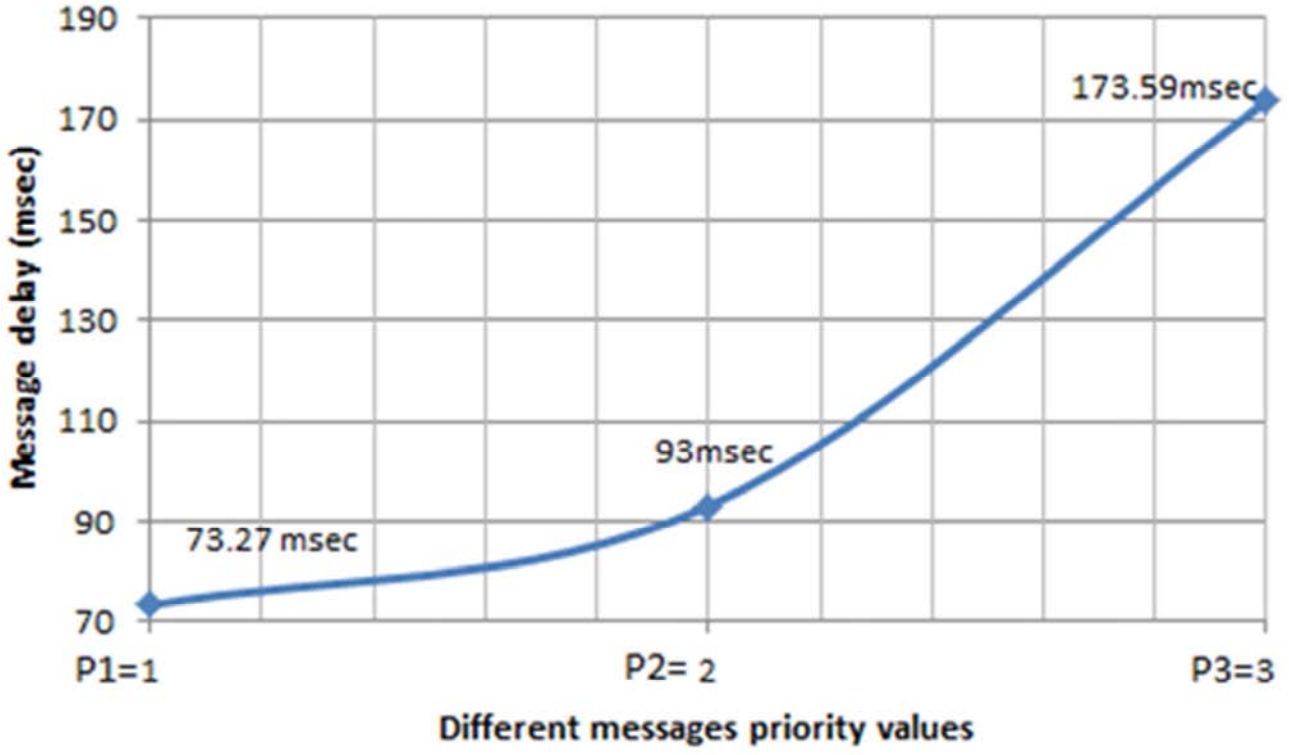

Figure 18. *TDSM Scheduling delay.*

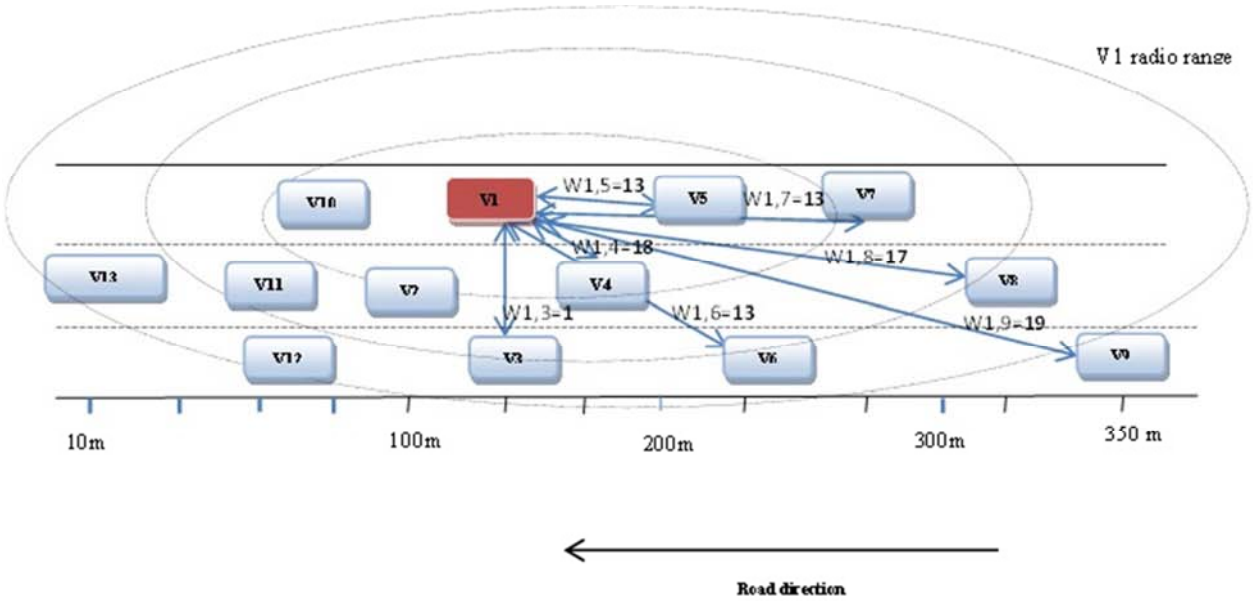

Figure 19. *A snapshot of vehicles distribution in a segment.*

Three cases for the random number (N) are shown in Table 5. In case 2, the forwarder needs to make a decision for trusting the warning message information by using binomial probability. For the mathematical evaluation on data samples, we took the probability of success P=0.9 (where P represents the proportion of interconnected vehicles out of the maximum number of vehicles in the entire 350 m segment), and assumed the radio range of node signal coverage as guaranteed to cover 350 m. We achieved the following results: From equation 7, we have distinguished that for a binomial distribution over 0.40, the decision can be considered as a trust decision, and the node can successfully retransmit the message farther. In this toy example, we examine ETS for all the nodes connected to V1 in Table 4, in order to analyze the efficiency of the endorsement of the incoming messages by the vehicles. The results are illustrated in Table 6.

Table 6 shows that the message received by the nodes with ID=V6 and V8 are trusted, while vehicles V2, V3, V5, and V9 must make a decision to trust the incoming message. Nodes V4 and V7 have decided not to trust the incoming message, but rather to notify the misbehavior authority. In our example, the best forwarder V9 must make a decision on the incoming warning message, to either trust the message or not, before rebroadcasting farther to the next segment. To help in making the trust decision, we have depended on the binomial probability.

Table 4. *Neighboring nodes' status information available in V1.*

Link with V1	V2	V3	V4	V5	V6	V7	V8	V9
Positions	10	35	60	80	100	130	170	190
Velocity	60	60	70	65	75	60	80	90
No. of vehicles connected to Vi (C)	30	10	32	18	11	19	22	25

Table 5. *Data trust results when based on ETS scheme.*

Case no.	(E) according to (N)	Trust results
1	$E \geq N$	Trust
2	$N/2 < E < N$	Make a decision
3	$E \leq N/2$	Not trust/ notify misbehavior authority

Table 6. *Decision-making result at each node.*

V-ID	V2	V3	V4	V5	V6	V7	V8	V9
(C)	30	10	32	18	11	19	22	24
(E)	15	8	7	12	10	6	22	14
(N)	16	9	16	13	10	13	20	15
P result	0.33	0.39	≈0	0.37	-	≈0	-	0.34
Decision	Make	Make	Not trust	Make	Trust	Not trust	Trust	Make

5. Conclusion

The connection between the neighboring nodes is established by exchanging hello packets. Messages' dissemination protocols are restricted by three main problems; the broadcast storm problem, the hidden node problem, and

data trustworthiness. These restrictions should be addressed when building a broadcast scheme. Moreover, in order to achieve promising message delivery rates, two major objectives must be considered: avoiding packets drop during transmitting and controlling message redundancy rate. However, it is still challenging to avoid network overhead while maintaining a high delivery rate and decreasing the reachability time delay.

In this paper, we have proposed a technique for the dissemination of warning messages through multi-hop V2V communication (out of the internet coverage), based on three main contributions. F*irst*, we introduced a new broadcast approach based on five main steps which are: (1) using a pair of information packets containing the beacon and the message information <B, M> to minimize the number of packets sent during an emergency; (2) dividing the digital roadmap into small segments of fixed sizes; (3) arranging the nodes even if the situation is normal depending on the weight of the connected links, which is estimated at each time interval; (4) depending on a directional broadcast method, which implies a consideration of backward message transmission; and, (5) selecting the forwarder node with respect to the highest link weight value. The selected forwarder is responsible for rebroadcasting the message farther to the next segment. *Second*, our study suggests a scheme for queuing the incoming alert messages, with respect to some parameters, in order to send the message with the high priority first. *Third*, the study adds two algorithms for trusting the information of a message, based on probability. The first trust scheme is to trust the transmitted message in the same road segment (BTS), while the second is initiated when the message is rebroadcasted by the forwarder to the nodes within the next segment, to trust it before disseminating it farther (ETS).

The simulation results of TDSM have shown that almost 0% packets drop could be achieved by using the resending technique. In addition, we could overcome the board storm problem by: changing the resending time T_B, upon average distance between the connected vehicles. The message dissemination will be stopped when the forwarder replies with an ACK. Besides, the use of a pair of information <B, M>, to be transmitted at every T_B, has achieved promising message redundancy rates through each segment. The hidden node problem has been avoided through the directional broadcasting technique, and the ACK reply. We have achieved promising reachability rates in different road densities due to the dependence on message priority when enqueuing the incoming messages. This also has ensured that messages with high priority are sent first.

References

[1] Rola Naja, "Wireless Vehicular Networks for Car Collision Avoidance," eBook by Springer Science and Business Media, New York, 2013.

[2] Tony K. Mak, Kenneth P. Laberteaux, and Raja Sengupta, "A Multi-channel VANET Providing Concurrent Safety and Commercial Services," Working Papers, California Partners for Advanced Transit and Highways (PATH), 2005.

[3] Yousefi, Saleh, Mahmoud Siadat Mousavi, and Mahmood Fathy. "Vehicular ad hoc networks (VANETs): challenges and perspectives." In ITS Telecommunications Proceedings, 2006 6th International Conference on, pp. 761-766. IEEE, 2006.

[4] Nema, Megha, Shalini Stalin, and Vijay Lokhande. "Analysis of Attacks and Challenges in VANET." International Journal of Emerging Technology and Advanced Engineering 4, no. 7 (2014).

[5] Maihofer, Christian, "A survey of geocast routing protocols." IEEE Communications Surveys & Tutorials 6, no. 2 (2004).

[6] Kumar, Rakesh, and Mayank Dave, "A review of various VANET data dissemination protocols." International Journal of u-and e-Service, Science and Technology 5, no. 3, pp. 27-44 (2012).

[7] Hsieh, Yi-Ling, and Kuochen Wang, "Dynamic overlay multicast for live multimedia streaming in urban VANETs." Computer Networks 56, no. 16, pp. 3609-3628, (2012).

[8] Yunpeng Zang, Lothar Stibor, Hans-Jürgen Reumerman, and Hui Chen, "Wireless Local Danger Warning Using Inter-Vehicle Communications in Highway Scenarios, " Wireless Conference, 14th European. IEEE, June 2008.

[9] M. Bilal, P. M. L. Chan and P. Pillai, "A Fastest Multi-Hop Routing Scheme for Information Dissemination in Vehicular Communication Systems." IEEE International Conference on Software, Telecommunications and Computer Networks, Split, Dubrovnik, Sept. 2010.

[10] Xuewen Wu, Shiming Song, and Huibin Wang, "A novel position based multi-hop broadcast protocol for vehicular ad hoc networks." Journal of networks, Vol. 6, NO. 1, Jan., 2011.

[11] Zeadally, S., Hunt, R., Chen, Y. S., Irwin, A. and Hassan, A., "Vehicular ad hoc networks (VANETS): status, results, and challenges." Telecommunication Systems, *50* (4), pp. 217-241, 2012.

[12] Hanaa S. Basheer, and Carole Bassil, "A Review of Broadcasting Safety Data in V2V: Weaknesses and Requirements" Elsevier Journal of ad hoc networks, press. Vol. 65, pp. 13-25, October 2017.

[13] Raya, Maxim, and Jean-Pierre Hubaux, "The security of VANETs," Proceedings of the 2nd ACM international workshop on Vehicular ad hoc networks, ACM, 2005.

[14] M. Raya, P. Papadimitratos, V. D. Gligor, et al, "On data-centric trust establishment in ephemeral ad hoc networks," In proceedings of INFOCOM. The 27th Conf. on Computer Communications, Phoenix, USA, pp. 1238-1246, Apr. 2008.

[15] Hanaa S. Basheer, Carole Bassil, and Bilal Chebaro, "Toward using data trust model in VANETs." Applied Research in Computer Science and Engineering (ICAR), International Conference on (pp. 1-2). IEEE, October, 2015.

[16] Rehman, Osama M. Hussain, Hadj Bourdoucen, and Mohamed Ould-Khaoua, "Improving reachability of multi-hop alert messages dissemination in VANETs." Information and Communication Technology Convergence (ICTC), 2014 International Conference on. IEEE, 2014.

[17] Venkata Manoj D, M. M. Manohara Pai, Radhika M. Pai, and Joseph Mouzna, "Traffic monitoring and routing in VANETs: a cluster based approach," IEEE 11th Inter. Conf. on ITS Telecommunications, 2011.

[18] Korkmaz, Gokhan, Eylem Ekici, Fusun Ozguner, and Umit Ozguner. "Urban multi-hop broadcast protocol for inter-vehicle communication systems." In Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks, pp. 76-85. ACM, 2004.

[19] Da Li, Hongyu Huang, Xu Li, Minglu Li, and Feilong Tang, "A Distance-based Directional Broadcast Protocol for Urban Vehicular Ad Hoc Network Wireless Communications," Networking and Mobile Computing, WiCom Inter. Conf. IEEE, 2007.

[20] Rehman O, Ould-Khaoua M, Bourdoucen H. "An adaptive relay nodes selection scheme for multi-hop broadcast in VANETs." Computer Communications. Vol. 1, No. 87, Aug. 2016.

[21] Neeraj Kumer, Joel Rodrigues, and Jaime Lloret, "Replication-Aware Data Dissemination for Vehicular Ad Hoc Networks Using Location Determine.*"* Mobile Networks and Applications 20, No. 2 pp. 251-267, 2015.

[22] Achour, I., Bejaoui, T., Busson, A. and Tabbane, S., "SEAD: A simple and efficient adaptive data dissemination protocol in vehicular ad-hoc networks.**"** Wireless Networks 22.5, pp. 1673-1683, 2016.

[23] Lin, D., Kang, J., Squicciarini, A., Wu, Y., Gurung, S. and Tonguz, O., "MoZo: A Moving Zone Based Routing Protocol Using Pure V2V Communication in VANETs." IEEE Transactions on Mobile Computing 16 (5), pp. 1357-1370, 2017.

[24] Schwartz, Ramon S., Kallol Das, Hans Scholten, and Paul Havinga, "Exploiting beacons for scalable broadcast data dissemination in VANETs." Proceedings of the 9th ACM international workshop on Vehicular inter-networking, systems and applications, 2012.

[25] Technical Specification, Intelligent transportation systems (ITS); Vehicular communication; Geo-Networking; Part 4: geographical addressing and forwarding for point to point and point to multipoint communications, ETSI TS 102 636-4 v1. 1. 1, 2011.

[26] Sanaa Taha and Xuemin Shen, "Secure IP Mobility Management for VANET." eBook by Springer Cham Heidelberg New York Dordrecht London, 2013.

[27] Hanaa S. Basheer, Carol Bassil, and Bilal Chebaro, "Bayesian Trust Scheme: A Decentralized Safety Message Trust Method in Multi-hop V2V networks." Journal of Communication, Vol. 12, No. 4, April 2017.

[28] Sheldon Ross, "A First Course in Probability." 8th edition, textbook by Pearson Education, Inc., USA, 2010.

[29] Teerawat Issariyakul and Ekram Hossain, "Introduction to network simulator NS2." eBook by Springer Science and Business Media, New York, 2009.

[30] Behrisch, Michael, Laura Bieker, Jakob Erdmann, and Daniel Krajzewicz, "Sumo simulation of urban mobility-an overview." The 3rd International Conference on Advances in System Simulation (SIMUL). 2011.

[31] Sharef, B. T., Alsaqour, R. A. and Ismail, M., "Vehicular communication ad hoc routing protocols: A survey.**"** Journal of network and computer applications, 40, pp. 363-396, 2014.

[32] B. Karp and H. T. Kung, "GPSR: Greedy Perimeter Stateless Routing for Wireless Sensor Networks." Proc. MobiCom 2000, Boston, MA, Aug. 2000.

[33] Immers, L. H., and S. Logghe, "Traffic flow theory." Faculty of Engineering, Department of Civil Engineering, Section Traffic and Infrastructure, Kasteelpark Arenberg, vol. 40, no. 21, 2002.