

To be defined

Mario De Felice, Andrea Baiocchi, Francesca Cuomo, Gaetano Fusco and Chiara Colombaroni
University of Roma “Sapienza”, Rome, Italy
Email: mario.defelice@uniroma1.it

Abstract—little reminder: we have SAME (SAmpled Measure-ment Estimation), which samples vehicles every R_{max} m (in our case $R_{max}=827$ m). Clearly R_{max} is the maximum transmission range, so if there is not a vehicle that far, the protocol samples the vehicle that maximizes the distance from the previous sampled vehicle and its R_{max} : as a matter of fact, 700m is a good sampling average. Very light and accurate over 5s estimations.

Timer-based Ordered Measurement Estimation (TOME) samples all the vehicles on the road. Very accurate, but pretty heavy.

Index Terms—Traffic measurement; VANET; dissemination protocol; accident detection

I. INTRODUCTION

One challenging application in the field of the Intelligent Transport Systems (ITS) is the traffic monitoring to develop information strategies that help travelers in their travel choice. Advanced Traveler Information Systems (ATIS) and Advanced Traffic Management Systems (ATMS) may take advantage by the use of new technologies like Vehicular Ad-Hoc Networks (VANETs) that allow Dedicated Short Range Communications (DSRC) of vehicles in the 5.9 GHz band. Applications of VANETs to ATIS and ATMS provide these systems with a monitoring subsystem that exploits equipped vehicles as probes in the traffic stream and, at the same time, with a communication system that allows vehicles to exchange information with each other regarding current traffic speed or any other useful message on traffic conditions. In this work we propose two road traffic monitoring approaches based on VANETs. To gauge the feasibility of the approach, we have set up a realistic simulation of a main expressway in Rome, Italy, calibrated by exploiting information provided by an extensive sample of floating car data. Joint vehicular mobility and communication network simulations have been carried out, pointing out that the designed protocols can collect quite accurate average speed estimates for different sections of the road with a very limited effort. No monitoring infrastructure is needed, except for a single Road Side Unit (RSU) on a road span of almost 70 km; collection times are within few seconds, the required bitrate is very low (0.08 kbps) and relative errors are about 2% on the long-term average (7% if real time, in the worst case) with one approach; with the other one, we have higher accuracy (mostly 100%), collection times in about 40s, but a higher bit rate (~ 40 -50 kbps). Furthermore, we tested our approaches when an incident occurs on two of the 3 lanes of the highway. The protocols are both able to quickly report the event at the RSU side and in particular, one of them can do it real time. The impact of the market penetration rate of

VANET devices on the traffic monitoring capabilities in case of incident is also provided, such that our approach still works when not all the vehicles are equipped with DSRC devices.

II. RELATED WORK

In the last years, a broad literature arose from VANET applications to different ITS subsystems, ranging from ADAS (cooperative collision warning), to ATMS/ATIS (virtual traffic lights, incident detections) and ATIS (advanced speed control, route guidance). In the following, we will focus on traffic monitoring and incident detection. Traditional incident detection algorithms developed in 70s like California and Payne’s [1] are based on occupancy measures at fixed road sections and try to recognize anomalous conditions by comparing upstream and downstream traffic density measures; that is, by observing the effects of the incident on traffic flow.

Statistical algorithms detect significant differences between observed detector data and traffic characteristics predicted by prior probabilities [2], time-series and filtering analysis ([3][4]).

In the following years, more complex mathematical methods were introduced. McMaster algorithm applied the Catastrophe theory to recognize an abrupt interruption of the regular pattern in the flow-speed-occupancy space [5]. In the latest years, several methods were developed that apply artificial Intelligence techniques, including neural networks ([6][7]), fuzzy logic [8] and a combination of these two techniques ([9][10]).

Techniques based on the use of surveillance cameras take advantage of temporal variations of traffic characteristics in addition to spatial ones. These methods provide their best effectiveness when stopped vehicles are in the surveillance area. The performances of the aforementioned algorithms depend on the balance of the thresholds chosen for incident identification. If thresholds that provide false alarm lower than 2% are chosen, the mean time to detect an incident ranges from about 1 minute to 6-8 minutes [11].

Several authors have shown that vehicle-generated data can provide reliable estimates of traffic conditions, including identifying incidents and congestion ([12][13]).

Dia and Thomas [14] devised a neural network algorithm that uses a fusion of data from loop detectors and probe vehicles identified by fixed devices. Larger advantages can be obtained, however, by collecting data directly from floating cars; that is, vehicles that move on the road network. Geisler et al. [15] presented an evaluation framework for traffic information systems based on data streams from mobile phones and

applied it in two case studies, namely queue-end detection and traffic state estimation, simulated by VISSIM microsimulation software in the idealistic case of two highways links of 5 km length.

Ma et al. [16] envisioned a real-time travel incident detection method based on Artificial Intelligence paradigms that exploits vehicle-infrastructure integration. They analyzed their method through a simulation experiment on a small freeway network simulated by Paramics, which showed that the proposed framework outperforms traditional incident detection methods based on point traffic measures like California algorithm. Leontiadis et al. [17] investigated opportunities for implementing VANET-based ITS systems that performs a fully distributed information system and compared different traffic prediction algorithms in a simulation of downtown area of Portland, Oregon (USA).

While traditional techniques mostly rely on monitoring infrastructure placed on the road, in this work we use a different approach, where monitoring infrastructure is almost totally missing (one infrastructure node on 68 km), and instead we exploit DSRC features like multi-hop [18]. We can thus yield faster results for real time traffic monitoring and car incident detection, where we scale from the order of minutes, in the previous literature, to few seconds in our study.

III. THE VANET MONITORING SYSTEM

Several solutions [19] have been proposed for message dissemination in a VANET with the goal of extending the coverage area reached by message flows distributed by a RSU (or an On Board Unit-OBUs) thanks to vehicle-to-vehicle multi-hop communications. A key issue is to avoid the broadcast storm problem [20]. To this aim, we exploit a backbone based approach [21] where a set of vehicles that are situated along the road are dynamically self elected to act as relay nodes. Many protocols in this category are based on timers. All nodes receiving a given packet delay the relaying of that packet in order to choose the node in the best position and to inhibit the other nodes. The DDT [22] approach is based on a simple and quite effective way for reducing redundant rebroadcasts and the consequent medium contentions and collisions. It assigns the tasks of relaying packets in each neighborhood solely to the receiver that is located farthest away from the sender. DDT assumes that each node is equipped with a GPS and an embedded system connected to vehicle sensor devices, so that both node position and speed can be estimated. Let P_A and v_A denote the vehicle position (coordinates) and speed, respectively.

Leveraging the DDT idea, we define a so called Distance Based Forwarding (DBF) dissemination protocol. DBF operations is based on the Forwarding and Inhibition Rules. Let A be a vehicle located at a point of coordinates P_A , forwarding a message with sequence number k and hop count h , at time t . Any other vehicle V , at position P_V within transmission range R of A , receives the message sent by A . Let k_V the biggest message sequence number already seen and completely dealt with by V .

- **Forwarding Rule.** By checking that $k \leq k_V$, V can discard old or duplicated messages. If the message is new (i.e., $k > k_V$) V schedules its forwarding by setting a timer $T_{V,k} = T_{max}(1 - \overline{P_V P_A}/R_{max})$, where T_{max} is the maximum forwarding delay, R_{max} is an upper bound of the coverage radius of OBU transceivers, and $\overline{P_V P_A}$ is the distance between V and A . Hence, V schedules the forwarding of the message at time $t + T_{V,k}$ with sequence number k and hop count $h + 1$.
- **Inhibition Rule.** If during the time interval $(t, t + T_{V,k}]$, V receives another message with sequence number k and hop count h' , V checks that $h' = h + 1$. In that case, V drops the scheduled message and will not forward it. Otherwise, no inhibition takes place.

In this work we define a road traffic monitoring protocol by developing a new logic based on DBF.

A. Monitoring by sampling vehicles - SAME

In the following we define a collection logic called SAME (SAmpled Measurement Estimation). Let us consider two RSUs located along a road span, RSU_a and RSU_b . RSU_a originates a stream of messages, issuing one *call for measurement collection (cmc)* message every time interval T_{RSU} . The *cmc* message is passed over from vehicle to vehicle by using a DBF logic, until it reaches RSU_b , that is the final sink of the collected measurements. Messages are issued by the originating RSU with a sequence number, incremented by 1 at each new message, and with h set to 0. Figure 1 gives a detail on how the DBF works to forward the messages. This protocol assigns the tasks of relaying messages in each neighborhood solely to the receiver that is located farthest away from the sender (node V in the picture). The logic is based on retransmission delays that are inversely proportional to the distance from the sender as described at the beginning of Section III. In this way the first far away vehicle that received a message (V in the Figure) forwards it and inhibits all the other vehicles (B , C and D) that have longer timers. DBF uses vehicular GPS positions, which are inserted in the header of broadcast messages. Figure 2 presents an example of multi-hopping in the SAME logic from RSU_a to RSU_b . The red cars are vehicles that under the DBF logic are the forwarding nodes and hence are the only nodes sending their measured parameters to the RSU_a .

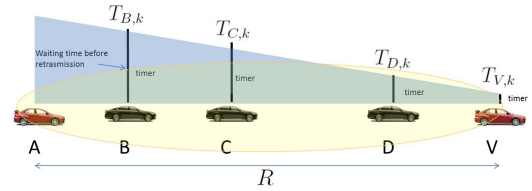


Fig. 1. DBF retransmission logic based on timers (black bars) inversely proportional to the distance from the sender. Oval box represents the maximum transmission range R of sender A .

The logic defined to monitor the traffic on the road with the VANET is implemented by having a distributed collection

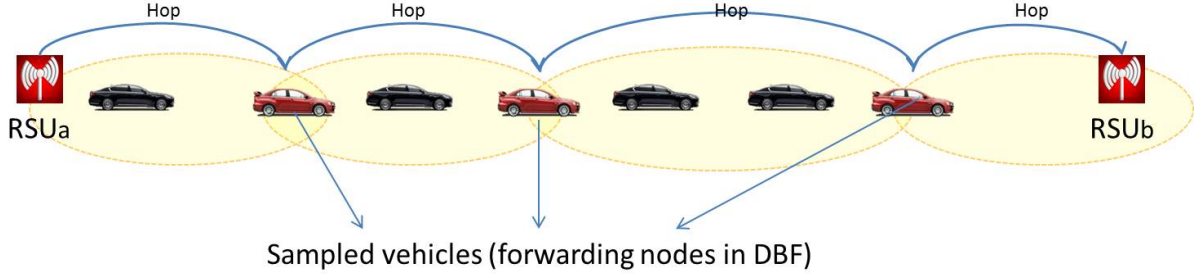


Fig. 2. An example of multi-hopping in the SAME logic from RSU_a to RSU_b . Only red cars are the sampled vehicles. Oval boxes represent maximum transmission ranges.

of a vector $\mathbf{m} = (m_1, \dots, m_n)$ where the i th element is a 4-tuple $m_i = \langle P_i, v_i, d_i, t_i \rangle$. Here P_i contains the geographical coordinates of the i -th sampled vehicle, v_i contains its current speed and d_i the direction, t_i is a timestamp with the current time. The size of \mathbf{m} depends on the number of vehicles that collect this information, that is the number of vehicles forwarding the message (or the number of hops) from the source RSU up to the sink RSU. The message initially issued by RSU_a has an empty payload. Each sampled vehicle, namely a vehicle acting as a forwarding node according to DBF, appends its 3-tuple to the current payload of the message. When the message reaches RSU_b , it carries all n collected measurements. n is related to the average hop length and to the length of the monitored road span. A fast polling is possible, since each forwarding hop takes a time in the order of ms and typical hop lengths can be several hundred meters. The traveling speed of the monitoring messages can be thus in the order of 50 km/s, three orders of magnitude more than vehicle speed. Measurements are collected at the RSU_b and can be used to track the traffic average speed and density in each sector of the monitored road span. Polling is repeated every τ_{RSU} seconds. τ_{RSU} is to be chosen so as to detect vehicular traffic variations quickly and reliably (that entails collecting tens of samples over a time scale comparable with that of vehicle motion, namely minutes) and to keep the collecting protocol performance near to optimal by avoiding communication channel congestion (which implies taking $\tau_{RSU} \gg \tau_{prop}$, where τ_{prop} is the time required for the message to travel a distance much bigger than the transmission range of a OBU; it is enough to take τ_{RSU} in the order of seconds).

B. Exhaustive measurement collection logic - TOME

DBF and similar multi-hopping protocols lend themselves naturally to sampled measurement collection, where samples are taken according to the distance covered at each hop. An exhaustive collection of vehicle measurements can be achieved by devising a variation of such protocol; the biggest difference is that all vehicles are assumed to send their own message, instead of having only a selection of relaying vehicles that forward the measurement message. We propose the Timer-based Ordered Measurement Estimation (TOME) protocol.

The protocol is based on an initiator station, namely an RSU, polling the system with *cmc* messages with period τ_{RSU} . τ_{RSU} is much bigger than the single hop time, so as to avoid MAC level congestion. The Collection Road Segment (CRS) is defined as the span of road for which the protocol aims at giving a single estimate of the average speed and density of vehicles. Let d_0 denote the target CRS length. For each CRS a node plays the role of the CRS Initiator (CRSI), i.e., it resets the message accumulator fields and starts off a new CRS.

Messages are made up of three parts: i) a header, that lists the sequence number k , the identity of the sending node S and its position P_S ; ii) the list \mathcal{L} of the measurements consolidated from previous CRSs; iii) the current CRS record $\langle CRSI.id, CRSI.pos, AS, NV \rangle$. $CRSI.id$ and $CRSI.pos$ are the CRSI identity and position of the vehicle playing the role of initiator of the CRS. AS is an accumulator field, whose value is the current sum of the speeds of vehicles that have been passing the message to one another within the CRS. NV is an integer counter, carrying the number of vehicles that have contributed to the accumulator AS . We denote formally a message as $\mathcal{M} = [k, S, P_S; \mathcal{L}; \langle CRSI.id, CRSI.pos, AS, NV \rangle]$.

At the end of the CRS s , the attained values AS_s and NV_s are frozen and stored into the list, i.e., $\mathcal{L} = \mathcal{L} || \langle P_s, AS_s, NV_s \rangle$, where P_s is the position of the first node of CRS s . The notation $||$ means appending the record appearing on the right to the list denoted at the left of the sign. The consolidation of the record appended to the list is done by the node that starts the new CRS $s + 1$, namely the node in position P_{s+1} .

Each node maintains state variables. For node A we define:

- k_A , the biggest sequence number read from incoming messages;
- the current measurement tuple $\mathcal{T}_A = \langle CRSI.id_A, CRSI.pos_A, AS_A, NV_A \rangle$, as updated after the reception of the most recent relevant TOME protocol message;
- TA, k the current value of the timer of A as for the k th message.

The basic idea of TOME is that each vehicle receiving a collection message will schedule the emission of its own message copy. The copy includes any accumulated speed sum

heard of until emission time, plus the contribution of the vehicle's own speed. The only exception is when a vehicle deems to be a new CRSI. In that case the most recent contribution, received before the scheduled emission time, is frozen into the list and a new accumulation starts off.

An example of the operation of TOME is given in Figure 3. Vehicle *A* is the CRSI, timers are highlighted as vertical bars. The black part of the bar represents the timer duration. Every vehicle in the CRS has its transmission range R (R_A , R_B and R_C in the figure). All vehicles in the CRS send their measurements.

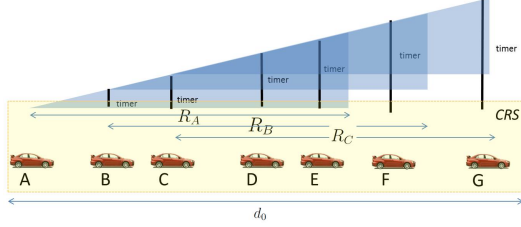


Fig. 3. An example of data collection in case of TOME over one Collection Road Segment. Retransmission timers are directly proportional to the distance. The rectangular box represents the CSI.

It can be noted that timer values are kept as chosen when receiving the first message with a new sequence number. Subsequent messages with the same sequence number do not cause the timer to be updated. This leads to *ordered termination of the timers* according to the distance from the CRSI. This property stems from the monotone rule chosen to set the initial values of timers, according to the distance of the receiving node from the CRSI. For each CRS, the RSU receives at last the collected sum of speeds V in the segment and the overall number of vehicles N , hence it can estimate the average speed (V/N) and the average vehicle density as $\delta = N/d_0$. Given V and δ , the average flow of the MCS is derived as $\phi = V\delta$.

Let us consider a node B with state k_B and $\mathcal{T}_B = \langle X, P_X, v, n \rangle$, when it receives the message $\mathcal{M} = [k, A, P_A | \mathcal{L} | \langle Y, P_Y, V, N \rangle]$.

If $k < k_B$, the new message is ignored, as an old, out of date message.

If $k = k_B$, B checks if $Y = X$. In that case, B has received a message from a vehicle inside the same CRS as B is. So, if $N > n$, B updates its state as follows: $\mathcal{T}_B \leftarrow \langle X, P_X, V, N \rangle$ and reschedules its timer as $T_{B,k} = T_{max} \overline{P_A P_B} / R_{max}$. If instead it is $N \leq n$, B does nothing and ignores the received message.

If $k = k_B$, but it is $Y \neq X$, then B checks whether $\overline{P_Y P_B} < \overline{P_X P_A}$ and $\overline{P_Y P_X} < \overline{P_X P_B}$. In that case B reassigns itself to the CRS initiated by Y and updates its tuple to $\mathcal{T}_B = \langle Y, P_Y, V, N \rangle$ and timer to $T_{B,k} = T_{max} \overline{P_A P_B} / R_{max}$. In case it is found that $\overline{P_Y P_B} \geq \overline{P_X P_B}$ or $\overline{P_Y P_X} \geq \overline{P_X P_A}$, B does nothing and ignores the message.

If $k > k_B$, a new measurement cycle is being run and B updates its state as $k_B = k$. As for the tuple \mathcal{T}_B , the key point

is to check whether B has to start a new CRS or it belongs to the current CRS. Hence B checks if $\overline{P_Y P_B} > d_0$. In that case, B sets $\mathcal{T}_B = \langle B, P_B, 0, 0 \rangle$. If instead it is $\overline{P_Y P_B} \leq d_0$, B sets $\mathcal{T}_B = \langle Y, P_Y, V, N \rangle$. In any case, a countdown timer is started with the initial value $T_{B,k} = T_{max} \overline{P_A P_B} / R_{max}$.

Finally, when the timer $T_{B,k}$ expires and the state of B is k_B and $\langle X, P_X, v, n \rangle$, B sends out a message $\mathcal{M} = [k_B, B, P_B | \mathcal{L} | \langle X, P_X, v + v_B, n + 1 \rangle]$, where v_B is the current speed of B .

An example of TOME behavior over multiple CRSs is given in Figure 4. When the CRSI of CRS1 sends the message (A in this case) all nodes with a distance $\overline{P_Y P_B} \leq d_0$ will retransmit the message by adding their speed values and positions. On the contrary when a node is out of the CRS (H in the example) it re-elects itself as CRSI, updates the list of other CRSs with the last measurements arrived and starts a new measurements collection over the new segment. The result is that for CRS1 the averaged speed value is over 7 vehicles while for CRS2 over 4 cars.

The speed of propagation of the measurement collection message is R/T_{max} . For typical values of the involved parameters, the order of magnitude of this speed is tens of thousands of km/h, so that it is from two to three orders of magnitude bigger than vehicle speed. As a matter of example, with $T_{max} = 500$ ms, $R_{max} = 830$ m, to cover the GRA (about 68 km), it takes about 41 sec.

As for the size of the measurement collection message, we assume that a position can be represented with 16 bits, the accumulated average speed as a floating number with 32 bits and the number of vehicles per segment as a 16 bit integer. Then 80 bits = 10 bytes are sufficient for a single record of the list. The number of MCSs depends on d_0 and the length L of the observed road, namely it is $\lceil L/d_0 \rceil$. If the measurement collection process involves no more than 100 road segments, 1000 bytes are enough to hold the whole list. Therefore, it can be expected that the message length is at most somewhat more than 1000 bytes, which is well compatible with message sizes sent over the DSRC interface.

Structures of the messages propagated in case of SOME and TOME are reported in Figure 5. In case of TOME the message is the one crossing the second CRS and transmitted by the node I ; H is the CRSI of this second segment while the CRSI of the first segment is A .

IV. APPLICATION OF MONITORING PROTOCOLS TO A URBAN HIGHWAY

In order to set up a realistic simulation, we exploit a large database of about 104 millions of GPS traces collected by about 80,000 equipped vehicles that made about 9 millions trips during the month of May 2010 in the metropolitan area of Roma (Italy). We focused on a subset of 50,220 vehicles on a 68 km long ring-shaped expressway embracing the city called GRA, which collects and distributes long-haul traffic entering and exiting from the city (see Fig. 6). Each vehicle sends its information record every 30 seconds. A variety of information is provided by each record, including the vehicle

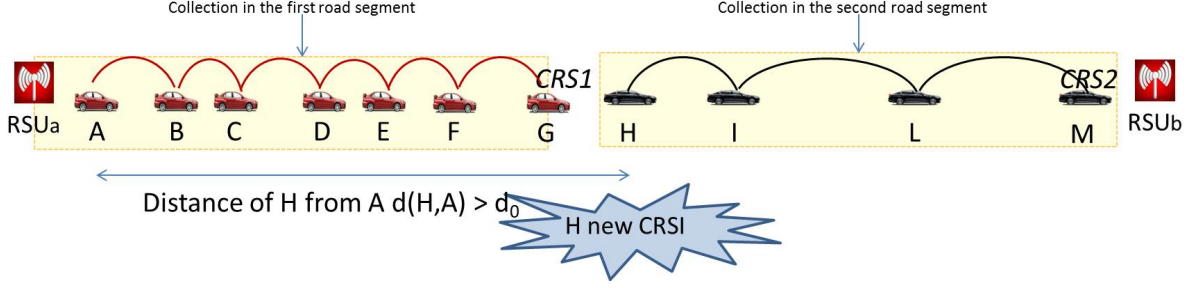


Fig. 4. TOME behavior over two CRSs. CSRI in the first segment is A while in the second segment is H .

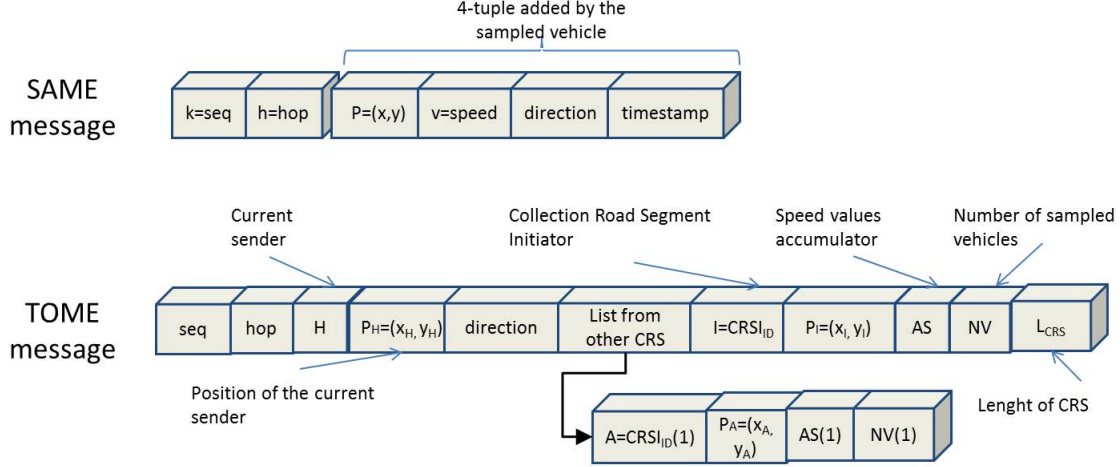


Fig. 5. Messages propagated in case of SOME and TOME.

ID, geographical coordinates, speed and quality of GPS signal. Our FCD (Floating Car Data) has been cleaned of the records with low GPS signal quality in order to consider only highly reliable data.

In order to analyse the data, we have divided the GRA in 29 different segments of length L_j , $j = 1, \dots, 29$, where the main exits from the GRA highway are the starting and ending points of each segment. Vehicles were then divided in two sets according to their traffic direction: clockwise and counterclockwise. Four different time periods of four hours each have been considered, starting from 7 am until 11 pm. Inter-vehicle distance distribution and speed distribution were obtained for each of the four time periods. This analysis showed that the highest density of vehicles is in the time period between 3 pm and 7 pm, which has the largest number of detected vehicles (9732 vehicles).

To set up a realistic mobility simulation of the urban traffic on the GRA, the available data sample have been inferred to the universe of vehicles by assuming a random uniform sampling. Let Δt be the sampling interval (30 sec in our study), v_i the average speed of vehicle i , n_j the estimated number of vehicles traveling in the j -th segment, $g_j(t_1, t_2)$ the number of detected GPS signals in the j -th segment during the observation time interval $[t_1, t_2]$, L_j the length of the j -

th segment, a the probe vehicles penetration rate ($a \approx 2.3\%$ in our study) and q_j the estimated flow on the j -th segment. Then:

$$n_j = \sum_{i=1}^{g_j(t_1, t_2)} \frac{v_i \Delta t}{L_j} \quad (1)$$

and the resulting flow is $q_j = \frac{n_j}{a(t_2 - t_1)}$. The above inference relations have been applied to the traffic flows in the peak period and have been used to enter into each segment j a flow of vehicles of mean intensity q_j . The FCD have been used to estimate the OD matrix, where Origins and Destinations correspond to the exits of GRA. Vehicle attributes have been tuned so that the average speed values measured per lane and per segment with the mobility simulation software match the corresponding values estimated with the inferred experimental data.

V. PERFORMANCE EVALUATION

A. Simulation model

For our case study of urban highway, we have imported the map of the Rome GRA from Open Street Map [23] to SUMO [24], where we have generated and verified the traffic flows from our OD matrix analysis in Section IV. We have implemented our VANET monitoring logics, as described in Section

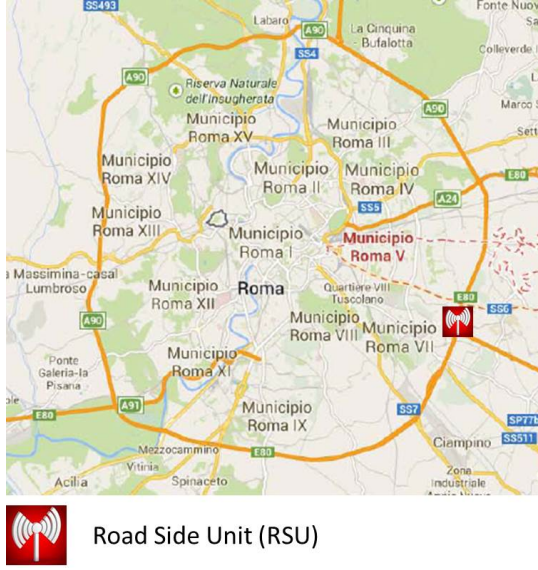


Fig. 6. The scenario we are working on. The RSU is highlighted in red

III), by adding code to NS2 [25] to realise the Forwarding Layer protocols on top of the IEEE 802.11p MAC/PHY layers already built in NS2. Communication network simulations of NS2 are fed by the outcome of SUMO to give node positions, sampled once per second. Between two consecutive sampling times, NS2 moves nodes according to linear uniform motion.

Since it will take years to deploy the VANET technology extensively, we assume different market penetration percentages of VANET DSRC radio equipment on board vehicles. Because of this, every new vehicle that is instantiated in our simulations, is equipped with this technology with probability $p \in [0, 1]$, that represents the market penetration rate of the VANET technology. So, on average, only $p \cdot 100\%$ of the vehicles are able to communicate with each other, while the others do not perform any transmission/reception operation, although influencing the vehicular micro-mobility simulated by SUMO.

We set up two simulation scenarios: i) regular traffic conditions, as given in the peak period identified in Section IV, based on the collected FCD on Rome GRA; ii) incident scenario. In the latter one, the flows are as in the regular scenario, except that in a specific highway sector (#14), just before time $t = 2000 \text{ sec}$, we raised the speed of some vehicles (the dense high speed cloud in Fig. 7(d)) thus ending up with a car incident that obstructs two out of the three lanes of the road in the considered traveling direction, until the end of the simulation.

Tab. I lists the main simulation parameters.

B. Traffic monitoring in regular conditions

We assume regular traffic conditions on the considered urban highway and we feed vehicular traffic according to the statistics derived from measurements in the peak time interval, as explained in Section IV. A single RSU is placed as shown

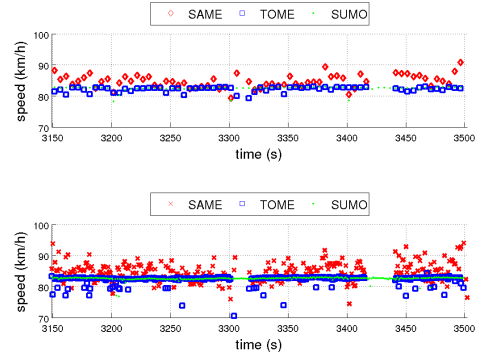
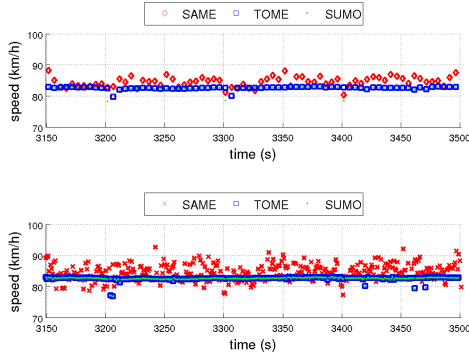
TABLE I
SIMULATION PARAMETER VALUES

Parameter	Value
Road length (km)	68.2
Number of lanes per traveling direction	3
Average vehicle density (veh/km)	31.02
SUMO simulation duration (s)	3600
Network simulation duration (s)	350-400s
Frequency of generated messages (msg/s)	1
R_{max} for each tx power value (m)	827
Max forwarding delay, T_{max} (ms)	100
Link Rate (Mbit/s)	6
MAC, PHY parameters	IEEE 802.11p
Propagation Model	Two ray ground

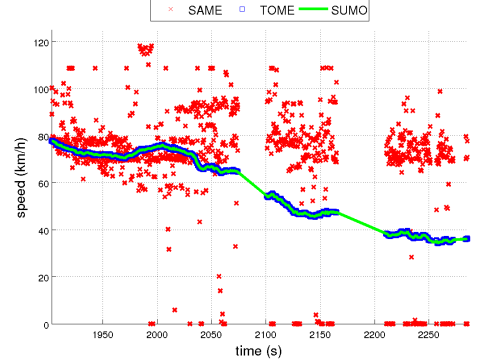
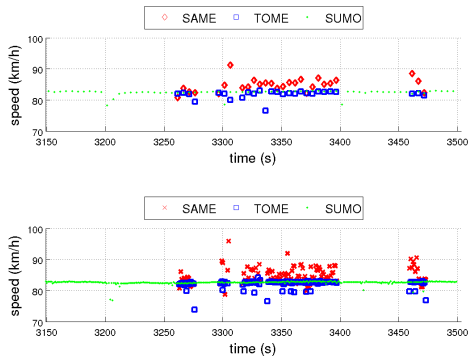
in Figure 6 and it acts contemporarily as *cmc* originator and sink. It sends out *cmc* with a time period $\tau_{RSU} = 1 \text{ sec}$. Figure 7(a), 7(b) and 7(c) plot the average speed measured over the entire urban highway through SAME and TOME and as given by the simulation software SUMO (ground truth). Measurements arriving at the RSU after taking a full trip along the urban highway ring are averaged out over a time window of 5 sec (on top) and 1 sec (bottom). The three graphics assume different fractions of vehicles equipped with a VANET OBU, namely $p = 1, 0.75$ and 0.5 .

By considering the top plots in Figures 7(a), 7(b) and 7(c), it is clear that TOME achieves a high accuracy degree in the estimation of the vehicle speed, since it collects speed samples from essentially all vehicles. Remarkably, also SAME turns out to be quite accurate, even though it just samples one vehicle every R_{max} , which is one order of magnitude less vehicles than TOME. Overall, the estimate by SAME is based on about 90 samples per round. For the averaging time of 5 sec, at most about 450 vehicle samples are taken. This is apparently enough to provide an accurate estimation of the average speed in regular traffic conditions. The effect of limited availability of OBU equipment on board vehicles is that some messages get lost due to the temporary disconnection of the VANET chain along the ring highway. The disconnection extent over time grows as the market penetration rate gets smaller. During the presumably long time when VANET equipment is being deployed, heterogeneous networking, exploiting the cellular network should be a solution in order to prevent too frequent disconnections and allow a regular monitoring of the highway. Once OBU spreading overcomes sensitively 50% of vehicles, VANET alone can start providing a reliable means of collecting measurements. Another means to improve the VANET reliability is to deploy few more RSUs, even if it is reasonable the RSU density shall be much lower than cellular base station density.

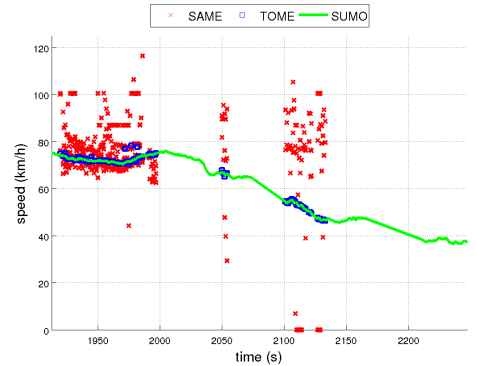
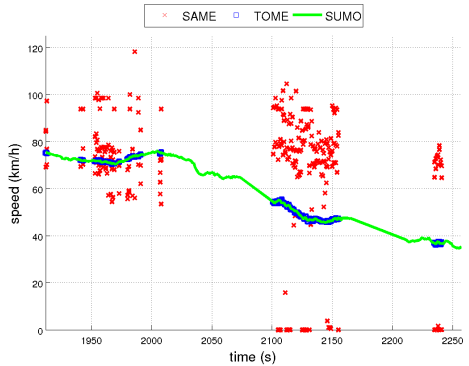
These findings are confirmed by the bottom plots in Figures 7(a), 7(b) and 7(c), where the same quantities of the top graphs are plotted, except that the average is performed over 1 sec. It is evident that SAME tends to yield a less stable result than TOME, somewhat overestimating the true average value, but still it keeps the error below 10% in almost any condition or



(a) 100% of OBU-equipped vehicles, regular traffic conditions (b) 75% of OBU-equipped vehicles, regular traffic conditions



(c) 50% of OBU-equipped vehicles, regular traffic conditions (d) 100% of OBU-equipped vehicles, incident just before $t=2000s$



(e) 75% of OBU-equipped vehicles, incident just before $t=2000s$ (f) 50% of OBU-equipped vehicles, incident just before $t=2000s$

Fig. 7. Speed measures on the whole highway, when varying the market penetration rate of the system, in case of regular traffic conditions (a,b,c), with results aggregated every 5 sec (on top) and every second (bottom) and speed measures of sector #14 of the highway in case of incident (d,e,f)

market penetration rate. Taking into account that we target real time vehicular traffic monitoring, we have to strike a balance between accuracy and delay. From our results, it appears that 5 *sec* can yield a good compromise.

Similar results are obtained by analyzing each highway sector, with more variability for SAME, due to the limited number of samples it collects (few units per sector, a sector being 2.4 *km* long on average).

A last consideration should be said about the required bitrate both for SAME and TOME: in the first case it is almost null (0.08 kbps), which yields the possibility of a permanent nearly invisible service, running on the control channel or on any non-dedicated sub-channel; in the second case, the bitrate is about 56 kbps, better suited for a channel explicitly dedicated to traffic monitoring services.

C. Incident detection through VANET monitoring

In this section we analyze the behavior of our protocols when dealing with an incident (Figure 7(d), 7(e) and 7(f)). These figures refer to only one sector of the GRA, namely sector #14, that is the one where the incident occurs. Moreover, SAME points are the individual speed samples taken from sampled vehicles within the tagged sector and transported by the protocol.

It is clear that in this case the market penetration rate is a very important factor, since after the accident occurs, the vehicle flow is strongly slowed down and becomes irregular in the interested sector. In terms of communications working, this means that connectivity is not always granted. In fact, even with $p = 1$, we have blank spots in Fig. 7(d), as for $2160 < t < 2200$ *sec*.

By looking at Fig. 7(d), we get a very clear information about what is going on from SAME. As soon as an incident occurs, immediately we observe two aspects:

- 1) higher variance in the monitored speed.
- 2) a sudden increase of (almost) zero speed vehicles.

These two aspects combined together immediately trigger an alarm, detecting the presence of an incident. The average speed of the vehicles from our ground truth and from TOME, instead, needs some time to drop down, because at the beginning the problem is very localized and the observed sector is 2 *Km* long, so vehicles that overcome the incident, accelerate and take the same speed they had before they encountered the incident, or even a higher one, because of the reduced density downstream the bottleneck formed by the stopped vehicles. This means that only after 50 – 100 *sec* TOME really gets aware that an anomaly has occurred (it is possible to notice the descending slope after the incident in the figure). Moreover, the fact that SAME detects many zero speed values is a clear signal that a stable queue has been formed upstream a sudden bottleneck: this fact identifies an incident. As market penetration rate drops down, the capability to reveal an incident is delayed. Finally, it is important to notice that SAME takes one order of magnitude less time than TOME to collect and deliver such information (about 4 *sec* vs 41 *sec* respectively).

In order to help showing the SAME capability to reveal an incident, we can automate the incident detection process (e.g., by building an app that automatically yields an alert signal at the traffic monitoring center). The idea is based on the number of zero speed vehicles in a short time frame. Let us define $T_{observation}$, which represents the period of time between a *cmc* message and the following one. Say that $T_{observation} = 1$ *sec*, and let $z[k]$ be the number of speed samples that fall below a threshold that represents “blocked” vehicles at the k -th $T_{observation}$ (e.g., we could set a threshold at 10 or 20 *km/h*). Then, an alert is triggered if the following condition is verified:

$$e + \sum_{i=k-9}^{k-5} z[i] < 2 \sum_{j=k-4}^k z[j]$$

where e is a sensibility constant that we set equal to 5, needed to protect the system from false positives when only few vehicles slow down. This calculation is performed every time a new *cmc* message is received (thus every $T_{observation}$). By applying this procedure, we can get aware of the incident in 4 *sec* for $p = 1$, 103 *sec* for $p = 0.75$ and 115 *sec* for $p = 0.5$, transmission times included.

VI. NEW PART: TO BE INTEGRATED WITH THE REST

We simulated a car incident on every sector of our scenario, for a total of 29 sectors, by blocking 2 out of 3 lanes of the highway at a precise time instant (one incident for every simulation). Our SAME protocol is running on the network for monitoring purpose and we measure the car incident detection time as the time that goes from the time instant in which the incident actually takes place to the instant in which the RSU detects the accident; hence all the transmission delays and the observation time to understand that an accident took place (see sec. ??) are already included.

VII. PERFORMANCE EVALUATION

In Fig. 8, we plotted the queue length directly from the SUMO simulation output, over the 2 lanes blocked by the car incident, as a result of the average queue length detected on every simulation, on the congested sector. We also plotted the incident notification time of our scheme, SAME, for the different market penetration rates, defined in sec. VI in relation to the queue matured on the road segment.

As we can see, Lane 0, which is the slowest lane on the right, has a very constant que grow rate, since most of the vehicles that arrive at the incident location just brake and enqueue behind the other ones. Lane 1, instead, has a different behavior, since after a first phase (from $t = 0$ to $t \approx 120$) where it grows just like Lane 0, then it starts being more inconstant, with faster increases and decreases, since vehicles try to overcome the incident by choosing the third lane on the left, which is free from the accident, but still used by the vehicles that flow on the road from behind. With a visual simulation inspection, this translates into watching vehicles that try to depart from Lane 1 to Lane 2 to overcome the incident and the blocked portion of the road.

It is very interesting to notice that SAME can actually inform the RSU of the incident when the queue is no more bigger than 25-30 m, which is a very interesting result, since this value can be granted also when the market penetration rate falls from 100% to 75%. Even when such market penetration rate value falls down to 50% of the vehicles equipped with this technology, the line does not exceed 70m per lane on the average.

This means that an accident can be detected and a reaction can be triggered when its consequences are still at a very early stage, before that the situation degenerates into a huge traffic jam.

In Fig. 9, instead, we evaluate the distribution of the incident detection time, grouped by 20s time windows. Again, we obtained those results as an average computed on the results of all the simulations on the different road segments where we simulated the incident.

It appears a clear and encouraging distribution pattern for all the market penetration rates. In particular, in Fig. 9(a), where it is equal to 100%, we show that more of the 50% of all the incidents are detected by the RSU within the first 20s after the accident actually happens and if we wait for 40s, about 80% of the accidents are revealed and reported. Again, this means that in combination with the results of Fig. 8, within 20s and a 25m queue on the average, a reaction can be triggered, like sending emergency vehicles on the place and redirecting the traffic over other routes. When the market penetration rate decreases to 75% (Fig. 9(b)), we still observe a very good reactivity of our system to the obstruction. In fact, in this case, within 40s about 75% of the accidents are revealed and reported, with the same detection time trend observed in Fig. 9(a). What makes this case different with respect to the previous one is that the notification is more spread over time: in some cases it may take up to 100s.

This happens because when the market penetration rate drops down, only few vehicles can overtake the accident, especially in the first time instants after the problem occurred. This means that there may be an occasional hole in the vehicles distribution, such that the wireless multi-hop chain breaks and the information cannot travel over the whole ring and reach the RSU back (notice that in this case, the other travel direction is not used to relay packets). The lesser the market penetration rate, the greater the probability that this event occurs, as we can see in Fig. 9(c), where the market penetration rate is 50%. In this case, in fact, there is no detection before 40s and only after that time frame we can observe a distribution which is similar to the previous cases.

REFERENCES

- [1] H. Payne and S. Tignor, "Freeway incident-detection algorithms based on decision trees with states," *Transportation Research Record*, no. 682, 1978.
- [2] C. L. Dudek, C. J. Messer, and N. B. Nuckles, "Incident detection on urban freeways," Tech. Rep., 1974.
- [3] S. A. Ahmed and A. R. Cook, "Application of time-series analysis techniques to freeway incident detection," *Transportation Research Record*, no. 841, 1982.

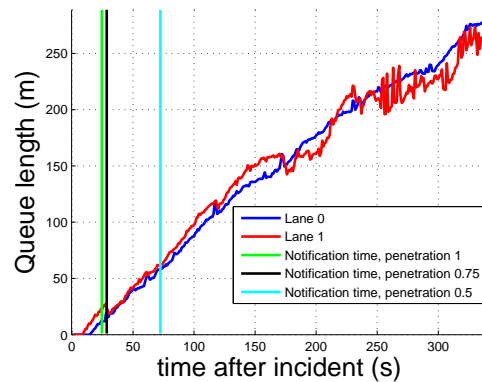
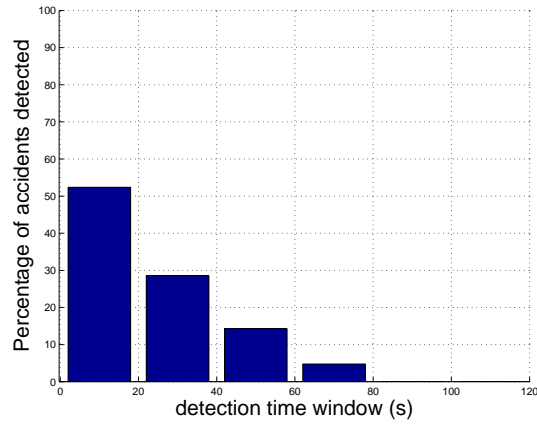
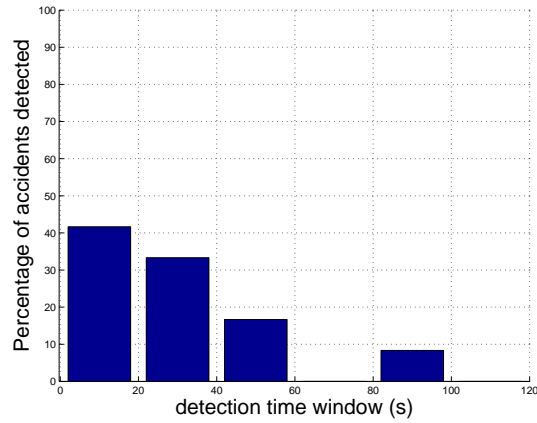


Fig. 8. Average queue length for both the lanes blocked by the accident. The vertical lines display the algorithm notification time as a function of the DSRC market penetration rate

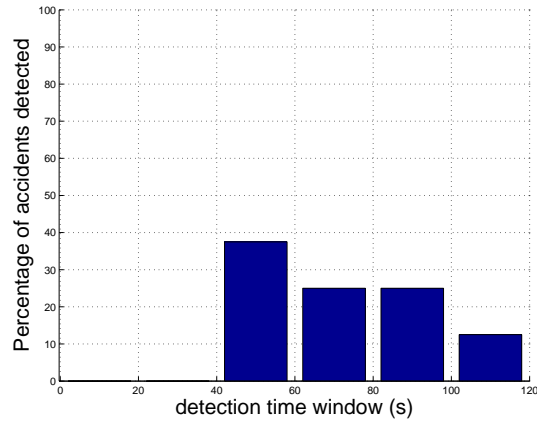
- [4] Y. J. Stephanedes and A. P. Chassiakos, "Freeway incident detection through filtering," *Transportation Research Part C: Emerging Technologies*, vol. 1, no. 3, pp. 219–233, 1993.
- [5] B. N. Persaud, F. L. Hall, and L. M. Hall, "Congestion identification aspects of the mcmastr incident detection algorithm," *Transportation Research Record*, no. 1287, 1990.
- [6] Y. J. Stephanedes and X. Liu, "Artificial neural networks for freeway incident detection," *Transportation research record*, no. 1494, pp. 91–97, 1995.
- [7] H. Adeli and A. Samant, "An adaptive conjugate gradient neural network-wavelet model for traffic incident detection," *Computer-Aided Civil and Infrastructure Engineering*, vol. 15, no. 4, pp. 251–260, 2000.
- [8] C.-K. Lin and G.-L. Chang, "Development of a fuzzy-expert system for incident detection and classification," *Mathematical and computer modelling*, vol. 27, no. 9, pp. 9–25, 1998.
- [9] C.-H. Hsiao, C.-T. Lin, and M. Cassidy, "Application of fuzzy logic and neural networks to automatically detect freeway traffic incidents," *Journal of Transportation Engineering*, vol. 120, no. 5, pp. 753–772, 1994.
- [10] S. S. Ishak and H. M. Al-Deek, "Fuzzy art neural network model for automated detection of freeway incidents," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1634, no. 1, pp. 56–63, 1998.
- [11] H. S. Mahmassani, C. Haas, S. Zhou, and J. Peterman, "Evaluation of incident detection methodologies," *Work*, p. 1, 1795.
- [12] M. W. Sermons and F. S. Koppelman, "Use of vehicle positioning data for arterial incident detection," *Transportation Research Part C: Emerging Technologies*, vol. 4, no. 2, pp. 87–96, 1996.
- [13] R. Long Cheu, C. Xie, and D.-H. Lee, "Probe vehicle population and sample size for arterial speed estimation," *Computer-Aided Civil and Infrastructure Engineering*, vol. 17, no. 1, pp. 53–60, 2002.
- [14] H. Dia and K. Thomas, "Development and evaluation of arterial incident detection models using fusion of simulated probe vehicle and loop detector data," *Information Fusion*, vol. 12, no. 1, pp. 20–27, 2011.
- [15] S. Geisler, C. Quix, S. Schiffer, and M. Jarke, "An evaluation framework for traffic information systems based on data streams," *Transportation Research Part C: Emerging Technologies*, vol. 23, pp. 29–55, 2012.
- [16] Y. Ma, M. Chowdhury, A. Sadek, and M. Jelihani, "Real-time highway traffic condition assessment framework using vehicle-infrastructure integration (vii) with artificial intelligence (ai)," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 10, no. 4, pp. 615–627, 2009.
- [17] I. Leontiadis, G. Marfia, D. Mack, G. Pau, C. Mascolo, and M. Gerla, "On the effectiveness of an opportunistic traffic management system for vehicular networks," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 12, no. 4, pp. 1537–1548, 2011.
- [18] J. B. Kenney, "Dedicated short-range communications (dsrc) standards in the united states," *Proceedings of the IEEE*, vol. 99, no. 7, pp. 1162–1182, 2011.
- [19] S. Panichpapiboon and W. Pattara-Atikom, "A review of information



(a) Penetration Rate=100%



(b) Penetration Rate=75%



(c) Penetration Rate=50%

Fig. 9. Percentage of accidents detected in 20s time windows, by considering several market penetration rate values

Workshop (ITA), 2013, 2013, pp. 1–9.

- [22] M.-T. Sun, W.-C. Feng, T.-H. Lai, K. Yamada, H. Okada, and K. Fujimura, "GPS-based message broadcast for adaptive inter-vehicle communications," in *Vehicular Technology Conference, 2000. IEEE VTS-Fall VTC 2000. 52nd*, vol. 6, 2000, pp. 2685–2692 vol.6.
- [23] M. Haklay and P. Weber, "Openstreetmap: User-generated street maps," *Pervasive Computing, IEEE*, vol. 7, no. 4, pp. 12–18, 2008.
- [24] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, "Sumo-simulation of urban mobility-an overview," in *SIMUL 2011, The Third International Conference on Advances in System Simulation*, 2011, pp. 55–60.
- [25] K. Fall and K. Varadhan, *ns notes and documents*, The VINT Project, UC Berkeley, LBL, USC/ISI, and Xerox PARC, February 2000.

dissemination protocols for vehicular ad hoc networks," *IEEE Communications Surveys Tutorials*, vol. 14, no. 3, pp. 784–798, 8 2012.

- [20] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. Sheu, "The broadcast storm problem in a mobile ad hoc network," in *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, ser. MobiCom '99. New York, NY, USA: ACM, 1999, pp. 151–162.
- [21] I. Rubin, A. Baiocchi, F. Cuomo, and P. Salvo, "GPS aided inter-vehicular wireless networking," in *Information Theory and Applications*