

Distributed Road Traffic Congestion Quantification Using Cooperative VANETs

Milos Milojevic and Veselin Rakocevic

City University London

{milos.milojevic.1, v.rakocevic}@city.ac.uk}

Abstract— The well-known traffic congestion problem in urban environments has negative impact on many areas including economy, environment, health and lifestyle. Recently, a number of solutions based on vehicle-to-vehicle communications were proposed for traffic congestion detection and management. In this paper we present an algorithm designed to enable each vehicle in the network to detect and quantify the level of traffic congestion in completely distributed way, independent of any supporting infrastructure and additional information such as traffic data from local authorities. Based on observations of traffic congestion by every vehicle, and by adapting the broadcast interval, it enables dissemination of the traffic information to other vehicles. The algorithm also makes every vehicle aware about the congestion level on the streets that are spatially separated from their current location by several streets. Its robustness keeps the vehicle's overall knowledge about congestion consistent, despite the short-term changes in vehicle's motion. Since the quantification of congestion is based on per-vehicle basis, the algorithm is able to operate even when only 10% of vehicles in the network are VANET enabled. Data aggregation and adaptive broadcasting are used to ensure that vehicles do not send redundant information about the traffic congestion. The simulations are conducted in Veins framework based on OMNeT++ network simulator and SUMO vehicular mobility simulator.

Keywords— *traffic congestion detection and quantification; vehicular ad hoc networks; intelligent transport systems; cooperation; data aggregation; dissemination;*

I. INTRODUCTION

Negative effects of traffic congestion in urban environments are well-known and they impact many areas of life including economy, environment, health and lifestyle. Traffic congestion can be caused by different factors from accidents to weather conditions, but when congestion happens the road infrastructure becomes too small to handle the large demand. Recently there have been several solutions proposed for traffic congestion detection and management, based on vehicle-to-vehicle communications and 802.11p standard [1].

One of the first papers dealing with distributed traffic congestion detection and management is [2], where traffic information system called SOTIS is presented. In SOTIS, each vehicle analyses traffic conditions based on the messages received from other vehicles, which include information about current speed, position, road identification and time. Periodic broadcasting is used as a way to disseminate messages and SOTIS was evaluated by simulation, only in the highway scenario, which evaluates only the delays of sent messages. No

data about traffic information, measurements and analysis is presented, neither any accuracy-related information. Traffic View [3] is another early published paper considering traffic information dissemination. It focuses on data aggregation in order to distribute the messages which contain information about the average speed of vehicles on the road, position and broadcast time. This way, vehicles are aware about other vehicles on the road. However, it does not provide and disseminate any information about intensity or volume of the traffic on the road. This approach was simulated only based on 802.11b standard and in highway environment. Another work related to distributed V2V traffic congestion detection and forecasting algorithm is presented in [4]. Authors define the road is in congested state if the travel times of vehicles are exceeding normal travel times under free flow traffic. Additionally, the scheme requires that each road segment needs to be observed for a day and that vehicles send their traversal times to a centralized entity. The results show the analysis of traversal time of vehicles in different simulation scenarios, but do not relate to intensity and volume of the traffic. In [5] authors presented cooperative approach for congestion detection (CoTEC) which is based on fuzzy logic, where vehicles periodically broadcast messages. The congestion is detected based on vehicle's speed and the received broadcast messages from other vehicles, together with external metrics called "level of service" (LOS) for classification of traffic congestion developed by Skycomp. This system defines metrics based on aerial surveys of different highways. Congestion detection algorithm has been evaluated in highway scenario. Authors in [6] developed a distributed V2V system, where vehicles send messages containing information based on travel time on the road section. The traffic estimation is evaluated on the basis of trip time analysis, but it is not evaluated how the scheme influences the dissemination of the messages and their accuracy. The authors concluded that real-time and up to date traffic information can reduce the traffic congestion in realistic scenario. Another cooperative solution for traffic congestion detection based on VANETs is presented in [7], where authors use event-driven architecture (EDA) to detect different levels of traffic jams. It showed good results although the system performance highly depends on the VANET penetration rate and the scheme uses periodic broadcasting. In [8] authors presented V2V-based congestion detection scheme which detects the traffic jam based on travel times of the vehicles. It uses geo-cast flooding-based protocol which uses request and response messages. Authors of [9] present a scheme based on periodically broadcasted beacons which contain speed, which are then used to calculate if the

road is congested or not. Paper [10] presents traffic congestion detection mechanism based on VANETs. The scheme shows good results in the case of low VANET penetration rate however it requires initial observation of the streets with external device.

As previously mentioned, solving the traffic congestion problem by using VANETs has been one of the hot research topics recently. The presented solutions based on V2V communications present the following important limitations:

- Some of the proposed solutions [4] [5] [10], rely on extra information about traffic conditions obtained either from third party companies or local authorities, which are then feed to the algorithms. The extra information sometimes is gathered by centralized entities such as traffic centers, and moreover the schemes in this case also depend on the accuracy of such information.
- Certain schemes, like [2] [7] [9], assume periodic broadcasting as a way of sending the messages without considering adaptation of broadcast interval. This is especially important because in traffic jams it might lead to broadcast storm, collision and inefficient message dissemination [11]-[13].
- Some schemes, [2] [3] [4] [6], only defer congested from non-congested states, and they do not provide any information about the intensity and duration of congestion, but just offer analysis of travel times of vehicles. Additionally, the authors mostly assume that vehicles exchange messages containing data such as speed, direction or coordinates which receiving vehicles then further need to process to determine if and where congestion exists.
- Some authors the failed to show an insight how their schemes disseminate traffic related information [6], and no analysis of accuracy and delay of received information from other vehicles is given.

Some of the papers in related work have one of previously described problems, but some of them have more or even all of them. We propose an algorithm designed to enable each vehicle in the network to detect and quantify the level of traffic congestion in completely distributed way, independent of any supporting infrastructure and additional information such as traffic data from local authorities. It is based on observations of traffic conditions by every vehicle and adaptive broadcasting as a way to disseminate the traffic information to other vehicles. All the vehicles are also aware about the congestion level on the streets spatially separated from their current location. Its robustness makes vehicles aware of the overall congestion level in certain street, despite the short-term changes in vehicle's mobility. The congestion quantification process is based on per-vehicle basis, making the algorithm able to operate even in case of low VANET penetration rate. Data aggregation and adaptive broadcasting, ensures that vehicles do not send redundant information about the traffic congestion.

The rest of the paper is organized as follows: Section II shows the algorithm, while we present results of extensive

simulations and testing in section III. The Section IV shows analysis of the low VANET penetration rate impact. Conclusions together with future steps in our research are presented in section V.

II. THE ALGORITHM

Having in mind the previously described problems and the limitation of some of the existing proposals for congestion detection by VANETs, we envisaged our algorithm to have several characteristics and capabilities:

- It needs to be completely distributed and independent from any kind of infrastructures and external systems. This means that we want to create an algorithm which will enable every vehicle to detect and quantify the congestion on its own without necessary help from others. In our opinion this is important because not all the vehicles will be VANET enabled and able to communicate with others.
- It needs to be efficient in terms of message exchange and needs to deploy data aggregation mechanism in order to reduce the number of messages exchanged. This is especially important in order to reduce the number of sent messages while in traffic congestion because every vehicles would send similar information about the congestion. On the other hand it needs to disseminate the messages to all vehicles even if they are not in proximity.
- Each vehicle needs to attempt to obtain the knowledge about the congestion level in other streets, even the ones it does not drive through.

Therefore we divided our algorithm in two mechanisms: congestion detection and quantification, and information dissemination, both of which will be presented.

A. Congestion Detection and Quantification

Firstly, it needs to be defined what traffic congestion is, what causes it and how can it be measured. Even though everyone from drivers to pedestrians can recognize traffic congestion, it has to be formally defined. One of the most common definitions of traffic congestion used in literature is the one from [15], where congestion is defined as the travel time or delay in excess of that normally incurred under light or free-flow travel conditions. For VANETs research is also very important to define how to measure the traffic congestion. In [16] authors presented causes of traffic congestion in urban environments and extensive survey of ways of measuring it, and concluded the paper with the comparison of different measurement metrics. Some of the most widely used measures are speed, travel time and delay, volume, level of service, demand and capacity, cost, etc. The authors concluded that congestion is a function of reduction in speed, and that the setting of a threshold that is directly related to travel speed is most appropriate to use as a metric of traffic congestion. We agree that using the speed as congestion indicator in distributed VANETs is most appropriate because each vehicle can measure it with no additional infrastructure or external systems required.

Therefore according to our algorithm, every vehicle measures its own speed and time during which the speed is lower or higher than the threshold V_t . In order to enable the quantification of vehicles mobility by measuring the speed, we introduce the congestion level C_L . C_L can have discrete number of values, in our case from 1 to 10. The values from 2 to 10 are reserved for situations when the speed is lower than the threshold, while the value 1 is left for case the speed is higher than V_t . Apart from speed, the C_L is determined by time T , during which vehicle has speed greater or lower than the threshold, according to:

$$\text{when } V_c > V_t \text{ and } T > 10s \text{ then } C_L = 1, \quad (1)$$

$$\text{when } V_c \leq V_t \text{ and } T = \eta \cdot 20s, \eta = \{2, 3, \dots, 10\} \text{ then } C_L = \eta. \quad (2)$$

According to the equations, vehicle will calculate congestion level only when its speed has been below or above the threshold for the specified period of time. Moreover, in the case when speed is lower than V_t , it needs to stay that way for η consecutive time intervals of 20s. The length of time intervals and number of congestion levels can be set to handle any type of scenario, but for this purpose we adopted to have ten values of congestion levels and set the time intervals to 10s when V_c is greater than V_t , and 20s when its lower. The result of congestion quantification of one vehicle is shown in Fig.1, where the speed is shown together with congestion level. The V_t in this case is set to 6 meters per second, and as soon as vehicles speed becomes lower than V_t for period of time longer than 20s, the congestion level starts to increase. As soon as the speed passes V_t the congestion level drops to lowest level.

The positive side of such approach compared to schemes based on the traversal time measurement is that vehicles need to have upfront information about some default values of traversal times for each street. Since this is not practically feasible, especially having in mind that standard free-flow traversal time change during the day, our approach makes congestion quantification independent from such assumptions. It is important to emphasize that vehicle's speed is directly influenced by the level of traffic congestion around it, and is a reason more to use it for quantification of traffic congestion. Additionally, since our scheme quantifies the level of congestion before sending it to the neighboring vehicles, the information in the message is already processed and results in less processing times of messages at receiving vehicles, which is particularly important for delay-sensitive applications.

B. Information Dissemination

Now that each vehicle is able to quantify its motion and therefore the congestion level around it, it is important to share this information with other vehicles in order to make the congestion detection and quantification algorithm complete. Before we explain the dissemination algorithm, it needs to be pointed out that we assume that each vehicle is equipped with a GPS device and on-board unit (OBU) in order to determine its location. Also we assume that each street section has its own identification A_{id} which is always known to each vehicle. Example of couple of street sections is shown on Fig. 2. Finally, all vehicles have their own databases where they store messages received from other vehicles and sort them according to the A_{id} .

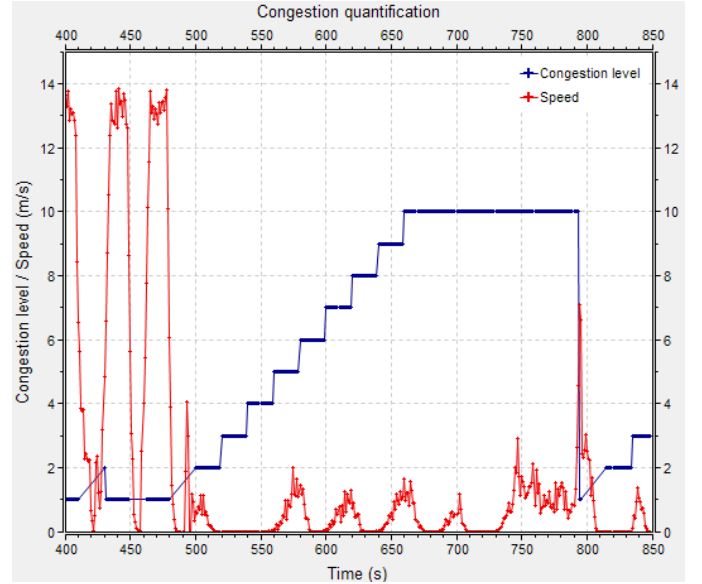


Fig. 1. Congestion quantification result and congestion level of a vehicle against its speed.

We propose to use adaptive broadcasting scheme that adjusts the broadcast interval according to the difference of congestion level for the current street that vehicle calculated on its own and the level from the database that includes values received from other vehicles as well. According to our scheme each vehicle will broadcast the message containing the values of congestion level of the street where it is currently located and for the previous street it was located before the current street. Therefore the data in the message looks like this:

$$(A_{idk}, C_{Lk}, A_{id(k-1)}, C_{L(k-1)})$$

Where A_{idk} and $A_{id(k-1)}$ are identifications of current and previous street section respectively, while C_{Lk} and $C_{L(k-1)}$ are the congestion levels for those street sections. This message will be sent only when the C_L is greater or equal to C_D , which is congestion value from the database for the same street section A_{id} . This is due to the nature of traffic congestion and its inertia, because congestion cannot actually disappear instantly and requires some time to disappear even when the input of vehicles suddenly drops.

Finally, when vehicle receives the message it is important to store it in the database. There are many ways of storing it in the database, for example to store the latest data only. This is not really safe because the received information about congestion depends solely on one vehicle's message, which can be wrong, malicious, etc. Therefore we chose that each received message contributes towards overall knowledge about the congestion. Data aggregation mechanisms are used in such cases and we chose simple averaging as an aggregation method. When vehicle receives the message about congestion level on some street it will store it in the database by calculating the average of current value in the database and the received value. Similarly, when vehicle determines the congestion level on its own, the value will be stored in the database by calculating the average of determined value and the value from the database. By this aggregation and

broadcasting mechanism we achieved the overall aggregation effect, instead of all vehicles sending almost the same information about the same street section. This enables that relevant traffic information is transmitted more efficiently contributing towards less collisions and solving the broadcast storm problem.

III. SIMULATION SETUP AND EVALUATION RESULTS

We conducted extensive simulations in various scenarios in order to evaluate our scheme for congestion quantification. For this purpose, we used bi-directionally coupled OMNeT++ 4.3 [17] network simulator and SUMO 17.0.1 [18] traffic mobility simulator. The integration of these two simulators is achieved in Veins 2.1 [19] framework, which is also specifically designed for vehicular networks and supports 802.11p standard. Additionally it also provides real-time feedback between network and traffic simulators, which is especially important for realistic simulations.

A. Simulation Scenario Setup

Firstly, we designed road traffic scenario with SUMO simulator by generating road network of one square kilometre consisted of five streets orthogonally intersecting another five streets, where each of them is 1km in length with two lanes in each direction, 2m wide each. This gives 80 street sections, each 250m long with two lanes inside, as shown in Fig.2, and having unique identification used in the algorithm, while vehicles are enabled to go in all three directions at intersections (left, right and straight). We created mobility scenario using VACaMobil [20] module based on the Veins framework which enables generation of random routes in SUMO for vehicles in simulation and also keeps the number of vehicles in the simulation constant. We generated 10,000 random routes on the road network each consisting of several consecutive street sections and set the number of vehicles to be constantly kept at 300. When vehicles reach the end of the route whilst simulation is running they exit the simulation and the new one is generated instead. We set the duration of simulation to 1000s, and the total number of vehicles generated was 1,500. The maximum speed was set to 50km/h, but since SUMO simulates the traffic lights as well, the speed was changing with time, while the length of the car was set to 5m. Having in mind this fact, we calculated that the maximum number of vehicles per street section is 90, which we get when we divide 250m length of the section with the length of vehicle and safety distance of 0.5m added. This number of maximum vehicles per section is important for the results, which is explained in the following section.

After the generation of the mobility routes, the network simulation was set up in the Veins framework and VACaMobil module. We implemented our algorithm as an application layer that sends data packets, while we disabled sending beacons in order to simplify simulation process due to large number of vehicles and results evaluation complexity. As per literature about VANETs, the applications are periodically broadcasting data or beacons and most of the scheme use fixed broadcasting interval. Since our algorithm assumes the adaptation of that interval, its length will depend on the level of congestion on the road and also the content of the database. Since the scope of the

paper focuses on congestion quantification algorithm and its accuracy, the full evaluation of broadcasting algorithm and its impact on networking parameters is not provided in this instance. Finally, in order to fully test the capabilities of our algorithm, we implemented it in scenario when penetration rate is only 10%, meaning that the number of vehicles on the road stays the same, but the number of vehicles participating in congestion detection is only 30, out of 300 vehicles in total. Finally, we set the speed threshold V_t for congestion detection to 21.1km/h.

B. Results

Firstly, we show the result of the congestion quantification algorithm which is being done by every vehicle in the simulation. Fig. 1 shows how the algorithm works and the congestion level of one vehicle is shown, and can be seen that it follows the vehicles mobility and at certain point when the speed has been lower than the V_t threshold for longer period of time, the congestion level starts to increase. As time elapses the speed of the vehicle has surpassed the threshold which means that the congestion becomes more severe and therefore congestion level is increasing. In this particular example, the level reached the maximum and stayed there until speed became greater than V_t . This shows that each vehicle is able to quantify its movement based only on its current speed and to detect the level of traffic precisely, which can be shared with other vehicles in the network.

Since every vehicle can detect the level of congestion, it can share this information with other vehicles according to our algorithm. When one of the vehicles receives the information it stores it in its database for that particular street section. In Fig. 3 we show the comparison of database and own congestion detection result achieved by single vehicle.

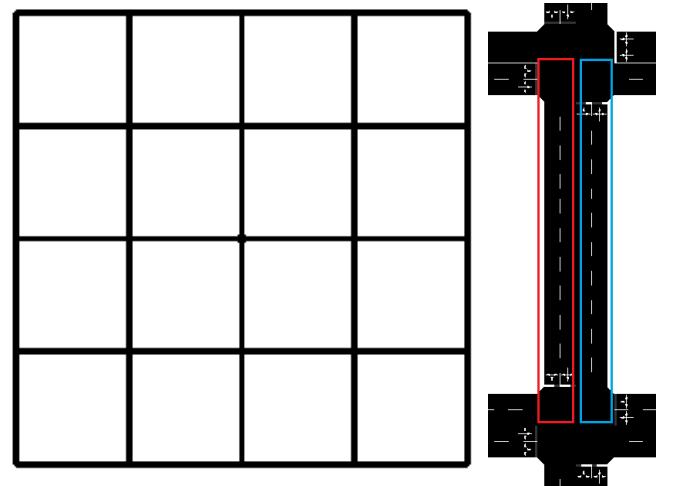


Fig. 2. The road network consisting of 80 street sections, and two street sections of the road network one in each direction, where each section has two lanes in same direction, as shown on the right.

The result refers to one vehicle and its entire route, which in this case consisted of six street sections. The route and the real number of vehicles present on the street sections on the route, is shown in Fig. 4, in order to compare it with the results of the congestion detection from Fig. 3. As it can be seen, that

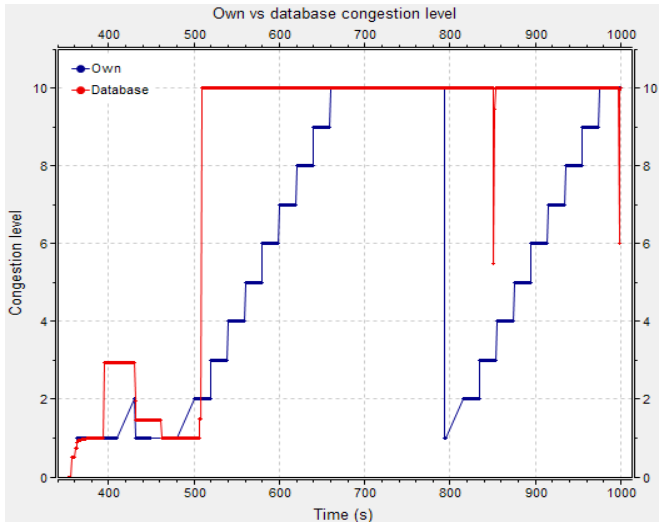


Fig. 3. Congestion level that vehicle calculated on its own and congestion level in the database received from other vehicles.

Street 1 and Street 2 have a relatively small number of vehicles on the road. At around 400s, the number of vehicles on the street increases as the vehicle enters Street 3. In this street the number of vehicles was higher, but decreased later and all this was detected by our algorithm as well. Later vehicle went through Street 4 and Street 5 where number of vehicles was low, and this also corresponds to the results of congestion detection. Finally vehicle entered the Street 6 where number of vehicles was very high, which was also detected by our algorithm. As can be seen the vehicle's database was already filled with this information, since it received this information from other vehicles. Also vehicle's own congestion level eventually resulted the same way, when enough time passed in the same condition. Around 800s the vehicle's congestion level suddenly dropped due to speed increase, however since this was obviously short-term change, value of database remained

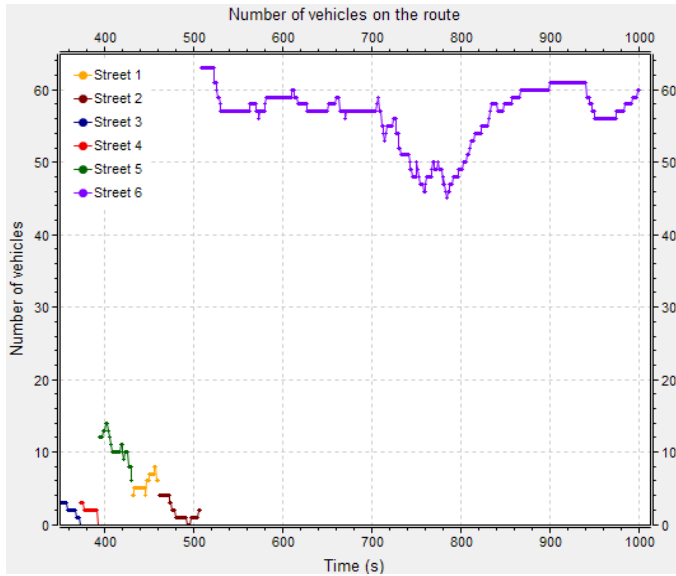


Fig. 4. The real number of vehicles on the route consisting of six streets sections.

the same, proving that even when a vehicle's short-term speed changes might influence its own congestion level but not the actual picture about the congestion. This shows that our algorithm has robustness which is extremely important having in mind very dynamic vehicle's mobility in urban environment. It is also important to notice that the number of vehicles on the Street 6 goes up to 64, which is about 71% of the street capacity, of 90 vehicles, meaning that the street was almost fully filled with vehicles. To see the overall performance of the algorithm we show the average congestion level from all vehicles in the simulation for three chosen streets and compare them with the real number of vehicles those streets. In Fig. 5, the streets with low, medium and high traffic we compare to the congestion detection results for same streets from Fig. 6.

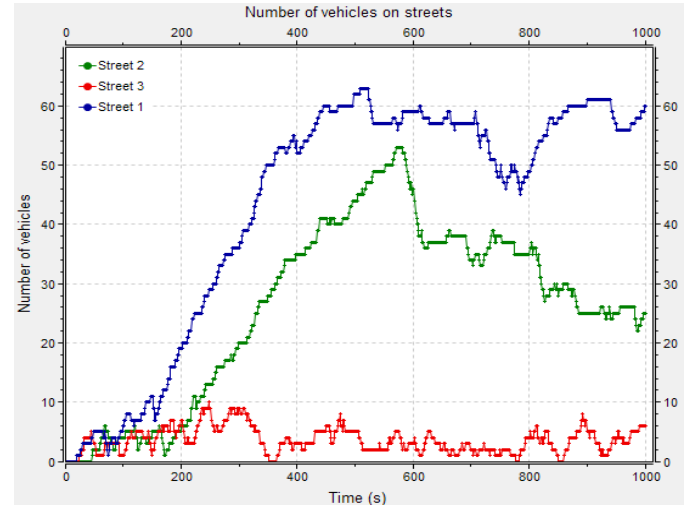


Fig. 5. The real number of vehicles on three streets, with low, medium and high traffic demand.

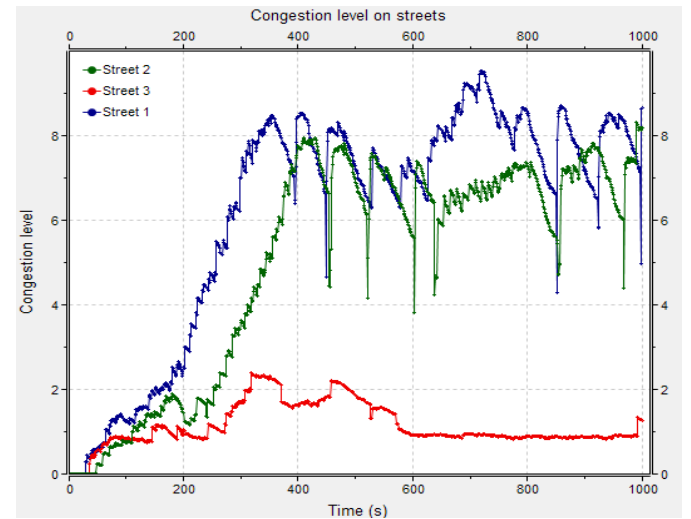


Fig. 6. Average congestion levels for three streets calculated as an average of databases of all vehicles in the simulation.

As per Fig. 6 it can be seen that the congestion level corresponds to the actual number of vehicles on these streets from Fig. 4 and that relative position of congestion levels corresponds to relative positions of traffic volume on those

streets. To show how spatial distance between the vehicles impacts the accuracy of the information we present the results on Fig. 7. There we show databases of five vehicles for the same street section, but each vehicle being at different location at the time. One of them passes through that street, the other passing through the street that is neighboring to the street, for which we refer to as 1 hop away, where hop represents one street section. Similarly we show 2, 3 and 4 hops away neighbors as well. It is obvious that the vehicle in 1-hop-away street will receive the correct information in real-time with no delay, while others will receive the information with a delay, but this information will eventually become correct. Finally, we show Fig. 8 and Fig. 9 where the average number of sent packets per vehicle is presented during the time of simulation. Fig. 8 shows the results when our algorithm is used to adapt the broadcast interval, while Fig. 9 shows the results when periodic broadcasting is used. The comparison of Fig. 8 and Fig. 9 show significantly lower number of sent packets per node, when our algorithm is used.

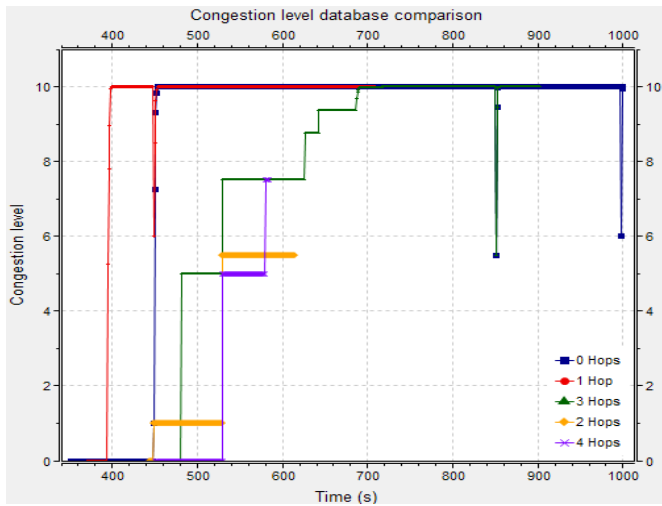


Fig. 7. Congestion levels about the same street of vehicles going through that street and from vehicles going through streets that are 1, 2, 3 and 4 hops away.

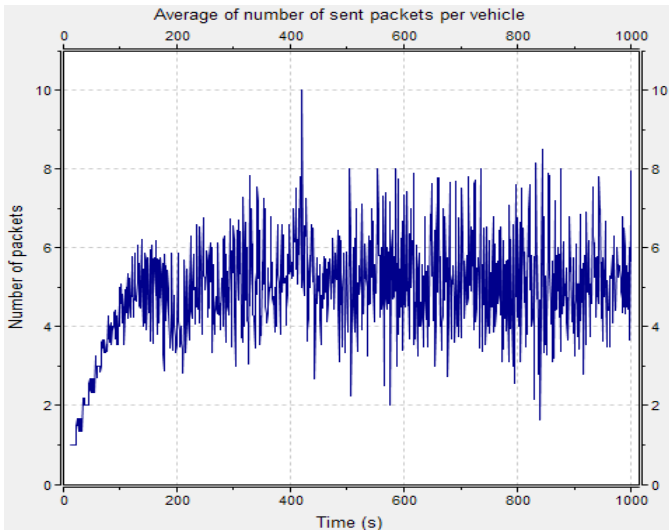


Fig. 8. Average number of sent packets per vehicle during simulation.

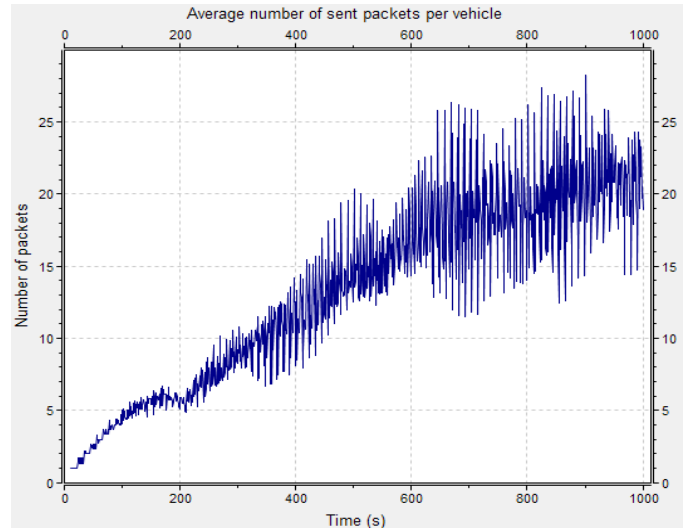


Fig. 9. Average number of sent packets per vehicle-periodic broadcasting.

IV. IMPACT OF LOW VANET PENETRATION RATE

As previously stated one of the main contributions of our algorithm is that it is able to work in case of low VANET penetration rates. This is especially important because when implementation of the V2V systems begins, not all vehicles will be able to communicate and contribute towards congestion detection and quantification. Due to the fact that each vehicle is able to quantify congestion on its own, independently from others in the network, our algorithm will enable the vehicles which are equipped with OBUs to be aware about the level of traffic congestion around them. However, each vehicle will have better picture about traffic congestion when more vehicles are participating in the network. Therefore, we tested the algorithm to see how algorithm behaves in the low VANET penetration rate situation to check if the accuracy is satisfactory. In the following analysis we set the penetration rate to 10%, meaning that during the simulation only 30 vehicles will be able to communicate with each other and to detect the congestion, while the total number of vehicles in the network will remain the same, 300.

First we present Fig. 10 where we show own congestion level of the vehicle and the level of congestion from the database. Here it can be seen that vehicle will receive the congestion information before it finds out about the congestion on its own. Fig. 11 shows the number of vehicles per street sections on the route that vehicle travelled on. On streets 1, 2, 3, and 4 the number of vehicles is very low which is confirmed by the vehicle's own congestion measurement and the database congestion level as well. The street 6 also has a small number of vehicles however the congestion algorithm shows that both own and database level started increasing which is not correct and happened due to low penetration rate. Then vehicle enters street 7 which crowded with vehicles and own congestion level detects it after some time. On the other hand database congestion level has slight delay of around 10 second before it reaches the maximum level. This is again due to very low penetration rate, because there was no, or was very few vehicles in that street before the vehicle entered it. The results

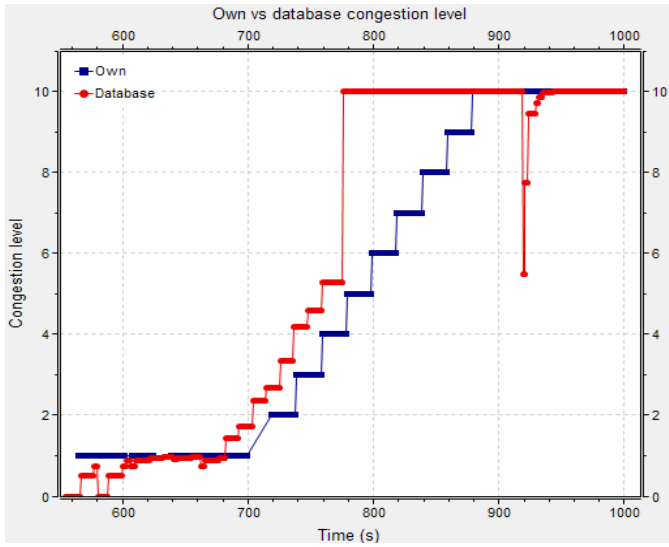


Fig. 10. Congestion level that vehicle calculated on its own and congestion level in the database received from other vehicles for 10% VANET penetration rate.

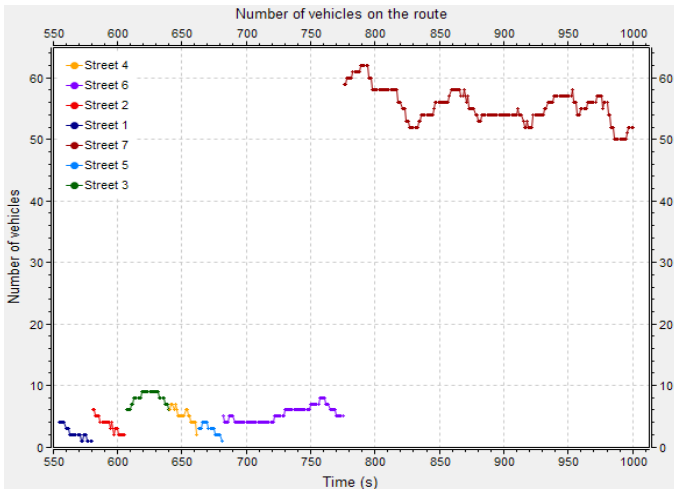


Fig. 11. The real number of vehicles on the route consisting of seven street sections.

showed the low VANET penetration rate does not influence vehicle's own congestion level calculation, but influences the knowledge about the congestion in the database, in this case leading to slight temporarily inaccuracy. This is because database is filled with messages from other vehicles as well, whose number is in this case only 30.

To see the overall performance Fig. 12 and Fig. 13 respectively show the average value of congestion databases of all vehicles for streets 1, 2 and 3, and the real number of vehicles on these streets during the simulation. As per those figures, the algorithm performed well on two streets, with low and medium number of vehicles. However in case of Street 2 which is the busiest, average value of the detected congestion in Fig.12 does not correspond entirely to real number of vehicles on Fig. 13 during whole simulation. Apart from that time period the results correspond to the real number of vehicles on the street. Fig. 14 shows the value of the congestion

level for one street and four vehicles, with the one of them going through the street, while others being 1, 2 and 3 hops away. It is shown that vehicles travelling through the streets that are 1 and 2 hops away from the observed street will receive correct information in real-time. However, the vehicle travelling through the street being 3 hops away, will receive delayed information which converges towards correct one, but vehicle left the simulation before it reached the correct value. Therefore, the delay is the consequence of low number of vehicles being able to communicate in the network.

Fig. 14 shows the average number of sent packets per vehicle during the simulation. The results show that since there are ten times fewer vehicles participating in overall congestion detection, each vehicle needs to send more packets. Compared to the previous case when all vehicles participated in congestion detection, each vehicle sent approximately two times more packets. This is, however still considerably less number of sent packets per vehicle than in case of simple periodic broadcasting as shown on Fig. 9.

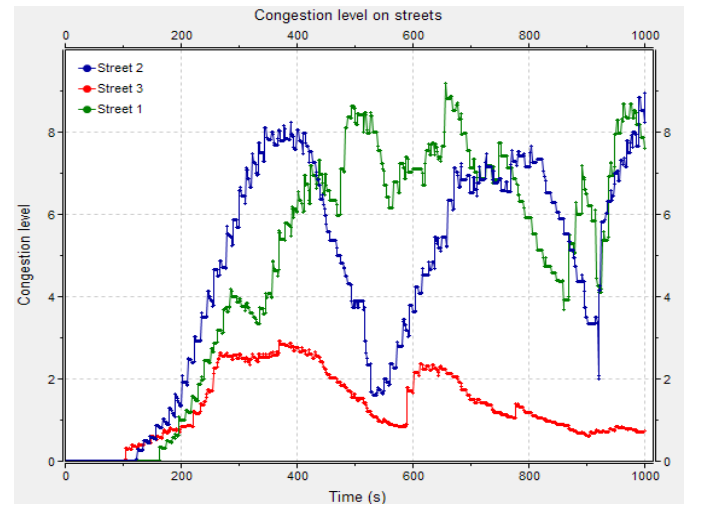


Fig. 12. Average congestion levels for three streets calculated as an average of databases of all vehicles in the simulation.

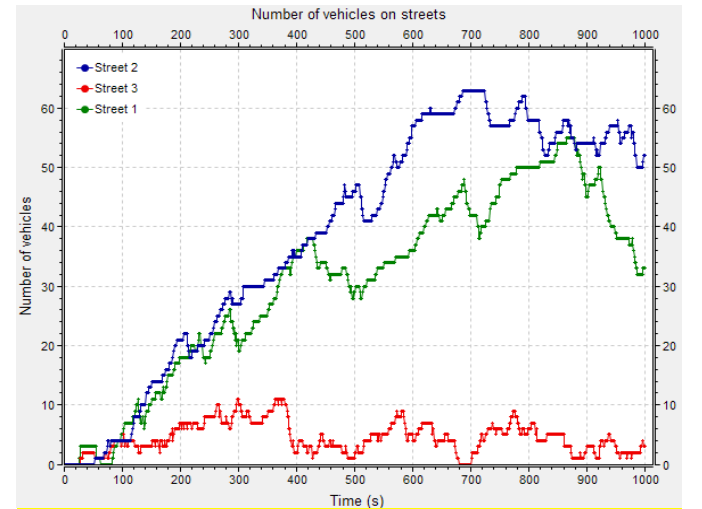


Fig. 13. The real number of vehicles on three streets, with low, medium and high traffic demand.

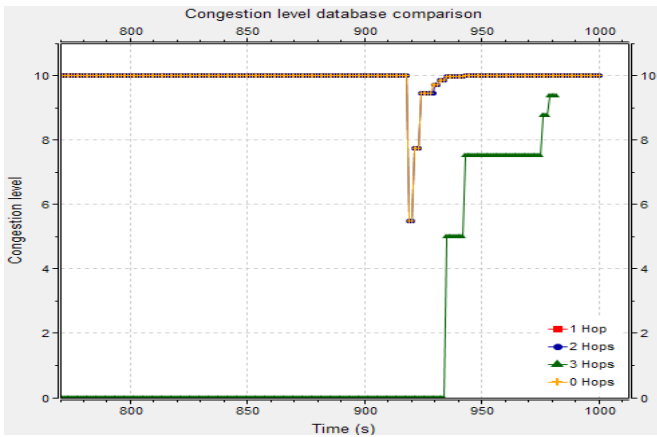


Fig. 14. Congestion levels about the same street of vehicles going through that street and from vehicles going through streets that are 1, 2 and 3 hops away.

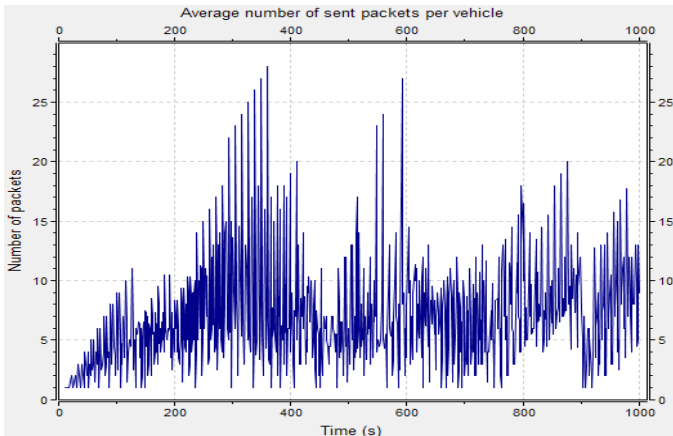


Fig. 15. Average number of sent packets per vehicle during simulation

V. CONCLUSION AND FUTURE WORK

In this paper we presented an algorithm designed to enable each vehicle in the network to detect and quantify the level of traffic congestion in completely distributed way, independent of any supporting infrastructure and additional information such as traffic data from local authorities. It is based on observations of traffic conditions by each vehicle and adaptive broadcasting as a way to disseminate the traffic information to all vehicles. The algorithm enables vehicles to be aware about the congestion level on the streets that are spatially separated from their current location by several streets. It is robust enough to keep the overall knowledge about congestion in certain street consistent despite the short-term changes in vehicle's mobility. Since the quantification of congestion is based on per-vehicle basis, the algorithm is able to operate even in case of low VANET penetration rate of 10%. Finally, because of data aggregation and adaptive broadcasting, the algorithm ensures that each vehicle sends fewer packets than in case of periodic broadcasting, which contributes towards reducing the network load and collisions in the wireless channel. For the future work we plan to further investigate the adaptation of broadcast interval and compare it to other adaptive broadcast schemes used in VANETs. Additionally, we

plan to implement the congestion quantification algorithm in vehicle routing and examine how it reflects on trip time.

REFERENCES

- [1] 802.11p-2010 - IEEE Standard for Information technology-- Local and metropolitan area networks-- Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 6: Wireless Access in Vehicular Environments, 2010.
- [2] L. Wischhof, A. Ebner, H. Rohling, M. Lott and R. Halfmann, "SOTIS - a self-organizing traffic information system," IEEE Vehicular Technology Conference 2003, VTC 2003-Spring.
- [3] T. Nadeem, S. Dashtinezhad, C. Liao, and L. Iftode, "TrafficView: a scalable traffic monitoring system," IEEE International Conference on Mobile Data Management, 2004.
- [4] G. Marfia, and M. Rocchetti, "Vehicular congestion detection and short-term forecasting: a new model with results," IEEE Transactions on Vehicular Technology, Vol. 60, Issue 7, 2011.
- [5] R. Bauza, J. Gozalvez, and J. Sanchez-Soriano, "Road traffic congestion detection through cooperative vehicle-to-vehicle communications," IEEE 35th Conference on Local Computer Networks (LCN), 2010.
- [6] 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", IEEE Transactions on Intelligent Transportation Systems, Vol. 12, Issue 4, 2011.
- [7] F. Terroso-Saenz, M. Valdes-Vela, Cristina Sotomayor-Martinez, R. Toledo-Moreo, A. F. Gomez-Skarmeta, "A cooperative approach to traffic congestion detection with complex event processing and VANET," IEEE Transactions on Intelligent Transportation Systems, Vol. 13, No.2, June 2012.
- [8] A. Lakas and M. Cheqfah, "Detection and dissipation of road traffic congestion using vehicular communication," Mediterranean Microwave Symposium (MMS), 2009.
- [9] F. Knorr, D. Baslet, M. Schreckenberg, M. Mauve, "Reducing traffic jams via VANETs," IEEE Transactions on Vehicular Technology, Vol.6, No. 8, October 2012.
- [10] C. Wang and H. Tsai, "Detecting urban traffic congestion with single vehicle," International Conference on Connected Vehicles and Expo '13.
- [11] N. Wisitpongphan, O. K. Tonguz, J. S. Parikh, P. Mudalige, F. Bai, and V. Sadekar, "Broadcast storm mitigation techniques in vehicular ad hoc networks", IEEE Wireless Communications, Vol. 14, Issue 6, 2007.
- [12] X. Ma, J. Zhang, and T. Wu, "Reliability analysis of one-hop safety-critical broadcast services in VANETs", IEEE Transactions on Vehicular Technology, Vol. 60, Issue 8, 2011.
- [13] T. Kosch, C. J. Adler, S. Eichler, C. Schroth, "The scalability problem of vehicular ad hoc networks and how to solve it," IEEE Wireless Communications, October 2006, Issue 5, Vol. 13.
- [14] M. Milojevic and V. Rakocovic, "Short Paper: Distributed vehicular traffic congestion detection algorithm for urban environments," IEEE Vehicular Networking Conference 2013.
- [15] T. J. Lomax, "Quantifying congestion," NCHRP Report 398, Volume 1, 1997.
- [16] A. M. Rao and K. R. Rao, "Measuring urban traffic congestion - a review", International Journal for Traffic and Transport Engineering, 2012, 2(4).
- [17] OMNeT++, www.omnetpp.org
- [18] D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker, "Recent Development and Applications of SUMO - Simulation of Urban Mobility," International Journal On Advances in Systems and Measurements, 5 (3&4):128-138, December 2012
- [19] C. Sommer, R. German, and F. Dressler, "Bidirectionally coupled network and road traffic simulation for improved IVC analysis," IEEE Transactions on Mobile Computing, vol. 10 (1), pp. 3-15, 2011.
- [20] M. Baguena, S. M. Tornell, A. Torres, C. T. Calafate, J. Cano, P. Manzoni, "VACaMobil: VANET car mobility manager for OMNeT++," IEEE International Conference on Communications 2013.