



# Performance Evaluation of Citywide Intersections Traffic Control Algorithm inVANETs-Based

Sarah Hasan<sup>(✉)</sup> and Mourad Elhadef

College of Engineering, Abu Dhabi University, Abu Dhabi, UAE  
1003299@students.adu.ac.ae

**Abstract.** The massive improvement in wireless communications pertains a real time and accurate delivery of information, which makes it possible to remotely control and manage a wide number of applications and services. The ability to connect to mobile and fast-moving nodes can aid in providing or obtaining information from vehicles that in turn provides a diverse picking in the development of vehicular communications and control. This includes all type of vehicular communications such as Vehicle to Infrastructure (V2I) and the Vehicle-to-Vehicle (V2V) communications. In this paper, we try to enhance traffic flow at intersections by simulating an improved VANET-based control algorithm. The study focuses on the flow of traffic across multiple adjacent intersections in a city where each intersection is equipped with a Roadside Unit (RSU). We believe that the communication between RSU at a given intersection and nearby vehicles, RSU and other surrounding RSUs (RSU2RSU) will affect the flow of the vehicles positively. We also consider the need to minimize the required time to cross an intersection particularly if a vehicle is an emergency vehicle.

**Keywords:** Vehicular ad-hoc networks · Intersection traffic control · Smart city · Intelligent transportation system · Internet of vehicles

## 1 Introduction

The overwhelmed city roads are a priority to study and analyze to improve the flow in the streets. One of the most important areas in traffic control is managing the flow at intersections. Traffic lights are the oldest form of traffic control at intersections. In their earlier version, traffic lights implement a prefixed time interval, which can lead to a delay especially if the green light is open for an empty street. The newer version of the prefixed time interval can be customized based on the load of a street and timing of the day or any other preset condition. This means that the timing for each intersection must be set separately nevertheless it also cannot perform well in case of change in the traffic flow due to any reason such as weather or holidays. The modern traffic lights implement an intelligent time-intervals by modifying the timing when the road load or condition changes. This type of traffic lights is known as reactive as it reacts to the new road condition and adjust the crossing time. The reaction is based on the collected road information by using different devices such as microwave detectors, loop detectors,

radars and laser beams, geomagnetic vehicle detectors, and video/image detectors. The main problem of reactive-traffic lights is the high cost of purchasing, fixing and maintaining the devices [1–4]. The modern intersections traffic control is based on using wireless communication technologies such as the IEEE802.11p standard and the IEEE1609.4 which tolerates Multi-Channel Operations with Wireless Access in Vehicular Environments (WAVE) [5]. The standardization of vehicular communication makes it easier to provide centralized-based and distributed-based traffic management. In the case of the intersection control, vehicles mostly communicate with RSU or/and with other vehicles to receive instructions [6–9]. Figure 1 summarizes some of the contributions in the area of intersection control [10–15]:

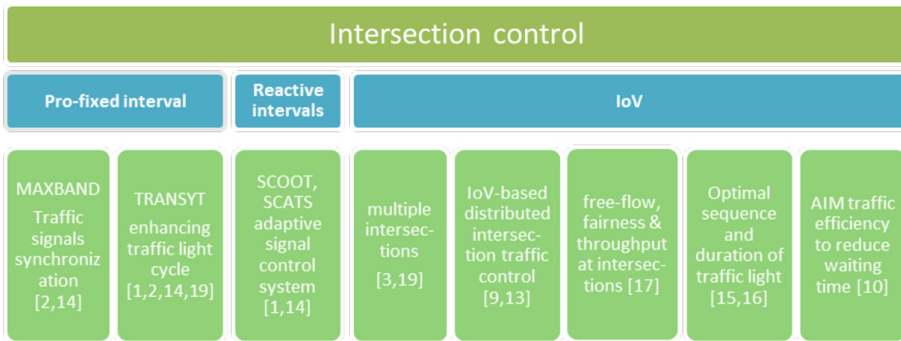


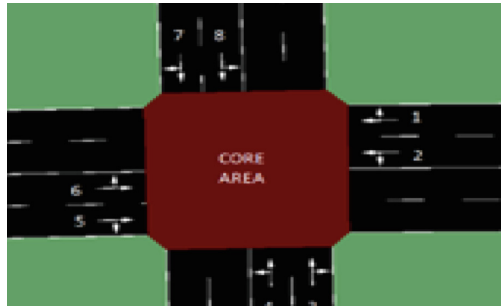
Fig. 1. Advancements in intersection control

To contribute to the development of intersections traffic control, we tested and improved the centralized intersection algorithm for controlling traffic at intersections by using VANET- based communications (V2I and RSU2RSU) in a citywide level. Our work is a continues improvement to the centralized algorithm in [16] which is based on a study suggested by [8]. We extended the scope of the study to include multiple number of RSUs in a citywide level to guarantee an efficient mutual exclusion; increase throughput; prioritize emergency and public transportations. The paper is organized as follow. Section 2 states preliminary elaboration of the inVANETs-based intersection control algorithm. Section 3 describes the model of the enhanced inVANETs-based intersection control algorithm in citywide. Section 4 provides the simulation results. Section 5 concludes and presents the future research directions.

## 2 Preliminary

The adapted system model is an adjusted version of the intersection control proposed by [8] and improved by [16]. The flow of the algorithm adheres to the mutual exclusion (MUTEX) where vehicles compete for the critical section, which is the intersection, or the core area (CA). Vehicles in a specific lane/s can access the CA as a group and lock the rest of the vehicles in conflicting lanes from entering the CA based on FCFS

approach. In the centralized algorithm in [8], the RSU is responsible for managing the entrance to the CA. Based on a request message received from the vehicles; the RSU can only permit the vehicle/s based on FCFS and a locking logic which can affect safety, fairness and liveness properties as the RSU can only give permission to cross the intersection but not denying it [8]. In this case, if an accident happened, the vehicles that were given the permission will continue having a state of crossing forever; cause collision; affect the liveness of the CA; and will cause junction bottleneck. To overcome issues found in [8], [16] suggested an adapted approach to resolve the safety problem and the liveness by changing the used locking schema and by adjusting the algorithm to set a timer to avoid starvation. The change in the locking schema is as in the Table 1 (Fig. 2):



**Fig. 2.** 4-Way traffic intersection

**Table 1.** New locking schema

Lanes	1	2	3	4	5	6	7	8
Lanes to be Locked	1, 6, 8	2, 4, 7	2, 3, 8	1, 4, 6	2, 4, 5	3, 6, 8	4, 6, 7	2, 5, 8

### 3 inVANETs-Based Intersection Traffic Control Algorithm in Citywide

The study follows the same locking schema as in [16]; nevertheless, we will implement it in two adjacent intersections in Abu Dhabi city. We extended the scope of the study to include multiple number of RSUs in a citywide level to guarantee an efficient mutual exclusion; increase throughput; prioritize emergency and public transportations. The inVANETs-based intersection control algorithm in a citywide has 2 main tasks; the vehicleTask and the controllerTask. The vehicleTask main states are IDLE, the vehicle is within the reach of the CA; WAITING, the vehicle sent a “request” to cross the CA and waits for the RSU’s respond; QUEUING, the vehicle received the permission to cross the CA; CROSSING, the position of the vehicles is inside the CA; URGENT,

request with higher priority to enter the CA; and DELAYED, vehicle lost the right of accessing the CA due to time-out, lane changed or urgent request.

The previously referred to states are triggered either by entering the CA or by receiving a message from the RSU. Figures 3 and 4 are the state machine of the vehicles' tasks and the controllers' tasks.

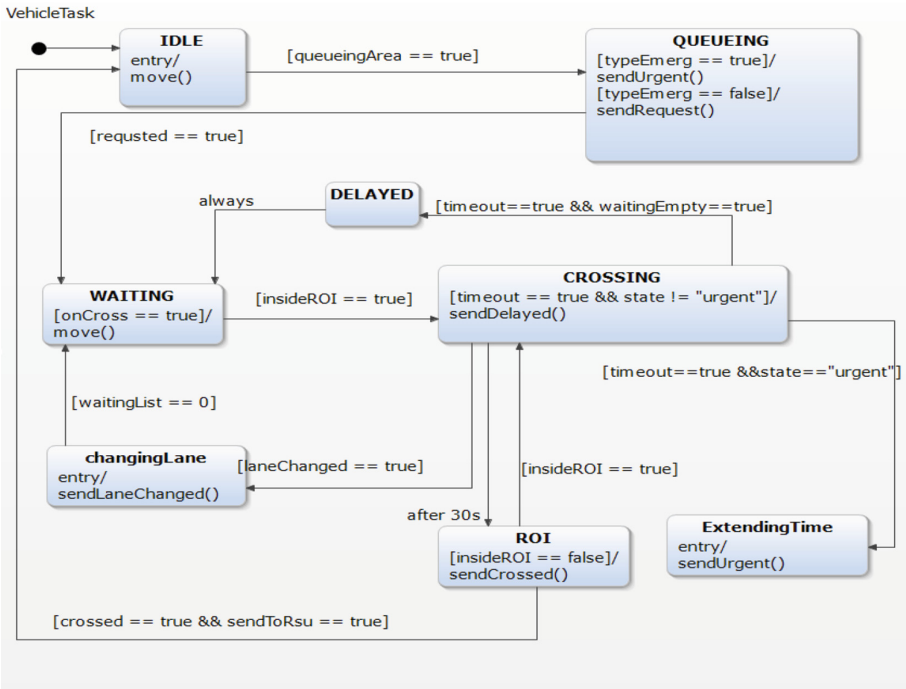


Fig. 3. Vehicle task

The desired scenario between the *vehicleTask* and the *controllerTask* is as follow. Vehicles arrives to the waiting area and sends a “request” message to the RSU. As soon as the RSU sends a replay to permit crossing, the vehicle crosses the CA. If the vehicle got the “cross” message and crossed the CA, the vehicle must send a “crossed” message to notify the RSU so the RSU removes it from the crossing list. There are other scenarios like when a vehicle changes its lane to a conflicting lane; or when a vehicle did not receive the “cross” message. In the cases of changed lane, the vehicle must inform the RSU about its current state by sending “newlane” or by sending the request message again. The vehicle then must wait for the RSU’s replay. In each case, the RSU must check for the availability of the CA. If the CA is empty or the CA is accessible by a vehicle/s in a concurrent lane then the RSU will allow the vehicle/s to cross; otherwise, the vehicle/s must wait for a specific time. In case of exceeding the expected required time to cross the CA, the RSU must check if there is a conflicting waiting vehicle. If there is a conflicting waiting vehicle, the RSU sends “delayed” to

stop the vehicles in the current crossing list from crossing the CA and to allow vehicles in the waiting list to fairly access the CA. In case of arrival of emergency vehicle, the algorithm proposes a mechanism to prioritize emergency vehicle in all cases. In this case, when the emergency vehicle sends an “urgentMsg” message, the RSU first checks if the CA is available. If not available, the RSU sends a “delayed” message to all vehicles conflicting with the emergency vehicle. The emergency vehicle can ask for extending crossing time if could not cross during the expected time by sending “emerExtendTime” message. If the vehicle crossed the intersection, it will send “emerCrossed”, but its state will remain Urgent until it reaches to its destination. In case of “urgentMsg” request from multiple emergency vehicles from conflicting lanes, the access to the CA is given based on the expected arrival time. If the expected arrival time is the same, then emergency vehicles will be served based on FCFS. The algorithm also gives the highest level of priority to the emergency vehicles then to public transportations over personal vehicles. For a better controlling schema, the information collected from vehicles such as speed, direction, and throughput will be shared and sent to other RSUs. This information will be used to predict the status of the road with Fuzzy Logic [17–20]. As well as, it will use Dijkstra algorithm to find the shortest path to a pre-set destination. The RSU will evaluate the results from both algorithms and the decision will be propagated to the vehicle/s. Similar approach was pointed by other researchers such as [21, 22].

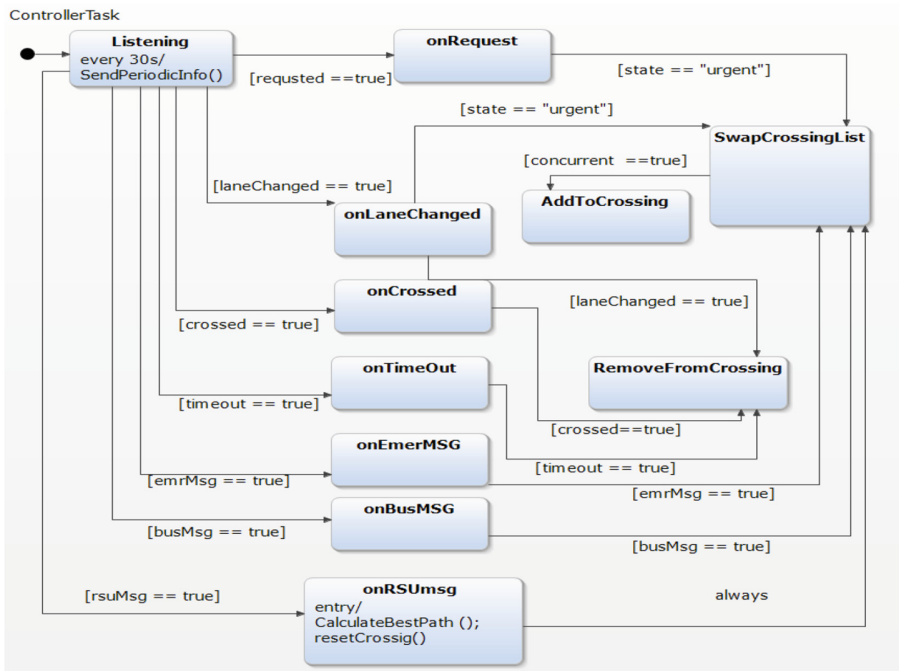


Fig. 4. Controller task

## 4 Simulation Results

The test consists of 2 intersections, 600 m apart from each other. Each intersection has 4 roadways with the same priority to have the right of way to access the CA based on the locking mechanism and FCFS to optimally traverse the CA. Each road consists of 2 lanes where the right lane can move forward or turn right; the left lane can move straight or left with maximum speed of 20 m/s. Each intersection has RSU to control the flow of the vehicles to the CA. The RSU receives request message/s from the vehicle/s and can allow or deny the request. The performed simulation adapted 2 different sets of 5 groups of volumes of vehicles. Each set is meant to study the effect of volume change; existence of emergency vehicle; RSUs exchanged messages to inspect the required time for a vehicle to cross the CA. The simulation is based on an open source framework (Veins) and a network simulator (Omnet++), a road traffic simulator (SUMO) and open street map (OSM). The framework uses the IEEE-802.11p and IEEE 1609.4 standards [23]. The feasibility of the algorithm to control the intersection was proved by [24] from a centralized perspective and by [9] from a distributed perspective. It was proven that the algorithm meets the fairness property for it allows various queue length and it fairly switches between the conflicting lanes. It was also proven that the waiting time is in a manageable range even when the queue length is long though it performs better with small queue length. The study [9, 24] also verified that the algorithm maintain high level of the throughput comparing to the tested results in [8]. The locking schema utilized the access to the CA and maintained high level of safety [16].

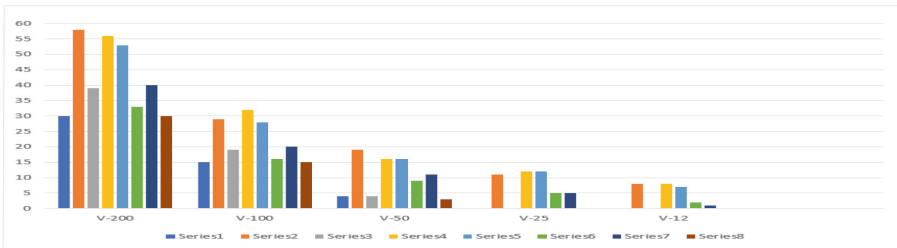
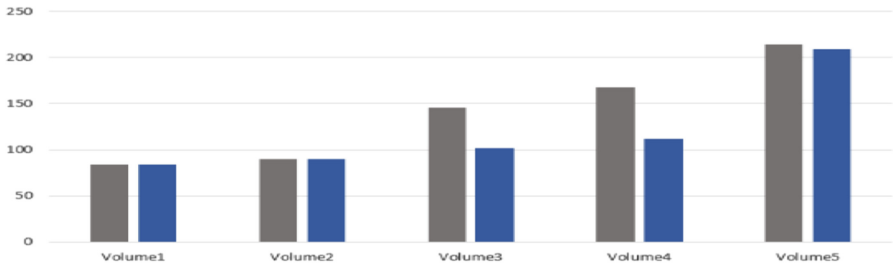


Fig. 5. Vehicle's load distribution

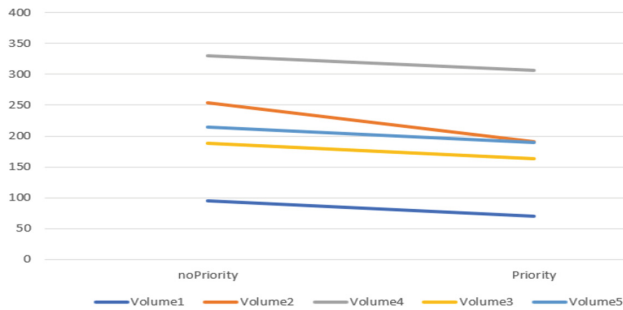
In Fig. 5 illustrates the load of the lanes and shows how likely a lane can be traversed by a vehicle. The figure indicates the number of vehicles in a road (the road consists of two lanes) and randomness distribution was taken in consideration to simulate real-life flow of traffic. Each road is initialized with different set of vehicles and different overall traffic volume. We can notice that some of the roads contain more than 50 vehicle and another road has no vehicles, in order to enhance the accuracy.

Figure 7 illustrates the average waiting time of an emergency vehicle. We can deduct from the graph that the waiting time of a vehicle is affected by the volume of the traffic as well as by whether the vehicle has a privilege (emergency vehicle or public transportation) or not. The waiting time will mostly decrease if the traffic load is low



**Fig. 6.** Average waiting time

and the “request” is from an urgent vehicle. Saying that we need to consider Fig. 6 to get a better understating of the vehicle waiting time. The graph illustrates not only the waiting time for the urgent vehicle but also the waiting time of all the vehicles. In contrast to the emergency’s vehicle waiting time, it is likely that the overall waiting time will increase when accompanied with urgent request. As giving the permission to an urgent vehicle could mean a “delay” for the normal vehicles which is an acceptable when considering the importance of giving the right of way to emergency. Also, the increase in traffic will rise the waiting time in all the scenarios as in the graph. The increased waiting time will decrease the overall throughput as the traffic increases.



**Fig. 7.** Emergency average waiting time

This is verified in Fig. 8 we can tell as the traffic load decreases, higher number of vehicles cross the CA per second. In the (T-12) scenarios where the traffic load is at its lowest volume, the number of vehicles passes the CA is 2 vehicle/s at least. This means if the crossing time of an intersection is set to be 1 min, around 30 vehicles/m to 50 vehicles/m can cross the CA if considering the drivers’ attitude, spacing between vehicles and if the car starts moving from speed of 0/m and accelerates up.



Fig. 8. Throughput

The number of the passed vehicles in (T-200) scenario is reduced to the half. Several reasons can justify this decreased affect like the swapping between conflicting lanes consumes more time to accelerate from stopping state; high chance of requesting the CA by urgent vehicle which means sending “delay” to the conflicting crossing lanes. Last but not least, Fig. 9 illustrates the number of messages sent or received by one of the controlling RSUs. We can notice that the number of request messages increase as the traffic load increases. If the load of the traffic is low or moderate, the increase in the number of cross messages will be associated with the increase in the number of request messages. Otherwise when the traffic load is very high, cross messages amount will be affected by the number of delay messages and the rsu messages. This is because when a vehicle is given the permission to access the CA, then due to a message from RSU, timeout, or emergency vehicle, the vehicle will go back to the WAITING state and will wait until it receives the cross-message from the RSU when a preset duration is met.

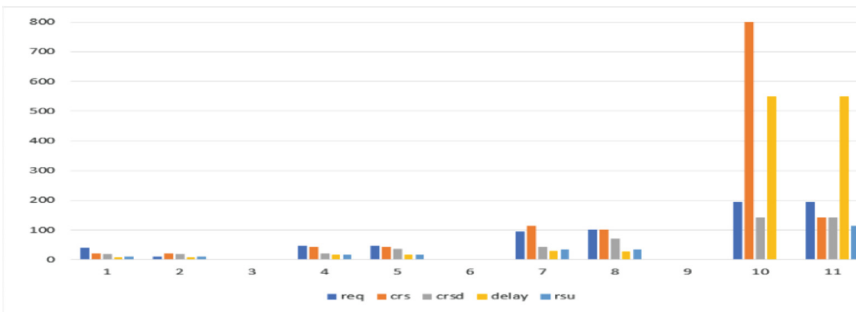


Fig. 9. Type and amount of sent messages

## 5 Conclusion

In this paper, we elaborate in depth about the enhanced intersection control algorithm suggested by [16] and improved by [8] and uses inVANET communications. We provide a performance evaluation for a suggested algorithm that claims its efficiency



and ability to handle intersections control. We tested number of different properties of traffic like different load of traffic; prioritized emergency vehicles; and the effect of the waiting time on the overall flow of the traffic. Based on these enquiries we set the simulation scenarios. The implementation of the algorithm evaluates the scenarios across 2 adjacent intersections, each has its own controller unit. The RSUs were communicating to inform each other about the traffic load moving between them. Also, in some cases, the RSUs sends information to the next RSU hub to inform it if an emergency vehicle is approaching so it can reschedule the CA availability when the emergency vehicles arrives. Based on the result we can deduce that the overall performance is satisfactory especially in normal load flow. However, when the traffic volume increases or when an emergency vehicle approached, we noticed that the waiting time increases as well. The algorithm favors the emergency vehicle and provide a quick access to serve it. Future investigations will study the feasibility with more numbers of vehicles where RSUs are able to find the best path for a given destination. We believe this will improve the overall flow and will solve the delay for the vehicles. Best path can be assumed based on Fuzzy logic and Dijkstra algorithm.

## References

1. Su, Y., Cai, H., Shi, J.: An improved realistic mobility model and mechanism for VANET based on SUMO and NS3 collaborative simulations. In: 20th IEEE International Conference on Parallel and Distributed Systems (ICPADS), Hsinchu, pp. 900–905 (2014)
2. Saini, T., Zahoor, S., Bedekar, M., Atote, B., Panicker, S.: Optimization of signal behavior through dynamic traffic control. proposed algorithm with traffic profiling. In: 2nd International Conference on Contemporary Computing and Informatics, pp. 598–602 (2016)
3. Chen, L.W., Chang, C.C.: Cooperative traffic control with green wave coordination for multiple intersections based on the internet of vehicles. *IEEE Trans. Syst. Man Cybern.: Syst.* **47**(7), 1321–1335 (2017)
4. Ma, D., Luo, X., Li, W., Jin, S., Guo, W., Wang, D.: Traffic demand estimation for lane groups at signal-controlled intersections using travel times from video-imaging detectors. *IET Intell. Transp. Syst.* **11**(4), 222–229 (2017)
5. Research and Innovative Technology Administration: Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks - Specific Requirements - Part 2: Logical Link Control. ISO 8802-2 IEEE 802.2, 1st (1989)
6. Benslimane, A., Taleb, T., Sivaraj, R.: Dynamic clustering-based adaptive mobile gateway management in integrated VANET-3G heterogeneous wireless networks. *IEEE J. Commun.* **29**(3), 559–570 (2011)
7. Zheng, B., Lin, C.W., Liang, H., Shiraiishi, S., Li, W., Zhu, Q.: Delay-aware design, analysis and verification of intelligent intersection management. In: IEEE International Conference on Smart Computing, Hong Kong, pp. 1–8 (2017)
8. Wu, W., Zhang, W., Luo, A., Cao, J.: Distributed mutual exclusion algorithms for intersection traffic control. *IEEE Trans. Parallel Distrib. Syst.* **26**(1), 65–74 (2015)
9. Saeed, I., Elhadeif, M.: Performance evaluation of an IoV-based intersection traffic control approach. In: IEEE Conference on IoT, Green Computing and Communications, Cyber, Physical and Social Computing, Smart Data, Blockchain, Computer and Information Technology, Congress on Cybermatics, Canada, pp. 1777–1784 (2018)

10. Bedekar, M., Atote, B., Panicker, S.: Centralized approach towards intelligent traffic signal control. In: International Conference on ICT for Competitive Strategies, p. 63 (2016)
11. Zhao, Y.F., Wang, F.Y., Gao, H., Zhu, F.H., Lv, Y.S., Ye, P.J.: Content-based recommendation for traffic signal control. In: IEEE 18th International Conference on Intelligent Transportation Systems, Las Palmas, pp. 1183–1188 (2015)
12. Pasin, M., Scheuermann, B., Moura, R.F.D.: VANET-based intersection control with a throughput/fairness tradeoff. In: 8th IFIP Wireless and Mobile Networking Conference, pp. 208–215 (2015)
13. Baselt, D., Knorr, F., Scheuermann, B., Schreckenberg, M., Mauve, M.: Merging lanes – fairness through communication. *Veh. Commun.* **1**(2), 97–104 (2014)
14. Burdett, R., Casey, B., Becker, K.H.: Optimising offsets and bandwidths in vehicle traffic networks. *ANZIAM J.* **55**, 77–108 (2014)
15. Hasan, S., Elhadef, M.: A citywide distributed inVANETs-based protocol for managing traffic. In: Park, J., Loia, V., Choo, K.K., Yi, G. (eds.) MUE/FutureTech-2018. LNEE, vol. 518, pp. 117–124. Springer, Singapore (2019). [https://doi.org/10.1007/978-981-13-1328-8\\_15](https://doi.org/10.1007/978-981-13-1328-8_15)
16. Elhadef, M.: An adaptable inVANETs-based intersection traffic control algorithm. In: IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence, pp. 2387–2392 (2015)
17. Pi, S., Sun, B.: Fuzzy controllers based multipath routing algorithm in MANET. In: International Conference on Applied Physics and Industrial Engineering, pp. 1178–1185 (2012)
18. Upadhayay, S., Sharma, M.: Performance evaluation of fuzzy routing algorithms for a new fuzzy mixed metric approach. *Int. J. Comput. Sci. Netw. Secur.* **8**(4), 21–28 (2008)
19. Doja, M., Alam, B., Sharma, V.: Analysis of reactive routing protocol using fuzzy inference system. In: AASRI Conference on Parallel and Distributed Computing Systems, pp. 164–169 (2013)
20. Gajjar, S., Sarkar, M., Dasgupta, K.: FAMACRO: fuzzy and ant colony optimization based MAC/routing cross-layer protocol for wireless sensor networks. In: International Conference on Information and Communication Technologies, vol. 8, pp. 1014–1021 (2015)
21. Ganda, J.: Simulation of routing option by using two layers fuzzy logic and Dijkstra's algorithm in MATLAB 7.0. *J. Electr. Electron. Eng.* **1**(1), 11–18 (2016)
22. Biswas, S.: Fuzzy real time Dijkstra's algorithm. *Int. J. Comput. Intell. Res.* **13**(4), 631–6404 (2017)
23. [veins.car2x.org](http://veins.car2x.org). <http://veins.car2x.org/features/>. Accessed 17 June 2019
24. Tabaza, H., Elhadef, M., Saeed, I.: Performance evaluation of an adaptable inVANETs-based traffic control. In: ICT Conferences, Society, and Human Beings, Porto (2019)