SDN based Network Traffic Routing in Vehicular Networks: A Scheme and Simulation Analysis

Jitendra Bhatia[1], Mohammad S. Obaidat[2,3,4,*], Tirath Savasaiya[1], Hardik Trivedi[5], Sudeep Tanwar[5,†] and Kuei-Fang Hsiao[6,7]

[1] *Vishwakarma Government Engineering College, Gujarat Technological University, Ahmedabad, Gujarat, India*

[2] *College of Computing and Informatics, University of Sharjah, U.A.E.*

[3] *KAIST, University of Jordan, Jordan*

[4] *University of Science and Technology Beijing, China*

[5] *Department of CSE, Institute of Technology, Nirma University, Ahmedabad, Gujarat, India*

[6] *Department of Information Systems, University of Sharjah, U.A.E.*

[7] *Department of Information Management, Ming Chuan University, Taiwan*

Keywords: VANET, Software Defined Networks, Routing, Traffic Optimization.

Abstract: This paper focuses on exploiting the service architecture that exercises the Software-Defined Network (SDN) concept in heterogeneous vehicular networks for handling data traffic using optimal path selection and routing in the situation like congestion in Vehicle Ad Hoc Networks (VANETs). In particular, we consider the scenario where a set of information is requested by a vehicular node from other vehicular nodes. Software-Defined Networking (SDN) have caught much attention in vehicular networks where decoupling of data and control plane enables the centralized control of the network; providing flexibility to a great extent. In this paper, we design an efficient network traffic routing algorithm assisted by SDN. Finally, we built a realistic traffic based simulation model. Simulation results show that the proposed protocol leveraged by the SDN framework outperforms the conventional network for data traffic management in terms of Round Trip Time (RTT) and Packet Delivery Ratio (PDR).

1 INTRODUCTION

The design and development of the routing algorithms that enhance the efficiency of the vehicular networks have been motivated by recent advancements in vehicular communications. VANET consists of vehicles where they behave as mobile nodes and make requests for some data as per the application requirement. An example of traditional VANET is shown in Fig.1where all the vehicles use GPS and OBU (On-Board Unit) equipped with sensors and communication hardware. Traditional IP networks are not sufficient to meet all the needs of the VANET requirements. A traditional network is unable to provide flexibility, scalability and is not capable of managing huge number of mobile nodes in VANET(Chahal et al., 2017). As traditional network has high latency; some of the VANET applications can not work properly. All VANET applications have a different kind of requirements like high packet delivery ratio, man-

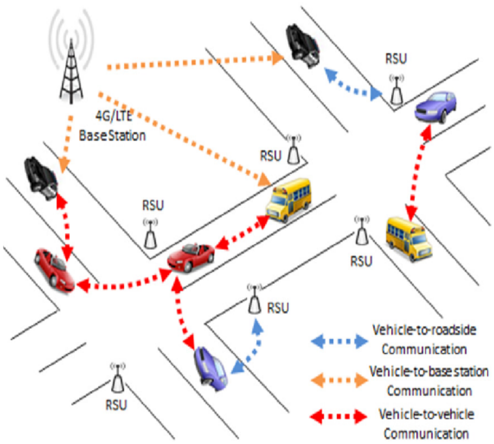

Figure 1: Traditional VANET(Ku et al., 2014a).

*Fellow of IEEE and Fellow of SCS
†Member, IEEE

Bhatia, J., Obaidat, M., Savasaiya, T., Trivedi, H., Tanwar, S. and Hsiao, K.
SDN based Network Traffic Routing in Vehicular Networks: A Scheme and Simulation Analysis.
DOI: 10.5220/0009911400130020
In *Proceedings of the 10th International Conference on Simulation and Modeling Methodologies, Technologies and Applications (SIMULTECH 2020)*, pages 13-20
ISBN: 978-989-758-444-2

aging a heterogeneous environment, better QoS and these requirements can't be served by the traditional network due to its limitation (Vora et al., 2018).

SDN is such a developing technology in which management of the network abstraction through the lower level of functionality is handled by the network administrator. SDN framework mainly comprises of two sections, First is control plane and second one is information plane(Liu et al., 2017) (Bhatia et al., 2018). The control plane is generally implemented in the server as a software component. The data forwarding mechanism is executed by the data plane available on the network device like switch and router. An Open Flow is a standard used for communication between infrastructure and control plane. Northbound and Southbound APIs can be used for communication and those APIs can also be changed according to requirements. The Road Side Unit (RSU) fully exercises the logically centralized controller (Bhatia et al., 2020). SDN concept was not fully fledge explored in VANET so far. The closest studies of SDN are a couple of implementations in VANET (Tanwar et al., 2018c).

Some changes in the traditional VANET architecture will help to integrate VANET with SDN. SDN-based VANET example is shown in Fig.2. In place of a base station, SDN supported open-flow switches that can be used for communicating with the SDN controller directly or through the Internet. Based on the requirements, RSU may communicate with a base station and a base station in further, may communicate with Controller or RSU can directly communicate with SDN controller. It is also possible to deploy the SDN controller on the cloud or on fog if the area, which is covered by the SDN controller is beyond the coverage (Tanwar et al., 2018a).

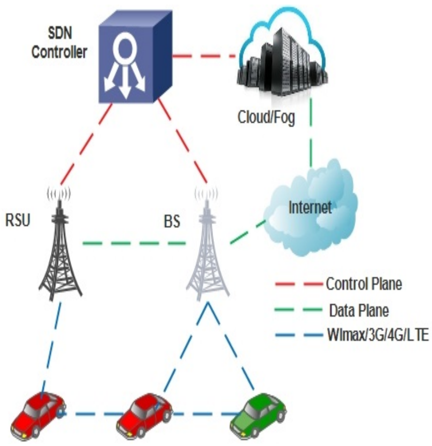

Figure 2: SDN-based VANET.

This work is dedicated to the efficient routing of vehicular application data traffic under the I2V communication environment. Communication efficiency between vehicles and RSU is the fundamental function of such services.. The principle ideas of the whole paper are as follows:

- Vehicles are likely to publish or subscribe to various infotainment and emergy messages. Motivated by this observation, the routing of those messages on RSU in such an I2V communication environment has been investigated. This is the study based on SDN for efficient routing of network traffic in VANETs with consideration of application requirement under communication constraints.
- Based on a global view of traffic information in the network, a simulation model has been developed. Under different traffic loads, performance evaluation of the proposed algorithm has been done. The efficiency of simulation is discussed in the result section.

An overview of the paper is as follows. First, the main notion on which our approach is based on is introduced. Then various network routing strategies and related works in VANETs have been reviewed by authors (Tanwar et al., 2019) (Tanwar et al., 2014). Furthermore, the author proposed the SDN enabled architecture for the network traffic routing for better response time and higher throughput. After that, based on the precisely described algorithm, problem analysis has been conducted. At the end, the simulation model and the experimental results are discussed by the authors. Finally, it is concluded with open issues that can be carried as future work.

2 RELATED WORKS

Past studies on data dissemination in VANETs emphasizes on improving communication reliability, Quality of Service (QoS) and the issues commonly dwell in PHY and MAC layers (Chaturvedi and Srivastava, 2017)(Bhatia et al., 2019). One of the prime issues is to manage the network traffic efficiently under communication constraints. To answer this question, lots of mechanisms came into existence (Wang et al., 2018) (Kühlmorgen et al., 2019) (Tanwar et al., 2018b) (Wu et al., 2019). Using SDN instead of the traditional network provides many benefits like better efficiency, congestion control, better PDR. The main idea after introducing the SDN in the field of VANET is that the traditional network is not able to provide the required QoS to VANET (Li et al., 2018) (Dave and Bhatia, 2013) (Wu et al., 2018).

Ku *et al.*(Ku et al., 2014b), proposed the idea of using SDN with VANET to make it available with flexibility and programmability. In the proposed idea, the OpenFlow controller communicates with OpenFlow switches. Two different communication channels are also used, Wi-Fi for the data plane and LTE for the Control Plane. The architecture was classified into three operation modes, one of them is centralized control mode, second is hierarchical control mode and the last one is distributed control mode (He et al., 2016b) (Bhatia et al.,).

Duan *et al.*(Duan et al., 2014) (Jindal et al., 2018) proposed a special architecture, Vehicular Cyber-Physical Systems (VCPS). The proposed architecture reduces packet delay time by 20%. The key components of the system were vehicles, RSU, OpenFlow switches, and global controller. Proposed approach works on the location-based routing protocol. Liu *et al.*(Liu et al., 2015), discussed the idea of GeoBroadcast in VANET. It supports periodic broadcast message. Controller overhead is reduced by the proposed approach. It uses less amount of bandwidth and latency is also reduced. In proposed architecture floodlight was used as a controller.

He *et al.*(He et al., 2016d), presented the idea of using multicast in VANET based on SDN on the basis of trajectory prediction, which reduces the burden of SDN control and data plane and also provides better multicast scheduling decisions. Dong *et al.* (Dong et al., 2016), presented SDN-based on-demand routing protocol SVAO. In Map based protocol, traffic is forwarded according to road topology. Road information is taken into the account in Map-Based protocols.

Ji *et al.*(Ji et al., 2016), proposed the approach of using geographic routing in SDN based VANET. Some, typical geographic routing protocols only use local information to make decision, which may lead to local maxima. Kumar *et al.*(Sahoo and Yunhasnawa, 2016)(Bhatia, 2015), proposed the idea of using cloud service in VANET based on SDN for better Packet Delivery Ratio (PDR) and less RTT. In SDN based VANET, many vehicle requests for data, cloud service can be used to meet these requirements. Luo *et al.* (Luo et al., 2016), introduced hybrid architecture in VANET for managing physical resources in it, to provide pre-warning collision and topology change in VANET. The proposed sdnMAC protocol is TDMA based. In proposed algorithm time is divided into equal frame sizes. Each RSU should at least acquire a one-time slot in one frame.

Liu *et al.*(Liu et al., 2016), idea of using Cooperative Data Scheduling (CDS) in hybrid SDN based VANET is proposed by authors. The ultimate goal is to serve the maximum number of request. In the proposed approach two service channels are used, one is for I2V communication and one is for V2V communication. At a time vehicles can only send or receive data. Each vehicle has its own record of requested items. At a time vehicle can transmit or receive only one data item in one scheduling period. He *et al.*(He et al., 2016c) proposed the idea of using SDN in VANET to schedule resources and reduce cost. Heterogeneity of the current VANET introduces two problems: one is interoperability and the other one is resource allocation. To solve these issues researchers design resource scheduling solutions for VANET.

Wang *et al.*(Wang et al., 2017) claims that using the traditional network in VANET introduces complexity. To overcome this issue they proposed SDN based IoV (SDIV). Using IoV (Internet of Vehicles) is also a great challenge because it has limited size flow table and it will be difficult to configure heterogeneous switches. He *et al.*(He et al., 2016a) proposed the idea of using fog computing in SDN based VANET to provide less latency and support mobility and location awareness. Modified Constrained Optimization Particle Swarm Optimization (MPSO-CO) algorithm is proposed by authors. To reduce the burden of cloud computing, idea of using cloud computing with fog computing and SDN is introduced (Bhatia et al., 2018)(Yao et al., 2018)(Shah et al., 2019).

In contrast to prior work, we have designed an SDN based network traffic rerouting under the overload condition. SDN leverages the global view of network traffic load on RSU, which helps to minimize the response time to request when network traffic load tends to higher.

3 PROPOSED APPROACH

In this section, the proposed algorithm is briefly explained. The basic idea is to exploit the SDN capabilities to analyze the network and reroute the traffic based on the network overhead. SDN controller has a global view of the network, which comprises of the parameters like a number of vehicular nodes in the network, bandwidth and total requirement of the nodes. The load factor is calculated by dividing bandwidth by requirement of a single application running on an application layer. Thus, if there are N nodes, it can be directly calculated as nodes requirement. Now the value of available nodes and load for an intermediate node is compared where the load is the maximum number of nodes which can be served by an intermediate device at a time. If the value of the load factor is less than number of available nodes, it indicates that the number of available nodes is more and the system is being overloaded. For SDN based vehicular network, the SDN controller has all the information about the network. So, we can get the nearest underloaded node called as a proxy node for the intermediate device which is overloaded. After getting proxy node, start rerouting traffic from that node. Now, the load factor is calculated for a proxy node and compared with load factor. If load factor is less then available nodes for proxy node then repeat above process for proxy node, else continue routing from that node.

After fixing time of interval, the value of load factor is recalculated because many applications run on the application layer. On the basis of the newly calculated load value, again conditions are checked whether current traffic can be served by an intermediate node or not. If traffic is more, then find proxy node and reroute traffic from that node. After checking conditions for the load factor of an intermediate node and for the proxy node, if both conditions are not matched, it signifies that traffic is decreased and the proxy node can be relieved, but if all the nodes are heavily overloaded, the performance of the system may be decreased and a warning message is displayed. In the result section, it is clear that our proposed algorithm outperforms traditional competing approach.

Algorithm 1: SDN based Traffic Rerouting protocol.

Step 1: Get $bandwidth, Nodes, Requirement$ ▷ Parameters required to calculate load factor
Step 2: Calculate $loadfactor$ for intermediate node through which traffic is being passed using the following function:

$$loadfactor = (bandwidth)/Requirement$$

▷ Load factor is maximum number of nodes that can be served by an intermediate node
Step 3: Compare Load Factor with number of available nodes to intermediate device.

if (Load$<=$ $AvailNodes$) **then**
    Go to Step 4.
else
    System is not overloaded, so keep observing.
end if

Step 4: Get nearest node of intermediate device which is underloaded and called as proxy node.
step 5: Find load factor for proxy node.
Step 6: Compare values of load factor and available nodes for the proxy node.
if ($Load <= AvailNodes$) **then**
    Repeat step 4, step 5 and step 6 for proxy node.
else
    System is safe and Continue routing from the proxy node.
end if

Step 7: Repeat Step 3
    ▷ At regular time interval value of load factor is recalculated.

Step 8: If all the nodes are heavily loaded then give warning that performance of the system may decrease.
Step 9: Exit

Computational Complexity. However, small overhead is incurred to find out the total number of nodes and requirement of the system at the fixed interval of time. At regular interval of time, the load value is calculated and if traffic is more, proxy node is started. After some time if load decreases, it relieves the proxy node and resets the counter value. If the counter value is increased more then load, it means now proxy node is also overloaded and performance of system decreases. This overhead is minor and can be ignored, but it still affects the performance. Protocol overhead is lower than before and comparatively achieved efficiency is higher which can be seen in the result section.

4 EXPERIMENTAL SETUP

4.1 Simulation Scenario

Two networks were taken to perform the simulation viz., simulation area of 1 KM and 2 KM is shown in Fig.3. In both networks, block size is of 200m*200m and four different scenarios were taken into consideration where number of nodes varies as 50,100,150 and 200. The details of simulation parameter are shown in the Table 1.

Table 1: Simulation Parameters.

Parameter	Value
Tools Used	NS-3,SUMO(Song et al., 2014)
Simulation Area	1KM, 2KM
Simulation Time	250 Second
Chanel bandwidth	300kbps
Proxy Node capacity	300kbps
Number of Vehicles	50-200
Vehicle Max Speed	14m/s
Open flow module	OFSwitch 1.3

For both networks, simulation was performed for different number of vehicles varying from 50 to 200. All vehicles moves randomly in network with random speed and maximum vehicle speed is set to 14m/s. All the nodes pings to a server from one route and at one stage traditional switch is unable to serve all nodes. However, in case of SDN, due to the global view of the network re-routing the traffic when network traffic increases is needed.

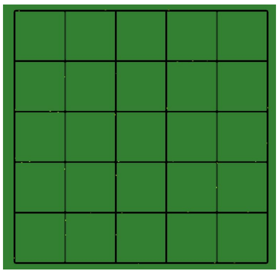

Figure 3: Grid Road Network 1 and 2 KM.

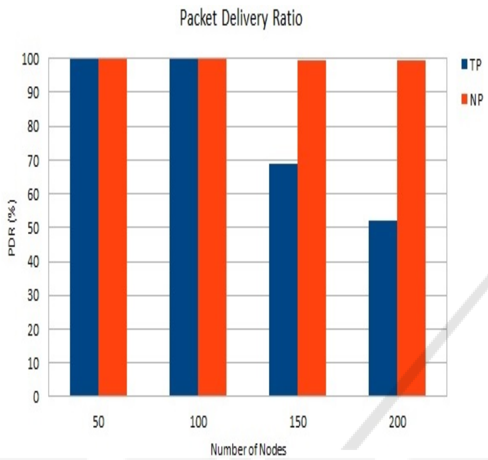

Figure 4: Packet Delivery Ratio for 1KM.

4.2 Evaluation Metrics

Fig.4 shows the Packet Delivery Ration (PDR)for 1KM area and for 50,110,150 and 200 nodes. As the number of node increases, it can be seen that in TP (Traditional protocol), PDR decreases and same is applicable for 2 KM area, which can be seen in Fig.5. But in proposed NP (New Protocol), PDR is almost 100%. Only few packets are lost. SDN conroller has an idea about the traffic load on particular node and it reroutes the traffic.

RTT is time taken by a packet to reach at the destination and come back to the source node. We have used ping application which uses ICMP protocol. We can see that as number of node increases, RTT increases too. Fig.6 shows the maximum RTT time for simulation area of 1 KM and for 50,100,150 and 200 nodes. where Fig.7 shows the maximum RTT time for simulation area 2 KM for 50,100,150 and 200 nodes. We can see that when number of node reaches at 150 and 200 in TP(Traditional Protocol), RTT also increases whereas in our proposed NP(New protocol) RTT remains same because of SDN and traffic rerouting paradigm.

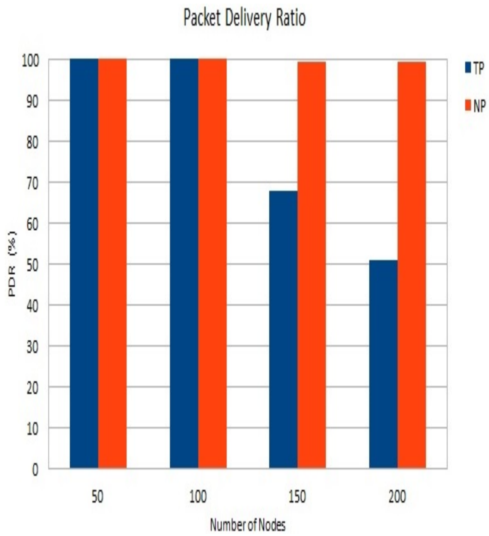

Figure 5: Packet Delivery Ratio for 2KM.

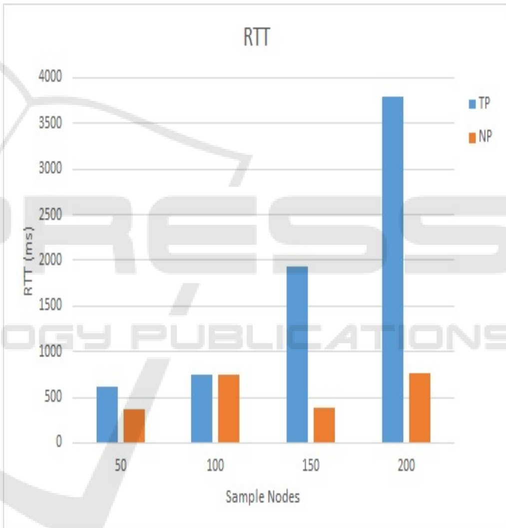

Figure 6: Max RTT Time for 1KM.

5 DISCUSSION

Two performance parameters were taken into consideration, i.e., PDR and RTT which can be seen in Fig.4 and Fig:5 respectively. In Fig.4 PDR for 1 KM area is shown. It is visible that for 50 and 100 nodes, Traditional Protocol serves well, but as number of nodes increases, PDR decreases whereas our proposed protocol yields almost the same results. Fig.5 shows PDR for 2 KM area in which, if we compare with 1 KM area it can be seen that PDR in 2 KM is slightly less then 1 KM because of the distance. Second parameter is RTT for which we have taken 10 sample nodes from all simulations and compared them. Fig.8 shows

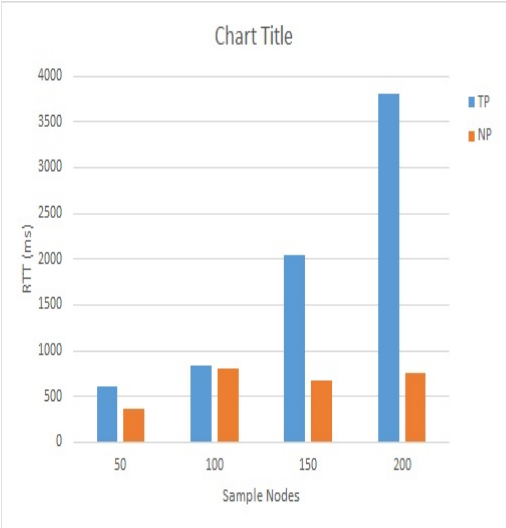

Figure 7: Max RTT Time for 2KM.

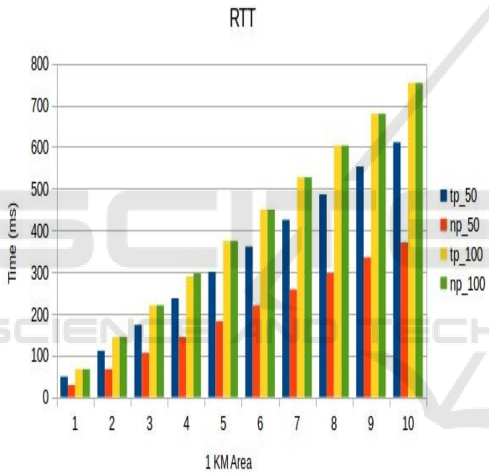

Figure 8: RTT Time for 1KM.

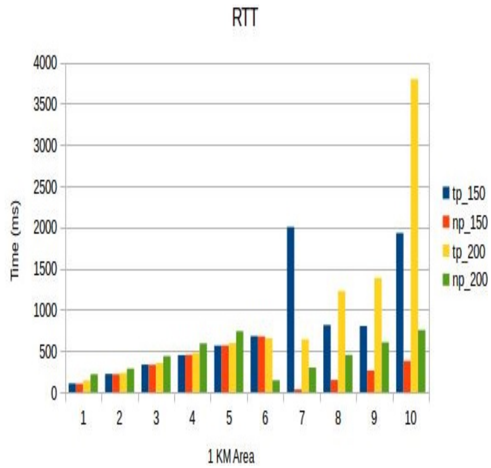

Figure 9: RTT Time for 1KM.

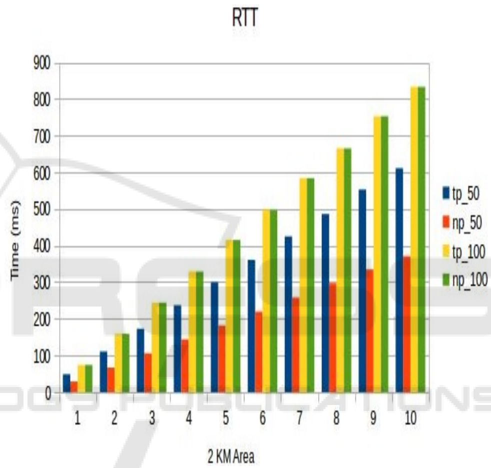

Figure 10: RTT Time for 2KM.

RTT for 1 KM for 2 different scenarios where number of nodes are 50 and 100 and RTT of Tradition protocol is compared with the newly proposed protocol. Fig.9 shows RTT for 1 Km area for 150 and 200 nodes. Fig.10 shows RTT for 2 KM area for number of nodes of 50, and 100 and Fig.11 shows RTT for 2 km area for 150 and 200 nodes.

It can be seen that both in 1 KM area and 2 KM area for 50 and 100 nodes almost both protocols gives the same result, but when traffic increases in case of 150 and 200 nodes, PDR decreases in traditional network and because of heavy traffic RTT time increases in traditional network. RTT is almost 3.8 seconds for 2 KM area and 200 nodes. Definitely all networks have some limitations like number of nodes which can be served or limited bandwidth, but When we use SDN openflow approach, controller comes to know when node is being heavily congested and it reroutes the traffic accordingly. SDN reroutes the traffic and another node behaves as proxy and much more better result can be gained by this paradigm. It can be seen in Fig:11 that traditional network's RTT is much more, but the maximum RTT time of the new proposed protocol is approximately 800 ms, which is much lower then second one. The results also depends on the capacity of proxy node. Actually, we can say that it depends on the node which is behaving as proxy for all the traffic.

In our simulation, we inferred that at rate of 300 Kbps bandwidth, traditional protocol is capable of serving around 100 to 105 nodes and after that, there is drastic performance drop. When traffic increases, SDN performs rerouting and uses another path. But if node count increases to a certain limit N or if traffic increases to certain limit T, it is possible that proposed new protocol is unable to serve all and performance is dropped.

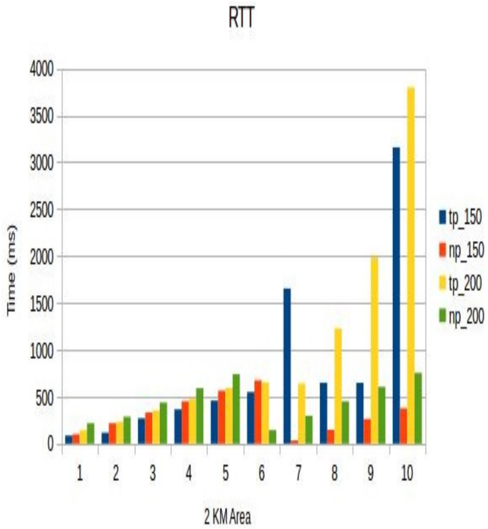

Figure 11: RTT Time for 2KM.

6 CONCLUSION AND FUTURE DIRECTIONS

SDN plays a very important role in VANET for network traffic routing. The issues of the traditional networks have been solved by SDN. Due to its flexibility and programmability new scope of technology has been risen. The proposed strategy yields low latency and high throughput. The impact of the different number of vehicular nodes has been analyzed in I2V networks by considering a grid network. The authors give a substantial analysis of both the requirement and constraint on network traffic routing. On this basis, an algorithm has been described. A dynamic algorithm for the same is intended to enhance throughtput and reducing response time. SDN enabled VANET architecture leverages the benefits of a global view of traffic information which assists our approach for handling the network traffic efficiently. Simulation results under different traffic loads demonstrate the superiority of the proposed approach. Designing a backup mechanism for the SDN controller in case of failure can be a motivation for future direction.

REFERENCES

Bhatia, J., Dave, R., Bhayani, H., Tanwar, S., and Nayyar, A. (2020). Sdn-based real-time urban traffic analysis in vanet environment. *Computer Communications*, 149:162–175.

Bhatia, J., Govani, R., and Bhavsar, M. (2018). Software defined networking: From theory to practice. In *2018 Fifth International Conference on Parallel, Distributed and Grid Computing (PDGC)*, pages 789–794.

Bhatia, J., Kakadia, P., Bhavsar, M., and Tanwar, S. (2019). Sdn-enabled network coding based secure data dissemination in vanet environment. *IEEE Internet of Things Journal*, pages 1–1.

Bhatia, J., Mehta, R., and Bhavsar, M. (2018). Variants of software defined network (sdn) based load balancing in cloud computing: A quick review. In *Future Internet Technologies and Trends*, pages 164–173, Cham. Springer International Publishing.

Bhatia, J., Modi, Y., Tanwar, S., and Bhavsar, M. Software defined vehicular networks: A comprehensive review. *International Journal of Communication Systems*, page e4005.

Bhatia, J. B. (2015). A dynamic model for load balancing in cloud infrastructure. *Nirma University Journal of Engineering and Technology (NUJET)*, 4(1):15.

Chahal, M., Harit, S., Mishra, K. K., Sangaiah, A. K., and Zheng, Z. (2017). A survey on software-defined networking in vehicular ad hoc networks: Challenges, applications and use cases. *Sustainable Cities and Society*.

Chaturvedi, M. and Srivastava, S. (2017). Multi-modal design of an intelligent transportation system. *IEEE Transactions on Intelligent Transportation Systems*, 18(8).

Dave, J. R. and Bhatia, J. (2013). Issues in static periodic broadcast in vanet. *International Journal of Advances in Engineering & Technology*, 6(4):1712.

Dong, B., Wu, W., Yang, Z., and Li, J. (2016). Software defined networking based on-demand routing protocol in vehicle ad hoc networks. In *Mobile Ad-Hoc and Sensor Networks (MSN), 2016 12th International Conference on*, pages 207–213. IEEE.

Duan, P., Peng, C., Zhu, Q., Shi, J., and Cai, H. (2014). Design and analysis of software defined vehicular cyber physical systems. In *Parallel and Distributed Systems (ICPADS), 2014 20th IEEE International Conference on*, pages 412–417. IEEE.

He, X., Ren, Z., Shi, C., and Fang, J. (2016a). A novel load balancing strategy of software-defined cloud/fog networking in the internet of vehicles. *China Communications*, 13(2):140–149.

He, Z., Cao, J., and Liu, X. (2016b). Sdvn: enabling rapid network innovation for heterogeneous vehicular communication. *IEEE network*, 30(4):10–15.

He, Z., Zhang, D., and Liang, J. (2016c). Cost-efficient sensory data transmission in heterogeneous software-defined vehicular networks. *IEEE Sensors Journal*, 16(20):7342–7354.

He, Z., Zhang, D., Zhu, S., Cao, J., and Liu, X. (2016d). Sdn enabled high performance multicast in vehicular networks. In *Vehicular Technology Conference (VTC-Fall), 2016 IEEE 84th*, pages 1–5. IEEE.

Ji, X., Yu, H., Fan, G., and Fu, W. (2016). Sdgr: An sdn-based geographic routing protocol for vanet. In *Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and*

IEEE Smart Data (SmartData), 2016 IEEE International Conference on, pages 276–281. IEEE.

Jindal, A., Aujla, G. S., Kumar, N., Chaudhary, R., Obaidat, M. S., and You, I. (2018). Sedative: Sdn-enabled deep learning architecture for network traffic control in vehicular cyber-physical systems. *IEEE Network*, 32(6):66–73.

Ku, I., Lu, Y., Gerla, M., Gomes, R. L., Ongaro, F., and Cerqueira, E. (2014a). Towards software-defined vanet: Architecture and services. *2014 13th Annual Mediterranean Ad Hoc Networking Workshop (MED-HOC-NET)*, pages 103–110.

Ku, I., Lu, Y., Gerla, M., Ongaro, F., Gomes, R. L., and Cerqueira, E. (2014b). Towards software-defined vanet: Architecture and services. In *Ad Hoc Networking Workshop (MED-HOC-NET), 2014 13th Annual Mediterranean*, pages 103–110. IEEE.

Kühlmorgen, S., Lu, H., Festag, A., Kenney, J., Gemsheim, S., and Fettweis, G. (2019). Evaluation of congestion-enabled forwarding with mixed data traffic in vehicular communications. *IEEE Transactions on Intelligent Transportation Systems*.

Li, F., Song, X., Chen, H., Li, X., and Wang, Y. (2018). Hierarchical routing for vehicular ad hoc networks via reinforcement learning. *IEEE Transactions on Vehicular Technology*, 68(2):1852–1865.

Liu, K., Feng, L., Dai, P., Lee, V. C., Son, S. H., and Cao, J. (2017). Coding-assisted broadcast scheduling via memetic computing in sdn-based vehicular networks. *IEEE Transactions on Intelligent Transportation Systems*, 19(8):2420–2431.

Liu, K., Ng, J. K., Lee, V. C., Son, S. H., and Stojmenovic, I. (2016). Cooperative data scheduling in hybrid vehicular ad hoc networks: Vanet as a software defined network. *IEEE/ACM transactions on networking*, 24(17):1759–1773.

Liu, Y.-C., Chen, C., and Chakraborty, S. (2015). A software defined network architecture for geobroadcast in vanets. In *Communications (ICC), 2015 IEEE International Conference on*, pages 6559–6564. IEEE.

Luo, G., Jia, S., Liu, Z., Zhu, K., and Zhang, L. (2016). sdnmac: a software defined networking based mac protocol in vanets. In *Quality of Service (IWQoS), 2016 IEEE/ACM 24th International Symposium on*, pages 1–2. IEEE.

Sahoo, P. K. and Yunhasnawa, Y. (2016). Ferrying vehicular data in cloud through software defined networking. In *Wireless and Mobile Computing, Networking and Communications (WiMob), 2016 IEEE 12th International Conference on*, pages 1–8. IEEE.

Shah, N. B., Shah, N. D., Bhatia, J., and Trivedi, H. (2019). Profiling-based effective resource utilization in cloud environment using divide and conquer method. In Fong, S., Akashe, S., and Mahalle, P. N., editors, *Information and Communication Technology for Competitive Strategies*, pages 495–508, Singapore. Springer Singapore.

Song, J., Wu, Y., Xu, Z., and Lin, X. (2014). Research on car-following model based on sumo. In *Advanced Infocomm Technology (ICAIT), 2014 IEEE 7th International Conference on*, pages 47–55. IEEE.

Tanwar, S., Kumar, N., and Niu, J.-W. (2014). Eemhr: Energy-efficient multilevel heterogeneous routing protocol for wireless sensor networks. *International Journal of Communication Systems*, 27(9):1289–1318.

Tanwar, S., Trivedi, H., and Priyank, T. (2018a). Software defined network-based vehicular adhoc networks for intelligent transportation system: Recent advances and future challenges. *Futuristic Trends in Network and Communication Technologies (FTNCT)*, 858:325–337.

Tanwar, S., Tyagi, S., Kumar, N., and Obaidat, M. S. (2019). La-mhr: Learning automata based multilevel heterogeneous routing for opportunistic shared spectrum access to enhance lifetime of wsn. *IEEE Systems Journal*, 13(1):313–323.

Tanwar, S., Tyagi, S., and Kumar, S. (2018b). The role of internet of things and smart grid for the development of a smart city. In *Intelligent Communication and Computational Technologies*, pages 23–33. Springer.

Tanwar, S., Vora, J., Tyagi, S., Kumar, N., and Obaidat, M. S. (2018c). A systematic review on security issues in vehicular ad hoc network. *Security and Privacy*, 1(5):1–23.

Vora, J., Kaneriya, S., Tanwar, S., and Tyagi, S. (2018). Performance evaluation of sdn based virtualization for data center networks. In *2018 3rd International Conference On Internet of Things: Smart Innovation and Usages (IoT-SIU)*, pages 1–5.

Wang, X., Ning, Z., Hu, X., Wang, L., Hu, B., Cheng, J., and Leung, V. C. (2018). Optimizing content dissemination for real-time traffic management in large-scale internet of vehicle systems. *IEEE Transactions on Vehicular Technology*, 68(2):1093–1105.

Wang, X., Wang, C., Zhang, J., Zhou, M., and Jiang, C. (2017). Improved rule installation for real-time query service in software-defined internet of vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 18(2):225–235.

Wu, G., Wang, J., Obaidat, M. S., Yao, L., and Hsiao, K.-F. (2019). Dynamic switch migration with non-cooperative game towards control plane scalability in sdn. *International Journal of Communication Systems*, 32(7):e3927.

Wu, G., Wang, L., Xu, Z., Yao, L., and Obaidat, M. S. (2018). A qos and cost aware fault tolerant scheme insult-controller sdns. In *2018 IEEE Global Communications Conference (GLOBECOM)*, pages 1–6. IEEE.

Yao, L., Zhao, X., Wu, G., and Obaidat, M. S. (2018). The community characteristic based controller deployment strategy for sdns. In *2018 IEEE Global Communications Conference (GLOBECOM)*, pages 1–6. IEEE.