Enhancing the throughput of a Node by Handling Congestion in Vanets using Efficient Queue Management

Kamlesh Chandra Purohit (purohit_kaml@rediffmail.com)
Graphic Era Deemed to be University https://orcid.org/0000-0002-3188-529X
Sanjay Jasola
Graphic Era Hill University

Research Article

Keywords: VANET, MANET, ITS, congestion, throughput, end_to_end delay and packet loss.

Posted Date: June 29th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-531774/v1

Abstract

Vehicular ad-hoc network (VANET) is one of the most challenging domain in current scenario to provide an Intelligent transport system (ITS). One of the important application area of mobile ad-hoc network (MANET) is VANET. Few major challenges in VANET environment like network congestion, routing and securitywhich must be resolved to provide an efficient service to the user. This paper explores the effect of network layer congestion in the overall performance of VANET and proposes a framework to handle congestion in the VANET by providing efficient queue management. The proposed approach named Node Based Throughput (NBTH) is intended to evaluate the throughput of a node. Method proposed in this paper is being analysed with various parameters like throughput end_to_end delay and packet loss with respect to density of nodes.

1. Introduction

The main characteristics of the Ad hoc networks are that they are infrastructure less networks described as a pool of wireless mobile nodes underprivileged of any stationary infrastructure. A sub category of ad-hoc network is Mobile Ad-hoc Network (MANET). Furthermore, a typical category of MANET, which is solely responsible and adaptation of diverse methodology for Intelligent Transport System (ITS) is Vehicular Ad-hoc Network (VANET) [1]. The fundamental feature of VANETs is to provide an efficient communication between vehicles and roadside apparatus.

In the present scenario VANET have come out with more acute research and development area which permits interaction between vehicle to vehicle and vehicle to Road Side Unit (RSUs) by using a wireless sensing device mounted in every vehicle in the network. Because of the very high and vital topology, VANET has become one of the most captivating research areas. It is not only providing safety but also security and reliability in vehicular system. Mainly two categories of communication which are Vehicle to vehicle and vehicle to infrastructure are supported by VANET. In vehicular communication, information generation and distributions occur with the vehicle to vehicle (V2V) and vehicle to infrastructure (V2I) [1, 2, 3]. Following are the distinct features of VANET:

- The capacity of moving vehicles is extremely expectable as all the vehicles are moving with only two ways on the same street.
- Vehicles are responsible for facilitating ample electric power to the wireless sensing apparatus which are already exists in the vehicles.
- In VANET broadcast communication is liable to provide information from source to destination rather than unicast communication.

The main problem when we work with network layer is congestion, therefore some approach should be there to manage congestion that exist in networks whether it is VANET, MANET or any other network. While handling congestion the primary thing that arrives into mind is, how to detect that there is congestion in network? After congestion being detected the close loop approach is the main methodology used for handling the congestion problem in network [4].When bestowed load surpasses the available capability of a node to manage the packets or link is incompetent to manage the quantity of packets given to it for delivery afterword's Network congestion occurs. The bandwidth of links is also condensed due to fading channels.

Network congestion is the root cause behind the inefficient use of channel bandwidth which reduce the effective data rates due to loss of data. It is also accountable for packets drops at the buffers, amplified delays, misused energy, that is why need a retransmission of signals. Furthermore, it would be highly biased for nodes whose data has to negotiate a substantial collection of hops. It will significantly reduce the accomplishment and lifetime of the network.

This paper proposes a node-based congestion detection approach. Nodes calculate their throughput and compare it with the expected throughput and accordingly detect congestion. Rest of the paper is organized as follows; next section discusses the related work in this area, Sect. 3 contains proposed approach. In Sect. 4 we have evaluated the proposed approach and finally we conclude our work in Sect. 5.

2. Related Work

Congestion is a special scenario where offered load for a link or a node excides its handling capacity. This paper dealing such nodes which are facing the problem of congestion due to too many packets in the network. A node may face congestion just because too many link failure or connection loss. In VANET most of the data transmitted by the nodes are broadcasted (single hop or multi hop), due to which congestion may arise in the network. Proposed method provides QoS through reducing the congestion in the network. Our approach based on the efficient use of queues or buffers available at the node. We have studied the problem of congestion in various environment like sensor network, mobile ad-hoc network, and vehicular ad-hoc network.

A congestion detection and avoidance approach for sensor network has been proposed in [5] which is an energy efficient scheme called CODA, it uses the mechanism of open loop, close loop and receiver based detection. The technique has been developed for sensor network which was not tested for the mobile nodes. A concept of utility based congestion controlled and packet forwarding is given in [6]. A utility factor is associated with each packet, in case of congestion a low utility factor packets are dropped. Assigning utility factor to each packet is not a fair scheme. A cross layer congestion control approach is given in [7] in which transport layer and physical layer jointly control the congestion using TCP – Reno2. In physical layer the transmission power changes according to condition of physical layer and in transport layer a window based flow control is used to handle the congestion. For the highly dynamic environment like VANET, TCP is not a good choice at transport layer. A priority based congestion control method was proposed in [8] which is an enhancement in the CODA [5]. Another improvement called decentralized, predictive congestion control (DPCC) scheme for WSN was proposed in [9] which uses an adaptive back-off interval selection and adaptive flow control approach to manage the unnecessary packets in the network. A weight is associated with each packet to maintain the fairness during the congestion.

All the above mention techniques are for the stationary nodes which are not tested for the mobile nodes. A hop_by_hop congestion control algorithm has been proposed in [10], which uses the both hop_by_hop control and end_to_end control to handle congestion. The proposed method is being tested for stationary nodes, which may not be feasible for mobile nodes. A priority based message scheduling approach is proposed in [11, 12] which provides a reliable and safe communication. The given approach dynamically schedule the messages to provide congestion control. The priorities are given, according to the message type. A hop_by_hop cross layer congestion control scheme proposed in [13] which is overplayed on the contention based MAC layer. It dynamically adjust the priority of the channel. To maximized the broadcast efficiency a congestion control strategy proposed in [14]. A heuristic technique is used to counter congestion in [15]. To detect congestion two methods are used first measurement based detection and event-driven detection in [16]. A coordinator based congestion control detection is proposed in [17], in which time slot based medium access method is used and scheduling is coordinated by a local node/vehicle. The proposed protocol takes beacon and emergency messages into account and is based on time-slot-based medium access. The whole network is virtually divided into segments; each segment is supervised by a local controller vehicle. By using different data rates and cutting the transmission time of beacon messages, this approach controls channel congestion. Reuse of unused time slots also improves bandwidth utilization.

A classification of existing congestion control techniques is given in [18] as well as a new dynamic approach called distributed fair transmit power adjustment for VANET (D-FTPA) is given. This technique dynamically adjusts the maximum beacon load or carrier sense threshold value. An improved hop_by_hop congestion control technique is proposed in [19] which uses a dynamic queue management for congestion control. To enhance the performance of VANET through decreasing control channel congestion has been proposed in [20]. It avoids the hidden terminal problem specially in multi-channel environment using RSU. A central and local data congestion control policy is suggested in [21], to regulate data congestion using roadside units (RSUs) at nodes. This approach consists of three components. Machine learning approach is applied to messages for gathering, filtering and clustering them. Clustering is done on the basis of message size, its validity, type also and k-mean clustering algorithm is applied to the messages. Data congestion control units at nodes calculate values for transmission rate, size of contention window and other parameters. At last RSUs at each node launch the values of parameters to immobile vehicles to lessen message crashes. A trust-based congestion control algorithm (TBCCA) has been proposed by [24] they have compared the priorities of each message transmitted by a node independently. The contribution of each node calculated based on the data disseminated by that node in the network. A trust parameter is being used to calculate the priority of the message. If number of nodes get increased this technique my phase a problem to assign the trust parameter. The various existing approached regarding vehicle_to_vehicle (V2V) congestion are analysed by [25]. They have analysed various parameters like data rate, power, etc to determine the congestion in the network. The improved awareness of the vehicle on road make things quite manageable if control parameters are used. This analysis was done only for V2V communication they did not consider the amount of data contributed by RSUs. A survey has been conducted by [26] to detect and control congestion in connected vehicles. Various solution have been proposed through an architecture which supports mainly urban and highway environment. Routing, statistical, Machine learning and location aware approaches may be well suited to deal with congestion. They have suggested a path to carry this work further. The node-based multicast congestion control approach proposed in [27] for the stationary nodes. The approach is based in adaptive join and leave of layers. The study in [28] suggests an intelligent vehicular traffic information system (IVTIS). This system builds system models corresponding to urban road networks. Author examines the gathering, broadcasting of information and automatic generation and update effect of road network congestion information.

The first lane-level solution for collision avoidance is proposed by Chen in [22] for preventing collisions, they have proposed a framework which is beacon-less, infrastructure-less and GPS-less cooperative collision avoidance (BIG-CCA) framework. To validate the viability of BIG-CCA, an android-based prototype is developed. A hybrid system is proposed in [23] where Device-to-Device communication is presented to increase delay performance of network. An algorithm is developed for cellular base stations for choosing the optimum set of device-to-device links. A performance lower bound for the algorithm is also derived in this research work. According to our literature survey the issue of congestion in VANET especially network layer congestion is still an unexplored and very challenging area compared to other issues like routing in VANET in which lots of work has been already carried out.

3. Proposed Method

The proposed work is divided into five subsections first section explains the congestion and its detection process, second subsection explains congestion control process, the third subsection explain the algorithm followed by an example which shows the comparison between propose work and the existing approaches, the forth sub section describes the experimental setup and the fifth subsection describes the mathematical proof of the proposed approach.

3.1. Congestion Detection: -

If node's receiving rate is higher than the forwarding or consumption rate, then the node get congested. Congestion avoidance is one of the biggest challenge for the VANET environment. In general, the approach used for congestion detection is acknowledgement-based congestion detection. In this approach a node send a packet to next node and if the next node send acknowledgement of the receipt of packet then there is no congestion and if it doesn't send acknowledgement then the node is said congested.

Figure 1 shows that how a node process the data, when all the neighbouring nodes send data in the IN_Q of a node.If IN_Q does not discard the data due to congestion it will be treated as pass. Data is then processed in a multilevel queue than placed into OUT_Q. Priority based data will be directly putted into the output queue. If IN_Q is full, data will be discarded. Following Table 1 shows the notations used in proposed approach. All the packets that are discarded due to insufficient IN_Q will be treated as a fail and pushed into bucket. The priority based packet will be directly forwarded to the OUT_Q. Different priority levels are assigned to the packets according to the adaptive differential service of IP packet for providing QoS.

Table 1
Notations Used in Proposed Approach

Symbol	Description
P_{siz}	Packet size
D_{rec}	Data received
D_{sen}	Data send
P_{avg}	average packet size
P_{loss}	Packet loss
IN_Q	In queue
OUT_Q	Out queue
E_{th}	Expected throughput
A_{th}	Actual throughput
P_{proc}	Processed packet.

3.2. Node-based Congestion control:

Our approach named node-based throughput computing (NBTH) approach is based on QoS policy named weighed random early delivery (WRED) Each node in VANET computes its throughput. As known to all, each node in VANET has its own input and output queues (named in our approach *IN_Q* and *OUT_Q*), see figure 2(a). The length of these two queues is fixed. For each node, we have fixed expected throughput (E_{tp}) which is based on its priority. Nodes in the network compute their throughputs that is called actual throughput (A_{tp}) of the node. Calculate the difference (*diff*) between E_{tp} and A_{tp}. This difference may be positive, negative or zero. If it is zero then no issue, the node is utilizing its full its capacity. If the *diff* is negative it means node is underutilize increase the priority of node so that it can receive more data from other nodes.And if the *diff* is positive it means the node is congested. This is the approach for congestion detection.

Now, next issue is how to handle network congestion. Cutting the traffic rate or packet rate is the only solution to deal with congestion. In proposed approach, whenever a node detects congestion it will broadcast a message (chock packet with highest priority) shown in Fig. 2(b), to its neighbouring nodes and neighbouring nodes will cut their packet sending rate by ¼ (see Fig. 4).

We have given highest priority to these messages because if the priority is not high there may be situation that the neighbouring node may also be congested and it will not listen to that message and keep forwarding packets to the congested node. Hence these messages are handled on priority bases. We have also assigned priorities to nodes in the network. Whenever a node want to transmit packet it will choose the highest priority node and if this node is congested then the sending node choose the next highest priority node to transmit the packet.

3.3. Congestion detection and control algorithm: -

Algorithmic description of the proposed approach is depicted in the following Figs. 3 and 4. Figure 3 shows congestion detection algorithm and Fig. 4 shows congestion control algorithm.

The proposed algorithm is compared with the Machine Learning – Congestion Control (ML-CC) algorithm proposed in [21], which is a cluster based approach to handle the congestion. For example, if we will compare proposed approach with the approach given in [21], which dynamically adjust the transmission range and transmission rate for the sender in a cluster using machine learning. In ML-CC the optimum results for throughput and average delay have been found when there are 4 clusters. For more than or less than 4 clusters the performance are being degraded. The proposed algorithm is based on the throughput of a node without any cluster so entire data will be considered and part of the network traffic. Thus there will be no rider like cluster or transmission range.

3.4. Experimental Setup: -

We have used NS-2.35, SUMO and MOVE to evaluate the purposed approach. In which MOVE, has been used to represent the road and vehicles in the network, SUMO is used to show the mobility pattern of the vehicles and NS-2.35 is used to simulate and analyze results through trace. For experimental setup, the different parameters used in the simulation and their brief description is shown in Table 2.

Table 2
Simulation parameters and their brief description.

Simulation Parameter	Value/ Range /Unit	Brief description
Number of vehicles	50, 100, 200, 300, 400	Density of the vehicles on the roads
Road Length	800m x 700m	Area for the vehicles to move in the road.
Number of lanes	4	2 lanes in each direction.
Maximum speed of a vehicle	20 m/s	Maximum speed of a vehicle could be 72 Kmph on an average
Simulation Time	500 Sec	Duration of a vehicle to move in the given setup.
Packet size	1024 bytes	Maximum payload of a data packet.
Sender data pattern	5 packets per sec	Packet rate for a vehicle to transmit.
Window Size	2048 bytes	Buffer available with a node.
Node placement type	Source of a Road	All vehicle start from the certain fixed point.
Traffic Light Time	60s	Traffic light duration at a junction.
MAC type	IEEE 802.11p	Wireless access in vehicular environment
Number of simulation	10	Number of experiment carried out per setup.

In a rage of 800 x 700 m^2as shown in Fig. 5, the area in which vehicle can move is near about 15% to 30 % which is basically the road area. The approximate length of road is 4 Km in which vehicles can move. Simulation parameters are compared with the parameters given in [21, 24]. The propagation method used for the scenario is also Nakagami propagation model, which is one of the well-suited model for urban scenarios [21, 24]. The number of vehicle which we have simulated are in a range of 50 to 400 vehicles in a cross section scenario. The cross-section scenario is being chosen because at the diagonal points the density of vehicles would be high due to which any node can face a congestion as shown in Fig. 5 and Fig. 6.

The cross-section points in the scenario shown by Fig. 5 and Fig. 6 clearly shows that the maximum number of vehicles are placed at a time. The maximum amount of data traffic faced by the priority node or vehicle at the cross-section point due to traffic light [21] or other vehicle weighting reasons. Figure 7 clearly mention that the maximum amount of communication is going on at cross-section points.

3.5. Discussion:

In node-based throughput approach if a vehicle or a node N_i receives n_{pi} packets from its n neighbours than the total data received by the node per unit of time in its IN_Q and the total data send by the node per unit of time in the OUT_Q is given by the Eqs. (1) and (2) respectively. Where $n_1$are the number of nodes from where N_i is receiving the data and n2 be the number of nodes to whom N_i is sending the data, where $0 < n_1 \leq n$ and $0 < n_2 \leq n$.

Where $P_{i_{asiz}}$ is the average packet size received by node N_i.

$$D_{rec} = \sum_{i=1}^{n_1} N_i * P_{i_{asiz}} \quad \text{-----------------------------------(1)}$$

$$D_{sen} = \sum_{i=1}^{n_2} N_i * P_{i_{asiz}} \quad \text{----------------------------------- (2)}$$

Each node compute its actual throughput by computing the amount of data processed by the node per unit of time is given in Eq. (3). A_{th} is the amount of data processed by a node form IN_Q to OUT_Q. Where μ is average processing capacity of a node.

$$A_{th} = \sum_{i=1}^{n_{pi}} \mu * P_{i_{siz}} \quad \text{-----------------------------------(3)}$$

Each node will be assigned an expected throughput (E_{th}). The E_{th} is computed using the amount of buffer associated with each node and its processing capacity. The input and output buffer size is divided into the number of average packet length and the processing capacity of the node per unit time. Let μ is the processing capacity of a node per unit time thus the expected throughput for a i^{th} node is given in Eq. 4.

$$E_{th} = \mu * P_{avg} \quad \text{-----------------------------------(4)}$$

A node will face a congestion if $A_{th} > E_{th}$ or in other words if a difference of expected throughput and actual throughput is positive than the node will be treated as a congested node and congestion control algorithm will be invoked.

4. Experimental Evaluation

4.1. Performance evaluation and simulation results:-

In this section performance of proposed NBTH approach is evaluated and compared with the congestion control with machine learning (ML-CC). We compared the proposed approach with two other approaches first one is when there is no congestion control strategy is being adopted hear it is named without nodes based throughput (WNBTH) and the second approach is ML-CC [21]. The same scenario has been created to compare all the three approaches.

4.1.1 Average throughput: -

In figure. 8, ML-CC dynamically adjust the transmission range and rate of a node. NBTH adjust the transmission rate only once the congestion has been detected. The throughput of node based throughput (NBTH) scheme is exhibiting better performance than without node based throughput (WNBTH) for the high node density of nodes. Thus, we can say that NBTH can reduce the congestion in a high-density environment. For 400 vehicles, the throughput of ML-CC, WNBTH and NBTH are 9.28, 8.49 and 9.46 Mbps respectively. The NBTH has shown slightly better results than the ML-CC for high density of nosed because in case of ML-CC after detection of congestion it adjust the transmission range. As the number of vehicles are increased the network faces congestion due to which WNBTH performance degrades. Initially for 50 and 100 nodes the throughput of ML-CC, WNBTH and NBTH are very much similar but as the number of vehicles increased in the road the performance of ML-CC and NBTH are relatively higher than the WNBTH.

4.1.2: - Average end_to_end delay:-

Figure 9 shows the end_to_end delay for the ML-CC, WNBTH and NBTH techniques. The end_to_end delay of NBTH scheme is better than the ML-CC and WNBTH for the high node density of nodes. Initially for 50 and 100 vehicles the end_to_end delivery of messages in ML-CC is very efficient because of the adjustment in the transmission range and rate. As the number of vehicles increased in the network, NBTH delivers data in more efficient way than the ML-CC or WNBTH. For 400 vehicles, the end_to_end delay for ML-CC, WNBTH and NBTH is 240, 298 and 225 ms respectively. Thus, the proposed method NBTH is able to make the efficient delivery of messages in comparison to the ML-CC and scenario where we have not implemented any congestion control mechanism like a case of WNBTH.

4.1.3. Packet Loss: -

Figure 10 shows the results of packet loss for the approach proposed in this paper named NBTH with the ML-CC and WNBTH techniques. The result shows that initially for 50 vehicles the packet loss for ML-CC is least while as it is very high for WNBTH. For high density of nodes, the packet loss of proposed method NBTH is better than the ML-CC strategy. The number of packet drops for the scenario where there is no congestion control strategy has been deployed is very high in comparison to the scenario where congestion control strategy is being deployed. The difference in the packet drops are quite high for the scenario where no congestion control strategy is being deployed such as WNBTH in comparison to the scenario where congestion control strategies are deployed such as NBTH and ML-CC. The packet drops for 400 vehicles in case of ML-CC, WNBTH and NBTH are 104128, 60s8523 and 101523 respectively. Thus, for high density of vehicles the proposed method has shown better results than the ML-CC.

5. Conclusion And Future Work

This paper proposes a Node Based Throughput (NBTH) approach to provide an efficient delivery of data in an urban environment for high density of nodes. The proposed method handle the congested network through efficient queue management with feedback mechanism. NBTH method assigns highest priority to that node which is having the best possible path to deliver the data. All the neighbouring node will pick the highest priority node in their network range to deliver their data. The proposed method results are being compared with the normal scenario called WNBTH and a machine learning based congestion controlled method named ML-CC. The proposed method NBTH has been evaluated in three metrics called throughput, end_to_end delay and packet drop. The throughput, end_to_end delay and packet loss of proposed algorithm depicted batter then the one of the existing solution termed ML-CC. NBTH shown a significate enhancement in the throughput and decreased end_to_end delay and packet loss, as compared to scenario where we have not deployed the congestion control mechanism. Therefor the proposed approach NBTH has shone batter throughput then the available solutions for congestion control.

Declarations

Funding: Not applicable

Conflicts of interest/Competing interests: No

Availability of data and material: As and when required.

Code availability: Not applicable

Ethics approval: Not applicable

Consent to participate: Not applicable

Consent for publication: As and when required.

References

1. Zeadally, S., Hunt, R., Chen, Y., Irwin, A., & Hassan, A. (2010). Vehicular ad hoc networks (VANETS): status, results, and challenges. *Telecommunication Systems, 50*(4), 217–241.
2. Xu, Q., Mak, T., & Sengupta, R., & Oct. (2004), "Vehicle-to-vehicle safety messaging in DSRC," in Proc. Ist ACM Workshop on Vehicular Ad Hoc Network, (Philadelphia, PA, USA).
3. Harsch, C., Festag, A., & Papadimitratos, P. (2007, October). Secure position-based routing for VANETs. In *2007 IEEE 66th Vehicular Technology Conference* (pp. 26–30). IEEE.
4. Tanenbaum, A. S. (2003). *"Computer networks, 4-th edition. " ed: Prentice Hall.*
5. Wan, C. Y., Eisenman, S. B., & Campbell, A. T. (2003, November). CODA: Congestion detection and avoidance in sensor networks. In *Proceedings of the 1st international conference on Embedded networked sensor systems* (pp. 266–279).
6. Wischhof, L., & Rohling, H. (2005, October). Congestion control in vehicular ad hoc networks. In *IEEE International Conference on Vehicular Electronics and Safety, 2005.* (pp. 58–63). IEEE.

7. Kliazovich, D., & Granelli, F. (2006). Cross-layer congestion control in ad hoc wireless networks. *Ad Hoc Networks, 4*(6), 687–708.
8. Wang, C., Sohraby, K., Lawrence, V., Li, B., & Hu, Y. (2006, June). Priority-based congestion control in wireless sensor networks. In *IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy* Computing *(SUTC'06)* (Vol. 1, pp. 8-pp). IEEE.
9. Zawodniok, M., & Jagannathan, S. (2007). Predictive congestion control protocol for wireless sensor networks. *IEEE Transactions on Wireless Communications, 6*(11), 3955–3963.
10. Yi, Y., & Shakkottai, S. (2007). Hop-by-hop congestion control over a wireless multi-hop network. *IEEE/ACM Transactions On Networking, 15*(1), 133–144.
11. Bouassida, M. S., & Shawky, M. (2008, November). On the congestion control within VANET. In *2008 1st IFIP Wireless Days* (pp. 1–5). IEEE.
12. Bouassida, M. S., & Shawky, M. (2010). A cooperative congestion control approach within VANETs: formal verification and performance evaluation. *EURASIP Journal on Wireless Communications and Networking*, *2010*, 1–13.
13. Wu, G., Xia, F., Yao, L., Zhang, Y., & Zhu, Y. (2011). A hop-by-hop cross-layer congestion control scheme for wireless sensor networks. *arXiv preprint arXiv:1201.0207*.
14. Ye, F., Yim, R., Roy, S., & Zhang, J. (2010). Efficiency and reliability of one-hop broadcasting in vehicular ad hoc networks. *IEEE Journal on Selected Areas in Communications, 29*(1), 151–160.
15. Taherkhani, N., & Pierre, S. (2012, October). Congestion control in vehicular ad hoc networks using meta-heuristic techniques. In *Proceedings of the second ACM international symposium on Design and analysis of intelligent vehicular networks and applications* (pp. 47–54).
16. Darus, M. Y., & Bakar, K. A. (2013). Congestion control algorithm in VANETs. *world applied sciences journal, 21*(7), 1057–1061.
17. Sahoo, J., Wu, E. H. K., Sahu, P. K., & Gerla, M. (2013). Congestion-controlled-coordinator-based MAC for safety-critical message transmission in VANETs. *IEEE transactions on intelligent transportation systems, 14*(3), 1423–1437.
18. Taherkhani, N., & Pierre, S. (2015). Improving dynamic and distributed congestion control in vehicular ad hoc networks. *Ad Hoc Networks, 33*, 112–125.
19. Yi, Y., & Shakkottai, S. (2007). Hop-by-Hop Congestion Control Over a Wireless Multi-Hop Network. *IEEE/ACM Transactions on Networking, 15*(1), 133–144.
20. Li, X., Hu, B. J., Chen, H., Andrieux, G., Wang, Y., & Wei, Z. H. (2015). An RSU-coordinated synchronous multi-channel MAC scheme for vehicular ad hoc networks. *IEEE access : practical innovations, open solutions, 3*, 2794–2802.
21. Taherkhani, N., & Pierre, S. (2016). Centralized and localized data congestion control strategy for vehicular ad hoc networks using a machine learning clustering algorithm. *IEEE Transactions on Intelligent Transportation Systems, 17*(11), 3275–3285.

22. Chen, L. W., & Chou, P. C. (2015). BIG-CCA: Beacon-less, infrastructure-less, and GPS-less cooperative collision avoidance based on vehicular sensor networks. *IEEE Transactions on Systems, Man, and Cybernetics: Systems, 46*(11), 1518–1528.
23. Cao, X., Liu, L., Cheng, Y., Cai, L. X., & Sun, C. (2016). On optimal device-to-device resource allocation for minimizing end-to-end delay in VANETs. *IEEE transactions on vehicular technology, 65*(10), 7905–7916.
24. Singh, R. P., & Singh, D. (2018, July). Trust Based Congestion Control Algorithm (TBCCA) in VANET. In *International Conference on Advanced Informatics for Computing Research* (pp. 524–535). Springer, Singapore.
25. Liu, X., & Jaekel, A. (2019). Congestion control in V2V safety communication: Problem, analysis, approaches. *Electronics, 8*(5), 540.
26. Paranjothi, A., Khan, M. S., & Zeadally, S. (2020). A survey on congestion detection and control in connected vehicles. *Ad Hoc Networks, 108*, 102277.
27. Singh, K., & Yadav, R. S. (2014). Efficient Multicast Congestion Control. *Wireless personal communications, 78*(2), 1159–1176.
28. Zhang, Q., Zheng, H., Lan, J., An, J., & Peng, H. (2016). An autonomous information collection and dissemination model for large-scale urban road networks. *IEEE Transactions on Intelligent Transportation Systems, 17*(4), 1085–1095.

Figures

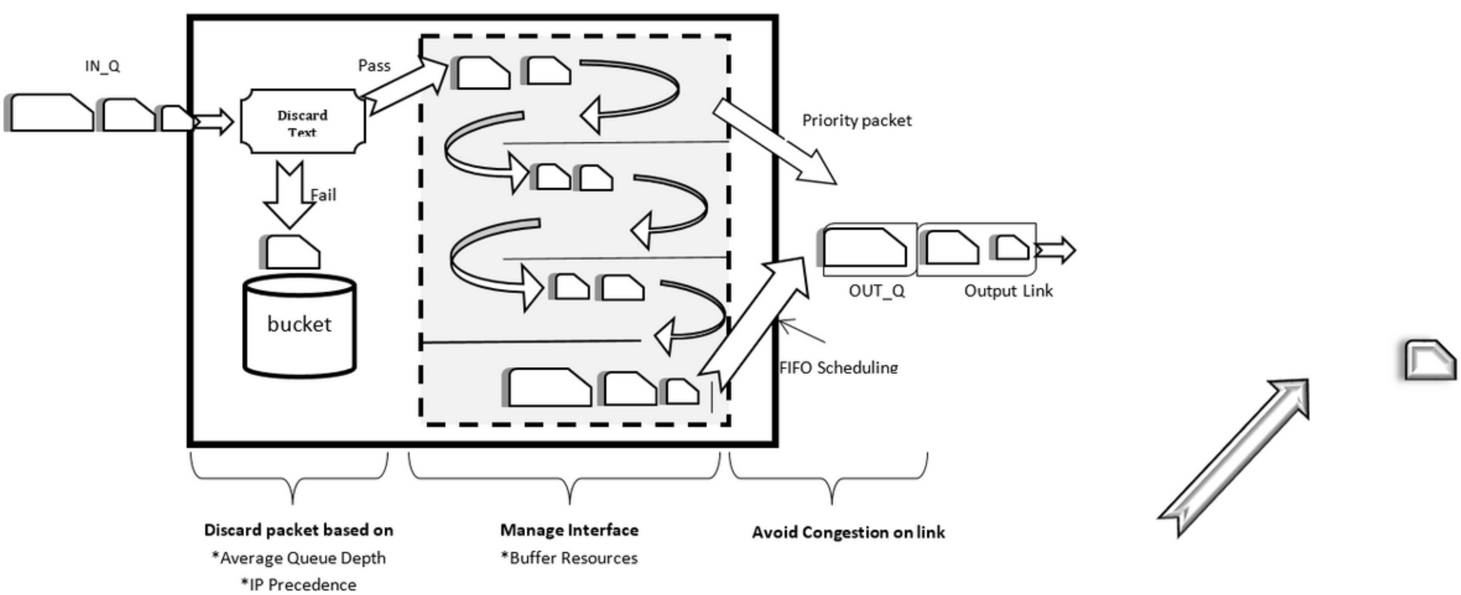

Figure 1

Congestion Queue of a node.

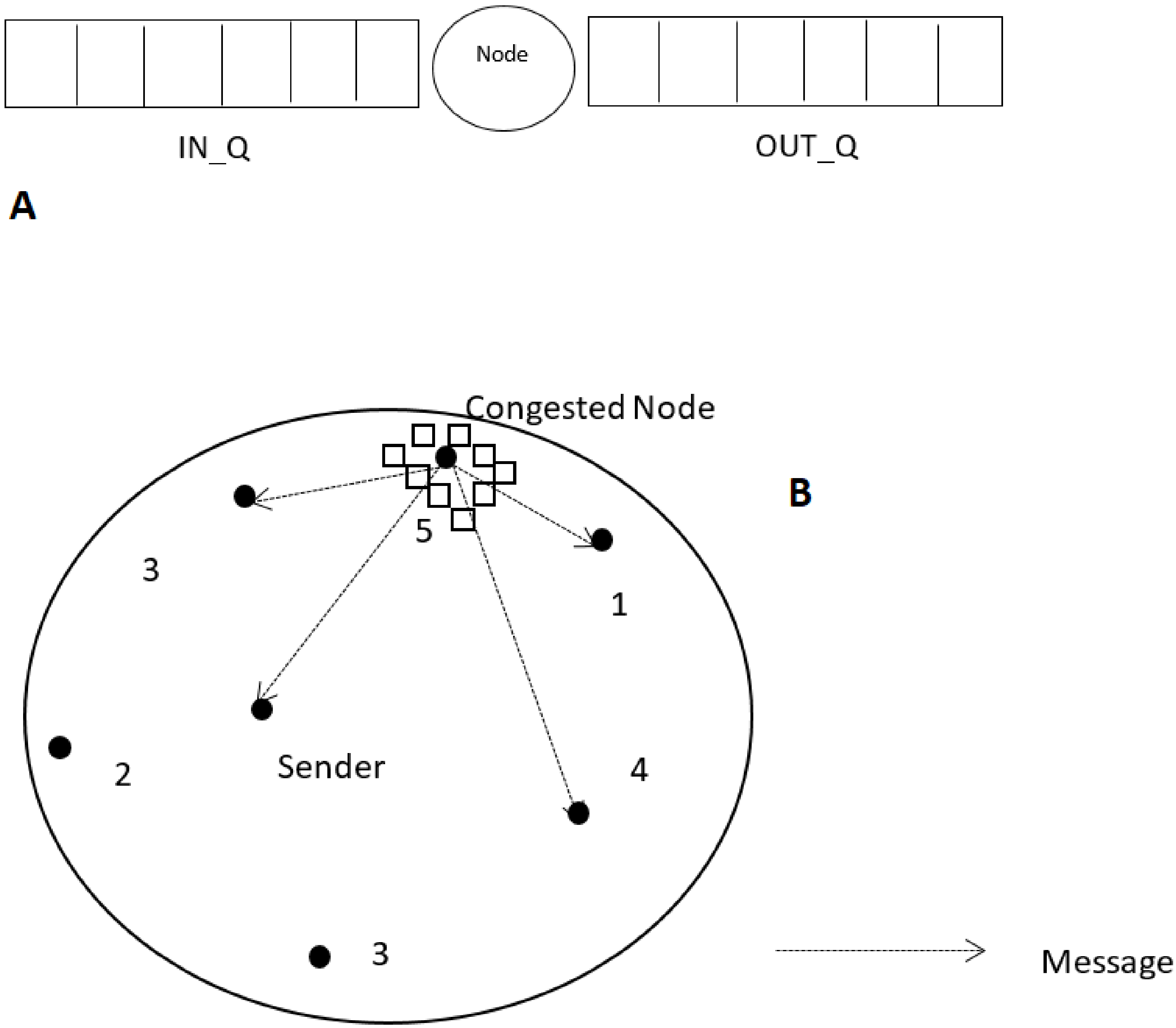

Figure 2

(a): Input and output queue for a node. (b): Congestion message by the cogested node

Step1: Start
Step2: Send HELLO message to all
Step3: Initialized the expected throughput (*expect_tp*) to each node.
Step4: if num_neighbor_node> 1 &&send_data_packet = = true
Step5: Compute the actual throughput (*actual_tp*) of the node.
Step6: calculate the difference (*diff*) between *expect_tp* and *actual_tp*.
Step7: if (*diff* == 0) then
Setp8: calculate the probability of transmission p for every k_i
Step9: Grade the node k_i with maximum probability.
Step10: send_data_packt using k_i
Step11: else if diff > 0
Step12: call the cong_cont_algo
Step13: else choose the peripheral pattern.
Step14: else send_data_packt using k_0
Step15: Stop

Figure 3

Congestion Detection Algorithm

Step1: Start
Step2: Send chock_packet to the all k_i neighbors
Step3: if chock_packet received
Step4: cut transmission rate by ¼ of the current transmission rate
Step5: go to step 2
Step6: Stop

Figure 4

Congestion Control Algorithm

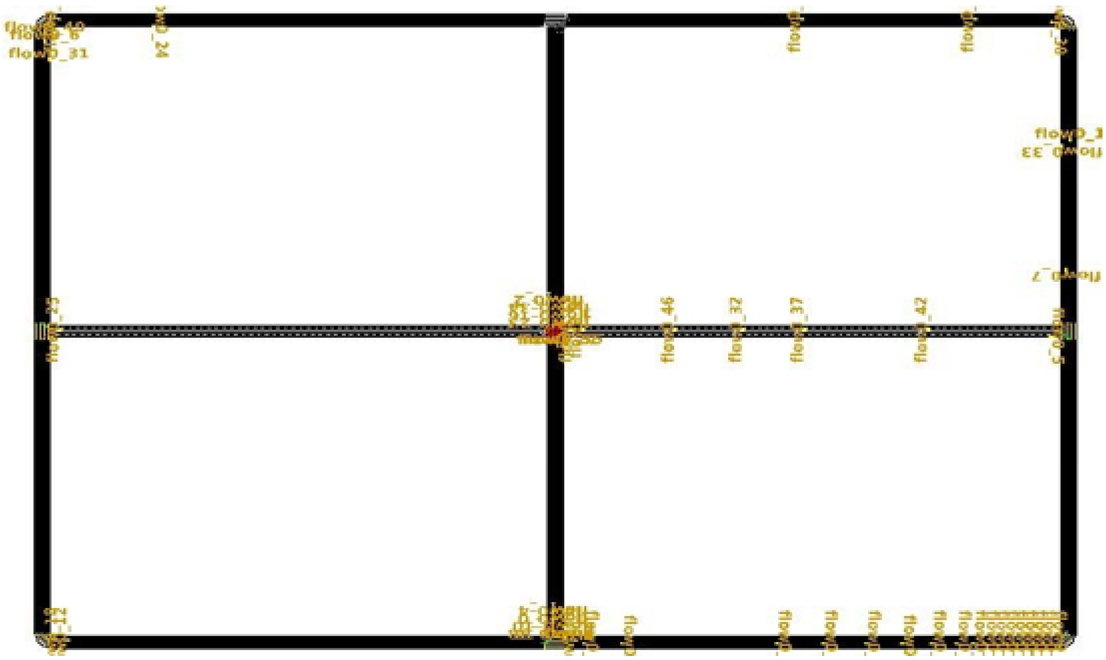

Figure 5

Cross section mobility scenario with 50 vehicles.

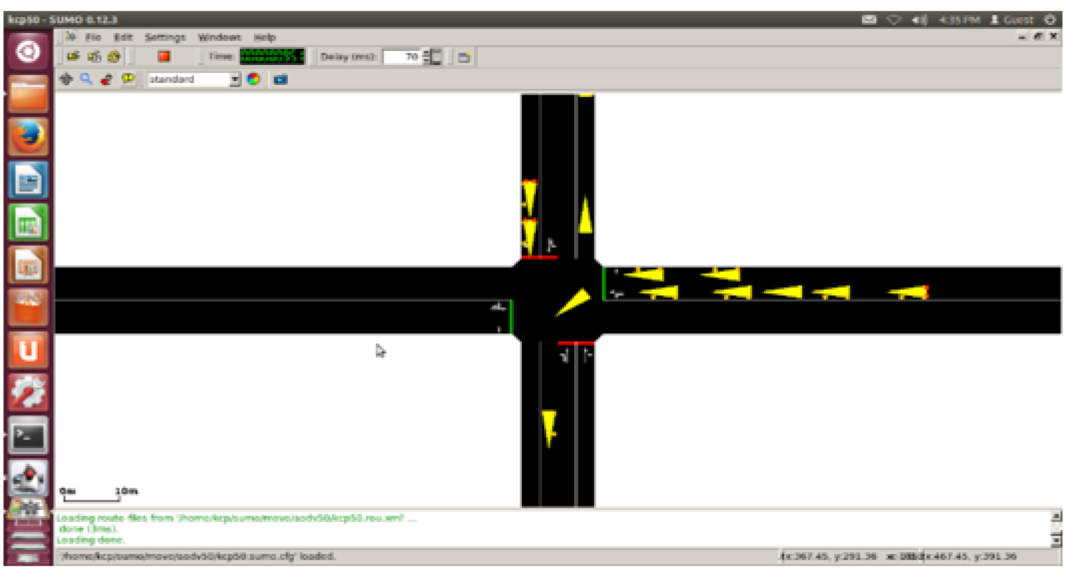

Figure 6

Vehicle movement and the traffic light at a cross-section.

Figure 7

Communication between the vehicles at a cross-section point in VANET.

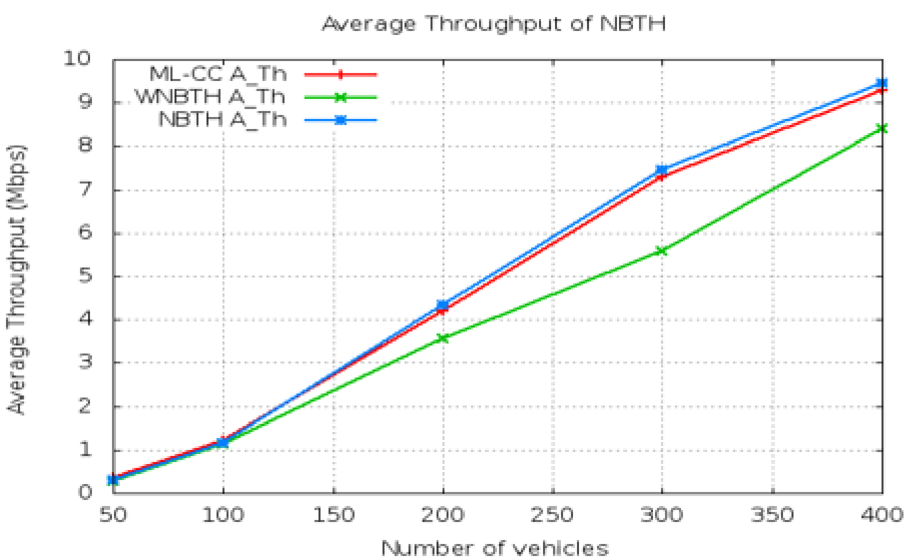

Figure 8

Average throughput of NBTH

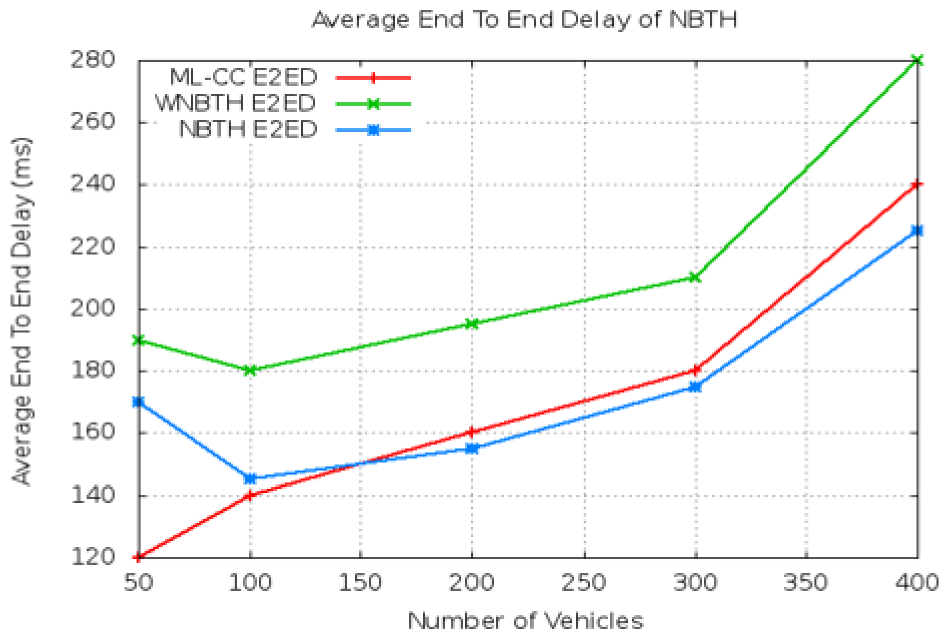

Figure 9

Average end_to_end delay of NBTH

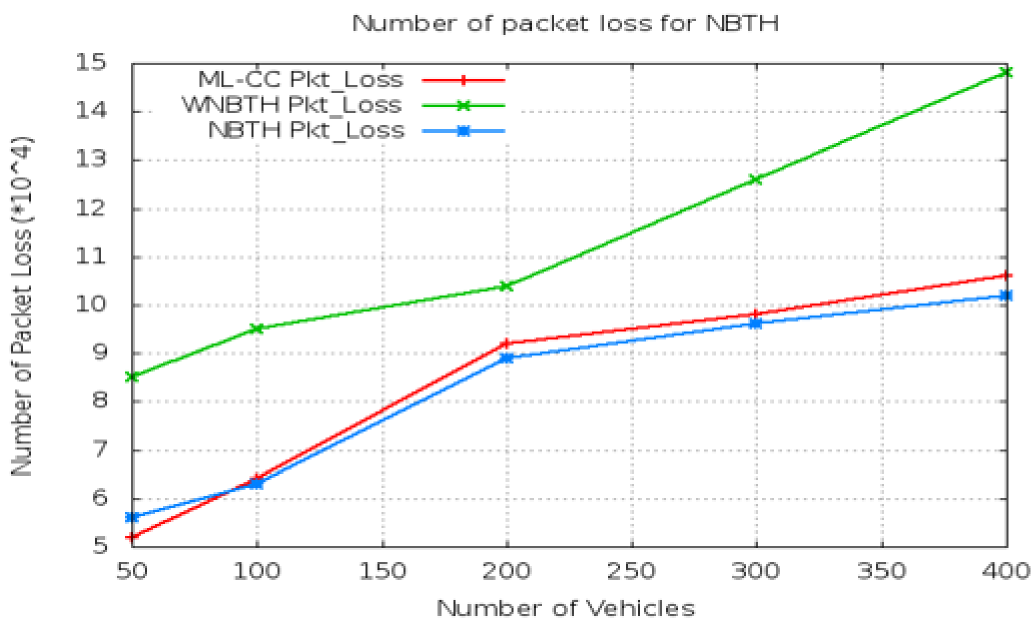

Figure 10

packet loss for NBTH