ISSN: 2349 - 4891
International
Journal of Recent Research and Applied Studies
(Multidisciplinary Open Access Refereed e-Journal)

Adaptive Congestion Window Management Mechanism for Highly Robust Vehicular Communication Environment

Nisha Angeline M[1], Shivaranjani M[2]Akash C Anand[3],Arunkumar S[4]Balaraman A[5], Jaishree V[6]

[1]Associate Professor and Head, [2,3,4,5,6]UG Students - Final Year, Department of Electronics and Communication Engineering, Velalar College of Engineering and Technology, Erode - 638012, Tamilnadu, India

Abstract

In recent years, inter vehicular networking technologies are gaining momentum due toitsmergerwithwirelessinternetthroughinfrastructuregateways.Alargequantityofinternetdata traffic heavily relies on transmission control protocol (TCP) due to reliable connection-oriented services. TCP with sluggish window growth and weakly shaped congestion control mechanism fails under multi-hop inter vehicular conditions due to disjoint links and wireless channel errors. This paper introduces inter vehicular access network (IVAN) TCP, designed to suppress the limitationsofcongestioncontrolandwindowincrementalgorithminthemulti-hopintervehicular wireless environment. The IVAN's flexible window increment algorithm newer growth pattern achieves a faster increase in the transmission rate. Furthermore, IVAN's congestion control algorithmreducesthetransmissionratebasedonsender'swindowutilityandrelayrouter'sexplicit congestion notification. The simulation results confirm the significant rise in IVAN's throughput performance, substantial reduction in the queue packet drop and packet latency against standard TCP variants under inter vehicularenvironment.

Keywords: Vehicular ad hoc network (VANET), Transmission control protocol (TCP), Slowstart (SS), Flexible window increment (FWI), Inter vehicular access network(IVAN)

1. Introduction

In this world the population increases day by day and usage of internet also increased.We may face lot of issues while using internet during traveling. To reduce them we prefer VANET. Nowadays, VANET has become a great area for research purpose. VANET is an intelligent productoftransportsystem.VANETstandsforVehicleadhocNetworkwhichmeansthemoving vehicles which are connected with each other using a wireless medium to communicate in urban orhighwaysisknownasVANET.ThemainpurposeofVANETistoprovideanadditionalsafety and comfort to the drivers. VANET prevents us from collisions, traffic violation, dynamic route scheduling, traffic monitoring, blind crossing and it also provides internet connectivity between vehicular nodes. VANET have come up with the idea of interaction with vehicle to vehicle and road

ISSN: 2349 - 4891
International
Journal of Recent Research and Applied Studies
(Multidisciplinary Open Access Refereed e-Journal)

side unit by using wireless sensor device which are mounted in every vehicle with Network medium. VANET are been classified into two. In VANET vehicular communication are used to generate the information or distributed the information with vehicle to vehicle (V2V) and vehicle to infrastructure(V2I).

OneofthemajorprotocolsusedfortheInternetprotocolsuiteistheTransmissionControl Protocol (TCP). It helps in establishing and maintaining the network communication through which they could exchange their data. It is initiated only for transmission of packets. TCP acts as an interface between client and server. Traditional TCP was developed in the 1970s. In 1983 adoptedasthestandardforARPANET.TheTCParesubdividedinto3stagestheyareConnection establishment, Data transfer and Connection termination. For reliable connection TCP follows synchronize (SYN) and acknowledge (ACK) method. It is been performed in 3 steps which are SYN,SYN-ACK,ACK.Majordrawbackiscommunicationfailurewiththehostandthenetwork, receiving duplicate acknowledgement leads to packet loss and routingproblems.MajorproblemsfacedinTCPduetoVANETarelackingininfrastructure,communicatio n failure, faces terminal issues, hidden terminals, uni-directional links and mobility of nodes with frequentpathbreakwithhugepacketloss.Averagepacketlosstakesplacewhenoneormorethan one packetissending from the computer network fails to reach the destination then packets are considered loosed. It's happened due to congestion of network. The packet loss is calculated in percentage with respect to the packets which are sending. The TCP could detect the packets and perform a retransmission process to make reliability of messaging. Random packet loss is been duetothequeueandComplexradioenvironment.Thepacketsaresentrandomlymayfacepacket loss in larger packet transmission on considering the time lag for sending the larger packets. Random Early Detection (RED) is used to prevent early packet loss. But it can be controlled in certain casesonly.

To overcome all the issue, we are going to design a TCP which handles connectionintelligently. The main contribution to the paperis

- New congestion control algorithm is designed forVANET.
- Window increment algorithm is modified before and after thecongestion.
- The performance of the proposed system is verified using NS2 and outcomes are longer with existing TCPvariant.

2. Literature Survey Newreno

The extension of TCP new Reno is a TCP Reno. Reno has overcome many limitations. The second variant of the TCP is a TCP Reno which has an in-built congestion algorithm. When packetlossoccursintheReno,itreducesthecountby50%insendersidewiththethreshvalueand it may allow the network to come out of the congestion state easily. There may cause critical backlog in Reno and its performance may affect. For detecting multiple packet losses in the same congestion window, it may take long time to process. When we received the segment there may be an immediate response from theReno.

2.1 Cubic

ISSN: 2349 - 4891
International
Journal of Recent Research and Applied Studies
(Multidisciplinary Open Access Refereed e-Journal)

The congestion control protocol for TCP is a Cubic (transmission control protocol) and inLinux it is the current default TCP algorithm. High speed variant of standard TCP is CUBIC. CUBICandmanyearlierTCPalgorithmshavemanydifferencesandCubichasreliedonthecadence of RTTs to increase the window size.

ISSN: 2349 - 4891
International
Journal of Recent Research and Applied Studies
(Multidisciplinary Open Access Refereed e-Journal)

To maintain the growth rate as matches as TCP. CUBIC enhance new TCP mode to change the situations. In cubic TCP mode when RTT is short it uses the same congestion control mechanism. For cubic the Binary Increase Congestion Control (BIC- TCP) [XHR04], a predecessor has been selected as the default TCP congestion controlAlgorithm by Linux.

2.2 Compound

It was developed by the Microsoft "vista" for operating system. It is the combination of two Congestion control mechanism one is TCP Reno and another one is TCP Vegas. Compound TCP has high bandwidth and low delay product. When their network buffers are small, they may even work well. CTCP design which satisfies the both efficiency requirements and TCP requirements at same time. Without affecting its performance, it also improves its connection throughput in CTCP congestion. In CUBIC to increase the convergence speed, we additionally add heuristic in CUBIC.

2.3 Vegas

It is a TCP congestion avoidance algorithm which gives important to packet delay, otherthanpacketloss,forsendingthepacketssignalsmayhelptodetermineitsrate.TCPVegasdetects congestion at an initial stage depends upon increasing Round-Trip Time (RTT) values of the packets in the connection congestion may detected only via packet loss. The Base RTT value of accurate calculation based on algorithms. When the RTT values are too small then throughput of the connection will be less than the applicable bandwidth. When the value is too large then it will connectionwillbeoverrunout.Foravoidingcongestivecollapse,itusesmulti-facetedcongestion control tricks inTCP.

2.4 Westwood

TCP Westwood (TCPW) which is modification done only in sender side of the TCP congestion window algorithm which may help to work well in TCP Reno in wired as well as wirelessnetworksmechanism.MostimportantfeaturesofWestwoodTCPareitdoesnotneedany inspection or interception of TCP pockets at intermediate nodes. For implementing TCP Westwood there are two methods first one is Countback and the second one is Recovering from sporadic error which may provide fast transmission and fast recovery. When random packet loss exist in few percent TCPW functionality will be low. TCP Westwood have well-known good controllable andfriendliness.

3. Existing System

The IVAN customized for inter vehicular wireless access environment requires four major modifications.Modification 1: The sender side cwnd growth algorithm i.e., FWI algorithm.

ISSN: 2349 - 4891
International
Journal of Recent Research and Applied Studies
(Multidisciplinary Open Access Refereed e-Journal)

Modification 2: The relay node's single queue dual marking (SQDM) algorithm. Modification 3: The receiver's congestion echo state variables.

ISSN: 2349 - 4891
International
Journal of Recent Research and Applied Studies
(Multidisciplinary Open Access Refereed e-Journal)

3.1 Flexible Window Increment (FWI) Algorithm

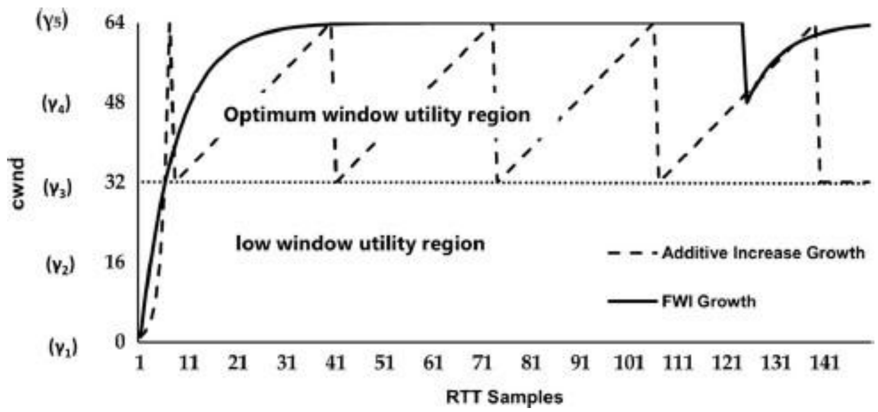


Figure 1 Exponential versus flexible window growth

The sender side FWI algorithm divides the entire cwnd growth into five thresholds based onthereceiver'sadvertisedwindowcapacity.ThesenderusesthethresholdstomonitorTCPflows growth level and initiates rate reduction based on the cwnd growth. The threshold γ1 represents theinitialwindow(int_wnd),γ5themaximumwindow(max_wnd),γ3centroidwindow,γ2lower midpoint between γ1 and γ3, and γ4 the upper midpoint between γ3 and γ5. The threshold above γ3referredasoptimumwindowutilityandbelowasalowerwindowutility.Intheoptimumwindow utility, the cwnd growth rate exceeds 50% of the receiver advertised capacity and vice versa in the lower utility region. For each successful ACK, the FWI updates the transmission rate by computing current cwnd, γ5, and increment parameter (inc_p). The inc_p determines thecwnd growth and takes the value 0.125, higher inc_p value results in bottleneck link congestion and smaller inc_p value leads to slower window increment rate. The SS phase cwnd update function and the window threshold levels computed using the followingequations.

$$cwnd_{i+1} = cwnd_i + (\gamma_5 - cwnd_i) * inc_p; \qquad 0 < cwnd < \gamma_5 \qquad (1)$$

$$\gamma_5 = \text{max\_wnd} \qquad (2)$$

$$\gamma_1 = \text{int\_wnd} \qquad (3)$$

$$\gamma_3 = \frac{\gamma_1 + \gamma_5}{2} \text{(centroid)} \qquad (4)$$

$$\gamma_2 = \frac{\gamma_1 + \gamma_3}{2} \text{(Lower midpoint)} \qquad (5)$$

$$\gamma_4 = \frac{\gamma_3 + \gamma_5}{2} \text{(Upper midpoint)} \qquad (6)$$

ISSN: 2349 - 4891

International
Journal of Recent Research and Applied Studies
(Multidisciplinary Open Access Refereed e-Journal)

ISSN: 2349 - 4891
International
Journal of Recent Research and Applied Studies
(Multidisciplinary Open Access Refereed e-Journal)

When the cwnd reaches the maximum receiver advertised capacity ($\gamma 5$), FWI ends the SS phase and moves to CA phase. In the maximum window condition i.e. cwnd = $\gamma 5$, the sender compares current RTT against the sum RTT i.e. the average of last three RTTs and sets the ca phase new ssthresh. The ssthresh and cwnd updated using the following equation.

$$\text{ssthresh} = \begin{cases} \gamma_5 * \frac{3}{4}; RTT < SUM\_RTT \\ \gamma_5 * \frac{1}{2}; RTT \geq SUM\_RTT \end{cases} \quad (7)$$

3.2 Single Queue Dual Marking (SQDM)Mechanism

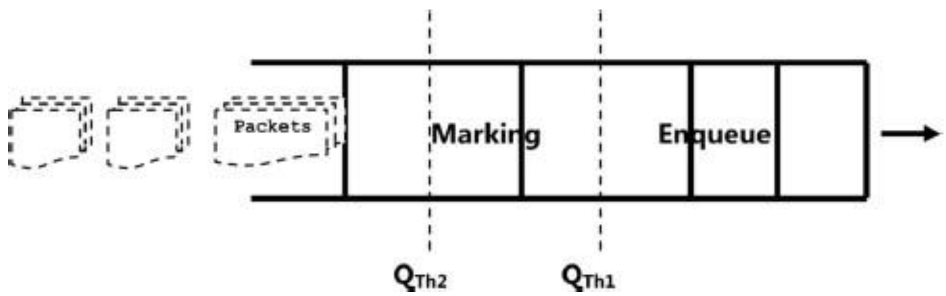


Figure 2 Queue Model: Single Queue Dual Marking (SQDM)

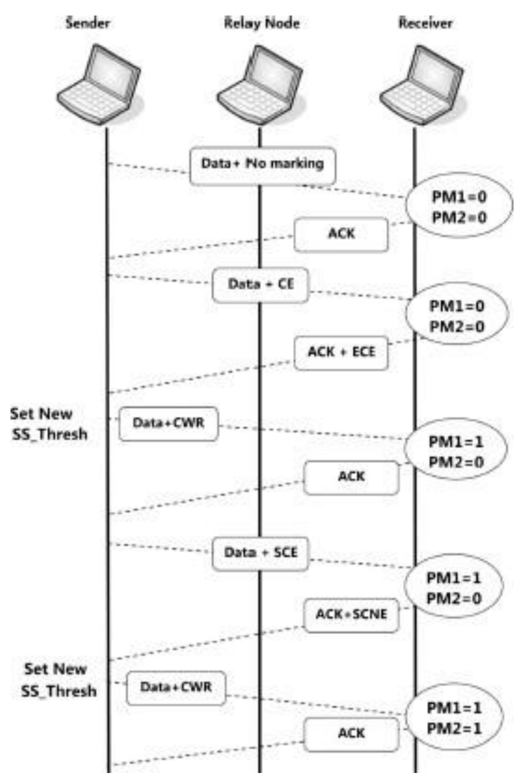
TheIVANcongestioncontrolalgorithmheavilydependsonrelayrouter'spacketmarking information. The idea behind SQDM carved from the NCE's ECN based queue mechanism with minor modifications in the threshold settings and congestion notification, i.e., two level notifications.TheexistingNCE'squeuingalgorithmwithsmallerqueuethresholdvaluefalselyinitiatesspuriouscongestionalertduringre-routing,resultsinneedlesstransmissionratereduction. The SQDM approach maintains two threshold levels ($Q_{Th1}$and Q_{Th2}) to notify the sender about therelaynodesqueuebuildup.TheQ_{Th1}fixedas50%ofaveragequeuesize,providesanadequate buffer capacity to store the packets during rerouting that occurs due to link breakage. The Q_{Th2}markinglevelchosenas75%ofthetotalaveragequeuesize(Q_{Total}),notifiesthesenderaboutthe critical queue buildup in the relayrouters.

$$Q_{(k+1)} = (1 - Q_{wt}) * Q_{Avg(k)} + Q_{wt} * Q_{cur(k)} \quad (8)$$

Where$Q_{cur}$is the current queue size and Q_{wt}be the queue weight, takes the value 0.2. For every packetarrival,SQDMchecksthestatusoftheQ$_{Avg_cur}$,iftheQ$_{Avg_cur}$exceedsQ$_{Th1}$,relayrouters marktheincomingpacketswiththecongestionexperienced(CE)fieldintheinternetprotocol(IP) headerthatalertsthesenderaboutincipientnetworkcongestion.

WhentheQ$_{Avg_cur}$exceedsQ$_{Th2}$, the relay router performs the second level marking by initiating a newly added severe congestion experience(SCE)flagintheIPpacketheader.Therelayrouterduallevelmarkingimprovessender alertness towards network congestion. In short, the sender will have an update on appropriate congestionlevelandreducesthetransmissionrateaccordingly.Furthermore,theSQDMalgorithm with improved marking threshold overcomes the spurious congestion notificationimpact.

3.3 Receiver EchoNotification



ISSN: 2349 - 4891
International
Journal of Recent Research and Applied Studies
(Multidisciplinary Open Access Refereed e-Journal)

Figure 3 TCP IVAN Flow Diagrams

The receiver maintains packet marking-1 (PM1) and packet marking-2 (PM2) variables that precisely echo the relay router's congestion notification to the sender. For non-congested condition, the receiver state variables remain inactive ('PM1 = PM2 = 00). For first level CEnotification, receiver enables ECE flag in the ACK packet to notify the incipient congestion. On receiving the first level CWR flag, the receiver stops first level echo and activates PM1 ('PM1 = 1 and PM2 = 00) i.e., no further marking of level 1 notification.

For second level SCE packet marking, receiver enables SCNE flag to notify the severe congestion. On receiving the second levelCWRresponseflag,receiveractivatesboththestatevariables('PM1=PM2=10).Thestate variables PM1 and PM2 reset to zero for no marking condition. The receiver congestion echoes based on two state variables swiftly react to queue buildup and supports two level congestion notifications.

4. Performance Analysis

4.1 Good put

In data transmission goodput measures how fast and accurately useful data travels in the network and arrives at its desired location.

Goodput = Size of a transmitted file / the time it takes to transferthefile (9)

4.2 *Throughput*

In data transmission, throughput is the amount of data moved successfully from one place to another place in a given time period, and typically it's measured in bits per second (bps), as in megabits per second (Mbps) or gigabits per second (Gbps).

Max throughput = RCV buffer size/RTT (10)

4.3 Delay

ISSN: 2349 - 4891
International
Journal of Recent Research and Applied Studies
(Multidisciplinary Open Access Refereed e-Journal)

Delay is a design and performance characteristic of telecommunications network. It specifies the latency for a bit of data to travel across the network from one communication end point to another. It is typically measured in multiples or fractions of a second.

Delay = Length between sender and receiver / Size ofdatapacket (11)

4.4 Packet loss

Packet loss describes lost packets of data not reaching their destination after being transmitted from one end to another end across a network.

Packet loss = Number of lost packets / Number ofreceivedpacket (12)

4.5 Error Rate

Depending upon the different values error rate the goodput, throughput, delay and packet lossvaluesmaybechange.Whentheerrorrateincreasesthroughputincreasesandwhenerrorrate decreases the throughput decreases. In case of goodput the when error rate increases goodput increases likely when the error rate decreases the goodput also decreases. Similarly, when we see thepacketdroptheerrorrateincreasesthepacket dropdecreasesatthesametimewhenerrorrate decrease packet drop increases. While error rate increases the delay decreases likely when error rate in a network decreases delay isincreases.

ISSN: 2349 - 4891
International
Journal of Recent Research and Applied Studies
(Multidisciplinary Open Access Refereed e-Journal)

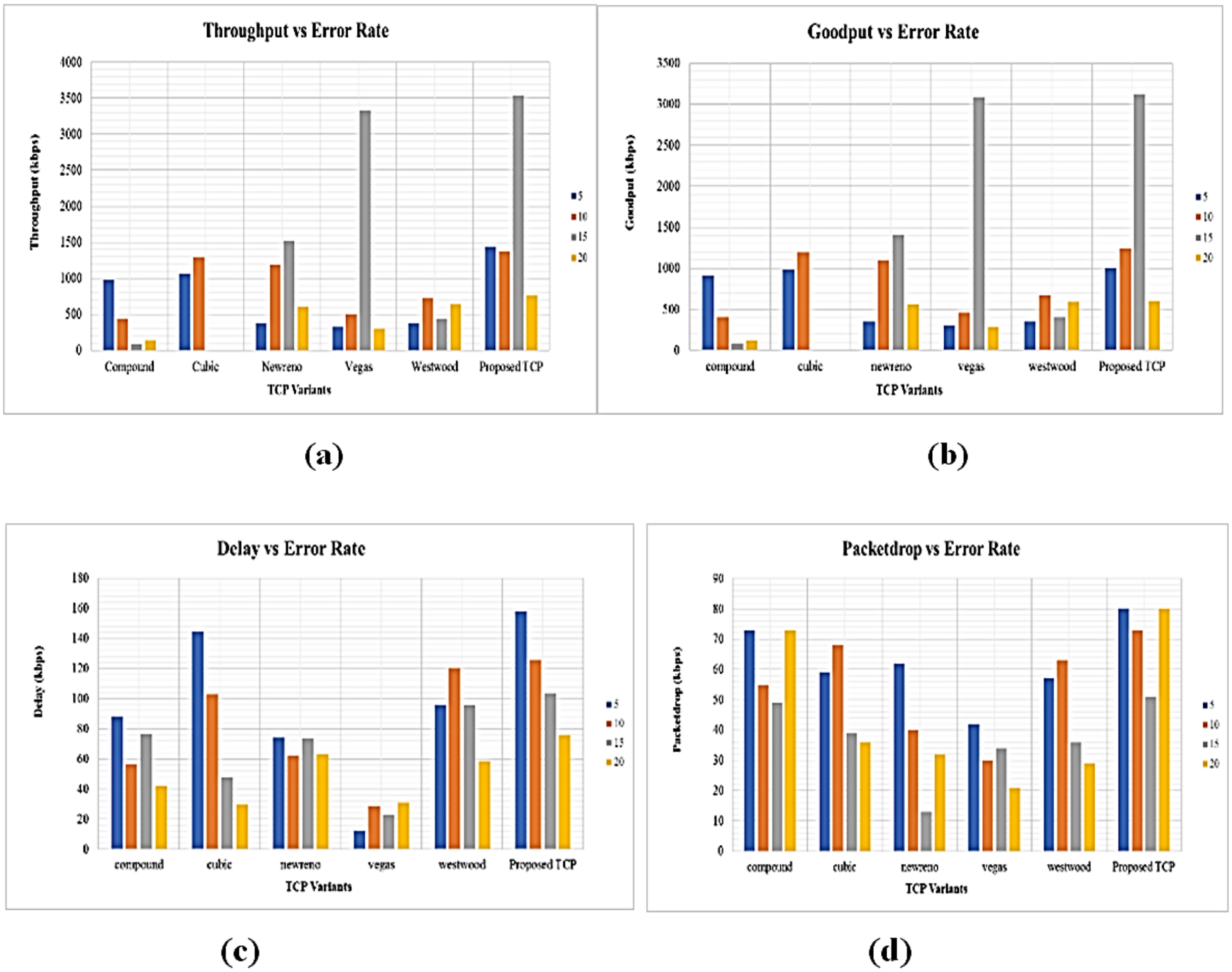


Figure 4 Error Rate vs (a) Throughput (b) Goodput (c) Delay (d) Packet drop

4.6 Number of Nodes

ISSN: 2349 - 4891
International
Journal of Recent Research and Applied Studies
(Multidisciplinary Open Access Refereed e-Journal)

Dependinguponthenumberofnodesusedinanetworkthevaluesofgoodput,throughput, delay and packet loss maybe change. When the Number of nodes increases throughput decreases and when number of nodes decreases the throughput increases. In case of goodput the number of nodes increases goodput decreases but when nodes are decreases the goodput also increases. Similarly, when we see the packet drop the nodes increases the packet drop also increases at the same time when nodes decrease packet drop also decreases. While number of nodes increases the delay decreases and number of nodes decreases delay isincreases.

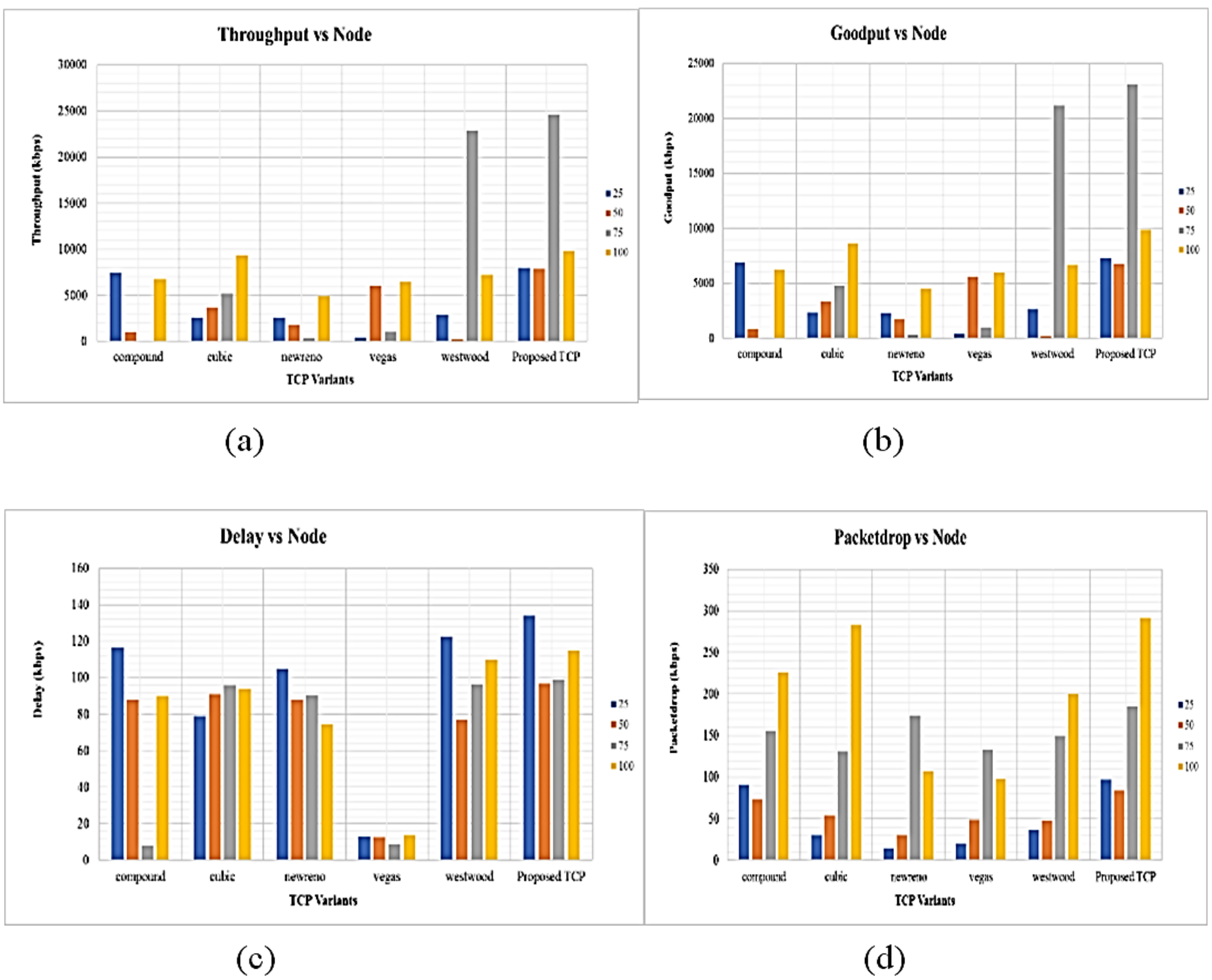


Figure 5 Number of Nodes vs (a) Throughput (b) Goodput (c) Delay (d) Packet drop

ISSN: 2349 - 4891
International
Journal of Recent Research and Applied Studies
(Multidisciplinary Open Access Refereed e-Journal)

4.7 Speed

Depending upon the speed of the network the values of goodput, throughput, delay and packet loss maybe change. When the speed of the network increases throughput decreases and whenspeeddecreasesthethroughputincreases.Incaseofgoodputthewhenspeedofthenetwork increases goodput decreases but when the speed decreases the goodput also increases. Similarly, when we see the packet drop the network speed increases the packet drop also increases at the same time when network speed decrease packet drop also decreases. While network speed increases the delay decreases and likely when speed of network decreases delay isincreases.

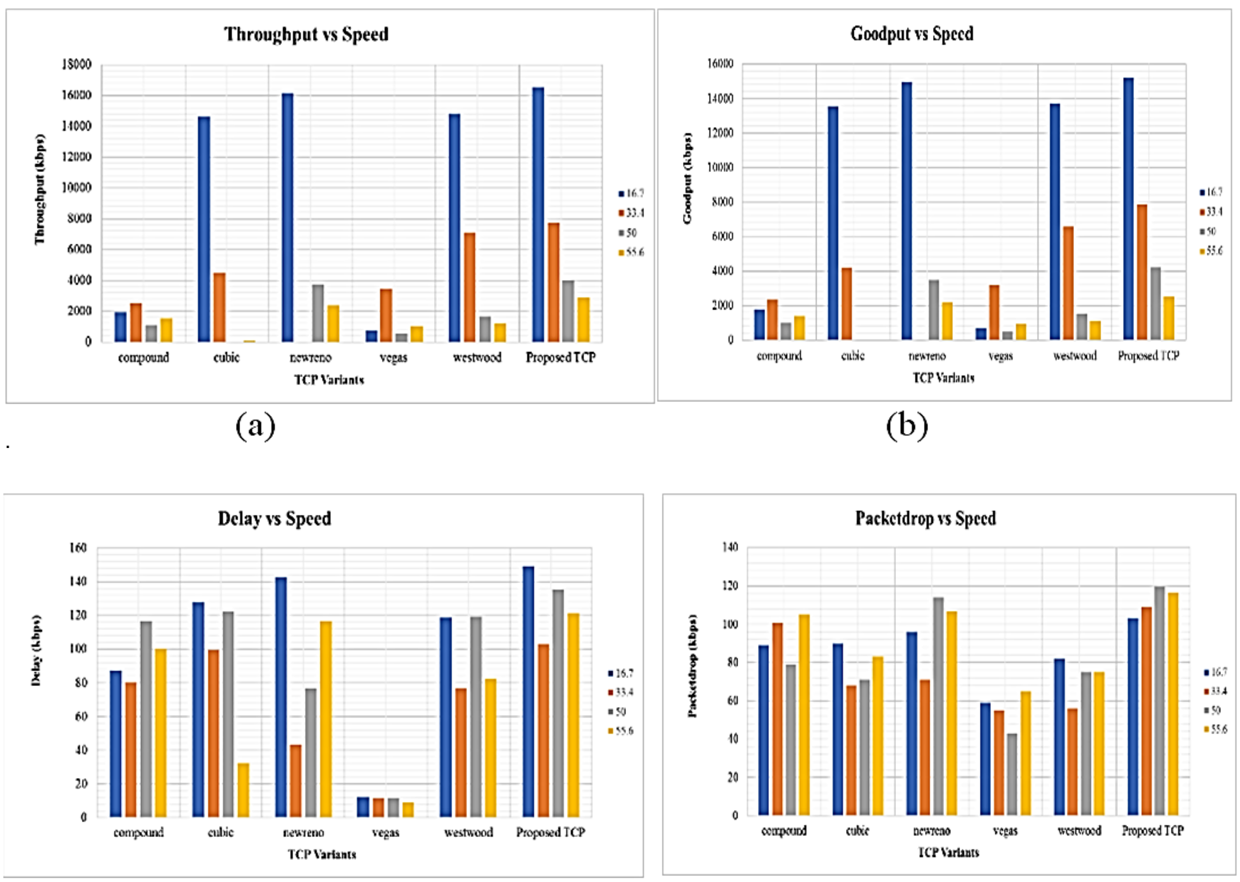


Figure 6 Speeds vs. (a) Throughput (b) Goodput (c) Delay (d) Packet drop

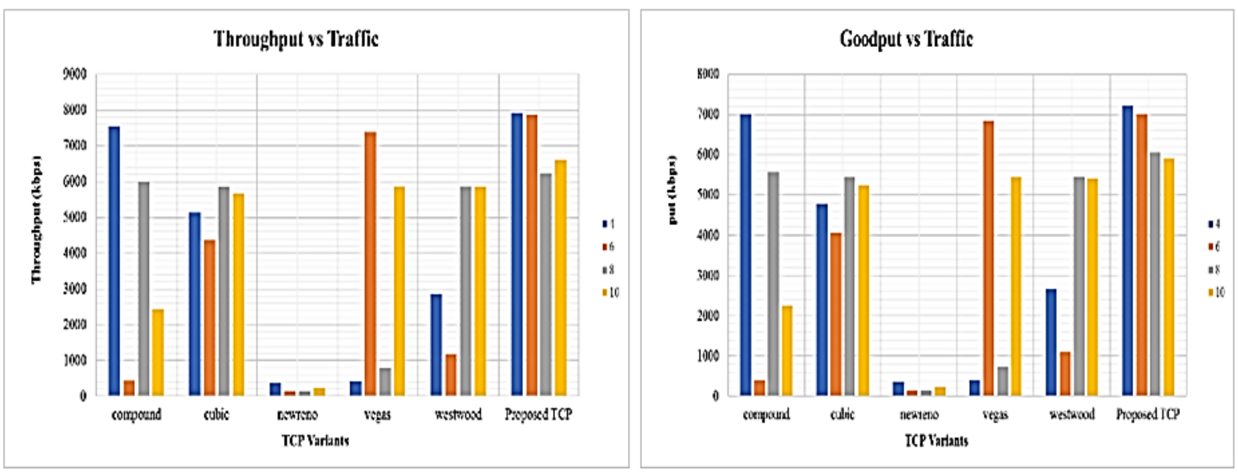
ISSN: 2349 - 4891

International
Journal of Recent Research and Applied Studies
(Multidisciplinary Open Access Refereed e-Journal)

ISSN: 2349 - 4891
International
Journal of Recent Research and Applied Studies
(Multidisciplinary Open Access Refereed e-Journal)

4.8 Traffic

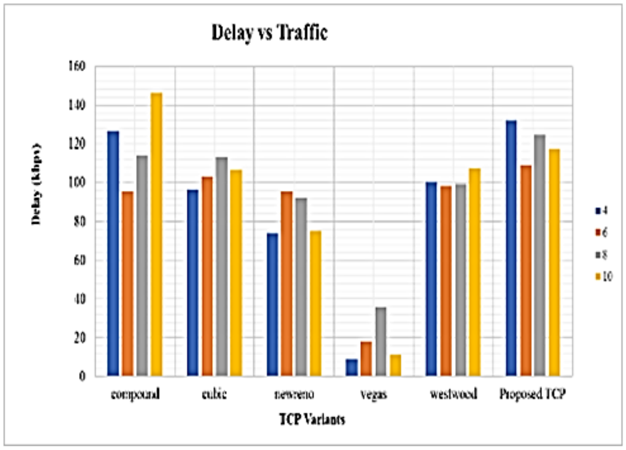
Depending upon the number of traffic nodes present in a network the values of goodput, throughput, and delay and packet loss maybe change. When the Number of traffic nodes increases throughputdecreasesandwhennumberoftrafficnodesdecreasesthethroughputincreases.Incase of goodput the number of traffic nodes increases goodput decreases but when nodes aredecreases the goodput also increases. Similarly, when we see the packet drop the traffic nodes increases the packetdropalsoincreasesatthesametimewhennodesdecreasepacketdropalsodecreases.While number of traffic nodes increases the delay increases and number of traffic nodes decreases delay isdecreases.

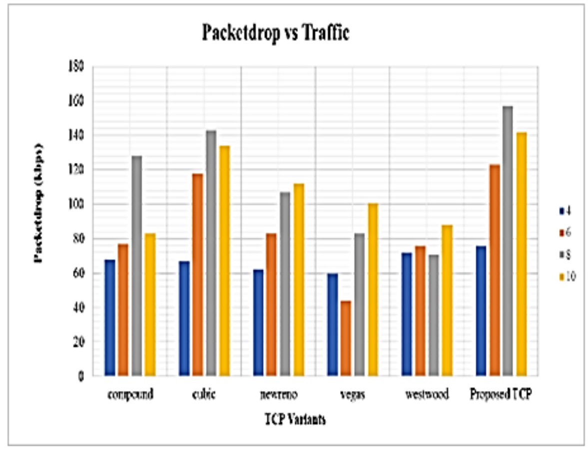


(a) (b)

ISSN: 2349 - 4891

International
Journal of Recent Research and Applied Studies
(Multidisciplinary Open Access Refereed e-Journal)





(c) (d)

Figure 7 Delay vs. (a) Throughput (b) Goodput (c) Delay (d) Packet drop

5. Conclusions

Vehicular ad-hoc network (VANET) is one of the most challenging domains in current scenario to provide an intelligent transport system (ITS). One of the important application areas of mobile ad-hoc network (MANET) is VANET. The inter vehicular simulation outcome highlightstheproposedIVANTCPachievesimprovementinthroughputandsubstantialreduction in loss rate, Packet drops, and end-to-end packet latency reduction against Westwood and NCE. The standard approach control system design is to develop a linear model of the process for an operationconditionandtoDesignacontrollerhavingconstantparameter.Therefore,wearetrying to find a more precise TCP flow dynamic model than tonoise.

References

1. T. Limbasiya and D. Das, "Lightweight Secure Message Broadcasting Protocol for Vehicle-to-Vehicle Communication," in IEEE Systems Journal, vol. 14, no. 1, pp. 520- 529, March 2020.

2. S.A.Ahmad,A.Hajisami,H.Krishnan,F.Ahmed-ZaidandE.Moradi-Pari,"V2VSystem Congestion Control Validation and Performance," in IEEE Transactions on Vehicular Technology,vol.68,no.3,pp.2102-2110,March2019.

3. Liu, X., &Jaekel, A. (2019). Congestion control in V2V safety communication: Problem, analysis,Approaches. Electronics, 8(5), 540.

4. H.Jiang,Z.Zhang,J.DangandL.Wu,"ANovel3-DMassiveMIMOChannelModelfor Vehicle-to-Vehicle Communication Environments," in IEEE Transactions on Communications, vol. 66, no. 1, pp. 79-90, Jan. 2018.

5. 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.

6. 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.

7. Q. Zhang, H. Zheng, J. Lan, J. An and H. Peng, "An Autonomous Information Collection and Dissemination Model for Large-Scale Urban Road Networks," in IEEE Transactions on Intelligent Transportation Systems, vol. 17, no. 4, pp. 1085-1095, April 2016.

8. L. Chen and P. Chou, "BIG-CCA: Beacon-Less, Infrastructure-Less, and GPS-Less Cooperative Collision Avoidance Based on Vehicular Sensor Networks," in IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 46, no. 11, pp. 1518-1528, Nov. 2016.

9. Huang, P.-K., Lin, X., & Wang, C.-C. (2013). A low-complexity congestion control and scheduling Algorithm for multihop wireless networks with order-optimal per-flow delay. IEEE/ACM Transactions on Networking.

10. Henderson, et al. (2012). The NewReno Modification to TCP's Fast Recovery Algorithm.

RFC 6582, IETF Network WorkingGroup.

11. Taherkhani, N., & Pierre, S. (2012, October). Congestion control in vehicular ad hoc networksusingMeta-heuristictechniques.InProceedingsofthesecondACMinternational symposiumonDesignandanalysisofintelligentvehicularnetworksandapplications(pp. 47–54).

12. Wu, G., Xia, F., Yao, L., Zhang, Y., & Zhu, Y. (2011). A hop-by-hopcross-layer congestion control Scheme for wireless sensor networks. arXiv preprint arXiv:1201.0207.

13. G. Karagiannis et al., "Vehicular Networking: A Survey and Tutorial on Requirements, Architectures,Challenges,StandardsandSolutions,"inIEEECommunicationsSurveys& Tutorials, vol. 13, no. 4, pp. 584-616, Fourth Quarter 2011.

14. Networking: A survey and tutorial on requirements, architectures, challenges, standards and solutions. IEEE Communications Surveys &Tutorials.

15. Bouassida, M.S., Shawky, M. A Cooperative Congestion Control Approach within VANETs: Formal Verification and Performance Evaluation. J Wireless Com Network 2010, 712525 (2010).

16. Zeadally, S., Hunt, R., Chen, YS. *et al.* Vehicular ad hoc networks (VANETS): status, results,andchallenges.*TelecommunSyst***50,** 217–241(2012).

17. "IEEE Standard for Wireless Access in Vehicular Environments (WAVE) - Networking Services," in IEEE Std 1609.3-2010 (Revision of IEEE Std 1609.3-2007) , vol., no., pp.1-144, 30 Dec. 2010, doi:10.1109/IEEESTD.2010.5680697.

18. Y. Yi and S. Shakkottai, "Hop-by-Hop Congestion Control Over a Wireless Multi-Hop Network," in IEEE/ACM Transactions on Networking, vol. 15, no. 1, pp. 133-144, Feb. 2007.

19. Yun, JH. Cross-Layer Explicit Link Status Notification to Improve TCP Performance in WirelessNetworks.*J Wireless Com Network***2009,** 617818 (2009).

20. 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).

21. DembaA and D. P. F. Möller, "Vehicle-to-Vehicle Communication Technology," 2018 IEEEInternationalConferenceonElectro/InformationTechnology(EIT),2018,pp.0459- 0464.