International Journal of Engineering Applied Sciences and Technology, 2018
Vol. 3, Issue 7, ISSN No. 2455-2143, Pages 8-17
Published Online December 2018 in IJEAST (http://www.ijeast.com)

A STUDY AND ANALYSIS FOR CONGESTION CONTROL MODEL USING BBR IN WIRELESS NETWORKS SCENARIO

Girish Paliwal
Dept of AIIT
Amity University Jaipur Rajasthan

Swapnesh Taterh
Dept of AIIT
Amity University Jaipur Rajasthan

Narendra Singh Yadav
Dept of CS/IT
Manipal University Jaipur Rajasthan

Abstract: **Recent years, wireless network grow very rapidly mostly inside the homes. The number of home automation products and devices is connected with the wireless technique easily without wiring. These internet user expectations are much higher when they want to access the internet with high speed and minimum cost. It can be achieving by increasing the bandwidth or improving the congestion control algorithms. The many types of research based on the improved TCP congestion control algorithms. They have been building and improve TCP congestion control algorithm and proposed over different kinds of high-speed networks, get high throughput. Researcher searches the techniques to reduce the wireless user's data accessing delay from minutes to seconds. In this paper, we proposed a congestion control model based on the TCP BBR and TCP Reno and analysis it with respect to another congestion control algorithms. The congestion control algorithms are analyzed and study for the wireless network scenario and found it has impacted over another TCP congestion control algorithms.**

Keywords: **RTT-Round Trip Time, Congestion Control Algorithms, TCP-Transmission Control Protocol, New Reno, Westwood, Cubic, BBR-Bottleneck Bandwidth Round Trip Propagation, WSN-Wireless Sensor Network**

I. INTRODUCTION

Wireless sensor network (WSN) is broadly used at the different public spots. The wireless LAN provides the facility to self-organization of the connected number of sensor nodes within the monitoring range. They provide the free of internet accessing for the limited time period and with limited bandwidth or limited speed at some public service areas (Shah, Nazir, & Khan, 2016).The various wireless devices install for providing the facility. These devices are call access point and connected to the power supply, whereas the number of mobile devices having the limited battery power, memory, and processing capabilities. It is not practical to keep all the devices active for the monitoring purpose, due large number of active devices or nodes can consume higher energy as well as create bottleneck effect in the network(Singh & Sharma, 2015). The information related to the connected devices is collected from the install sink or access point and aggregated data can be analysis. The wireless Access points are using TCP (Transmission control protocol) for the internet within the home and building or public locations. The wireless connected devices are fixed or mobile. It is observed that the rural area showing the constant growth of the wireless sensor network whereas urban areas showing the growth as well as the high mobility of the wireless sensor devices. The working mechanism of MANET is based on the mobile nodes. All the nodes are assumed as trusted nodes, but some malicious nodes get affected in MANET working approach(Paliwal, Mudgal, & Taterh, 2015). It increases the density of wireless devices and the poor performance of TPC over the wireless networks attract researchers, innovates new solutions and some of the solutions have been presented to overcome the problems. Initially, researchers focused on designing and developing efficient routing schemes, now they realized that the traffic control mechanism plays an important role in future because the traffic at sink node become greater than its capacity(Ravindranath, Singh, Prasad, & Rao, 2016). In Wireless packet network this is a common problem. The congestion can be explained as too many packets try to access the same access point resource memory result is the number of packets is dropped. In other words, we can say that the send data into the network at the higher rate than the allocated to the network. Congestion occurs when a number of the node sending data packets to the same destination. To overcome

this problem the mechanism used is known as congestion control mechanism. The main objective of different congestion control algorithms to maximum utilization of the resources of WSN to achieve the maximum throughput.
We organized the paper as the following sections under the different headings the first section represented by Types of congestion control approaches. The next section emphasizes on the related work the various TCP variants. Another section starts with the heading Bottleneck Bandwidth Round Trip Propagation congestion control algorithm. Next section start with the heading Analysis of the TCP variants with TCP BBR further section is the proposed congestion control model and final section gives the conclusion the work.
The rest of this paper is organized as follows. In section II, authors give an overview of the background and related working approaches for wireless network TCP. In III, authors give a description of various TCP congestion control. Section IV Discussion and analyzes the TCP-BBR algorithm V compare TCP BBR with others, VI in this section proposed model analysis and results. Finally, The authors summarize the findings in section VII.

II. TYPES OF CONGESTION CONTROL APPROACHES

There are two ways to make a congestion free connection.

A. Congestion Prevention

This is the first way in which prevents the network from occurring congestion. Prevention is better than cure in this process the main focused on keep network less busy than its capacity. It is used as the preliminary step of the congestion control. In this technique, the congestion detected before it happens and aware the source nodes to slow down their transmission speed. Due to this the network not working according to their actual strength and its throughput less than its actual.

B. Congestion Control

Another way network works with its full capacity and waits when congestion occurs than control it. In this process, the congestion was detected and it is already slow down the network speed. To handle the congestion there are many congestion prevention and control algorithms have been proposed researcher. The congestion control algorithms mostly proposed the methods of active management of the buffer memory of maintaining the queue at the sink node or congested node. They set the congestion windows as a trigger to activate the congestion prevention algorithm and congestion control algorithms automatically. Existing congestion control algorithms have some limitation on the basis of optimization of the traffic load on the congested path node and link, traffic distribution to the alternative path is not based on traffic estimation and the priority of a packet processing is not actually based on the delay(Wala, Gupta, & Kumar, 2016).

III. THE VARIOUS TCP CONGESTION CONTROL RELATED WORK

The IEEE802.11 standards used for wireless networks are having some advantages over the wired network for a small to medium network group. Wireless Access Point (APs) is the central device through which internet can be accessed. After a decay tested and used of wireless technology in the houses offering the different services as home automation, security of home, online smart TV entertainment, control of energy wasting etc. The numbers of home automation equipment are connected with an Access Point and all devices sent data to a APs this is known IOT (Internet of Things). The most of the data are automated from the different sensors like camera, energy controller, and etc. not human generated data. So continue data transmission by different high resolution devices the access point router congested and its service affected due to the congestion(Baldo et al., 2010). The congestion control algorithms are required to quickly balance the network performance into the stable state. There is the number of congestion control algorithms are implemented at the transport layer and these are integrated with TCP. More than the three decades many extensions of TCP congestion control algorithms have been proposed and the resulting congestion control improvement.
TCP is mostly used and it is well developed also wildly available transmission control protocol. It is required to maintain the stability for the internet services to all kind of applications due to improving congestion control algorithms. TCP is much fast and effective but congestion degrades its performance. Congestion control algorithms use the AIMD (Additive Increase Multiplicative Decrease) mechanism to resolve the congestion to decrease the window size that is hurting the actual data rate of the channel(Floyd, Handley, & Padhye, 2000).
TCP congestion control was created in the 1980s interpreting packet loss as "congestion"(Cardwell, Cheng, Gunn, Yeganeh, & Jacobson, 2017). Today TCP's approaches to updating its congestion window under different congestion condition are:

- Loss-based TCP congestion control
- Delay-based congestion control
- Mixed loss-delay based TCP congestion control

TCP control the congestion by the changing of the sender window and buffer size, it is decided on the basis of the

International Journal of Engineering Applied Sciences and Technology, 2018
Vol. 3, Issue 7, ISSN No. 2455-2143, Pages 8-17
Published Online December 2018 in IJEAST (http://www.ijeast.com)

available buffer space in the receiver window and the network bandwidth. Figure 1 show, the simple congestion window growth in TCP of a slow start and congestion avoidance to calculate the ssthresh(Levasseur, Claypool, & Kinicki, 2014; Paul & Sarkar, 2013). TCP's general policy for congestion control having the three phases:

- Slow start
- Congestion avoidance
- Congestion detection

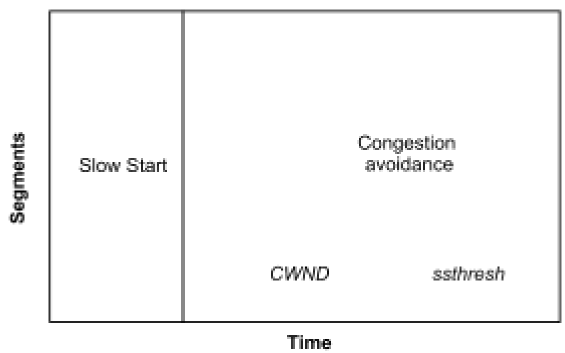


Figure 1 Simple congestion window growth representation in TCP

Almost, the entire TCP congestion controls algorithm work on the basis of these three phases. First sender slow start of data transmission and increases the transmission rate gradually, but transmission rate reach to the maximum threshold that why congestion occurs and detected. When congestion detected then reduced the data transmission rate to avoid congestion and get the maximum throughput(Almobaideen & Al-maitah, 2014; Paliwal & Taterh, 2018).

A. TCP Reno

TCP Reno implemented the mechanism of AIMD for the congestion window (cwnd). It incremented at every round trip time received an acknowledgment (ACK)(Nguyen, Gangadhar, Rahman, & Sterbenz, 2016). TCP Reno congestion control window calculated as:

Increase: $cwnd = cwnd + \frac{1}{cwnd}$

And when congestion occurs then the congestion window size decrease multiplicatively and TCP Reno calculates Congestion windows decrease as:

Decrease: $cwnd = cwnd - \frac{cwnd}{2}$

During the transmission the channel bandwidth not changed but congestion window decrease rapidly when the packet loss, the TCP Reno periodically changes the congestion window (cwnd) increase and decrease(Levasseur, et al., 2014). Figure, 2 show the TCP Reno congestion control algorithm flow its first start to sending data packet when the wireless channel available and received the acknowledgment (ACK). ACK received counter to start counting and calculate slow start thresh(ssthresh). DUP_ACK accumulated if they reached to the three than congestion window cwnd decreased suddenly half of the cwnd

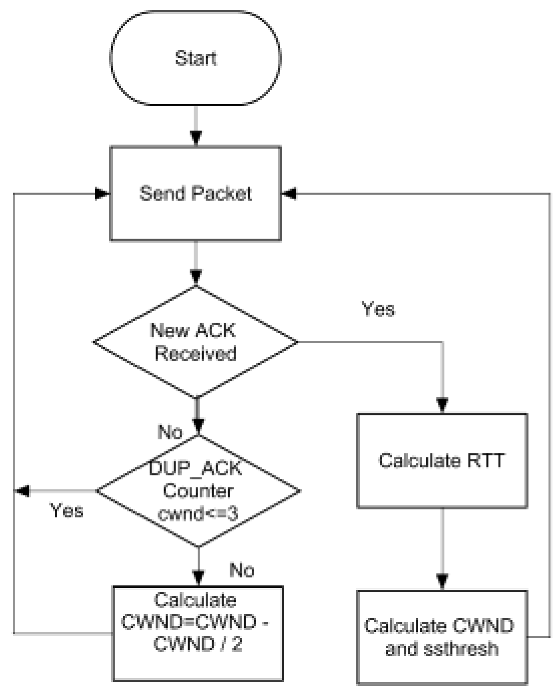


Figure 2 TCP Reno Congestion control algorithm Flow Diagram

In the above Figure 2 show the TCP Reno congestion control flow that might have little bit change of their actual implementation by the different researchers to produced better result.

B. TCP Cubic

TCP Cubic enhanced the congestion window control mechanism by cubic increased. It simplifies the TCP friendliness and round trip time function. CUBIC rapidly grow

International Journal of Engineering Applied Sciences and Technology, 2018
Vol. 3, Issue 7, ISSN No. 2455-2143, Pages 8-17
Published Online December 2018 in IJEAST (http://www.ijeast.com)

TCP cwnd for short RTT or low speed of the network. This growth function is based on the cubic function for the elapsed time to the last event. It provides a good scalability and stability. TCP CUBIC maintains the cwnd growth independent to the RTT(Poojary & Sharma, 2011). The congestion period is determined by the packet loss rate as well as RTT. When the loss rate is high and RTT short then CUBIC can work in the form of congestion control by the following function use to control the cwnd size(Gwak, Kim, & Kim, 2013):

$$cwnd_{cubic} = C(t-K)^3 + cwnd_{w\_\max}$$

Where C is the scaling factor, t is elapsed time, w_ max is the windows size before the last window reduction and K is the following equation

$$K = ((w\_\max - cwnd) \times C)^{\frac{1}{3}}$$

The TCP CUBIC has effectively utilized the capacity of the connected network. It provides the congestion window growth algorithm faster until not get the mid-point of the congestion window. After the mid-point of the congestion window, the TCP CUBIC increased the congestion window slowly until not get the next loss point(Levasseur, et al., 2014). TCP CUBIC maintains the last packet of lost point and it is very helpful to protect the congestion in future. The TCP CUBIC congestion control windows show in figure 3, in which the complete TCP CUBIC congestion control algorithm is divided into two major parts first one is concave and another is convex. Before the mid-point of congestion window, it is known concave and after the mid-point of congestion window it is more complex to increased the congestion window that why it is known as convex.

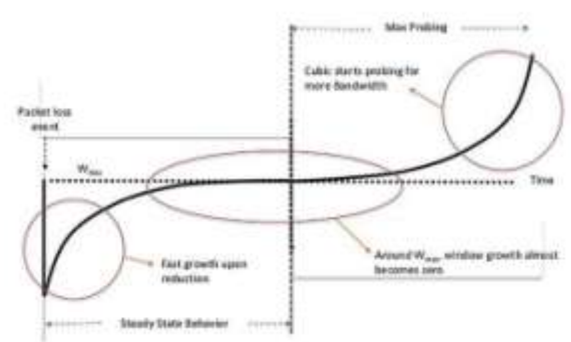


Figure 3 TCP CUBIC growth function work over cwnd

The TCP Cubic handled the DUP_ACK received. If three DUP_ACK received than the TCP is not in the fast recovery and set the lower congestion window and ssthresh.

C. TCP Westwood

TCP Westwood is a congestion control mechanism that introduced a novel approach for wired or wireless network scenario that controls congestion by the end to end bandwidth estimation mechanism. It starts slow threshold value furthermore introduced faster state in which TCP increased the transmission rate. TCP Westwood senders continuously calculated the connection bandwidth estimate (BE) that is important for network bottleneck bandwidth estimation(Mishra, Vankar, & Tahiliani, 2016). The bandwidth estimation based on the collection of data ACK received to the round trip time(Al-Hasanat, Seman, & Saadan, 2014).

Another important of the TCP Westwood is RTT round trip time. The RTT is estimated for each connection it is dependent the path length and the queue delay process and the congestion occur.The Setting of the congestion window and ssthresh after a packet loss indication is based on the estimation of bandwidth. At the first time congestion window dynamically set during the slow start phase and congestion avoidance not changed. It is increasing exponentially and linearly as increased in the TCP Reno and New Reno. In TCP Westwood a packet loss indicated by the three duplicate acknowledgments (DUP_ACK) and the expiration of time to live (RTT)(Bassil, 2012). In the Table 1 show, the congestion window control procedure:

Step 1: Starting phase of transmission set the congestion window, ssthresh and Duplicate Acknowledgment
    cwnd=1
    ssthresh=infinity
    DUP_ACK=0
Step 2: When new ACK arrive than calculate the new congestion window as
    If (cwnd < ssthresh) Than
        cwnd=cwnd+1
    Otherwise
        cwnd=ssthresh
Step 3: Accumulate the DUP_ACK when ACK arrived as

International Journal of Engineering Applied Sciences and Technology, 2018
Vol. 3, Issue 7, ISSN No. 2455-2143, Pages 8-17
Published Online December 2018 in IJEAST (http://www.ijeast.com)

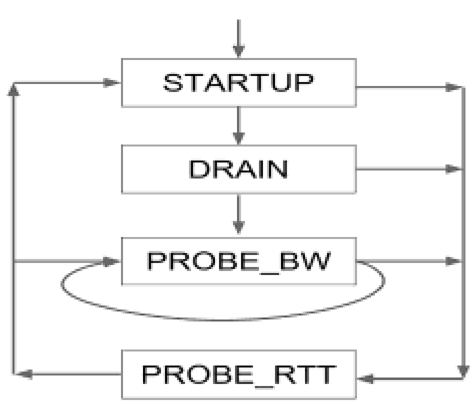
```
    If (DUP_ACK is True) Than
        DUP_ACK=DUP_ACK+1
Step 4: Change the congestion window according to decision
        parameter
    If (DUP_ACK ==3) Than
        cwnd=cwnd / 2
    Otherwise
        cwnd=1
        csthresh=(BE x RTT min ) / Seg_Size
Step 5: Exit from the algorithm when data complete sent
```

Table 1 TCP Westwood procedure of controlling congestion

It may be possible that the above table 1 show, TCP Westwood algorithm of controlling congestion have different code according to their implementation by different simulations. In this algorithm the most important to calculate the estimated bandwidth of the channel and calculating the ssthresh on the basis of minimum round trip time(Assasa & Widmer, 2016; Gangadhar, Nguyen, Umapathi, & Sterbenz, 2013).

IV. BOTTLENECK BANDWIDTH ROUND TRIP PROPAGATION CONGESTION CONTROL ALGORITHM

TCP BBR is recently developed in 2016 by the Google group. They are rethinking about the congestion control existing methods play a big role rather than using the existing such as loss or buffer occupancy. BBR start from Kleinrock formal model of congestion and its associated optimal operating point. They find out the loss based congestion control is not good for the network because they control congestion when the network triggered the packet loss.The bottlenecks buffers are large, loss base congestion control mechanism full the buffer that causes the buffer boat. If the bottleneck smaller than the loss based congestion control algorithm are misinterpreted that the losses as a signal of congestion due to this we get low throughput. The TCP BBR finds out an alternative solution against the loss base congestion control algorithm(Assasa & Widmer, 2016; Cardwell, et al., 2017).



In previous, we study about the congestion and bottleneck here again specify some properties of the congestion and bottleneck and it is important because of it determined the connection maximum data delivery rate. It is also important because its persistent queue formation, queue shrink only when transmission rate exceeds then the arrival rate. Figure 4 show, the state transition of the TCP BBR.

Figure 4 TCP BBR state transition diagram

The Google group proposed two physical constraints to bind the transport performance one is the RTprop (Round Trip Propagation) and another one is BtlBW (Bottleneck Bandwidth). These two constraints are more use full to find out the maximum transmission speed to avoiding the congestion. The TCP bottleneck bandwidth round trip propagation (TCP BBR) algorithm that shows the maximum delivery ratio calculated at each round trip time when congestion buffer windows increase the RTT than again start ssthresh and periodically check the available bandwidth of the channel.

If the network path is a physical connecting pipe then the Round Trip propagation would be its length and the Bottleneck Bandwidth its minimum diameter. Data in flight is the term that means the data transmitted but its acknowledgment not received. Data in flight is the term that used at the starting of congestion control algorithm at the starting time data sent but its ack not received yet.

The three kinds of filters applied to controlling the congestion one are RTP (Round Trip Propagation), BtlBW (Bottleneck Bandwidth) and the third constraint is the bottleneck buffer area. The effects of these constraints show on the application limit, bandwidth limit and buffer limit. Most of the loss based congestion control algorithms work on a limited region of the bandwidth this resign always the left hand side of the maximum bandwidth. The high cost is applied if the bottleneck buffer full in term of high delay and frequently packet losses. It is very clear that the left point of the maximum bottleneck Bandwidth is a better operating point then its maximum bandwidth.

Google group of research spends too much time each day to examining the different TCP headers those captured from all over the world. They first identify the characteristics of the path, with respect to RTP and BtlBW(Cardwell, et al., 2017).

International Journal of Engineering Applied Sciences and Technology, 2018
Vol. 3, Issue 7, ISSN No. 2455-2143, Pages 8-17
Published Online December 2018 in IJEAST (http://www.ijeast.com)

They find out the Round Trip propagation change when the path length change, Bandwidth decrease or the queuing delay increased. After three years of hard work of Google group, they proposed the better congestion control mechanism based on the measuring two parameters. Bottleneck Bandwidth and the Round Trip Propagation time or BBR.

BBR characteristics any connection runs with the highest throughput and lowest delay. When the bottleneck packet arrival speed is balanced to the Bottleneck Bandwidth and total data is equal to the Bottleneck bandwidth times of round trip propagation time. It shows as the following equation:

Bandwidth Delay Product: $BDP = BtlBW \times RTP$

It gives the guarantees that the Bottleneck maintains 100% delivery ratio. It is also having the solution to prevent bottleneck starvation with transmitting the enough data but not overfill the channel. The estimation of the BtlBW and RTP must be continuously calculated over the life of the connection because it is very during the connection. The TCP track the Round Trip Time (RTT) and its use for the loss detection at any time t. In the BBR proposed system

Round Trip Time: $RTT_t = RT_{prop_t} + n_t$

Where nt >=0 is known as “noise” introduced by queues along the path received delay ack strategy, ack aggregation etc. RTprop is calculated as propagation delay of the path it changes when path change

$$RT_{prop} = RT_{prop} + \min(n_t) = \min(RT_{prop})$$

Delivery Rate = delta delivery / *delta time*

The delivery rate must be <= the bottleneck rate. The data arrival amount knew so all the uncertainty in the dt and it must be greater than equal to the true arrival time interval.

BtlBW=Max(Delivery Rate t)

The Bottleneck bandwidth round trip propagation time that recorded with the total data delivered so that each ack arrival yield both RTT and delivery rate measurement that filters convert to RTP and BtlBW estimations whereas TCP must record the departure time of the each data packet to compute the RTT(Cheng & Cardwell, 2016). The BBR algorithm is having the two parts to matching the packet flow to the delivery path. In the first phase of the algorithm check the ack when received. Each acknowledgment provides the new RTT and the delivery rate measurement that updated the RTP and BtlBW estimated. In the second phase, the data packet sent with the specified data rate. In this match, the data packet arrival rate to bottleneck link departure rate, BBR pace every data packet. BBR must match the bottleneck rate which means the pacing of the packet. BBR minimizes the delay by spending most of its time on BDP in flight, the pace at BtlBW estimated. The steady state behavior show by the BBR because it is periodically spending time pacing gain greater than 1which increase the data sending rate but BtlBW is not changed that means make a queue at bottleneck increase the RTT which keeps delivery rate constant. All this achieve by compensating the pacing gain less than 1 for next RTP. If BtlBW increase then delivery rate increased(Cardwell, et al., 2017).

BBR synchronizes flow around the desirable events with an empty bottleneck queue in contrast loss based congestion control synchronizes around the undesirable event due to periodic queue growth, overflow of the queue, amplifying delay and packet loss.

BBR is supporting for the wireless communication system or cellular system adaptive bandwidth. In cellular system adapt per subscription bandwidth based partly on demand estimate that uses the queue of the packet o a destination for subscribers. It is a very big advantage of the BBR that it is tuned to create a very small queue, resulting connection getting stuck at the low rate. The currently 1.25xBtlBW peak gain no degradation is apparent(Cheng & Cardwell, 2016).

V. ANALYSIS OF DIFFERENT TCP VARIANTS WITH TCP BBR

The numbers of the TCP variants implemented already here our focused only that the analysis of techniques of the TCP congestion controls algorithms. Here we include only three already implemented TCP congestion control algorithms TCP RENO, TCP CUBIC and TCP WESTWOOD study in the light of newly released Google congestion control algorithm TCP BBR. The given Table 2 show, the comparison of the congestion control algorithms.

Protocol	Category	Design Goal	Analysis
Reno	Pkt Loss Based	Recovering packet losses in a one sending window	Standard AIMD factors(AI)=1 factors(AI)=1/2 for fast recovery
Cubic	Pkt Loss Based	Simplify Binary cubic window control	Use binary search method and congestion windows growth is a cubic function
Westwood	Pkt Loss Based	Improve throughput in the presence of packet losses	Calculate the estimated Bandwidth and control the congestion window for every RTT
BBR	Congestion Based, Round Trip Propagation	Congestion free connection	Calculate Delivery ratio of every Round Trip Propagation to control data transmission rate

Table 2 Analysis of the TCP protocols

In the given table 2, it is shown that TCP Reno, TCP Cubic and the TCP Westwood detect the congestion when packet delivery failed and then they control the congestion to reducing the packet forwarding window. These protocols achieve the maximum throughput according to their different way of controlling the congestion window. The TCP BBR algorithm is trying to get rid of the congestion to estimating the maximum throughput of the channel. In this regard, the BBR showed that it provided the congestion free connection but when congestion RTT increase it slow down the pacing of the packet forwarding it does not show what happened when in-flight data packet received Dup-ACK. To handle the situation we proposed the following congestion control model(Zhou et al., 2015).

VI. PROPOSED CONGESTION CONTROL MODEL

Here we have proposed a model for congestion control that is not yet to be implemented till this research paper written but this model is based on the already working congestion control models. The TCP Reno is working on the basis of the packet loss and the TCP BBR working on the round trip time. In our proposed model first of all congestion control algorithm determined the Estimated Bandwidth of the connection, second increased the congestion window randomly, find the minimum round trip time and if three duplicate acknowledgment received (DUP_ACK) than congestion window gain set to again 1 to control the congestion. This model is designed to achieve the high performance of networking. This is working on the best operating point of a network. Today loss based congestion control algorithm is not a good signal to measuring the best operating point of the network. These congestion control algorithms are not so much popular due to they are generating the buffer boat problem, without packet loss we are unable to find out the best operating point simultaneously using the RTT. It means the best operating point is that sending data packet rate equals to the maximum bandwidth available for the connection. This can maximize the throughput while minimizing the delay because it keeping the queue empty. The proposed method fast loss recovering high bandwidth and maintain the low queue. Figure 5 show, the proposed model flow diagram(Avallone & Di Stasi, 2016).

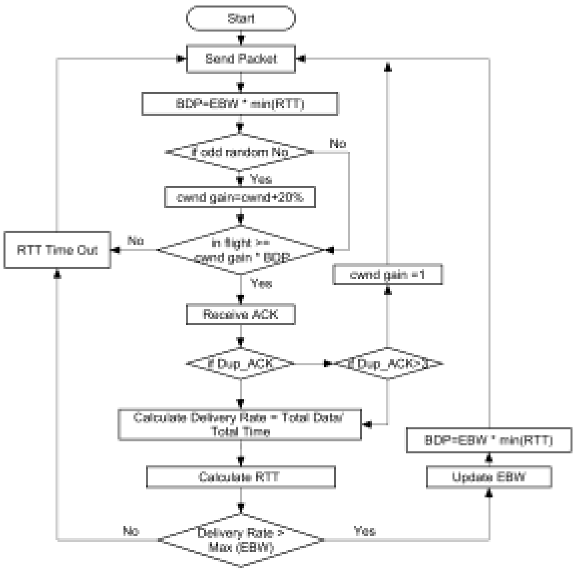


Figure 5 proposed model of congestion control algorithm

In this proposed congestion control model we want to fetch the maximum bandwidth of the currently available connection. When every time data packet transmitted than the congestion window increased by 20% at the random time period it is not applied periodically. The random number generator generates random number if the random number is odd than increase the congestion window by 20% until not get the maximum operating bandwidth and the packet sent continue. The number of packets is sent but not received any acknowledgment than this kind of send data is called in-flight. Wait for the ACK of in-flight data packets. After received the ACK it is checked that the ACK is not duplicate if ACK is duplicated then, first of all, check the duplicate packet received due to congestion or network error. If duplicate ACK received but it is a wireless network error than the process continue going on otherwise if three DUP_ACK received than the congestion window set to the one and sent data packet again. The cwnd set initial but maximum estimated bandwidth not changed because it is found after a long estimation process over the current connection.

If received ACK not duplicated than calculate the maximum estimated bandwidth of connection and minimum round trip time. Set the calculated data and again calculate the BDP to achieve the maximum throughput of the connection. The proposed model implementation is going on, to find out the estimated bandwidth and round trip time simultaneously. This model also focused the zero or minimum length queue. This model is working better than the previous implementation. In MANET the most important challenges to maintaining the link between mobile nodes in MANET and data transmission speed (Paliwal & Taterh, 2018).

VII. CONCLUSION

After the study of the different type of TCP congestion control algorithms, we find out the connection bandwidth suffers when slowing down the speed of transmission or overflow the queue buffer and congestion occur. The packets are dropped and controlling the congestion by the slow down the transmission speed by resetting the congestion window. All of the congestion control algorithms based either on the packet loss or RTT delay but in any case the congestion occurs then it is the trigger point for controlling the transmission of the packets but Google group take initiative and do much efforts to control the transmission speed according to the channel round trip propagation time that why TCP BBR much more sound then others. TCP BBR calculates the delivery ratio on every RTT and increases or decreases the transmission speed of data packet to achieve the maximum throughput of the channel. In our proposed model it is also handled the data packets when the in-flight packet received duplicate ACK than set the congestion window gain to one. We also introduce the congestion window pacing speed 20% at random time it is not required to repeat it periodically. Thus both the introducing factor increased the throughput and handles the congestion in every situation. In our point of view after making the correction in TCP BBR is much more suitable for the next generation internet connection either it is wired or wireless.

Conflict of Interest

The authors confirm that this article contents have no conflict of interest.

Acknowledgement

The authors acknowledge and express the gratitude for the support of department of AIIT, amity university jaipur rajasthan.

REFERENCES

[1] Al-Hasanat, M., Seman, K., & Saadan, K. (2014). Enhanced TCP Westwood slow start phase. *Transactions on Networks and Communications, 2*(5), 194-200.

[2] Almobaideen, W. A., & Al-maitah, N. O. (2014). TCP Karak: A New TCP AIMD Algorithm Based on Duplicated Acknowledgements for MANET. *International Journal of Communications, Network and System Sciences, 7*(09), 396.

[3] Assasa, H., & Widmer, J. (2016). *Implementation and Evaluation of a WLAN IEEE 802.11 ad Model in ns-3.* Paper presented at the Proceedings of the Workshop on ns-3.

[4] Avallone, S., & Di Stasi, G. (2016). Design and implementation of WiMesh: A tool for the performance evaluation of multi-radio wireless mesh networks. *Journal of Network and Computer Applications, 63*, 98-109.

[5] Baldo, N., Requena-Esteso, M., Núñez-Martínez, J., Portolès-Comeras, M., Nin-Guerrero, J., Dini, P., et al. (2010). *Validation of the IEEE 802.11 MAC model in the ns3 simulator using the EXTREME testbed.* Paper presented at the Proceedings of the 3rd International ICST Conference on Simulation Tools and Techniques.

[6] Bassil, Y. (2012). TCP congestion control scheme for wireless networks based on tcp reserved field and snr ratio. *arXiv preprint arXiv:1207.1098*.

[7] Cardwell, N., Cheng, Y., Gunn, C. S., Yeganeh, S. H., & Jacobson, V. (2017). BBR: congestion-based congestion control.

[8] Cheng, Y., & Cardwell, N. (2016). *Making Linux TCP Fast.* Paper presented at the Netdev Conference.

[9] Floyd, S., Handley, M., & Padhye, J. (2000). A comparison of equation-based and AIMD congestion control.

[10] Gangadhar, S., Nguyen, T. A. N., Umapathi, G., & Sterbenz, J. P. (2013). *TCP Westwood (+) protocol implementation in ns-3.* Paper presented at the Proceedings of the 6th International ICST Conference on Simulation Tools and Techniques.

[11] Gwak, Y., Kim, Y. Y., & Kim, R. Y. (2013). *WiCUBIC: Enhanced CUBIC TCP for mobile devices.* Paper presented at the 2013 IEEE International Conference on Consumer Electronics (ICCE).

[12] Levasseur, B., Claypool, M., & Kinicki, R. (2014). *A TCP CUBIC implementation in ns-3.* Paper presented at the Proceedings of the 2014 Workshop on ns-3.

[13] Mishra, D. K., Vankar, P., & Tahiliani, M. P. (2016). *TCP Evaluation Suite for ns-3.* Paper presented at the Proceedings of the Workshop on ns-3.

[14] Nguyen, T. A. N., Gangadhar, S., Rahman, M. M., & Sterbenz, J. P. (2016). *An Implementation of Scalable, Vegas, Veno, and YeAH Congestion Control Algorithms in ns-3.* Paper presented at the Proceedings of the Workshop on ns-3.

[15] Paliwal, G., Mudgal, A. P., & Taterh, S. (2015). *A study on various attacks of tcp/ip and security challenges in manet layer architecture.* Paper presented at the Proceedings of Fourth International Conference on Soft Computing for Problem Solving.

[16] Paliwal, G., & Taterh, S. (2018). Impact of Dense Network in MANET Routing Protocols AODV and DSDV Comparative Analysis Through NS3 *Soft Computing: Theories and Applications* (pp. 327-335): Springer.

[17] Paul, H., & Sarkar, P. (2013). A Survey: High Speed TCP Variants in Wireless Networks. *International Journal of Advance Research in Computer Science and Management Studies, 1*(7).

[18] Poojary, S., & Sharma, V. (2011). *Analytical model for congestion control and throughput with TCP CUBIC connections.* Paper presented at the 2011 IEEE Global Telecommunications Conference-GLOBECOM 2011.

[19] Ravindranath, N., Singh, I., Prasad, A., & Rao, V. (2016). Performance Evaluation of IEEE 802.11

ac and 802.11 n using NS3. *Indian Journal of Science and Technology, 9*(26).

[20] Shah, S. A., Nazir, B., & Khan, I. A. (2016). Congestion control algorithms in wireless sensor.

[21] Singh, S., & Sharma, R. M. (2015). Some aspects of coverage awareness in wireless sensor networks. *Procedia Computer Science, 70*, 160-165.

[22] Wala, G., Gupta, O., & Kumar, S. (2016). Congestion Avoidance in packet networks using network simulator-3 (NS-3): OJCST.

[23] Zhou, J., Wu, Q., Li, Z., Uhlig, S., Steenkiste, P., Chen, J., et al. (2015). *Demystifying and mitigating tcp stalls at the server side.* Paper presented at the Proceedings of the 11th ACM Conference on Emerging Networking Experiments and Technologies.