International Journal For Technological Research In Engineering
Volume 2, Issue 10, June-2015 ISSN (Online): 2347 - 4718

SURVEY ON CHALLENGES AND ISSUES IN CONGESTION CONTROL FOR END TO END DELIVERY IN WIRELESS NETWORK

Satishkumar D. Prajapati[1], Dhaval J. Varia[2]
[1]PG Student, [2]Assistant Professor, Computer Science and Engineering
Government Engineering College Modasa, Gujarat, India

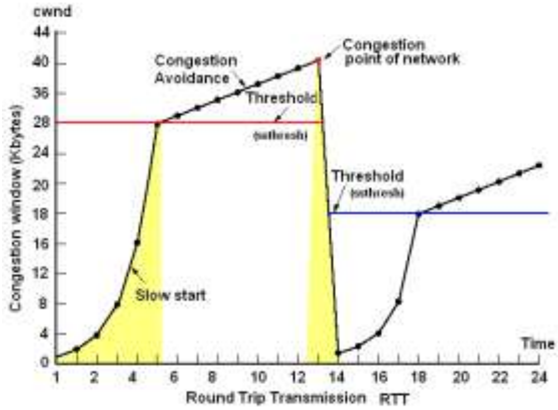
Abstract: Today Internet carries most of the traffic as TCP data which travel across heterogeneous platforms including both wired and wireless networks. With wireless networks, it is a challenge to select proper congestion window size. Segment loss can be because of transmission error or mobility and handoffs is mistaken as cause of congestion. So it is difficult to distinguish the reason for segment loss due to congestion in network or not. New parameters are required specially to control the aggressive ness of the cwnd size in slow start and congestion avoidance phase. Various variants of TCP congestion control has been studied and westwood bandwidth estimation is suggested as parameter to decide cwnd size.

Index Terms: Congestion, TCP, cwnd, slow start, ssthresh, congestion avoidance. (key words)

I. INTRODUCTION

Increasing number of wireless hot spots is providing people to be wireless connected in almost every place now a days. Large amount of static users are using the IEEE 802.11 wireless access technology for checking emails, web surfing, video and audio streaming as well as P2P file sharing. Tcp is the heart of the Internet highway today.90 % of the internet traffic is Tcp in wireless environment.[1] Base for the window based congestion control mechanism was laid down by Van Jacobson[2]. He introduced first congestion control strategy with congestion window, cwnd, maintained at sender side. Cwnd is defined as number of packets in flight which are unacknowledged. [2]. Based on how to increase and maintain value of cwnd many proposals has been evolved [2] Most of the developed protocols were targeting wired networks. Initial network conditions are changed; today we are dealing with broadband networks.[3]. Conventional congestion control mechanism of TCP is divided in four main phases.

A. Slow-start: It is the mechanism to start slowly but reach early in the equilibrium state of network to avoid congestion. Initial cwnd is low but it increase exponentially in the power of two. With every round trip time the cwnd is doubled until it reaches to threshold, known as ssthresh. Figure 1 illustrates the at value 28 Kbytes cwnd reaches to ssthresh and exit from the slow-start phase.



B. Congestion Avoidance: When cwnd is greater than ssthresh, the algorithm is in congestion avoidance phase.In this , cwnd increases linearly with one segment only after every round trip time. When congestion is detected via packet loss then algorithm applies sudden decrease in the value of cwnd. It also set threshold ssthresh to half of the current value popular known as AIMD(additive increase multiplicative decrease) mechanism[AIMD] . Here 13^{th} is the unlucky transmission when congestion occurs. So cwnd is decreased and set to 1 and new threshold is half of current cwnd value.

C. Fast Retransmission: TCP –Tahoe[2] work with cumulative ack and provide retransmission timer mechanism which very coarse and cause waste of resources at the point of congestion. So when three duplicate acknowledgement arrives at sender side then it is considered as packet loss and packet is retransmitted before time is expired.

D. Fast Recovery: TCP-Reno[2] provide fast recovery which work hand in hand with fast retransmission phase. Instead of going in slow-start phase and starting from the cwnd 1 , it enter into congestion avoidance phase when three – duplicate acknowledgement arrives at sender side.It also reduce the slow start threshold to half.

II. LITERATURE SURVEY

Wherever Times is specified, Times Roman or Times New Roman may be used. If neither is available on your word processor, please use the font closest in appearance to Times. Avoid using bit-mapped fonts. True Type 1 or Open Type fonts are required. Please embed all fonts, in particular symbol fonts, as well, for math, etc.

Based on the detection of the congestion, the TCP congestion control algorithms are divided into different families of algorithm.[4]

1. Loss-based Algorithms
   TCP Reno[2]
   TCP newReno
   TCP Sack
   HSTCP and others
2. Delay –based Algorithms
   TCP Vegas
   Fast TCP and others
3. Mixed – loss-delay – based Algorithms
   Compound TCP
   TCP Africa and others
4. Explicit congestion Notification
   XCP
5. Bandwidth Estimation based algorithms[5-6]
   Westwood
   LIAD and other variants of TCP

Slow-start , Congestion Avoidance and Fast retransmission phases collectively known as TCP Tahoe while this three with Fast recovery known as TCP Reno. Fast recovery phase is modified to facilitate multiple losses of packets in one round trip time which is known as TCP new Reno. All this three loss based mechanism do not take any assistance from the receiver side. In place of using cumulative ack, Selective acknowledgement is implemented at the receiver side. The version is popular as TCP sack. Other algorithms like HSTCP, BIC-TCP,CUBIC-TCP and other loss based algorithms are also proposed but lacking some fairness and bandwidth over the wireless network as well as high bandwidth and long delay networks[6-7]

TCP Vegas keeps track of when each packet was sent and calculates an estimate of RTT for each transmission. This is done by monitoring how long it took each ACK to get back to the sender. Whenever a duplicate ACK is received, it performs the following check:

if (current RTT > RTT estimate)

If above condition satisfied, it retransmits the packet without waiting for 3 DUPACK or a coarse time out as in Reno. Hence, Vegas solves the problem of not detecting lost packets when the window is very small *i.e.* less than three and could not receive enough duplicate ACKs[7]. Other than TCP Vegas, there are TCP LP, Fast TCP and other variants this category also faces fairness and friendliness issue in heterogeneous network. So most of the wireless networks are long delay and high bandwidth product networks ,the delay based algorithm inherently lacking the high throughput in this environment.[7] Though mixed delay and loss based approach has been proposed. Most acceptable one is Compound TCP, which soon become standard in windows operating system. Other variants in this direction are TCP illions, TCP Africa. CUBIC can also be considered as loss delay based algorithm.[7] ECN takes help from the active queue management(QAM) of routers. When queue length reaches to specified threshold it modifies the (CE) bit in the TCP packet. While in split –TCP , the base station node is provided with extra amount of buffer. Base station has intermediate agent which retransmit the lost segments from there rather than from the original source of packet. Since the ECN-based and split-based TCP schemes, both needs some assistance from the intermediate nodes (i.e., intermediate routers or the wireless access point) on the TCP connection, these schemes are very expensive for deploying the required components on every intermediate router or wireless access point. This study thus focuses on the issue of the end-to-end TCP congestion control, in which we adopt the case of without any help from intermediate nodes for its efficient feasible implement.[6]

A. Bandwidth Estimation Based algorithms.

TCP Westwood (TCPW)[5] is a well known, sender-side modification of the TCP congestion control algorithm that improves upon the performance of TCP Reno in wired as well as wireless networks. The improvement is most significant in wireless networks with lossy links. An important distinguishing feature of TCP Westwood with respect to previous wireless TCP “extensions” is that it does not require inspection and/or interception of TCP packets at intermediate (proxy) nodes. The key idea of TCP Westwood is to use the “bandwidth estimate” to directly control the congestion window and the slow start threshold. Suppose that an Acknowledgement is received at the source at time tk, which notifies that dk bytes have been delivered at the TCP receiver. Sample bandwidth of the connection is measured by the equation shown as bk = dk/Δk, where Δk = tk − tk-1 and tk-1 is the time the previous ACK was arrived. Available bandwidth estimation is taken by given formula:

$$B_i = \alpha_i \cdot B_i + (1 - \alpha_i)\left(\frac{b_i + b_{i-1}}{2}\right) \quad (1)$$

where αi is a co-efficient for Tustin approximation, Bi-1 is previous bandwidth estimation value and bi-1 is previous sample bandwidth value. Other details for this can be found in[5] When 3-dupack or time out event occurs , the ssthresh is calculated by using above bandwidth estimation in following way:

ssthresh= (BWE * RTTmin) / Seg_size.

In[8]a novel approach to distinguish the loss of the segment occurred in the network because of the congestion or bit error. The traditional TCP westwood does not differentiate the issue of loss. Whether it is through congestion or through link error. So it will reduce the congestion window and degrade the performance. This paper shows novel approach for that by putting indication of the congestion in the header of the TCP segment. The receiver will continuous monitor the rate at which the successive Segments are coming at the receiver side. Based on that receiver set the flag of the acknowledgement So that it can know about the congestion.

In [9] use of bandwidth estimation in the congestion avoidance phase of the TCP westwood algorithm so that in place of linear increment it can improve increment rate. For that it took bandwidth estimation at every acknowledgement and compare it with the previous bandwidth estimation by taking the ratio of both values. If ratio is high it will increment cwnd by two segments in place of conventional 1 segment. So high cwnd values can be achieved to increase the throughput in congestion avoidance phase. If the ratio is lower than the 0.5 in many cases then algorithm will not increase it's traffic. So no increment is there in the congestion window value. Beside this paper propose refinement over the retransmission time value setting as well.By combining effect of this to modification testing is done over the topology and compared with westwood as well as other variants of TCP like reno , vegas and sack. In TCP LIAD(Logarithmic Increase adaptive decrease) uses two different threshold define whether to be in slow –start phase or in congestion avoidance phase. More focus on state transition of the cwnd. It calculate shrink factor defined by

Shrink Factor =RTTmin / RTTcurrent;

Based on calculated shrink factor it decide the value of cwnd for the both phases. More details in[6]

III. COMPARISION

SR.NO	Method	Comments
1	TCP tahoe[2]	First proposal to congestion control at TCP level but having problem low bandwidth utilization because of the coarse time out.
2	TCP Reno[2]	Over coming the shortcomings of the TCP tahoe. Introduced new phases like fast recovery and fast retransmission. Very well in wired networks but not so well in heterogeneous environment.
3	TCP newReno[7]	Making the fast Retransmission and fast recovery phase more stronger. The algorithm can identify multiple n-dupack so handle multiple retransmission in the single window scenario.
4	TCP SACK[7]	Involve receiver in the maintaining congestion free environment in network. It introduces the selective acknowledgements at the receiver side. Though selective repeat at transport layer is difficult to implement and maintain but it is a standard protocol for many simulation and implementations.
5	TCP Vegas[7]	Congestion window is updated based on the end to end delay observed during the transmission rather than packet loss in the earlier mechanisms. But having degraded performance in long delay based networks like most of the wireless networks.
6	TCP WestWood[5]	Instead of directly putting half value as new ssthresh and starting from 1 in earlier reno it takes estimated bandwidth in account to set the new threshold of the congestion window size.
7	TCP LIAD[6]	Uses to different threshold to define whether the we are in slow start phase or in congestion avoidance phase. More focus on improving transition on the state change from the slow start to congestion avoidance phase. Uses logarithmic increase as well as adaptive decrease in congestion avoidance phase.
8	Westwood with Flag[8]	Need receiver side modification
9	TCP Westwood(new)[9]	Use bandwidth estimation ratio in Congestion avoidance only and increment factor is untold.

IV. CHALLENGES AND ISSUES IN WIRELESS

Packet Loss:It is difficult to identify in wireless environment whether the loss of packet is because of congestion or due to link error. Congestion control algorithm takes every loss of packet as cause of congestion and degrade the congestion window as well as early enter in congestion avoidance phase. Multiple losses may repeatedly reduce the slow-start threshold and cause the slower congestion avoidance phase to take over immediately which leads to large throughput degradation[10]

Mobility of Nodes: It was noted that TCP throughput drops significantly when node movements cause link failures.It occurs due to TCP's inability to distinguish the difference between link failure and congestion.[11]

Hand offs: Since handoffs momentarily disrupt the connectivity with adverse effect on TCP performance, hierarchical cellular systems must be carefully designed to avoid increasing the gravity handoff induced problems.The small area and high data rates of microcells will lead to more frequent handoffs and potentially increased losses during each handoff. [10]

StartUp cwnd: Most of the algorithm start with initially very low value of cwnd . Approx mately one or two segment size is preffered to start in slow start phase. Starting very low many time lead very low utilization of available bandwidth. So proper startup mechanism or new parameter are needed in this direction.

Decrease in cwnd : Algorithms like TCP LIAD provides not only sudden decrement but also adaptive decrement in the current cwnd value when there is less available bandwidth in network. But advancement needed to decide and refine how much decrement in cwnd is needed. Cwnd decrement and increment need to be controlled by some new parametes.

N-Dupacks : N-dupack event occurs when n successive duplicate acknowledgement are received at sender side. The sender will assume n- dupack as loss of packet and perform fast retransmission and fast recovery. Issue is the value of n, that number of duplicate acks considered for loss notification. Mostly it is 3 but now with faster networks better results are achieved with n=2 and for some scenario n=4. So, new parameter are required to decide the value of n.[4]

V. CONCLUSION AND FUTURE WORK

During the study of various TCP congestion control algorithms, it was observed that bandwidth estimation based algorithms will prove better to distinguish the link error and congestion. Various other algorithms can be used as mixed approaches to proved enhanced TCP performance in wireless environment. With increased speed and long delay many other issues come across in the wireless networks like handoff and mobility of nodes, affecting TCP throughput. In future research new parametes or variants are needed to decide the aggressive ness of the increment and decrement of the cwnd in wireless networks. Some new fields may be introduced with negative information to enhance the performace in end to end delivery. Bandwidth utilization notification or bandwidth availability notifications might be introduced in futute work. Our aim to carry the study further and use available information at TCP to enhance it's performance without taking help from lower layers of

TCP/IP suit in wireless networks.

REFERENCES

[1] Toni Janevaski, and Ivan Petrov, “Performances of Internet Transport protocols in wireless Environment,” TELSIKS, IEEE,Serbia, vol. 9, pp. 126–129, October 2009.
[2] Van Jacobson, “Congestion control and avoidance ”ACM SIGGCOMM,Computer Communication Review, vol.18(4) ,1988 , pp.314–329..
[3] Toni Janevaski, and Ivan Petrov, “Improved Slow Start Algorithms” 21 st Telecommunication Forum TELFOR,Serbia, 2013, pp. 26-28.
[4] Saleem-ullah Lar,Xiaofeng Liao, and Songtao Guo, “Modeling TCP New Reno Slow – Start and Congestion Avoidance using Simulation Approach” IJCSNS, International Journal of Coputer Science and Network Security., vol 11, January,2011 pp. 117-124.
[5] Claudio Casetti, Mario Gerla, Saverio Mascolo, M.Y. Sanadidi and Ren Wang, “TCP Westwood : End –to – end Congestion Control for Wired/Wireless networks”, Wireless networks , vol 8, Kluwe Academics publications.,Nethereland, 2002, pp. 467-479.
[6] Ben-Jye-Change, Shu-Yu Lin, Jun-yu Jin ,“LIAD : Adaptive bandwidth prediction based Logarithmic Increase Adaptive Decrease for TCP congestion control in heterogeneous wireless networks,” ScienceDirect ,Elsevier,Computer network vol. 53, pp. 2566–2585, 2009.
[7] K.I.Oyeyinka, A.O. Oluwatope, A.T. Akinwale, O. Folorunso, G.A. Aderounmu, O.O. Abiona, “TCP window based congestion control slow – start approach” ,Communication and networks, vol.3 pp.85-98, 2011
[8] Diwan G. Raimagia, Charvi N. Chanda, “A novel approach to enhance the performance of TCP westwood on wireless link” IEEE, Nirma University International conference in Engineering(NUiCON) pp. 1-6, 2013.
[9] Sheema Hagag, Ayeman-ul Sayed, “Enhanced TCP Westwood Congestion avoidace mechanism(TCP westwood new)”, International Journal of Computer Application,IEEE, Vol. 45, 2012
[10] George C. Polyzos, George Xylomenos, Peteri Mahonen ,Mika Saaranem, “ TCP Performance Issues over Wireless Links”, IEEE Communication Magazine, Vol 39, pp 52-58 , 2001
[11] Gavin Holland , Nitin Vaidya, “Analysis of TCP Performance over Wireless Adhoc Netwoks”, Wireless Networks, Kluwer Academics Publishers,Vol 8 . pp. 275-288, 2002
[12] G.A. Abed, Mahmaod Ismail, Kasmiran Jumari, “Comparision and analysis of Congestion Window for HS-TCP, Full-TCP, and TCP-Linux in Long Term Evolution System model” ,IEEE Conference on Open System pp.25-28, 2011

