An unambiguous TCP method for dynamic rate control to maximise bandwidth utilization in high-rate networks.

Priya Kohli,
Asst. Professor, SOC (School of computing),
GEHU-Dehradun Campus
DOI: 10.48047/jcr.07.09.575

Abstract: – When used to high-bandwidth interval creation networks designed to transmit association information at tens of thousands of Megabits per second (Mbps), the tried-and-true TCP has shown to be woefully inefficient. When congestion reaches a certain threshold, the algorithm used to control it in the Transmission Control Protocol reduces the size of the congestion window to half of its current value, piqueing interest in an additive increase methodology that could be too slow for the alluring ease of a vastly greater magnitude of usable bandwidth. This paper unveils a revised version of the TCP protocol that improves upon its predecessor in a number of key respects, including stability, fairness, bandwidth utilisation, performance, and throughput, and is designed for use in high-speed networks.

Keywords: AIMD, Sliding Window, RTT, and Congestion Reduce, High Rate Networks.

Introduction

The Transmission Control Protocol (TCP) has been reused for the Internet on the transport layer for at least two decades. Internet repetition has enhanced by unusual commands of extents. The application climate has changed considerably for the worse. The original plan by which the proposition's final number was calculated is now obsolete. As far as the Internet is concerned, TCP still holds the ace of spades as far as the TCP/IP protocol stack is concerned. Since connection loads are always changing, TCP requires continuous enhancement to ensure its continued operation [1, 2, 3]. Reformed application procedures allow for modern rate regulator processes to be developed [2, 4, 5, 8]. While the modern Internet makes use of a number of congestion regulation schemes, TCP is still the most widely utilised protocol at the transport layer. According to [4]'s authors, a novel proposal for congestion regulator that can monitor a new congestion drop down and posture the difficulty compliance of congestion reply is both feasible and advised. Transmission Control Protocol operates on Layer 4 of the seven-layer Open Systems Interconnection (OSI) network paradigm. While assuming just a thin layer of abstraction between the IP and lower layers, this protocol offers a variety of features, including dependability, sequence delivery, and a byte-stream facility with organised movement. A combination of algorithms may achieve the aforementioned. Transmission Control Protocol now uses the four algorithms to act as a congestion regulator. These protocols use a number of different timers and algorithms, including delayed start, congestion avoidance, rapid retransmission, and rapid recovery. To quickly transmit a fresh beginning stream to step, slow start works by stretching the window exponentially. In a continuous condition, the motions continually use a combination of congestion avoidance and quick retransmit. The Additive

Increase Multiplicative Decrease (AIMD) method is often used in the window of congestion. Congestion window is extended while no damages may be tried and the effective acknowledgment has been obtained. When a packet is lost, the congestion window is reduced to a smaller value so that the blocked linking buffers may be removed. Existing networks provide a rare but manageable danger to the AIMD approach.

Working of TCP

TCP is an automated and dependable transport convention since the source makes use of information provided by the destination during the confirmations procedure. There is no guarantee that the switches will always respond in the same way. The idea behind this opening is that bottlenecks in the system are to blame if packages fail to arrive at their destination in the same order in which they were sent. While this claim holds merit in conventional settings, it is prompted by the modern organisational climate [3]. TCP uses the AIMD clog control computation provided by sliding window design by Van Jacobson and others [1]. The field of network coding has seen a lot of research and development over the years, with several publications covering topics like coding strategies [3, 4, 5, 6], network utility optimisation via network coding [7, 8, 9, 10, 11], and network coding implementation [12, 14], among others. Since wireless channels are naturally broadcast and ideal for the application of network coding, the majority of the aforementioned studies are conducted through wireless networks. The two main types of network coding are intra-session network coding and inter-session network coding. While only packets belonging to the same session are coded in intra-session network coding, packets belonging to distinct sessions are coded jointly in inter-session network coding. Multicast is the sole use of intra-session network coding, and it has been shown that linear network coding may reach the multicast rate area [3]. Inter-session network coding is more difficult, and how best to optimise it remains an unanswered subject. However, in real-world wired and wireless networks, many more simultaneous flows occur, and even simple network coding techniques may provide considerable performance gains [14]. This highlights the greater value and appeal of building and optimising inter-session network coding for real-world networks. Using XOR (Exclusive Disjunction, also known as Exclusive OR) as the operation between packets in an 802.11-based wireless ad hoc network, opportunistic network coding was suggested and first implemented [14]. This resulted in a significant speed boost. According to [14], the speed boost from using TCP with COPE on an 802.11 network is minimal at best. Most wireless applications, however, still depend on the older version of TCP in order to talk to TCP-dominated wired hosts, and it's probable that TCP will continue to be the dominant transport protocol for 802.11 network clients [15]. This highlights the significance and value of studying and enhancing TCP performance in multihop ad hoc networks. Due to the unique characteristics of wireless networks, such as hidden terminal and exposed terminal difficulties, transmission faults, topology fluctuations, and routing instability, etc. [16], the performance of traditional TCP across multihop ad hoc networks is less than ideal. TCP performance in multihop ad hoc networks has been studied extensively over the last several years [16, 17, 18, 19, 20, 21, 22] by taking into account the real-world constraints of wireless ad hoc networks. Recent research [23, 24, 25] has also looked at the benefits of combining TCP with network coding. To far, however, there has been no investigation into the possibility that TCP's rate control mechanism and the routing protocol are to blame for the protocol's performance decline [15, 26, 27].

The Transmission Control Protocol, often known as TCP, is considered to be one of the core protocols that make up the Internet Protocol Suite. The Transmission Control Protocol (TCP) is a host-to-host protocol that was developed specifically for use in packet-switched computer network communications and the integrated systems that use these networks [40]. TCP, that stands for "transmission control protocol," regulates the flow of data between computers when they are connected to a network. This guarantees that the data is sent in a reliable and timely manner. Traffic congestion, traffic load balancing as well as and other unexpected network behaviour require that TCP include specific characteristics to identify these difficulties, transmit again the lost information rearrange out-of-order info, and even assist mitigate network congestion to minimise the recurrence of the other problems. These features may be found in TCP version 4. As a result, TCP is responsible for managing connections, controlling congestion, and managing the flow and data. The primary objective of this research project is to zero in on the four interdependent algorithms that comprise congestion management. These algorithms are known as Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery.

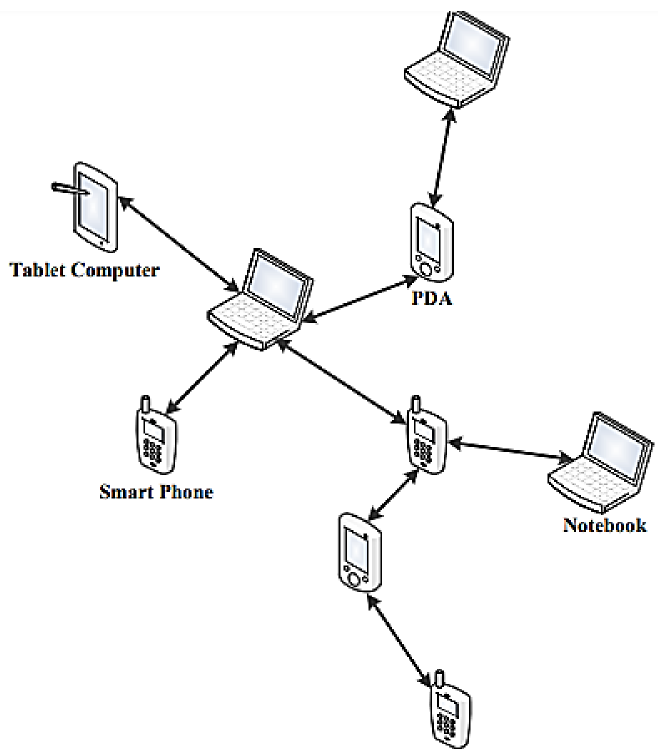


Figure 1: A Classic Wireless Ad Hoc Network

As we have previously shown, the TCP protocol's capabilities and algorithms were largely developed for and tested on wired networks, which is where they operate most effectively. However, as wireless networks and gadgets have progressed, they are no longer well-suited for usage with TCP due to these advancements. Because wired TCP protocols are unable to differentiate between packet losses caused by congestion and link failures, they treat most packet failures as if they were caused by capacity and decrease the sending rate to relieve the issue. Wireless TCP protocols are able to differentiate between packet losses caused by congestion and connection errors. This, in turn, results in a significant reduction in the size of the congestion window. Despite this, packet loss is rather prevalent in wireless networks due to causes other than congestion. These problems include things like transmitting error, flexibility, fading, training, hand off, and so on. There is a chance that the wireless devices will automatically recover from this little setback on their own. If congestion management isn't implemented before recovery, then there will be a considerable decrease in TCP performance. As a direct consequence of this, the dimensions of the window of contention shifts around rather often, and the radio link is employed only to a very limited extent. A significant amount of research has been done looking for possible answers to these problems. In order to address these problems, you may choose to either keep error losses a secret from the person who sent them or explain the cause for the packet loss to the sender. It is necessary to make certain adjustments to the network in order to implement split-connection answers (such as I-TCP and M-TCP), but end-to-end and link layer solutions (which need to be modified on either the client or the server) do not call for these kinds of adjustments. complete the solutions that have been offered.

Literature Review

Saiyin Hou et.al.,(2021) under single-link data transmission, the TCP accelerator mechanism has been extensively utilised to improve system performance and channel utilisation under conditions like extended latency and high error codes. The classic TCP acceleration method has a low acceleration impact on coordinated transmission, despite the fact that an increasing number of network devices are able to coordinate data transfer across several network interfaces. In order to boost TCP information transfer speed during multi-link collaborative transmission, this research offers a TCP acceleration mechanism well-suited for heterogeneity network collaboration transmission situations..

Rahul Pradeep et.al.,(2021) Due to TCP's inability to determine the cause of packet loss, traditional end-to-end Congestion Control techniques cannot be easily applied to heterogeneous wireless networks. In wireless networks, several simultaneous broadcasts produce interference or frame collisions, which in turn reduces throughput. There are several ML strategies for congestion management, but neither supervised nor unsupervised learning can teach you how to implement the best policy. As a result, it is necessary to create a model that dynamically interacts with the surrounding environment in order to accurately anticipate the best congestion window. To overcome these obstacles, we provide a model based on reinforcement learning that uses the Actor-Critic approach and Temporal Difference learning to make real-time changes to the congestion window. Experiments show that the suggested learning model delivers 40% higher throughput than the state-of-the-art methods while keeping the transmission delay to a minimum.

Jianpeng Xu et.al.,(2020) To provide service dependability in HSR networks, multipath transmission control protocol (MPTCP) is a very promising technology. However, the performance of MPTCP is negatively impacted by the occurrence of wireless losses induced by frequent handoffs or other non-congestion losses in HSR networks. In this letter, we provide a new congestion management strategy called RVeno to enhance the performance of MPTCP while aggregating Long-Term Evolution for Railway (LTE-R) and WiFi communication paths in HSR networks. Throughput findings from simulations show that RVeno is superior than the current crop of MPTCP congestion management algorithms..

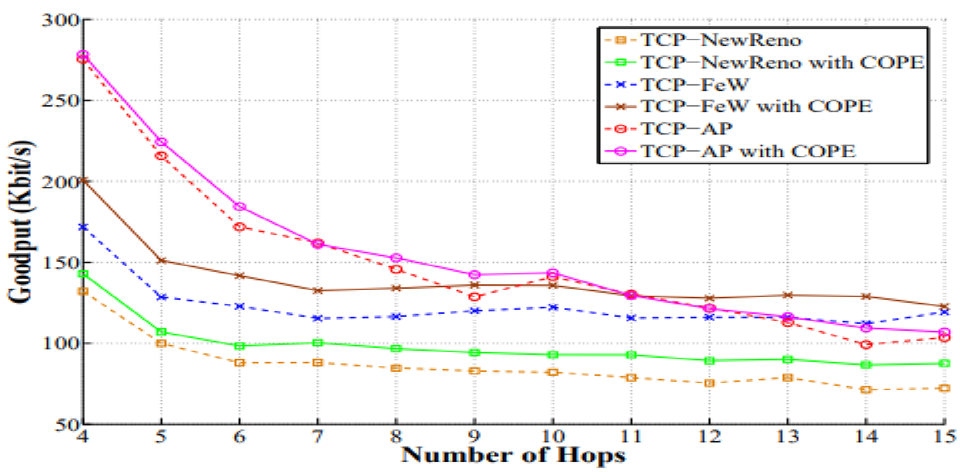
Kaoutar Bazi et.al.,(2020) Congestion in wireless mesh networks caused by the Transmission Control Protocol (TCP) is still an issue that the networking world has yet to crack. Several algorithms for congestion management have been explored as potential solutions. Congestion in a network makes it harder for TCP connections to provide fairness, stability, efficient bandwidth allocation, adequate throughput, or even low latency. In this study, we make extensive use of analyses and comparisons of TCP variations, with an emphasis on those built for high-speed networks, to address this topic in depth and draw conclusions and make recommendations for improvement..

Present Weakness of TCP

The necessary method for congestion in TCP postulation is widely used, and in the last few years, a great deal of training [2, 4, 5, 6] has been made accessible to examine it. Several researchers have laboured to perfect TCP's algorithm for dealing with congestion. Due to its predictable mechanism for avoiding congestion, Transmission Control Protocol cannot serve the whole achievable maximum data rate on high bandwidth. The legitimacy of TCP might be compromised in these conditions. The problem is that TCP has a hard time distinguishing between high traffic and a slow connection. This means that as long as there is no more data loss, the Transmission Control Protocol system will continue on the route to expansion in the window of congestion, so increasing the circulation frequency[9]. It's intriguing because of the potential for packet rejection due to congestion on the narrow channel. Once a congestion warning has quickly reached the source, any out-of-the-ordinary data should resolve inside the congestion window's normal range, as measured by the shortest possible round-trip and acknowledgment times. If a warning can travel quickly from its origin to its target, then out-of-the-ordinary data should fall inside the norm during a congested period, as measured by the round-trip time of the acknowledgments. The evaluation of the congestion avoidance tool verifies that the bursts in the Transmission Control Protocol congestion algorithm are separated into two distinct time intervals. First, a sluggish start, and second, avoiding traffic. In the first stage, the method doubles the packet count up until packet loss occurs. The standard congestion controller technique in Transmission Control Protocol reduces the congestion window to half its original size later on in light of packet damage. Again, the congestion window would be slashed if Transmission Control Protocol detects data loss. This is an example of a technique known as "multiplicative decrease" that prevents data packets from tumbling down. Important concerns are to the occurrence of sluggish start times during the uncommon later window modifications. To understand the blockage queue scope and the router's capabilities in the new Transmission Control Protocol scheme, we must discard our previous narrow assessments and refrain from using exponential increase in window size due to the risk of losing data at the origin. Instead, it must use an adaptive TCP algorithm to enlarge its window in order to keep from losing a disproportionately large number of packets. It sends data at full speed after determining it to be the optimal rate. The method also has the capacity to predict congestion in real time and adjust the packet transfer rate to reflect the newly available bandwidth.

Network Traffic Flow Categorization

Here we assume in the aforementioned architecture that "n" senders, denoted by S1, S2,..., up to Sn, are exchanging data with "n" receivers, denoted by D1, D2,..., up to Dn. This updated protocol works alongside the router and incorporates fewer changes from the standard TCP. Once the packets have been sent from the source to the router, the Store and Forward standard will be used to continue the conversation. This is because packets are stored in the in queue before being moved to the out queue if the outbound communication channel becomes unmanageable for forward broadcast of the gathered packets. For now, let's assume that there are n distinct queues in in queue that incorporate with one another, each of them conveying sender that is, the broadcast by sender S1 will be set up in the q1 of in queue, the broadcast by sender S2 will be set up inside q2 of in queue, and so on. The Round Robin method will be used to choose packets from the outgoing queue. That is, q1, q2, etc., designate a packet. Up until this point, there has been zero empirical evidence of congestion inside the network, and hence zero evidence of packet damage. Acknowledgement packets instruct the directed sources to reduce their transfer speeds when a packet loss occurs instantly, as depicted in figure 1. The router then enters wait mode, during which it does the aforementioned work in a distributed fashion for a length of time determined in advance using the Que_occupancy table. If a source is determined to be a disobedient source after its expected pause step ends and the source fails to comply through the rate reduction, all packets from that source will be dropped from the in queue. Disobedient sources' unpaid bandwidth is added to the total available bandwidth, allowing for more bandwidth to be allocated to new sources with high communication demands [12].



Number of Sources (1, 2, 3,... n): This is a numeric reference to the many sources (S1, S2, etc.). Sn. The transmitting device's IP address, also known as the "Source IP Address." The IP address of the device that will be receiving the data.

Present Price: High rate Network communication package vendor and the directing source agree on the distribution frequency.

New_Sending_Rate: As the interceding router becomes involved in the congestion through packet dropping, a new Distribution rate is carefully calculated for each distribution source based on the number of packets from that source currently waiting in the in_queue relative to the total Que_Occupancy table.

Occupancy. Wait_Time: Before the stage router reports congestion, this field in the Que_Occupancy table is blank. When packets begin to bounce, however, the Congestion detection mechanism comes to life, and the New_SendingRate discard calculation is performed for all sources. Initially, the Que_Occupancy table is updated to reflect the new New_Sending_Rate, which is subsequently sent to the distribution sources through acknowledgment packets.

Traffic from Performing sources:

Every Source device that sends out packets using the QoS [11], [13], & [10] conditions decided upon will be considered a Behaving source if, in response to heavy traffic, it slows down his own present conveyance rates in order to wait for acknowledgement packets from the congested devices. Non-Preforming source traffic: Non-Performing sources include any and all Source devices where a packet has not been transferred due to evident problems governed by Quality of Service (QoS), even after the congested device has acknowledged its receipt of the packet. For the sake of propagating further packets, these non-performing devices endure, which might lead to a drop owing to a high ratio of queue occupancy and, ultimately, network congestion.

Conclusion

A method for adjusting the behaviour of the sliding window technique used by the transmission control protocol (TCP) in PtMP wireless networks. In order to regulate the TCP transmission rate, the procedure involves checking the consistency of packet transport over a wireless telecommunications connection and, if necessary, adjusting the operation of a TCP sliding window algorithm. A link layer over the wireless connection is used to convey an acknowledgment of the packet's transmission, and a link layer over the wireless link is used to retransmit a lost packet. When it comes to media access control (MAC), the link layer might be packet-centric and QoS-aware. In order to prevent the TCP sliding window method from adjusting the TCP transmission rate, it is suppressed during the modifying stage.

References

1. Jacobson, V. Congestion avoidance and control. In Proceedings of SIGCOMM '88, Stanford, CA, Aug. 1988.
2. Sally Floyd, HighSpeed TCP for Large Congestion Windows and Quick-Start for TCP and IP, Yokohama IETF, July 18, 2002, Available at http://www.icir.org/floyd/hstcp.html
3. Sumitha Bhandarkar, PhD thesis submitted to the Office of Graduate Studies of Texas A & M University (TAMU), 2006.
4. Dina Katabi, Mark Handley, Charlie Rohrs. Congestion Control for High Bandwidth-Delay Product Networks. SIGCOMM „02, Pittsburgh, Pa, Aug. 2002.

5. Tom Kelly, Scalable TCP: Improving Performance in Highspeed Wide Area Networks, ACM SIGCOMM Computer CommunicationReview, Feb" 2003. http://.citeseer.ist.psu.edu/kelly03scalable.html
6. C. Jin, D. Wei, S. H. Low, G. Buhrmaster, J. Bunn, D. H. Choe, R. L. A. Cottrell, J. C. Doyle, W. Feng, O. Martin, H. Newman, F. Paganini, S. Ravot, S. Singh, FAST TCP: From Theory to Experiments, Dec. 6, 2003. http://netlab.caltech.edu/FAST/publications.html
7. G. Jin, Feedback adaptive control and feedback asymptotic convergence algorithms for measuring network bandwidth. LBNL-53165.
8. C.Socrates, P.M.Beulah Devamalar, R.Kannamma Sridharan, Congestion Control for Packet SwitchedNetworks: A Survey, IJSR, Vol. 4, Issue 12 December 2014.
9. Gurmeen Kaur, Kamaljeet Kaint, Rakesh Kumar, Congestion Control in MANET Using Various Approaches:A Review. IJARCSSE Vol 6, Issue 6 June 2016.
10. R. Guerin and V. Peris. "Quality-of-service in packet networks: Basic mechanisms and directions" Computer Networks, 31(3):169–189, February 1999.
11. Weibin Zhao, David Olshefski and Henning Schulzrinne "Internet Quality of Service: an Overview" Columbia University.
12. G. L. Nemhauser " Introduction to Dynamic Programming" John Wiley, New York, 1966.
13. P. Ferguson and G. Huston. "Quality of Service: Delivering QoS in the Internet and the Corporate Network" Wiley Computer Books, New York, NY, 1998
14. H. Safa, O. Mirza, and H. Artail, "A Dynamic Energy Efficient Clustering Algorithm for MANETs," pp. 51-56, 2008
15. W. Jiang, Z. Li, C. Zeng, and H. Jin, "Load Balancing Routing Algorithm for Ad Hoc Networks," pp. 334-339, 2009
16. S. Muthuramalingam, R. Rajaram, and G. Padmavathy, "Clustering Algorithm to Reduce the Overheads and to Balance the Load Using an Hybrid Algorithm in Gauss Markov Model," International Journal of Computer and Electrical Engineering, vol. 2, pp. 1793-8163, 2010
17. C.-C. Tseng, Y.-F. Tsai, L.-H. Chang, H.-C. Wang, K.-C. Ting, and F.-C. Kuo, "Organizing balanced and power-efficient clustered architecture for wireless ad hoc networks," in TENCON 2011-2011 IEEE Region 10 Conference, 2011, pp. 103-107
18. H. Cheng and S. Yang, "Genetic algorithms with elitism-based immigrants for dynamic load balanced clustering problem in mobile ad hoc networks," in Computational Intelligence in Dynamic and Uncertain Environments (CIDUE), 2011 IEEE Symposium on, 2011, pp. 1-7
19. A.Chodorek and R.R. Chodorek, "Streaming Video over TFRC with Linear Throughput Equation," Advances inElectronics and Telecommunications, vol/issue: 1(2), pp.26-29, 2010
20. M.A. Talaat, et al., "Enhanced TCP-friendly rate control for supporting video traffic over internet," CJECE, vol/issue:36(3), pp.135-140, 2013.ISSN: 0840-8688, Publisher: IEEE, DOI: 10.1109/CJECE.2013.6704695
21. A.SathiaseelanandG.Fairhurst, "TCP-Friendly Rate Control (TFRC) for bursty media flows,"Computer Communications, vol/issue:34(15), pp.1836-1847, 2011