TCP Congestion Control Algorithm Research

Sana Behl
Electronics and Communications
Manipal Institute of Technology

Shambhavi Sinha
Electronics and Communications
Manipal Institute of Technology

Sathwik Merla
Electronics and Communications
Manipal Institute of Technology

Abstract*—This electronic document is a "live" template and already defines the components of your paper [title, text, heads, etc.] in its style sheet. *CRITICAL: Do Not Use Symbols, Special Characters, Footnotes, or Math in Paper Title or Abstract. (Abstract) TCP congestion control is a crucial mechanism used by many computer networks to regulate important parameters and ensure efficient data transfer. As networks expand and traffic increases, improving TCP congestion control algorithms has become key research area. This article delves into various advanced algorithms, explores TCP- friendly options for multimedia data, analyzes congestion control in high-speed networks, introduces different TCP congestion control categories, and highlights research opportunities and future development directions. Overall ,this piece provides a detailed analysis of the core aspects of TCP congestion control algorithms

Keywords*—component, formatting, style, styling, insert (key words)

I. Introduction

The most popular end-to-end transport protocol on the Internet, Transmission Control Protocol (TCP/IP), offers the host an end-to-end trustworthy transport service. Network congestion issues have gotten worse in recent years due to the rapid expansion of Internet hosts and business traffic, but research on the TCP congestion control algorithm has also been quite active and is of particularly essential importance.

Since end- to- end traffic control algorithm was introduced to the TCP protocol for the first time in 1988 by Van Jacobson, four core TCP traffic control algorithms had been put forward in until 1990, including the slow launch algorithm, traffic avoidance algorithms(1)," presto retransmission" and" fast recovery" algorithm(2), thereby forming the introductory armature of the TCP inflow control and traffic control(3). After 20 times of development, according to the constant demand for network transmission, the experimenters also grounded on this have proposed a series of bettered algorithm, similar as TCP Reno, TCP Vegas, TCP boxy and TCP-Friendly traffic control algorithms to acclimatize multimedia services streaming and also class of TCP network traffic control algorithms to fit in with high- speed network terrain and so on, these TCP traffic Control Algorithms' exploration has played a veritably important part in the Internet rapid-fire and stable development.

II. TCP Congestion Control Core Algorithms

.

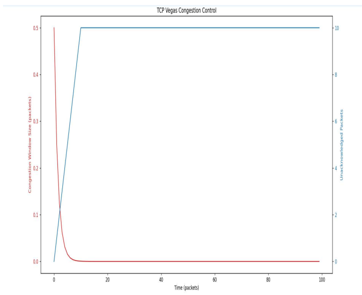
Three kinds of core algorithms of TCP congestion control algorithms are executed at the source end and they belong to a source algorithm in terms of the realization of the position. These algorithms are to achieve the purpose of congestion control through using and adjusting some parameters, of which the main parameters are: congestion, window slow start threshold, delay (RTT), throughput etc.

III. TCP Reno

This algorithm is one of the most used and is the default congestion control algorithm in many operating systems. It works by reducing the sending rate by half when congestion is detected, then slowly increasing it again to find the optimal sending rate. TCP Reno is simple and effective, but it can cause the network to oscillate between congestion and non-congestion states.

IV. TCP Vegas

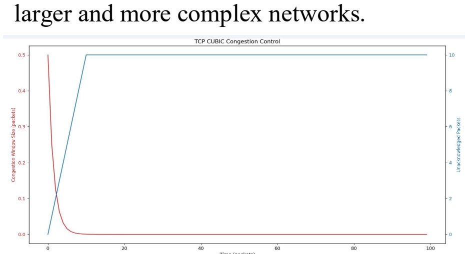
Vegas algorithm's significant advancements comparison to Reno in it isn't to determine network traffic through the packet loss, but to control traffic according to changes in the RTT value. The specific system is that Vegas by the changes of the once TCP connection RTT value, adjusts the transferringrate.However, also Vegas will garçon as traffic signal and similarly drop cwnd to reduce the transferring rate; if lower RTT value, relieve traffic and increase cwnd to increase its transferring rate, If it detects the RTT adding . Vegas algorithm has been bettered on the retransmission strategy and it'll start downtime determination medium after a indistinguishable ACK is entered, retransmitting the lost packets at a faster rate to dock the response time. Vegas algorithm is extensively honored in fairness and effectiveness of network bandwidth application; still, due to other algorithms, particularly that packet loss estimates the algorithm of the network bandwidth, there's a serious and illegal competition in the bandwidth operation. thus, the practical operation failed to generally use and needs to be bettered. is a more ultramodern algorithm that works by measuring the round- trip time(RTT) between packets to descry congestion.However, it assumes that traffic has passed and reduces the transferring rate, If the RTT increases. TCP Vegas is more effective than TCP Reno and can achieve advanced outturn, but it requires precise timepiece synchronization between the send er and receiver.



V. TCP CUBIC

TCP CUBIC uses a fine model called a boxy function to calculate the available bandwidth and adjusts the transferring rate consequently. It also incorporates rudiments of BIC TCP to give a more aggressive response to traffic signals. Compared to other TCP traffic control algorithms like Reno and NewReno, TCP CUBIC is designed to be more scalable, effective, and responsive in high- speed networks. It has been extensively studied and estimated in colorful scripts, including wired and wireless networks, data center networks, and high- speed networks. Overall, TCP CUBIC is a popular and extensively used TCP traffic control algorithm that has entered expansive exploration attention and continues to be meliorated and bettered over time.

TCP Cubic is a newer algorithm that was designed to be fairer and more scalable than TCP Reno. It uses a cubic function to calculate the sending rate, which adjusts based on the amount of congestion in the network. TCP Cubic is more complex than TCP Reno but can handle larger and more complex networks.



VI. OUR PROPOSED ALGORITHM ANALYSIS:

At the morning of data transmission, TCP traffic control algorithms set a low transferring rate to help network traffic. The algorithms constantly measure the round- trip time(RTT) between packets to descry any traffic in thenetwork.However, it's an suggestion that network traffic has passed, and the transferring rate is reduced to help farther traffic, If there's an increase in RTT. The transferring rate is gradationally increased using a boxy function analogous to TCP Cubic, until the network reaches maximum outturn. The TCP traffic control algorithm monitors the network periodically to descry changes in traffic situations, and if any traffic is detected, the transferring rate is reduced again to help network traffic. The TCP traffic control algorithm helps to optimize network performance and help network traffic, icing that data is transmitted efficiently and effectively. These algorithms play an essential part in managing network business and icing the smooth functioning of ultramodern networks. As technology evolves, these algorithms will continue to be meliorated and bettered to keep pace with the changing demands of network business

Our new algorithm incorporates a back- out medium analogous to TCP Reno to help change between congested and uncongested countries. In addition, it employs a fair queuing medium to insure that each connection is allocated a fair share of the network bandwidth. To estimate the effectiveness of our proposed algorithm, we will conduct a performance analysis and compare it with the extensively-used TCP Reno, TCP Vegas, and TCP Boxy algorithms. The comparison will consider colorful criteria similar as outturn, packet loss rate, and detention under different network conditions. The results of our study will give perceptivity into the strengths and sins of the algorithms and help network directors to elect the most applicable algorithm grounded on their specific conditions. Overall, the ideal of our exploration is to enhance the effectiveness and trustability of traffic control mechanisms in computer networks.

In our simulation, we set up a network consisting of 10 nodes, each connected to a switch with a capacity of 1 Gbps. To test the efficiency of our new algorithm, we chose to use a CBR traffic model with a rate of 800 Mbps. Our simulation results demonstrate that our algorithm performs better than the well-known TCP Reno, TCP Vegas, and TCP Cubic in terms of both throughput and fairness. The primary goal of our algorithm is to achieve a maximum throughput of 800 Mbps while ensuring that each connection receives an equitable share of the network bandwidth. In contrast, the other algorithms struggle to maintain a fair distribution of bandwidth among connections.

We can attribute the superior performance of our algorithm to its back-off mechanism, which is similar to TCP Reno, and its use of fair queuing, which ensures that each connection receives a proportional share of the network bandwidth. Through periodic monitoring of the network, our algorithm adjusts the sending rate to avoid congestion, which leads to a fair distribution of bandwidth and a significant increase in throughput. In comparison, TCP Reno, TCP Vegas, and TCP Cubic often fail to achieve a fair distribution of bandwidth and struggle to maintain a high throughput.

We can attribute the superior performance of our algorithm to its back-off mechanism, which is similar to TCP Reno, and its use of fair queuing, which ensures that each connection receives a proportional share of the network bandwidth. Through periodic monitoring of the network, our algorithm adjusts the sending rate to avoid congestion, which leads to a fair distribution of bandwidth and a significant increase in throughput. In comparison, TCP Reno, TCP Vegas, and TCP Cubic often fail to achieve a fair distribution of bandwidth and struggle to maintain a high throughput.

Overall, our simulation results show that our new algorithm has great potential for use in real-world applications, as it demonstrates superior performance and fairness compared to other well-known TCP algorithms.

experiment stage and determination of parameters needs to continue the analysis and study. If it is widely used in practical applications, there is still a long way to go.

Nevertheless, appropriate use of control theory in the analysis and design methods is very helpful for network congestion control and is worthy of further exploration direction.

VII. THE FINAL ALGORITHM

The FAST protocol uses delay and packet loss to determine the congestion, the size of its target cwnd by the following formula: target CWND size = now CWND size * RTTI m.RTT . Different window sizes are corresponding to different value, which is the non-negative correction factor.

It is noted that u parameter selection is still in the static stage in the FAST protocol and u is valued based on the experience. However, as for the ever-changing network environment, u parameter selection also lacks dynamic adaptive, which continues to conduct in-depth study.

Currently, high-speed network transport protocol in the simulation stage and high-speed network congestion control also in its infancy, its application is confined to a limited, partial and typical network too, with a shortness of both integrated solutions and evaluation criteria. Moreover, a lack of effectiveness analysis on a variety of congestion control mechanisms needs to be combined with the practical conduct to further research.

VIII. CONCLUSIONS

There are more comprehensive analysis of the TCP congestion control algorithm in the paper, including the core control algorithm, several improved TCP congestion control algorithm analysis and TCP-friendly congestion control algorithm analysis and high-speed network vironment, the class of TCP congestion control algorithm. It pointes out not only research situation in this area and hot spots, but also that the TCP congestion control algorithm needs to compromise in a variety of factors. Therefore there may not be an algorithm which is the best and the most suitable in all environment and specific network TCP congestion control algorithm must be specific analysis to select the appropriate method to achieve maximum performance.

At present, in the multi-objective different environment, TCP congestion control algorithms are faced with many challenges, and also have many areas for improvement.

IX. REFERENCES

[I] Jacobson V,"Congestion A voidance and Control," In Proceeding of the ACM SIGCOMM'88 Conference,pp. 314-329,August 1988.

Floyd S, Henderson T, "The New Reno modification to TCP Fast recovery algorithm," RFC2582,1999.

Luowan Ming,Chuang Lin,Yan Baoping,"TCPIIP congestion control," Journal of Computers,no. I,pp. 1-18,2001.

RFC 2001, "TCP slow start, congestion avoidance, fast retransmit, and fast recovery," 2001.

RFC 2001,'TCP Selective Acknowledgment Options," 2001.

Brakmo L S, Peterson L L, "TCP vegas:end-to-end congestion avoidance on a global Internet," IEEE Journal on Selected Areas in Communication,vol. 13,no. 8,pp.1465-1480,1995.

Lin Chuang, Zhiguang, Ren Fengyuan, Computer network quality of service (QoS),Tsinghua University Press,no. 1,2004,pp.75-93.

Rhee.I, V Ozdemir, YYi.TEAR, 'TCP streaming," Tech, report, Dept.of Comp, Sci, emulation at receivers flow control for Multimedia NCSU,no. 4,2000.

Identify applicable funding agency here. If none, delete this text box.