Mr. Gaganpreet Singh et al Int. Journal of Engineering Research and Applications www.ijera.com
ISSN : 2248-9622, Vol. 4, Issue 1(Version 1), January 2014, pp.335-337

RESEARCH ARTICLE OPEN ACCESS

Rate Control Protocol for Fast Flows: A Survey

Mr. Gaganpreet Singh, Mr. Sukhvir Singh
U.I.E.T, Panjab University, Chandigarh
U.I.E.T,Panjab University, Chandigarh

Abstract
In today's world, congestion control is a main objective to maximize fairness, utilization and throughput of the Internet. Every protocol has its own features to handle the congestion. The most widely used protocol over the Internet is Transfer Control Protocol. It aims at reliable and in order delivery of bytes to the higher layer and it also protect the network from congestive control. Other congestion control protocols are XCP and RCP. These new protocols are advancement over TCP. We study new congestion control protocol like Rate Control Protocol that make flows complete frequently as compared to TCP and other version of TCP and XCP. In this paper we have presented a comparison between TCP, XCP and RCP, which shows that RCP is a superior choice to use over the Internet to make flows complete quickly.

Keywords: Transmission control protocol, Explicit control protocol, Rate control protocol

I. Introduction

Transmission Control Protocol is a set of instructions used along with the Internet Protocol (IP) to send data in the form of message units between computers over the Internet. While IP takes care of handling the actual delivery of the data, TCP takes care of keeping track of the individual units of data that a message is divided into for efficient routing through the Internet. In order to accomplish the congestion control of messages over Internet, need of high throughput and fairness new algorithms were proposed like XCP and RCP. These algorithms are widely used to make flow completion as fast as possible then earlier algorithms.

In the next section we will discuss all of these protocols in detail.

II. Transmission Control Protocol

For congestion control TCP is widely used. TCP fulfills two important functions. The primary function involves a reliable and in order delivery of bytes to the higher application layer. It builds on the unreliable, connectionless IP service, providing a service that is reliable by transmitting lost or corrupted data until the data is successfully received at the destination. It also delivers bytes in order (reorders out-of-order data and eliminates duplicates before delivering to the application process), multiplexes and de-multiplexes traffic from different processes on an end-host, and performs flow control (prevents a sender from overwhelming a receiver by specifying a limit on the amount of data that can be sent). TCP's another function is to perform congestion control and protect the network from a congestive collapse. TCP uses adaptive congestion control mechanisms that react to congestion events (such as packet loss or delay) by limiting the sender's transmission rate.

TCP congestion control works on an end-to-end basis, where each connection, before starting, begins with a query: What should be the data transmission rate for the current network path? It does not receive an explicit answer for this question, but each connection determines the sending rate by probing the network path and modulating its rate based on perceived congestion, through packet-loss and delay. The connection rate is proportional to TCP's sliding window (swnd is the limit on the amount of outstanding data in flight), which is set as the minimum of the receiver advertised window (rwnd) and of the congestion window (cwnd changes dynamically based on feedback of network conditions). To determine the congestion window, TCP employs the following mechanisms, shown in Fig. 1.1:

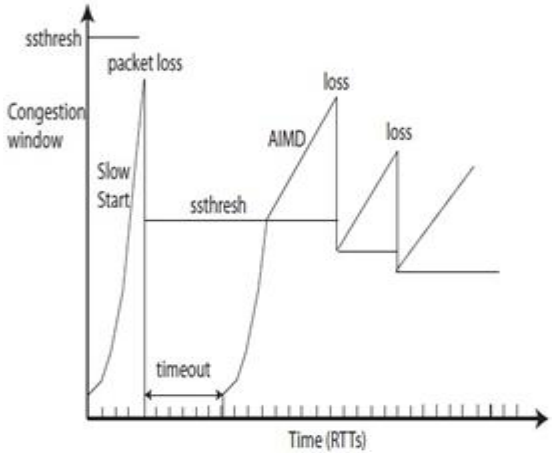

Figure 1.1: TCP's congestion control mechanisms: Slow-start and Addictive Increase Multiplicative Decrease (AIMD)[8].

Each TCP connection starts with a pre-configured small initial congestion window (no larger than 4 Maximum Segment Size (MSS)), and probes the network for available bandwidth using the Slow-Start procedure. TCP exits Slow-Start to enter the Congestion Avoidance phase, where it continues to probe for available bandwidth, but more cautiously than in Slow-Start. The four most commonly used variants are given below:

1. TCP Tahoe
2. TCP Reno
3. TCP New Reno
4. TCP SACK

Problems with TCP [4][5][6]:

1. In a high bandwidth-delay product environment TCP's additive increase of one packet per round-trip time means flows will take a long time to acquire any spare capacity.
2. TCP relies on packet-loss feedback to modulate its sending rate. The end-to-end loss probability needs to be impractically small for a TCP flow to be able to sustain a large equilibrium window, making it hard for a high-speed connection to obtain a large throughput in a high bandwidth-delay environment.
3. TCP gets confused by lossy links. It uses packet loss as a binary indicator of conges-tion, treating lossy links, such as wireless, as congested networks and under-utilizing them.
4. TCP shares bandwidth inversely proportional to the round-trip times.
5. TCP flows with long round-trip times (RTT), such as those going through satellite links, have difficulty obtaining their fair share of bandwidth on a bottleneck link.
6. TCP's Slow-Start makes short flows last much longer than necessary. Even if a flow is capable of completing within one round-trip time, TCP's Slow-Start makes it take multiple round-trip times to find its fair share rate.

III. eXplicit Control Protocol

Using the protocol for large data transfers in high bandwidth-delay networks is known as eXplicit Control Protocol [3]. XCP works by involving the routers in congestion control. The network explicitly tells the receiver the state of congestion and how to react to it. This allows senders to adjust their windows based on the precise feedback information. XCP carries the per-flow congestion state in packets, requiring no per-flow state in routers. XCP senders specify a desired throughput increase in packet congestion headers, which the routers modify to give a bandwidth increment or decrement based on the link congestion conditions. The novelty in XCP is the concept of decoupling the link efficiency control from the flow fairness control. XCP controls link utilization by adjusting its aggressiveness to the spare bandwidth and the feedback control delay, thus achieving stability and efficiency even for large bandwidth-delay product networks. It controls fairness by (conservatively) managing the bandwidth distribution among flows.

More specifically, it reclaims bandwidth from flows with rates above their fair share and distributes it to flows with lower rates. New XCP flows start with a small window size and thereafter receive a window increment/decrement. At any time, XCP flows can have different window sizes, different round-trip times, and different rates. XCP continuously tries to converge to the point where the link is efficiently utilized and all flows have their fair-share rate.

IV. Rate Control Protocol [8]

In the basic RCP algorithm a router maintains a single rate, R(t), for every link. The router "stamps" R(t) on every passing packet (unless it already carries a slower value). The receiver sends the value back to the sender so that it knows the slowest (or bottleneck) rate along the path. In this way, the sender quickly finds out the rate it should be using (without the need for slow-start). The router updates R(t) approximately once per roundtrip time (RTT), and strives to emulate processor sharing among flows. There are four main features of RCP that make it an appealing and practical congestion control algorithm.

- RCP is inherently fair (all flows at a bottleneck receive the same rate).
- RCP's flow completion times are often one to two orders of magnitude shorter than those of TCP-Sack and XCP, and close to what flows would have achieved if they were ideally processor-shared. This is because RCP allows flows to jump-start to the correct rate (because even connection set-up packets are stamped with the fair-share rate). Even short-lived flows that perform badly under TCP (because they never leave slow-start) will finish quickly with RCP. And equally importantly, RCP allows flows to adapt quickly to dynamic network conditions in that it quickly grabs spare capacity when available and backs off by the right amount when there is congestion, so flows do not waste RTTs in figuring out their transmission rate.
- There is no per-flow state or per-flow queuing.

- The per-packet computations at a RCP router are simple.

V. COMPARISON

Table 1 shows the comparison of all of the protocols mentioned in previous sections.

Table 1: Comparison of High-Speed TCP, XCP and RCP

Pros and Cons	**High speed TCP**	**XCP**	**RCP**
1. Processor sharing • ▪ Performance invariant of flow size distribution	NO	NO	YES
▪ Mix of flow : short flow completion time	NO	NO	YES
▪ Long flows:100% link utilization	YES	YES	YES
▪ Fair sharing	YES	YES	YES
2. Stable	YES	YES	YES
3. Close to zero queuing delay	NO	YES	NO
4. Efficient use of high bandwidth delay	YES	YES	YES
5. Proportional bandwidth sharing	NO	YES	YES
6. Any network condition	NO	NO	YES
7. Any traffic mix	NO	NO	NO
8. Police flows	NO	NO	YES
9. No per-flow state or queue	YES	YES	YES
10. Per-packet computation in routers	YES	NO	NO

VI. CONCLUSION

In this review, we have found that flow completion time is very important metric for congestion control [7]. Unlike TCP and XCP, RCP is a new congestion control algorithm used for fast download. It is replacement of TCP, XCP, STCP and fast TCP. We have also found that it is useful for typical flow of typical user over the Internet. Lastly RCP performs better than high speed TCP and XCP.

REFERENCES

[1] S.Floyd, M.Allman, A.Jain, P.Sarolahti, “Quick-Start for TCP and IP,” RFC4782, January 2007.

[2] M. Allman, S. Floyd, C. Partridge, “Increasing TCP’s Initial Window,” RFC 3390, October 2002.

[3] D. Katabi, M. Handley, C. Rohrs, “Internet Congestion Control for High Bandwidth-Delay Product Networks,” Proceedings of ACM Sigcomm 2002 , Pittsburgh, August,2002.

[4] S. Floyd, “HighSpeed TCP for Large Congestion Windows,” RFC 3649, December 2003.

[5] Y. Tian, K. Xu, N. Ansari, “TCP in Wireless Environments: Problems and Solutions,”IEEE(Radio)Communications Magazine, Vol. 43, No. 3, pp. S27 - S32 , March 2005.

[6] T.R. Henderson, R.H. Katz, “TCP Performance over Satellite Channels,” UCB ComputerScienceTechnical Report 99-1083, December 1999.

[7] Nandita Dukkipati ,Nick McKeown , “Why Flow-Completion Time is the Right Metric for Congestion Control”, ACM SIGCOMM Computer Communication review,Volume 36,Number 1,January 2006.

[8] Nandita Dukkipati ,”Rate Control Protocol(RCP): Congestion Control to Make Flows Complete Quickly” , A dissertation ,department of electrical engineering,October,2007.