Fairness Evaluation and Comparison of Current Congestion Control Techniques

Usman Ahmad*1, MD Asri bin Ngadi2, Ismail Fauzi bin Isnin3
Department of Computer Science, Faculty of Computing,
UniversitiTeknologi Malaysia (UTM), 81310 Johor, Malaysia.
*Corresponding author, email : usman.ahmad82@yahoo.com1, dr.asri@utm.my2, ismailfeuzi@utm.my3

Abstract

Transmission Control Protocol (TCP) is used by many applications on the Internet for the reliable data transmission. TCP does not able to utilize the available link bandwidth quickly and efficiently in High bandwidth short RTT and high bandwidth long RTT networks. Many congestion control techniques also known as TCP variants are developed to solve these problems in different network environments. In this paper an experimental analysis is done for the performance evaluation of TCP CUBIC, TCP Compound, TCP Reno and High speed TCP in term of Inter and Intra Protocol fairness by using Network Simulator 2 (NS-2). Results show that the performance of TCP CUBIC pathetically down and TCP Compound and TCP Reno shows good performance interm of protocol fairness. However, these congestion control techniques still need more improvement for the utilization of available link bandwidth in high bandwidth long RTT networks and other network resources.

Keywords: Protocol fairness, congestion control, TCP

Copyright © 2016 Institute of Advanced Engineering and Science. All rights reserved.

1. Introduction

Transmission control protocol (TCP) [1] provides reliable data communication over the network. Most of the applications on the Internet use TCP for the data communication. TCP Controls the transmission of data by increasing and decreasing the sender side sliding window also called congestion window (cwnd) and receiver side congestion window (rwnd) [2]. The size of congestion window depends upon the condition of the network. When there is no congestion over the network path, congestion window increases its size exponentially but in the presence of congestion it increases its size slowly. Sender side controls the rate of data transmission by a variable called congestion window (cwnd), which decides how many data need to send according to the network conditions. The receiver side also informs the sender side about its buffer capacity by a variable called (rwnd). With the help of these two parameters, TCP sends maximum amount of data over the network.

Many congestion control techniques are developed to solve the congestion issues. TCP CUBIC [3], TCP Compound [4], TCP Reno and High Speed TCP (HS-TCP) [5] are current congestion control techniques. TCP CUBIC does not perform well due it’s aggressive growth and reduction in cwnd in high bandwidth long distance networks and facing problem related inter and intra protocol fairness due to less reduction in size of cwndafter each packet loss [6-7].

2. Evaluation of High Speed Congestion Control Protocols

In this section performance evaluation of TCP CUBIC, TCP Compound, TCP Reno and HighSpeed TCP is carried out in term of protocol fairness. These congestion control techniques are recently tasted for the implementation in Linux operating system and new publically available in NS-2. [8]. For the performance evaluation, NS-2 simulator is used with dumbbell topology as shown in Figure 1. Two flows with FTP data are taken of each congestion control technique for simulation with short and long RTTs and with different combinations of bottleneck and link bandwidths. Complete description about simulation parameters are shown in Table 1.

Received September 14, 2015; Revised December 5, 2015; Accepted December 16, 2015
Table 1. Simulation Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variant</td>
<td>TCP CUBIC, TCP Compound, TCP Reno, HS-TCP</td>
</tr>
<tr>
<td>Bottleneck Bandwidth</td>
<td>[50, 100, 150, 200, 250, 300,350,400,450,500] Mbps</td>
</tr>
<tr>
<td>Link Bandwidth</td>
<td>[100, 200, 300, 400, 500, 600, 700, 800, 900, 1000] Mbps</td>
</tr>
<tr>
<td>Flow 1 RTT</td>
<td>100ms</td>
</tr>
<tr>
<td>Flow 2 RTT</td>
<td>[10, 30, 50,70, 90, 110, 130, 150,170, 190 ......, 310] ms</td>
</tr>
<tr>
<td>BDP Q Size</td>
<td>0.2, 0.4 ,0.6 ,0.8 ,1.0 ,1.2 ,1.4 ,1.6 ,1.8 ,2.0</td>
</tr>
<tr>
<td>Background Traffic</td>
<td>nil</td>
</tr>
<tr>
<td>Test duration</td>
<td>300 seconds</td>
</tr>
<tr>
<td>Test repetition</td>
<td>2 times</td>
</tr>
</tbody>
</table>

2.1. TCP Reno

CP Reno is proposed by Van Jacobson and an improved version of TCP Tahoe. TCP Reno is a loss based congestion control technique and uses packet losses for the estimation of link bandwidth. When there is no packet loss then it increases its congestion window size after each RTT and after each packet loss, it reduces its size by one half of its original size. This is also called additive increase multiplicative decrease [9-10].

2.2. High Speed TCP

High Speed TCP (HSTCP) [5] is an enhanced version of TCP Reno congestion control technique with large size of congestion window. In HS-TCP new increase and decrease parameter are proposed where cwnd is the size of congestion window. HS TCP uses packet loss based additive increase and multiplicative decrease (AIMD) algorithm for the detection of congestion.

2.3. TCP CUBIC

TCP CUBIC [3] is the current congestion control technique in Linux operating system. TCP CUBIC uses the cubic growth function for the congestion window. TCP CUBIC is also inspired the techniques proposed in [11] and [12]. The growth of congestion window can be calculated by Equation 1, where W is the size of congestion window, C is the scaling factor, t is the time when last congestion window reduced and K = \( \frac{1}{C} W_{\text{max}} + \beta / C \), where \( \beta \) is the congestion window decrease factor.

\[
W_{\text{CUBIC}} = C(t - K^3) + W_{\text{max}}
\] (1)

2.4. TCP Compound

TCP Compound (CTCP) [4] is proposed to solve the problem related to link efficiency and RTT fairness. It uses the two types of congestion windows. First is legacy loss based congestion window, which is used is TCP Reno (cwnd) and second is delay based congestion window (dwnd) based on TCP Vegas. Thus the congestion window of TCP Compound is the combination of two congestion windows. CTCP is the current congestion control technique in Microsoft Windows operating system.

3. Experimental Setup

Network simulator 2 (NS-2) [8] and Hamilton suite is used for the performance evaluation of above said TCP congestion control techniques interm of protocol fairness. All tests run according to the simulation parameters as shown in table 1. Network topology having six nodes is used for the performance evaluation as shown in Figure 1, by changing the link bandwidth, round trip time (RTT) and buffer queue size as describe in Table 1. S1 and S2 are the sender nodes R1 and R2 are receiving nodes and D1 and D2 are the bottleneck routers. FTP traffic is used between sender and receiver. Awk is used to get useful data from NS-2 trace files. At the end empirical data is used to analyze the performance of congestion control techniques through SPSS.
4. Performance Metric

For the performance evaluation of congestion control techniques, inter and intra Protocol fairness and discussed in this section.

4.1. Protocol Fairness

Protocol fairness is a performance metric of TCP congestion control technique, which shows how fair the TCP flows with each other regarding allocation of link bandwidth [13] [14]. There are two types of Protocol fairness, first is intra Protocol fairness and second is Inter Protocol fairness. Intra Protocol fairness means that the TCP flows having same propagation delay (RTT) between flows. Inter Protocol fairness means that flows having different propagation delay (RTT) among the flows. Protocol fairness is calculated by using Jain’s formula as shown in Equation 2

\[ f(x_1, x_2, x_3, ..., x_n) = \frac{\left( \sum_i^n x_i \right)^2}{n \sum_i^n x_i^2} \]  

5. Result and Discussion

Results regarding protocol fairness of different congestion control techniques are discussed and presented graphically below.

5.1. Protocol Fairness Analyses

In this section Inter and Intra Protocol fairness is analyzed in high bandwidth short RTT and high bandwidth long RTT networks at 500Mbps and 1000Mbps bandwidths. TCP Compound which is the current congestion control technique in Microsoft operating system and TCP Reno which is known as Standard TCP shows the highest inter and intra protocol fairness in both high bandwidth short RTT and high bandwidth long RTT networks. Figures 4, 5, 8 and 9 show, TCP CUBIC which is the current congestion control technique in Linux operating system shows lowest inter and intra protocol fairness in high bandwidth long RTT networks as compare to other congestion control techniques. In Figures 2 and 6 inter protocol fairness of TCP Compound and TCP Reno is better as compared to High Speed TCP and TCP CUBIC in high bandwidth short distance networks. However above said congestion control techniques show high intra protocol fairness in high bandwidth short RTT networks as shown in Figure 2 and 7. Figures 4, 8 and 9 show that, TCP Compound and TCP Reno show high fairness and clear performance gap among the protocol fairness of High Speed TCP and TCP CUBIC. Finally it can be concluded that all four congestion control techniques show high protocol fairness, when their flows are configured in high bandwidth short RTT networks and comparatively show low Protocol fairness in high bandwidth long RTT networks. Overall it is observed that TCP CUBIC has poor protocol fairness as compared to other state of the art congestion control techniques in high bandwidth short RTT and high bandwidth long RTT networks.
Figure 2. Inter Protocol fairness at 500Mbps

Figure 3. Intra Protocol fairness at 500Mbps

Figure 4. Inter Protocol fairness at 500Mbps

Figure 5. Intra Protocol fairness at 500Mbps

Figure 6. Inter Protocol fairness at 1000Mbps

Figure 7. Intra Protocol fairness at 1000Mbps
Fairness Evaluation and Comparison of Current Congestion Control Techniques

6. Conclusion

In this paper an experimental study is carried out in terms of inter and intra Protocol fairness of some recent TCP congestion control techniques. TCP Compound and TCP CUBIC are the default congestion control techniques in Microsoft Windows and Linux operating systems respectively. TCP Reno is the standard TCP congestion control technique and HS TCP is proposed for high bandwidth networks. The results indicate that TCP Compound and TCP Reno show better inter and intra protocol fairness as compared to TCP CUBIC and HS TCP. TCP CUBIC which is also the default congestion control technique in Linux operating system shows very poor Protocol fairness in both high bandwidth short RTT and high bandwidth long RTT networks.
RTT networks because after each packet loss the reduction parameter (β) of TCP CUBIC releases only 20% bandwidth for the other TCP flows for data transmission. The problem of inter and intra protocol fairness can be solved by tuning the reduction parameter (β) of congestion window so that after each packet loss TCP CUBIC can release more bandwidth for other incoming TCP flows.

References