Journal of Computer Science of Newports Institute of Communications and Economics
Volume 6, Issue-2015, ISBN: 2226-3683

Impact of Network Traffic Load on Performance of Queue Controllers

M. S. SHAH[1], [2]SHAFIQ UR REHMAN MASSAN[2],
LIAQAT ALI THEBO[3] AND MUKHTIAR ALI UNAR[4]

[1]Department of Computer Science Newports Institute of Communications and Economics, salehshah@gmail.com
[2]Shaheed Zulfikar Ali Bhutto Institute of Science and Technology, Karachi.srmassan@hotmail.com
[3.4] Mehran University of Engineering and Technology, Jamshoro, liaquat.thebo@faculty.muet.edu.pk

ABSTRACT

This study is performed to analyze the impact of network traffic load on performance of various Active Queue Management (AQM) controllers including traditional DropTail controller. This study represents four scenarios having identical network parameters except network traffic load. The performance of each controller is measured on multiple performance metrics.

Key Words: *DropTail, AQM, RED, Blue, PI, oDel.*

1. Introduction

Network congestion control is one of key research areas in the past, many controllers were introduced focusing on congestion control for resolving the problem. These controllers were based on some performance metrics. They performed well in the context of specified performance metrics but could not prove themselves when they were subjected to other parameters. This research is mainly focused on behavior analysis of small collection of well-known queue/AQM controllers i.e. Drop Tail, RED, Blue, Proportional Integral (PI) and Controlled Delay (Co Del) on various network traffic loads.

Four identical scenarios forming the similar bottleneck problem have been created. Each scenario is with comparable network parameters but having different network traffic load varying from a single to one thousand File Transfer Protocol (FTP) connections. The performance metrics in this research are throughput network overload, delay due to queue size, stability, required time to be stabilized, capacity to absorb Transmission Control Protocol (TCP) packet burst and the balance between different performance metrics.

2. Literature review

The network congestion, its avoidance & control remained a hot issue for researchers since the first network was established[1]. Rapid development, in the field of information technology, increased research requirements to optimize the solutions for various problems i.e. bottleneck issues[2]. Many controllers were introduced to resolve the problem based on a single or set of some selected parameters. Initially, buffers were introduced to cater the incoming packets that were greater than the link capacity. These buffers aligned the packets in queue and dropped all the packets when the buffer was full. This simple traditional scheme called Drop Tail [3] has two main problems, i.e. inequity and full queue. The inequity problem was resolved by introducing the Drop First and Drop Random Schemes.

The give full form of the abbreviation (RED) was introduced with new thought of Active Queue Management (AQM) to overcome the full queue problem of traditional queue controllers[4]. The idea behind AQM was to drop the packets before the full buffer to welcome the TCP packet bursts. The Full buffer also known as "buffer bloat" remained a controversial issue to overcome for AQMs as well [5]. The Blue controller was designed to reduce the network overload or packet loss/drop. It calculates the marking probability based on link ideal and the packet loss instead of making calculation on any type of queue sizes[6][7]. This lead to the design of the Blue controller with higher delays due to its long queues. Later, PI controller was implemented in AQM by considering its stable performance in various industries' problems [8]. This controller has performed stably in terms of its reference queue size, regardless of any network performance metric i.e. throughput, packet loss, the delay, etc. The Co Del controller was introduced to control delay only and reduce Round Trip Time (RTT)[9]. Its name was also based on Co for Control and Del for Delay[10]. It targets only to reduce the RTT but at what cost? At the cost of high packet loss and even in some cases reduced throughput.

Each controller has its own performance metric to evaluate like some focuses on queue size[8] and some on packet loss[6], some RTT [9]and some to reduce the queue size to absorb TCP packets burst [4]. Therefore, the collection of performance metrics is available to evaluate the responses and stabilities of controllers. The most commonly used are the throughput, packet loss, delay or RTT due to queue size and the capacity to absorb TCP packet bursts by maintaining short queues.

3. Methodology

In this study, five well known controllers i.e. Drop Tail, RED, Blue, PI and Co Del were selected to analyze their behavior/performance on different network traffic load varying from a single (01) to one thousand(1,000) FTP connections. Four network scenarios were established with 1, 10, 100 and 1,000 FTP

connections respectively, keeping all other network parameters identical i.e. 1 Mega Byte (1MB) bandwidth and 10 mili-seconds (10ms) link delay, in Network simulator-II (NS-II) environment as shown in Figure-I.

The performance of each controller is analyzed based on selected Key Performance Indicators (KPIs) from the wide range of the performance metrics in the field of network congestion control. To conduct this study and evaluate the responses of controller son various network traffic loads the selected KPIs are:

1. Throughput or good put or packets departed, network overload or packets dropped,
2. The delay [increased Round Trip Time (RTT)] due to waiting in queue to take turn for the departure or queue size
3. The capacity to absorb TCP burst
4. Stability and required time for stability
5. The balance between these performance metrics.

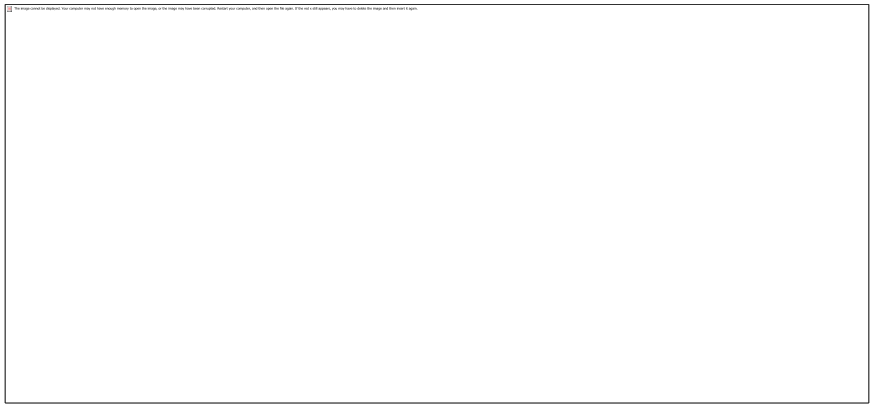

Figure 1: Network Scenarios where N=1, 10, 100 & 1000

6. Result & Discussions

The simulation periods/times of all scenarios were equal to 05 minutes/300 seconds. The performances of all controllers are

discussed with respect to their variation in performance to others. Each scenario is discussed separately according to the number from I to IV in ascending order. Scenario-I has a single FTP connection and all other scenarios (from II to IV) have the FTP connections equal to the ten multiples of their previous scenarios' FTP connections. At the end, the overall performance of all controllers is discussed.

4.1 Scenario-I: This scenario has been created to start and to identify the performance of controller on negligible traffic load of single FTP connection. The behavior of all controllers with respect to throughput, packet loss and queue size is presented in figures Fig-II, Fig-III and Fig-IV respectively.

Almost all controllers except CoDel have similar responses with high throughput, small queue size and very low packet loss. The CoDel performed worst in terms of throughput among all. Its throughput remained lowest continuously throughout the simulation. CoDel has tried to maintain very small queue size and has the largest oscillations between 0 to 9 packets against all others that maintained queue size between 25 to 26 packets. CoDel has the highest packet loss 459 packets in total throughout the simulation (from 1 to 2 packets continuously), as compared to all others with zero packet loss except a single packet loss by PI controller at approximately 145^{th} second of simulation.

Figure 2: Throughput on Single FTP Connection

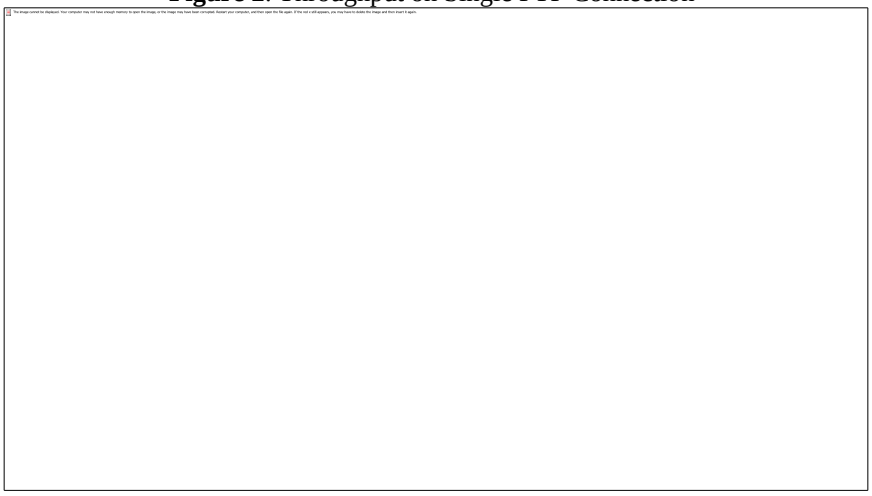

Figure 3: Packet Loss on Single FTP Connection

Figure 4: Queue size on Single FTP Connection

All controllers remained stable and capable to absorb TCP packet burst from the beginning to the end. Almost all controllers have tried to maximize throughput and minimize packet loss at the cost of higher delay except CoDel that tried to minimize the delay at the cost of lowest throughput and highest packet loss.

4.2 Scenario-II: In this scenario the network traffic load was ten FTP connections that is also very low but it was ten times higher than previous scenario. The throughputs, packet losses and queue sizes of all controllers in scenario-II are presented in Figure 5, Figure 6 and Figure 7 respectively.

The increased load did not affect the throughputs off all controllers. Almost all controllers had the same throughput except CoDel. Similar to the scenario-I, CoDel had least throughput but higher than its own in scenario-I. Throughout the simulation, Drop Tail and Blue had zero packet loss, while PI and RED had a limited packet loss of 676 and 1,255 packets respectively. CoDel had height packet loss of 29,486 packets and showed its poor performance in terms of packet loss.

Drop Tail and Blue controllers maintained highest queue sizes with approximately 475 packets. PI tried to minimize the error with its reference queue size of 200 packets. RED maintained higher queue

size with longest oscillation in its queue size till 200 seconds and highlighted its required time(approximately 200 seconds) to get stabilized. CoDel maintained the lowest queue size.

Figure 5: Throughput on Ten FTP Connections

Figure 6: Packet Loss on Ten FTP Connections

Figure 7: Queue size on Ten FTP Connections

CoDel maintained the lowest queue size among all controllers throughout the simulation. Initially PI had long queue with gradual decrease to its reference queue of 200 packets. Drop Tail had the longest oscillation in its queue size from approximately 200 packets to the maximum link capacity of 800 packets. Blue performed better than Drop Tail having a very long but stable queue size with an average capacity of 200 packets approximately, to absorb the TCP packet bursts. RED had similar behavior as discussed in scenario-II.
In this set-up, all controllers were capable to absorb TCP packet bursts. Blue performed like traditional controller Drop Tail with large queue size while CoDel has highest packet loss. RED was not stable up to 200 seconds. PI and RED performed better than others in terms of balanced performance.

4.3 Scenario-III: This is moderate situation with one hundred FTP connections. All controllers had similar response in terms of throughput. The responses of all controllers in terms of throughput, packet loss and queue size are presented in Figure 8, Figure 9 and Figure 10 respectively.

DropTail has long oscillation in packet loss throughout the simulation. The packet losses of all controller saugmented on

increased network traffic load. This indicated that the packet loss is directly proportional to the network traffic load. The behavior of Blue was different than Drop Tail unlike scenario-I and II. RED had long oscillation from packet loss of 0 to 100till first 200 seconds thereafter had low oscillation with higher packet loss than all others except CoDel.

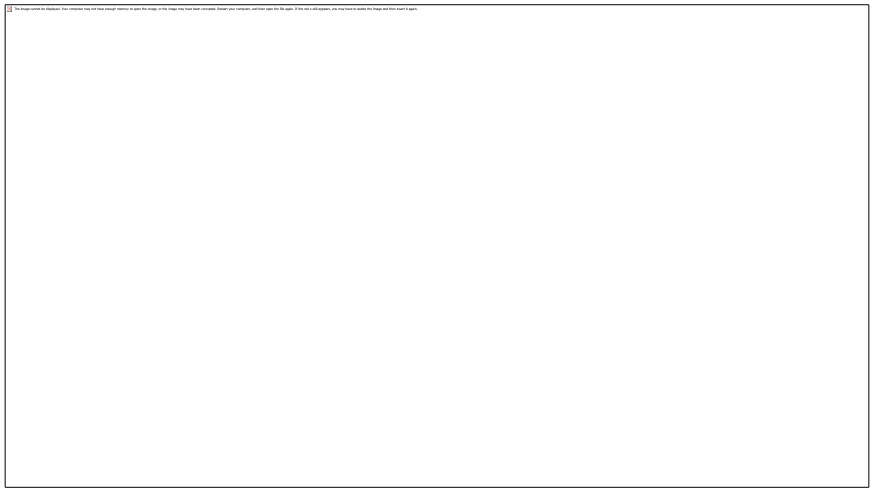

Figure 8: Throughput on One Hundred FTP Connections

Figure 9: Packet Loss on One Hundred FTP Connections

Figure 10: Queue Size on One Hundred FTP Connections

PI had the gradual increase in packet loss from 0 to 150 packets. Drop Tail, RED, Blue, PI and CoDel have dropped5,078, 31,512, 6,404, 33,691 and 85,187 packets respectively, throughout the simulation period. CoDel had the highest packet loss.

Almost All controllers had the capacity to absorb the TCP packet bursts except the traditional DropTail controller. The full queue problem of traditional DropTail controller has been clearly presented in this scenario. The Blue controller tried to minimize the packet loss in contrast to the CoDel that tried to minimize delay or RTT.

4.4 Scenario-IV: This is the last set-up with a huge network load of one thousand FTP connections. The recorded behaviors of all selected controllers in terms of throughput, packet loss and queue size are presented in Figure 11, Figure 12 and Figure13 respectively.

Throughput of all controllers remained same relative to each other. Drop Tail, RED, Blue, PI and CoDel have packet loss of 57,362, 105,500, 96,569, 117,894 and 156,150 respectively. The similarity was found in packet loss pattern of all controllers except RED's stability time with low packet loss for initial 135 seconds

approximately and increased packet loss after 135seconds approximately.

Figure 11: Throughput on One Thousand FTP Connections

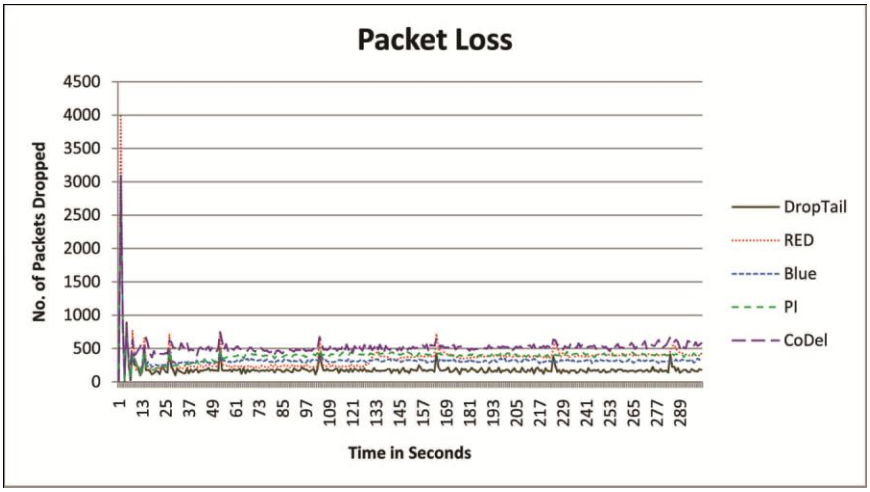

Figure 12: Packet Loss on One Thousand FTP Connections

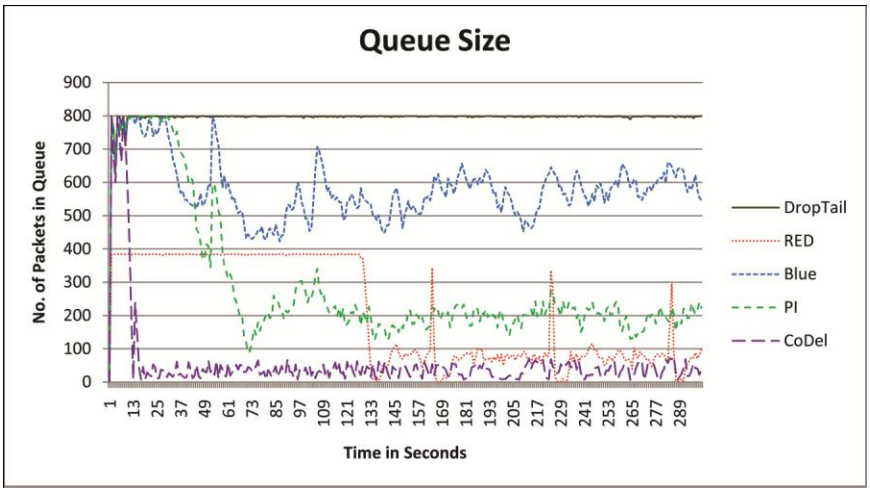

Figure 13: Queue Size on One Thousand FTP Connections

Drop Tail has maintained its queue size at the maximum capacity of the buffer. Blue left the average capacity of 200 packets to absorb TCP bursts. RED seemed to be unstable throughout the simulation especially for first 135 seconds approximately. Similar to the scenario-III, PI tried to reduce the difference between its queue size and the reference queue size of 200 packets. CoDel maintained a very short queue size irrespective of network traffic load.

All AQM controllers have the capacity to absorb TCP packet bursts except the traditional DropTail controller with almost zero capacity to welcome the TCP packet bursts.

4.5 All Scenarios: DropTail performed like other AQM controllers with single FTP connection and disclosed its full queue problem on high network traffic load. Blue performed similar to DropTail even on ten FTP connections. Blue tried to minimize the packet loss at the cost of higher delay but not at the cost of compromised throughput or without the capacity of absorbing TCP packets burst. RED needed time to stabilize but for a heavy traffic load the stability of RED was doubtful. PI always tried to maintain its queue size near to the reference queue regardless of network traffic load, maximized throughput, minimized packet loss or minimized

delay. CoDel only attempt to reduce delay or RTT at every cost factor i.e. highest packet loss and underutilized link.

The impact of network traffic load on the performance of network controller can be easily observed from the presented four scenarios. The behavior of all controllers clearly indicated the impact of network traffic load on their performance. The results proved that the network traffic load is directly proportional to the throughput, packet loss and the delay.

7. Conclusion and Future Work

Almost all controllers have been designed to overcome certain problems. DropTail was the default controller for the buffer to cater the packets that are above the link capacity along with well-known problems. RED was designed to overcome the full queue problem. Therefore, its focus is to drop the packets before buffer overflow. Blue was designed to minimize the packet loss and maximize the throughput. Therefore, it targets the particular parameters but does not care about higher delay. PI was implemented to maintain a queue size near to its reference queue size with minimum error. Therefore, it tries to reduce the error or difference between its queue size and reference queue size without considering any other nether network parameter. CoDel was designed to minimize the delay or RTT only; and does not care about its cost in terms of other network parameters i.e. link utilization, packet loss, etc.
This calls for further research in this area to introduce the state-of-the-art AQM controller that quickly stabilizes and performs the best on rapid changing and unpredicted network traffic load and other varying parameters under a wide range of performance metrics.

References

[1] C. V Hollot, V. Misra, D. Towsley, and W.-B. G. W.-B. Gong, (2001) "A control theoretic analysis of RED," in Proceedings IEEE INFOCOM Conference on Computer

Communications Twentieth Annual Joint Conference of the IEEE Computer and Communications Society Cat No01CH37213, vol. 3, no. 2, pp. 1510–1519.

[2] V. Jacobson, (1988) "Congestion avoidance and control," ACM SIGCOMM Comput. Commun. Rev., vol. 18, no. 4, pp. 314–329,.

[3] J. K. Sundararajan, F. Zhao, P. Youssef-Massaad, and M. Medard, (2004) "A modification to RED AQM for CIOQ switches," in IEEE Global Telecommunications Conference GLOBECOM 04, vol. 3, pp. 1708–1712.

[4] S. Floyd and V. Jacobson, (1993), "Random early detection gateways for congestion avoidance," IEEE/ACM Trans. Netw., vol. 1, no. 4, pp. 397–413,

[5] B. B. Ramakrishna, A. Prashant, and D. M. D. Shrinivasa, (2012), "A Survey on New Load Based Active Queue Management Mechanisms," ijser.org, vol. 3, no. 10, pp. 1–4,.

[6] W. Feng, K. G. Shin, D. D. Kandlur, and D. Saha, (2002) "The BLUE active queue management algorithms," IEEE/ACM Trans. Netw., vol. 10, no. 4, pp. 513–528,.

[7] W. Feng, D. Kandlur, D. Saha, and K. Shin, (1999) "BLUE: A new class of active queue management algorithms," Ann Arbor, pp. 1–27,.

[8] C. V. Hollot, V. Misra, D. Towsley, and W.-B. Gong, (2001) "On designing improved controllers for AQM routers supporting TCP flows," in INFOCOM Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. 2001, IEEE (Volume:3), vol. 3, pp. 1726–1734.

[9] T. Høiland-jørgensen, (2012) "Battling Bufferbloat: An

experimental comparison of four approaches to queue management in Linux," Roskilde University,.

[10] K. Nichols and V. Jacobson, (2012) "Controlling queue delay," Communications of the ACM, vol. 55, no. 7, pp. 42–50, Jul-.

[11] S. Floyd, R. Gummadi, and S. Shenker, (2001) "Adaptive RED: An algorithm for increasing the robustness of RED's active queue management,".

Ye Page Blank Rahe Ga Printing k Wakt is page ka number or header par jo line likhi

he usay delete kar dain

Journal Articles Right side se shuru hote hain is liye is page ko blank hi rakhna he