VOL. 12, NO. 3, FEBRUARY 2017 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

THE NETWORK PERFORMANCE OVER TCP PROTOCOL USING NS2

Ammar Abdulateef Hadi, Raed A. Alsaqour and Syaimak Abdul Shukor
School of Computer Science, Faculty of Information Science and Technology, University Kebangsaan Malaysia, Bangi, Selangor, Malaysia
E-Mail: alshida1973@yahoo.com

ABSTRACT

Transmission Control Protocol TCP is a congestion control transport protocol, utilizing by the applications which asks ensuring delivery. By the most common characteristics for TCP, congestion control, acknowledgment resend packet loos, where are prohibit from network collapse and trying to increase the performance of network. These attempts standing against a multiple situations which mark its presence in the network such as weakness of data flow which is happened in some point of the network such as bottleneck, however, sometimes bottleneck determining the efficiency of the network, which includes throughput, delay, packet delivery ratio PDR, and packet loss ratio PLR. Despite the all attempts, including the bandwidth increasing, packet size manipulation, and increase the queue size, that done on physical links of the network, without any tangible improvement unless a bottleneck treatment. In this paper, we study the performance metrics of TCP protocol including throughput, PDR, and PLR. Network simulation (NS2) version 2.35 is used to study the performance of TCP protocol with FTP network traffic load. The performance of TCP was tested by manipulating the network metrics like link bandwidth, packet size, and queue size. Results of the study showed that the high throughput performance done by increasing the packet size and also by providing high link bandwidth to a certain extent and begin to decline because of insufficient resources in certain connections, such as a bottleneck connection. The PLR is positively affected by increasing the queue size, whereas, the increasing of packet size is adversely affected on PDR.

Keywords: TCP, FTP, bottleneck, traffic load, throughput, packet size, queue size, congestion control, connection bandwidth.

INTRODUCTION

In recent several years, the computer networks are widely used by the companies and governments, and the first require of that kinds of users is reliable transmission of data, TCP is a transport layer protocol that is on a large scale of using by applications like FTP on the networks, and characterized by reliability[1]. TCP by congestion control algorithm identifies the performance of the protocol in parallel with the acknowledgment mechanism that prevents the duplicate and loos packets [2].This paper aims to study TCP protocol in wired networks, and benefit from the relative results for similar research and appropriateness in their work environment. We study the performance metrics of TCP protocol. Network simulation (NS2) version 2.35 is used to study the performance of TCP protocol FTP network traffic load. The performance of TCP was tested by manipulating the network metrics like connection bandwidth, packet size, and queue size. This paper is arranged as follows: sections 2 portray the TCP's Theoretical background section 3 Shows the results and evaluation and the conclusion was reported in section 4.

TCP's THEORETICAL BACKGROUND

The important treatment that TCP gave to the computer network against the congestion it was the mechanism of congestion control, which is keep the number of packets that send to the network under the ability of the network to process. However, as in [3]TCP's congestion control produces the congestion window, *cwnd,* and receiver window, *rwnd.* And the actual window size of sender, *sw*, was obtained by (1)

$$sw=\min(rwnd, cwnd) \quad (1)$$

rwnd is used to restrain the sender from transmit packets over the ability of the receiver, and *cwnd* is to preventing the sender from sending data more than the absorptive capacity of the network in the current load conditions. The *cwnd* is change to ensure the appropriate size of the congestion window that prevent the congestion and it is change when there is a signal referring to packet loos which be detected via time-out or duplicate ACK[4], also the sender uses that tow event as a notice for congestion in the network. For every connection there is are transmission time-out RTO, if RTO is expires and there is no ACK that Confirms receiving packets, the TCP move the segment to the front queue that is mean resend packet before RTO expire, RTO is updated by Round-Trip time RTT. When the packet has been lost, the receiver continue sending ACKs but with the same sequence, the sender there is a packet loos happened. TCP avoided the congestion by three algorithms, in slow start, congestion avoidance, and fast recovery. In the In slow start the *cwnd* increasing exponentially every RTT till it reaches a slow-start threshold value *ssthresh.* After that start the congestion avoidance algorithm in which the *cwnd* increase linearly, after three ACKs event happened will start the third algorithm and the value of *ssthresh* obtained by (2),

$$ssthresh=cwnd / 2 \quad (2)$$

The new value of *cwnd* obtained by (3)

$$cwnd = ssthresh + 3 \quad (3)$$

in Taho TCP treats the both signs of congestion, time-out and three duplicate in the same way, the cwnd goes to 1 and ssthresh goes to cwnd / 2, then restart with In slow start algorithm. In Reno TCP treats the both tow signs differently, after time-out TCP moves to slow star algorithm, after three duplicate ACKs TCP moves to third algorithm fast recovery[5, 6].

SIMULATION ENVIRONMENT

To study the performance of TCP ,we used ns-2.35 network simulation [7]which is developed for simulating wired network environment with physical, data link, and MAC layer, the simulation parameter as in Table-1.

Table-1. Simulation parameter.

Parameter	Value
Simulator	Ns-2 (v2.35)
Simulation time	200 s
Number of nodes	6 nodes
Routing protocol	TCP, UDP
Traffic model	FTP, CBR
Queue type	Drop-tail
Bandwidth (source to bottleneck link)	2 Mbps
Bandwidth (bottleneck-link)	10 Mbps
Bandwidth (bottleneck to destination like)	0.5 Mbps
Delay (source to bottleneck link)	10ms
Delay (Bottleneck-link)	100ms
Delay (bottleneck to destination like)	40 ms, 50ms
Packet size	We trace the packet size 250-3500 byte.
Queue size (bottleneck link)	20
TCP type	Tahoe or Reno TCP
Packet size	1460 or 512 (in bytes)
Starting times	CBR 0.1, FTB 10.0
Stop time	CBR 124.5, FTB 60.0

PERFORMANCE METRICS

We use network performance throughput, packet delivery ratio, packet loos ratio by our simulation to test TCP protocol. For test the performance we use the metrics given below:

- **packet delivery ratio (PDR)**

Packet delivery ratio *pdr*is the ratio of number of packets that received by the destination to the number of packets that send from the source, the *pdr* calculation as (4).

$$pdr = \frac{number\ of\ pakets\ received}{number\ of\ packets\ send} \times 100 \quad (4)$$

- **packet loos ratioPLR**

Is the Percentage of the delivered packets*dlvpck* to number of the total packets that send by the source, as (5). The delivered packets are the packets that success arrived to the destination, and is calculate as (6).

$$plr = \frac{dlvpck}{number\ of\ packets\ send} \quad (5)$$

$$\text{dlvpck} = \text{number of packets send} - \text{number of packets received} \quad (6)$$

- **Throughputs**

The throughput is the amount of data that received by destination, is calculated as (7).

$$data\ amount = packet\ size - header\ size \quad (7)$$

The average of the throughput*avrthr*is calculated by dividing the amount of data *data amount* over the duration time*d time* for each application, as (8).

$$avrth = \frac{data\ amount}{dtime} \quad (8)$$

The duration *dtime* is the time that the application is running, is calculated as (9).

$$dtime = stop\ time - start\ time \quad (9)$$

VOL. 12, NO. 3, FEBRUARY 2017 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

NETWORK METRICS

The network parameter setting during the simulation is traffic load, the amount of data packets transferred per second through the wired network.

THE RESULT OF SIMULATION

Throughput

The throughput is one of the performance metrics that depended on several network metrics. We will take the packet size and bandwidth, the simulation result show that the effecting of increasing packet size is directly proportional to increasing the throughput,The increase in packet size means increasing the amount of data delivered to the destination, In the event if this network not experiencing a high packet loos ratio, we noted that when the packet size 1500 byte, the amount of data was decreased decrease to level less than when the packet size was in case 1250 byte, 1000 byte even with the case when the packet size was 750 byte, This inappropriate case is due to the packet loos that happened when the packet was dropped because of Non-adequacy of nodes queues, bottleneck, expire RTO, RTT. Figure-1 Show the throughput affect by the packet size.

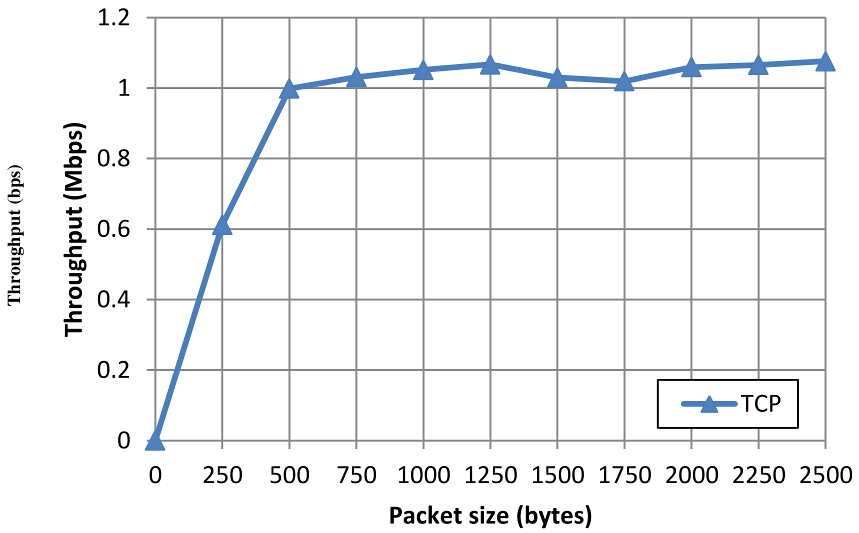

Figure-1. Throughput by manipulating the packet size.

That is leading us to a point, not all the maximizing cases of the packet size resulted an increase in throughput in network over TCP because the, the big size of packet is mean a big amount of data and if some packets lost which require TCP to resend thepacket loos, adding to ACKs from receiver all of that will consume the bandwidth consequently decreasing of data amount delivering to receiver, and decreasing the throughput. The bandwidth is another factor from the network metrics that have extrusive effect on throughput, referring to Figure-2. We will take the bottleneck as a point of the network to study the effect of bandwidth on the throughput. The increasing in the bandwidth generally leads to the observed increase in the amount of delivered data, observing the Figure-2. We notes the duplicate increase in throughput when the bandwidth increase from 0.3 (Mbps) to 0.6 (Mbps). And this improvement of throughput is continuing until it standing in front of an obstacle, it is the next node. We observed that the increase in throughput stopped when the bandwidth be 0.9 (Mbps), and throughput freeze at its stage without any improvement whatever the increase in bandwidth.

VOL. 12, NO. 3, FEBRUARY 2017 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

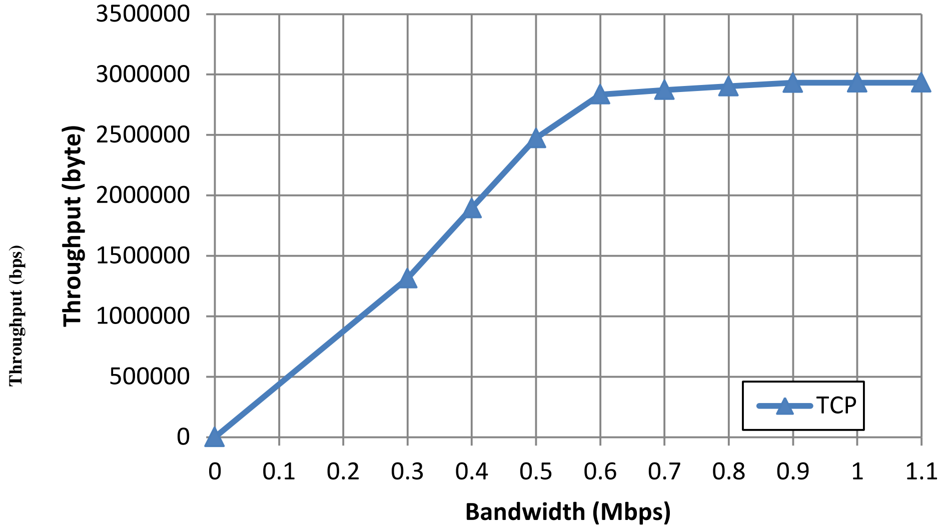

Figure-2. Throughput with different bottleneck bandwidth.

After manipulating the bandwidth of the link in the next link after the bottleneck we observe a significant improvement in the situation of throughput. That deals to the improvement of the network performance must do an enhance with all network, because if doing a network metrics upgrade to the some links in the network that makes the other links without upgrade be a bottleneck, and that will prevent the network performance in general.

Packet loos ratio PLR

The most important feature of TCP that is making it widely use, is the reliability[8]. The PLR in the networks is widespread, especially in network suffering from congestion. However, in our research we study the behaver of TCP when manipulate the queue size of node. The default value of the queue in NS2 topology is 50 packets. In our simulation, we observe the effect of queue size is relatively low, as Figure-3, only when the queue size less than 20 packets the packet loos ratio is adversely affected slightly. That deals to, the improvement of packet loos ratio is not depending on the queue size strongly.

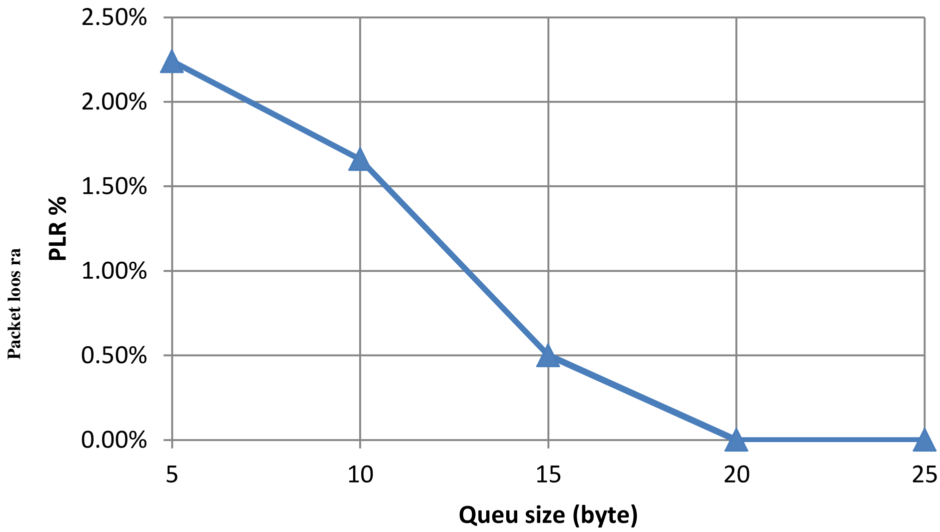

Figure-3. PLR with different queue size.

Packet delivery ratio PDR

We study the PDR as one of the performance metrics and the extent effected to packet size. We notes that when increase the packet size that make decreasing in number of packets because of distributing the number of bytes by fewer of packets. However, the increasing of the packet size in network suffering congestion situation that will adversely affected to throughput.In our simulation, we

VOL. 12, NO. 3, FEBRUARY 2017 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

observe that the increasing of packet size was resulted a decreasing in packet delivery ratio as in Figure-4. That is deals to the low length packets are delivered more than the high length packets. The high length packet need a high bandwidth link, as we see the throughput is increase in the network with high bandwidth.

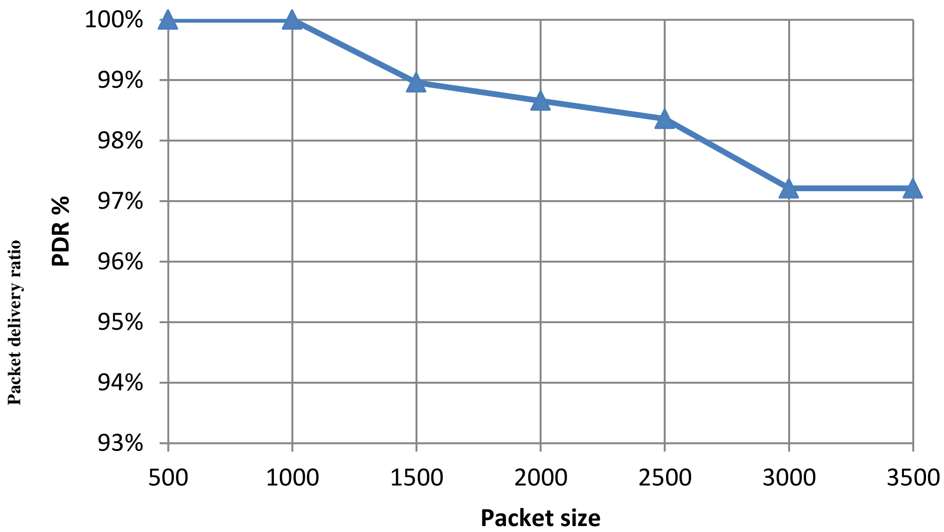

Figure-4. PDR with different packet size.

CONCLUSIONS

This paper evaluated the performance of TCP protocol using ns-2.35. Our study is based on network parameter: traffic load using FTP application. We conclude that, the network performance as a throughput depended on the high bandwidth links, the high bandwidth lead to a high result of throughput. The enhancement of the some point in network like a bottleneck is improving the performance of the network to a certain extent. Another observation is the improvement of packet loos ratio is not depending on the queue size strongly, and the low length packets are delivered more than the high length packets.

REFERENCES

[1] Kelly T., Scalable TCP: 2003. Improving performance in highspeed wide area networks. ACM SIGCOMM computer communication Review. 33(2): 83-91.

[2] Akase K., Obata H., Murase T., Hirano Y. and Ishida K. Setting methods of a slow start threshold in TCPs for guaranteeing a bandwidth over wireless LAN. IEICE Transaction on Communications. 93 (153-165).

[3] Allman M., Paxson V. and Blanton E. 2009. TCP congestion control. pp. 2070-1721.

[4] Johari R. and Tan D. K. H. 2001. End-to-end congestion control for the Internet: Delays and stability. Networking, IEEE/ACM Transactions on. 9(6): 818-832.

[5] Mathis M., Semke J., Mahdavi J. and Ott T. 1997. The macroscopic behavior of the TCP congestion avoidance algorithm. ACM SIGCOMM Computer Communication Review. 27(3): 67-82.

[6] Moraru B., Copaciu F., Lazar G. and Dobrota V. 2003. Practical analysis of tcp implementations: Tahoe, Reno, Newreno.

[7] Henderson T. R., Lacage M., Riley G. F., Dowell C. and Kopena J. 2008. Network simulations with the ns-3 simulator. SIGCOMM demonstration. p. 14.

[8] Tickoo O., Subramanian V., Kalyanaraman S. and Ramakrishnan K. 2005. LT-TCP: End-to-end framework to improve TCP performance over networks with lossy channels. Springer.