International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 06 Issue: 07 | July - 2022 Impact Factor: 7.185 ISSN: 2582-3930

An Efficient Modified Slow-Start Mechanism for TCP Performance Enhancement

Maya Diwakar[1], Anita Yadav[2]

[1]*Computer Science and Engineering Department & Harcourt Butler Technical University, Kanpur, Uttar Pradesh*
[2]*Computer Science and Engineering Department & Harcourt Butler Technical University, Kanpur, Uttar Pradesh*

---***---

Abstract - TCP protocol is widely used protocol in the transmission of data. Today is the world of digitization and all the works of government and individual are done online using the internet. While using the internet basically, we are sending and receiving the data. TCP is generally used for the wired network as well as for the wireless network also. TCP has its variants like TCP RENO, TCP VEGAS, TCP NEW RENO. We know that TCP is used everywhere in the network whenever there is a need of data transmission. Data packets are lost due to congestion and many other reasons like connection failure and limited bandwidth, random loss, handoff etc. This paper proposes the modified slow-start mechanism for increasing the performance TCP at the transport layer.

***Key Words*:** Wireless Networks, TCP, TCP RENO, Slow-Start, Throughput.

1.INTRODUCTION

TCP (Transmission Control Protocol) is used in both the wired and wireless networks. TCP is having many features other than the UDP (User Datagram Protocol). TCP protocol is used for the ordered and secure data delivery [3,4]. In TCP there is surety of Data delivering that the data is delivered and the acknowledged because in the present scenario sequencing delivery of the data is very much important because we are using many chatting applications in which the sequencing and sequence delivery of the data is the first priority otherwise the meaning of the message will change.

As we know that whenever the data get lost, we first thought that the data is lost due to the congestion. In the congestion control methods, there are three method slow-start, congestion detection and congestion avoidance.

Slow start algorithm is gradually increasing the number of packets on sender side [1]. In the slow start algorithm, there are two variables to control the flow of the data at the sender side and receiver side. First is the congestion window (cwnd) and the second is receiver advertising window (rwnd). Congestion window is related about the number of packets send by the sender before receiving the acknowledgement. The rwnd is the number of packets received by the receiver buffer. Sender has the controlling power to control the sending rate of the packets by setting the minimum cwnd and rwnd values.

2. RELATED WORK

Hanna A. Torkey[6] they were used three performance metrics to evaluate TCP protocol performance. The three metrics were throughput, fairness and losses. They used GT-ITM for realistic toplogy generator and NS-2 as a network simulator. After the Simulation result shows that TCP NEW RENO performed good among all three variants.

Ivan Petrov[7] they proposed an algorithm in which they were worked on the congestion window size in the slow start algorithm congestion window is growing by 1 after every RTT but here they were proposed three new algorithm for slow-start with the maximum cwnd size was 100. In the first algorithm cwnd is increased by 3 and upto 50 cwnd size then increased by 2 upto 75 cwnd and after the 100 cwnd it was using LSS algorithm. In second algorithm cwnd was increased by 2 after every RTT upto 25 cwnd and 25 to 100 cwnd size it increased by 1after every RTT and after the max cwnd that is 100 is uses LSS algorithm. In the third algorithm upto 50 cwnd, cwnd is increased by 2 and 50 to 100cwnd was increased by 1and after that is uses LSS algorithm. Result shows that the second algorithm was performed well due to the fast increment of cwnd at the starting and slow down after some time.

Xiaojing He [8] in this paper authors described the methods to enhancing the TCP performance over the wireless network like pure end-to-end Protocols, split-connection and link layer protocols. They also discussed about the measuring features of TCP Performance are deploy ability, Compatibility, Encryption and Flexibility.

Xiang Chen [9] and other given the broad survey on the TCP performance enhancement over the mobile an Adhoc networks using the various methods like TCP with feedback, TCP without feedback, TCP with two-layer enhancement. TCP with feedback and TCP without feedback having various methods which are discussed in the paper.

Saedi [10] and other revisited their own work CERL that is TCP Congestion Control Enhancement for Random Loss. They used the dynamic RTT. Threshold value of RTT was dynamic. Here the queue length was used for the differentiation of random loss and congestion. Simulation was done in ns-2 that shows that the CERL+ performs very good.

3. SLOW-START ALGORITHM

Slow-Start plays the important role to avoid the congestion in the network [5]. Slow-Start preempts the sending rate of the packet setting the value of cwnd. Initially TCP set the value of cwnd=1 segment. Value of cwnd is increased by 1 after every ACK (acknowledgement) received. Slow- Start increases the cwnd exponentially after every ACK received and it will stop when the value is equal to slow-start threshold (ssthresh). And the value of cwnd is half when the packet loss is detected. cwnd set to 1 again. Then the TCP sender again start the slow-start algorithm.[2]

For example:

Initially cwnd=1

After 1 RTT, cwnd = 2^(1) = 2

2 RTT, cwnd = 2^(2) = 4

3RTT, cwnd = 2^(3) = 9

……..

……..

n RTT, cwnd = 2^(n)

In the above example shows how the congestion window size is going to be increased exponentially.

4. MODIFICATION IN SLOW-START

For avoiding the congestion control slow start phase start transmission by taking cwnd =1. After every acknowledgement cwnd is increased by 2 segments. This is not growing exponentially and ends when it reaches to the slow start threshold value and again starts with cwnd= 1.

Explanation of cwnd as below-

Initially cwnd=1

After 1 RTT, cwnd = 2^(1) = 2

2 RTT, cwnd = 2^(3) = 8

3RTT, cwnd = 2^(5) = 32

……..

……..

n RTT, cwnd = 2^(n+2)

NowHere is the new method to set the slow start threshold value in the initial slow start phase. When the congestion window is growing in G.P.(Geometric Progression) then the sending rate of the packet is fast.

5. SIMULATION DESCRIPTION

There are various parameters are used in the simulation of new Slow-Start algorithm. In Table-I it shows the list of parameters.

TABLE I. Network Simulation Parameters

Methods	Values
Channel Type of network	Channel, Wireless channel
Radio-Propagation model	Propagation, Two-ray round
Network interface type	Phywirelessphy
MAC type of network	Mac/802.11
Link Layer Type of network	LL
Number of Nodes in wireless network	10
No of connections in network	5-10
Source Type	TCP (Reno, New Reno, Vegas)

6. PERFORMANCE PARAMETERS

We used various parameters like throughput, send packets, received packets and dropped packets to understand the behavior of new slow-start algorithm on various TCP types like TCP TAHOE, TCP RENO, TCP NEW RENO, TCP VEGAS.

7. SIMULATION RESULTS

I) NO OF DROP PACKETS

The drop packets are calculated by the difference of packets send by sender and packets received by receiver. We show the dropped packet of various TCPs after the modification of algorithm.

TABLE II. NUMBERS OF PACKET DROP BEFORE MODIFICATION(SLOW-START)

TCP TYPES	NO OF DROPPED PACKETS
TCP TAHOE	158
TCP RENO	402
TCP NEW RENO	255
TCP VEGAS	214

TABLE III. NUMBERS OF PACKET DROP AFTER MODIFICATION(SLOW-START)

TCP TYPES	NO OF DROPPED PACKETS
TCP TAHOE	163
TCP RENO	91
TCP NEW RENO	207
TCP VEGAS	1023

As we see from the above packet drop table TCP RENO and TCP NEW RENO is giving the best result after the modification in slow-start algorithm.

II) THROUGHPUT

Throughput is nothing but the measurement of the packets that pass through the network in single unit of time. Here we calculate the sender side throughput after modified slow-start algorithm.

TABLE IV. Throughput Before modification (Slow-Start)

TCP TYPE	THROUGHPUT
TCP TAHOE	281.81
TCP RENO	357.24
TCP NEW RENO	297.25
TCP VEGAS	319.97

TABLE V. Throughput After modification (Slow-Start)

TCP TYPE	THROUGHPUT
TCP TAHOE	78.10
TCP RENO	50.92
TCP NEW RENO	332.15
TCP VEGAS	444.57

According to the both throughput table we can see that the TCP VEGAS and TCP NEW RENO having the maximum throughput than another version of TCP.

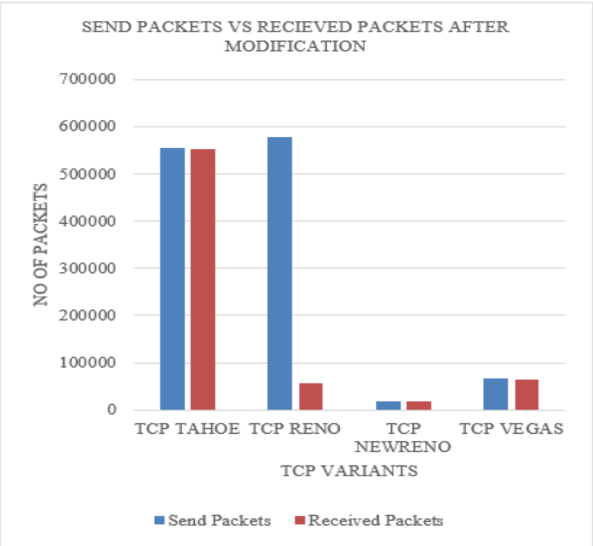

Fig. I. Numbers of Send Packets vs Received Packets Before modification (Slow-Start)

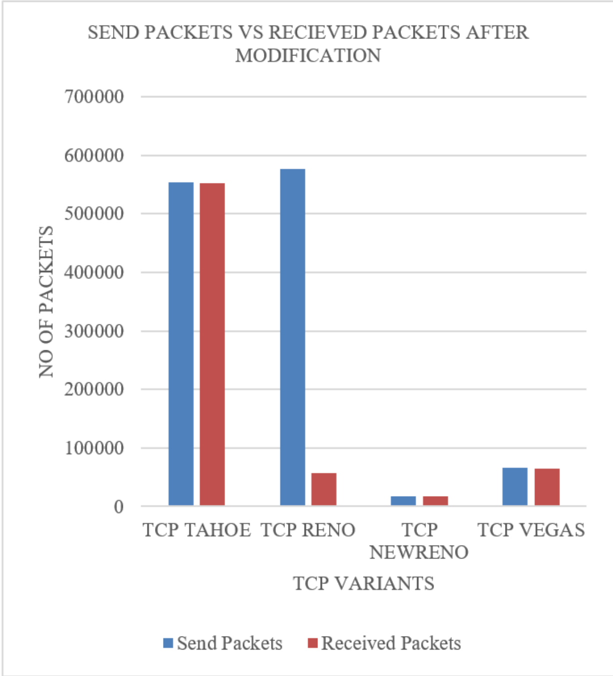

Fig I. Numbers of Send Packets vs Received Packets After modification (Slow-Start)

8. CONCLUSION

In this paper we explained the modified slow- start algorithm and its effect on the various TCP versions. By increasing the three-congestion window at every RTT (Round Trip Time). Size of cwnd is growing very fast. It is showing that the modified slow start algorithm is performing good for TCP VEGAS for the throughput. TCP VEGAS giving the maximum throughput that is 444.57. So, it is clearly showing that the modified slow-start is good for the TCP VEGAS.

FUTURE WORK

In the future it will be applied on other metrics in the network. In future, the cross-layer effect will used for the slow-start algorithm. Work can be done with the routing protocol to check the throughput of the TCP.

ACKNOWLEDGEMENT

We thanks to the all the references for their valuable suggestion for my research paper.

REFERENCES

1. Al-Hasanat, Mohanad, Kamaruzzaman Seman, and Kamarudien Saadan. "Enhanced TCP Westwood slow start phase." *Transactions on Networks and Communications* 2.5 (2014): 194-200.
2. Morshed, Md, Meftah Ur Rahman, and Md Islam. "An Empirical Study on variants of TCP over AODV routing protocol in MANET." *arXiv preprint arXiv:1109.0931* (2011).
3. Volkov, A. S., et al. "Applying of TCP-based protocols for mobile ad-hoc networks with PN signals in NS-3." *2018Systems of Signal Synchronization, Generating and Processing in Telecommunications (SYNCHROINFO)*. IEEE, 2018.
4. Bansal, Aruna, and Mridula Singh. "Enhancing MANET's performance: A transport layer solution." *2012 2nd IEEE International Conference on Parallel, Distributed and Grid Computing*. IEEE, 2012.
5. Francis, Breeson, et al. "Techniques for enhancing TCP performance in wireless networks." 2012 32nd International Conference on Distributed Computing Systems Workshops. IEEE, 2012.
6. Torkey, Hanaa A., Gamal M. Attiya, and I. Z. Morsi. "Performance Evaluation of End-to-End Congestion Control Protocols." *Minufiya Journal of Electronic Engineering Research (MJEER)* 18.2 (2008): 99-118.
7. Petrov, Ivan, and Toni Janevski. "Novel Slow Start Algorithm." *International Journal of Hybrid Information Technology* 6.5 (2013): 297-332.
8. He, Xiaojing, et al. "TCP performance evaluation over wireless networks." *Canadian Conference on Electrical and Computer Engineering 2004 (IEEE Cat. No. 04CH37513)*. Vol. 2. IEEE, 2004.
9. Holland, Gavin, and Nitin Vaidya. "Analysis of TCP performance over mobile ad hoc networks." *Wireless Networks* 8.2 (2002): 275-288.
10. Saedi, T., El-Ocla, H. TCP CERL+: revisiting TCP congestion control in wireless networks with random loss. *Wireless Netw* **27,** 423–440 (2021). https://doi.org/10.1007/s11276-020-

BIOGRAPHIES

She received M. Tech in Intelligent System from IIIT Allahabad, Prayagraj. B. Tech from State University of Uttar Pradesh. She is a research scholar of HBTU, Kanpur and working in the field of Networking.

She is Assistant Professor in HBTU, Kanpur. Her area of interest is networking, security and machine learning. Her research interests include Mobile Adhoc Networks, Computer Networks and Mobile Computing.