TCP RENO, SACK AND VEGAS PERFORMANCE ANALYSIS

B. Arunakumari[1]and Dr. P. Chennareddy[2]

[1]Department of Computer Science and Engineering
JNTUA College of Engineering, Pulivendula, Andhra Pradesh, India
[2]Professor of CSE Department
JNTUA College of Engineering, Pulivendula, Andhra Pradesh, India

ABSTRACT

TCP is the most widely used transport protocol by the Internet community. It is used by applications e-mail, WWW, and file transfer. TCP provides reliable transfer of stream of bytes between a pair of processes.There are different TCP implementations and each modifies the basic congestion control mechanismslightly. The sender reaction to congestion is different. But all the TCP implementations are originally designed and tuned to wired networks.Their working in wireless networks is an area of interest to researchers. Mobile networks like adhoc networks pose more challenges to the researchers. In this paper we do the performance analysis of TCP implementationsTCP Reno, TCP SACK and TCP Vegas in adhoc networks.Different mobility scenarios are considered. The metrics considered are Throughput, End-End delay, Packet delivery ratio and Jitter. NS-2.35 is used as simulator.

KEYWORDS

Adhoc network, TCP Reno, Sack, Vegas, performance.

1.INTRODUCTION

In mobile wireless networks, nodes transmit data without establishing a connection. The main benefit of mobile wireless networks is access to the application from anywhere and at anytime. The wireless network is preferred because the cost of wired network is more due to the high cost of Ethernet cables, hubs and switches.Wireless networks based on IEEE 802.11 [1] series of standards allowseamless access to services of the network. IEEE 802.11 supports infrastructure mode and adhoc mode. The adhoc mode of operation requires no additional infrastructure including access point and hence can be deployed on the fly without any additional cost.

TCP [2] is a transport protocolused by the applications which require guaranteed deliveryand is suitable for applications which require high reliability and which can tolerate variation in end-end delay and jitter. TCP ensures the order of the packets transmitted. TCP connection involves negotiating the parameters and establishing the connection. Data transfer begins after establishing the connection and terminated after the data transfer is completed.

TCPtransfers a stream of bytes by groupingthem into segments. TCP handles packet loss, packet errors, and also duplicate packets. A sequence number is assigned to each byte sent by the sender and receiver acknowledges the data received by using the sequence number of the byte last received successfully. Non-receipt of the ACK results in retransmission. Retransmission is controlled by the retransmission timers maintained by TCP. TCP implementations differ in their

DOI: 10.5121/ijci.2015.4211

reaction to reception of ACKs and also retransmission timer expiration. Checksum is used to detect errors in the data transmitted. TCP retransmits data segments in error. But the applications requirement to errors is different. Traditional applications like file transfer and e-mail require high reliability (no packet loss and errors) and hence use TCP. But multimedia applications can tolerate errors to a limited extent and retransmitted packet is of little use.

TCP Reno [3] is a modified version of the basic TCP protocol. TCP Reno requires immediate receiving of acknowledgment whenever a segment is received. Reno uses Fast Retransmission mechanism. When it receives 3 duplicate acknowledgments then it assumesthat the segment was lost and retransmits the lost segment without waiting for the retransmission time-out.

Vegas [4] isone of the stable implementations of TCP. Reno detects congestion and reacts to congestion by using congestion control mechanism. But Vegas tries to detect the occurrence of congestion before it happens. Vegas is capable of detecting congestion before packet loss occurs. Vegas calculate the expected rate and actual rate. The actual rateand estimated rateare compared. If the difference is large, then there is chance for congestion and hence congestion window size has to be decreased. If the difference is small then the congestion window size is increased. TCP SACK [5] is a congestion control mechanism which can selectively acknowledge the missing segment and hence selectively transmit the missing segment.

The performance of TCP depends on TCP implementation used. TCP Reno, Sack and Vegas exhibits different behaviour when congestion occurs. This paper compares and analyzes the TCP implementations Reno, Sack and Vegas by varying Pause time, Speed, and Data rate in wireless networks. The scope of this paper is to analyze the existing TCP implementations in wireless network. It doesn’t consider the interoperation of the different TCP implementations.

2. LITERATURE SURVEY

BalveerSingh [6], presented a paper on the performance of various TCP implementations in order to identify a best TCP Variant in the network. According to that, TCP Vegas is better than other TCP implementations. It concludes that TCP Vegas results in higher throughput compared to other TCP implementations. Sack is fair to Reno and it compets with Vegas. Vegas easily adaptsto the changing bandwidth.

Hala A. ElAarag and Mostafa A. Bassiouni[7], comparesTCP Reno, New Reno, and Sack. Topology is made of three nodes: a fixed host, a base station and the mobile host. TCP Reno shows high throughput, least transfer time and lower packet delay than Sack.

SandeepSandhu, AnirudhMenon [8],investigated TCP implementations performance due to node’s mobility and number of nodes. AODV is used as a routing protocol. From the results it is concluded that Vegas has stable End-End delay, performs far better than Reno when Packet delivery ratio is considered. Whennumber of nodes is varied, Throughput is high for TCP Vegas, and when mobility is varied TCP Vegas and Reno give better performance.

Dr. NeerajBhargava Dr. RituBhargava Manish Mathuria[9], performed analysis of different congestion control and avoidance algorithms. They compared TCP Tahoe, Reno, and Sack. TCP is evaluatedunder high and low mobility conditions. TCP protocols ensure better data transfer, reliability and congestion control.

Mohit P. Tahiliani, K. C. Shet, and T. G. Basavaraju[10], compared High-Speed TCP Variants in Multi-Hop Wireless Networks. High speed TCP variants consider round trip time(RTT) as indication of congestion. Apart from other metrics, new metric expected throughput is used for comparison.HSTCP, STCP, CUBIC and CTCP performance is studied in multi-hop wireless networks. They concluded that more routing overhead reduces the overall throughput of the network. Due to this increased overhead the number of collisions will increase and the performance of network is worse. The performance of TCP mainly depends on routing protocol.

MadihaKazmi, AzraShamim, NasirWahab, and FoziaAnwar[11], studied the behavior and performance of TCP over wired IP and MPLS(Multi Protocol Label Switching) network by TCP variants. Simulation was done by considering different number of flows such as single flow, two, four and eight flows for CBR traffic. Simulation of TCP Reno, Sack, Tahoe, New Reno, Vegas was done and throughput, delay, number of packets sent, received and lost are measured.TCP Vegas exhibits good performance and throughput also improved for Vegas when compared to other variants. The Performance of all TCP variants gradually decreases with increase in number of flows. In IP and MPLS network, TCP Vegas is a very good and reliable choice to use in real-time data transmission because of its stable jitter,and free end-to-end delay behavior.

M.A. Rahman, A.H.Kabir, K. A. M. Lutfullah, Z.Hassan and M. R. Amin[12], calculated the performance of TCP Tahoe, TCP Reno, TCP Newreno and TCP Vegas. From the results it is concluded that TCP Vegas dynamically increases or decreases data transmission according to the window size of sending packets where as Tahoe, Reno increase their window size until the detection of packet loss. Simulation, implementation concluded that TCP Vegas obtain higher throughput than other variants of TCP.

3.SIMULATION

3.1 Simulation Environment

In this paper simulation is done in wireless environment with 50 nodes,maximum connections are 8 and simulation time is 200 sec, area of simulation is 500 X 500 mm. By varying pause time, speed and data rate, metrics Packet delivery ratio, End-to-End delay, Normalized Routing overhead and loss rate are calculated. With variation in pause time and speed the Throughput and Jitter are calculated at different simulation times. The network simulator NS-2, version 2.35 is used for simulation.

3.2Metrics

The following are the various metrics that are considered:

- **Throughput:** Throughput is the rate at which destination receives data.
- **End-End delay:** It is time involved in delivery of data packets from source node to destination node.
- **Jitter:** Jitter is the difference in end-to-end delays of consecutive packets.
- **Packet loss rate:** Packet loss rate is the ratio between number of packets dropped and number of packets sent from source.
- **Normalized Routing Overhead:** Normalized Routing Overhead is defined as the total number of routing packets transmitted per data packets.
- **Packet Delivery Ratio:** It can be defined as the ratio of number of received data packets to the number of sent packets.

3.3 Simulation Results and Analysis

3.3.1Variation with Pause time

Number of nodes:50, Simulation time:200sec, Maximum speed: 10 m/sec, Data rate: 1Mbps

Table 1. Packet delivery ratio of Reno, SACK and Vegas

Pause time(sec)	Reno	Sack	Vegas
0	0.9787	0.9830	0.9946
10	0.9846	0.9801	0.9967
20	0.9889	0.9858	0.9992
50	0.9931	0.9747	0.9971
100	0.9886	0.9724	0.9981
Pause time(sec)	**Reno**	**Sack**	**Vegas**
150	0.9863	0.9876	0.9968
200	0.9893	0.9941	0.9995

Table 2. End-End delay (sec) of Reno, SACK and Vegas

Pause time(sec)	Reno	Sack	Vegas
0	0.36877	0.38065	0.06174
10	0.44485	0.18466	0.08116
20	0.21309	0.58748	0.02286
50	0.35695	0.44622	0.05399
100	0.38697	0.55799	0.04312
150	0.53830	0.72369	0.08745
200	0.18416	0.25086	0.01988

Table 3.Normalized routing overhead of Reno, SACK and Vegas

Pause time(sec)	Reno	Sack	Vegas
0	0.240	0.276	0.215
10	0.467	0.297	0.362
20	0.364	0.171	0.142
50	1.942	0.859	0.491
100	0.167	0.849	0.751
150	0.531	0.468	1.263
200	0.337	0.119	0.042

Table 4. Loss rate of Reno, SACK and Vegas

Pause time(sec)	Reno	Sack	Vegas
0	0.0213	0.0170	0.0054
10	0.0154	0.0199	0.0033
20	0.0111	0.0142	0.0008
50	0.0069	0.0253	0.0029
100	0.0114	0.0276	0.0019
150	0.0137	0.0124	0.0032
200	0.0107	0.0059	0.0005

3.3.2Variation with Speed

Number of nodes: 50, Simulation time: 200sec, Pause time: 100.0sec, Data rate: 1Mbps

Table 5. Packet delivery ratio of Reno, SACK and Vegas

Speed(m/sec)	Reno	Sack	Vegas
5	0.9838	0.9948	0.9997
10	0.9940	0.9968	0.9975
20	0.9956	0.9915	0.9986
30	0.9914	0.9935	0.9991

Table 6.End-End delay (sec) of Reno, SACK and Vegas

Speed(m/sec)	Reno	Sack	Vegas
5	0.75170	0.25652	0.03330
10	0.24810	0.52486	0.04958
20	0.19545	0.37033	0.06060
30	0.30076	0.44234	0.06791

Table 7.Normalized routing overhead of Reno, SACK and Vegas

Speed(m/sec)	Reno	Sack	Vegas
5	0.551	1.005	0.066
10	0.214	0.115	1.669
20	0.112	0.047	0.462
30	0.000	0.406	1.138

Table 8.Loss rate of Reno, SACK and Vegas

Speed(m/sec)	Reno	Sack	Vegas
5	0.0162	0.0052	0.0003
10	0.0060	0.0032	0.0025
20	0.0044	0.0085	0.0014
30	0.0086	0.0065	0.0009

3.3.3Variation with Data rate

Number of nodes: 50, Simulation time: 200sec, Maximum speed: 10 m/sec, Pausetime: 100sec.

Table 9. Packet delivery ratio of Reno, SACK and Vegas

Data rate(Mbps)	Reno	Sack	Vegas
1	0.9923	0.9940	0.9993
2	0.9922	0.9920	0.9990
5	0.9894	0.9928	0.9993
11	0.9929	0.9914	0.9994

Table 10. End-End delay(sec) of Reno, SACK and Vegas

Data rate(Mbps)	Reno	Sack	Vegas
1	0.19706	0.18742	0.01802
2	0.19011	0.19217	0.01936
5	0.20563	0.21294	0.02113
11	0.18611	0.18821	0.01888

Table 11.Normalized routing overhead of Reno, SACK and Vegas

Data rate(Mbps)	Reno	Sack	Vegas
1	0.022	0.024	0.028
2	0.078	0.028	0.027
5	0.113	0.048	0.020
11	0.028	0.035	0.018

Table 12.Loss rate of Reno, SACK and Vegas

Data rate(Mbps)	Reno	Sack	Vegas
1	0.0077	0.0060	0.0007
2	0.0078	0.0080	0.0010
5	0.0106	0.0072	0.0007
11	0.0071	0.0086	0.0006

3.3.4 Graphs for Throughput

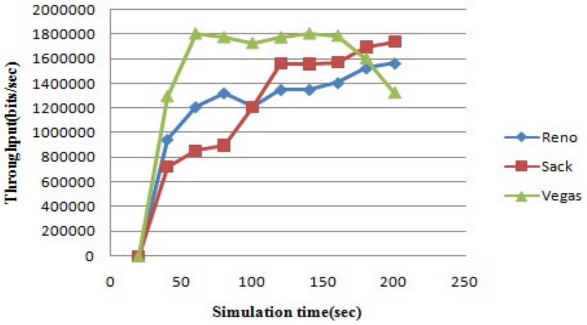

Figure 1. Throughput at Pause time-0sec, speed-5m/sec

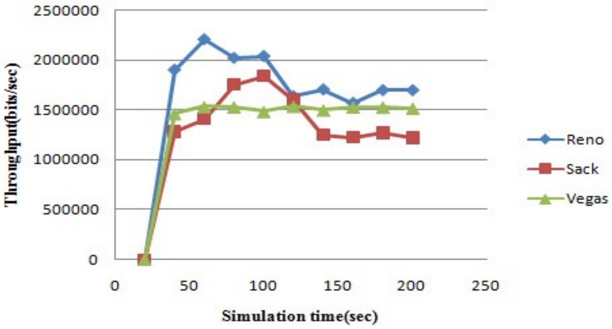

Figure2. Throughput at Pause time-0sec, speed-10m/sec

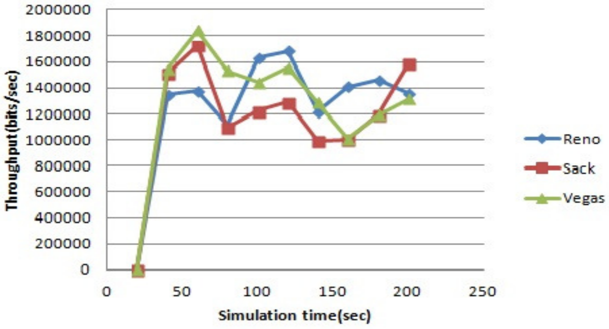

Figure 3. Throughput at Pause time-0sec, speed-30m/sec

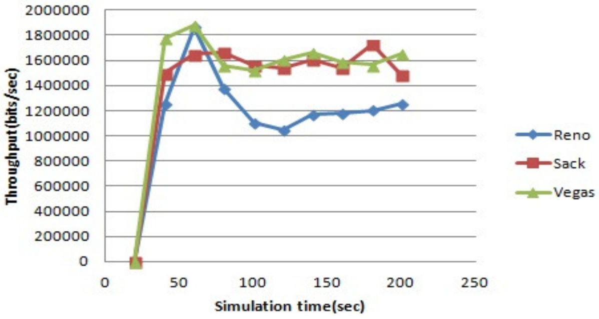

Figure 4.Throughput at Pause time-100sec, speed-5m/sec

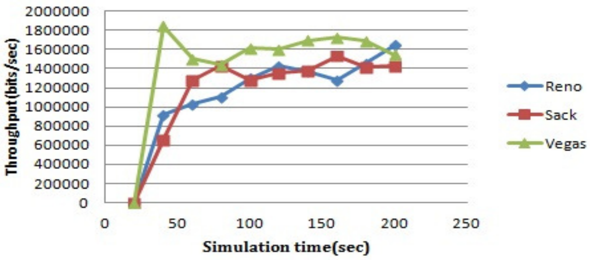

Figure5.Throughput at Pause time-100sec, speed-10m/sec

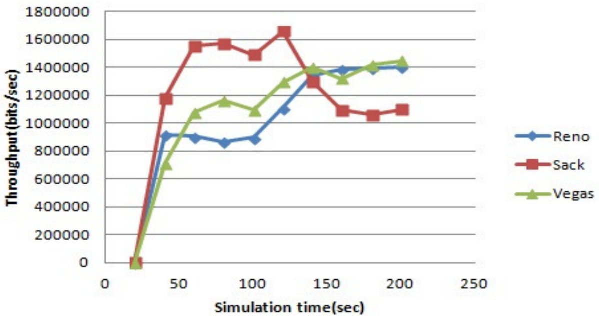

Figure6.Throughput at Pause time-100sec, speed-30m/sec

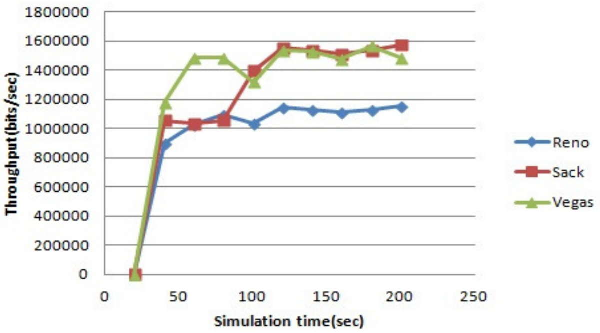

Figure 7.Throughput at Pause time-200sec, speed-5m/sec

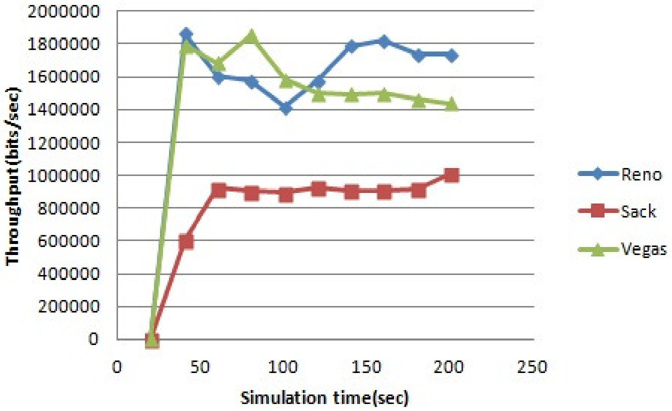

Figure 8.Throughput at Pause time-200sec, speed-10m/sec

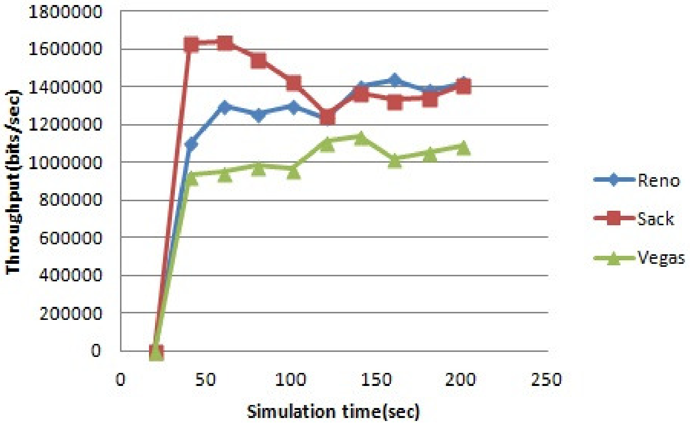

Figure9. Throughput at Pause time-200sec, speed-30m/sec

3.3.5 Graphs for Jitter

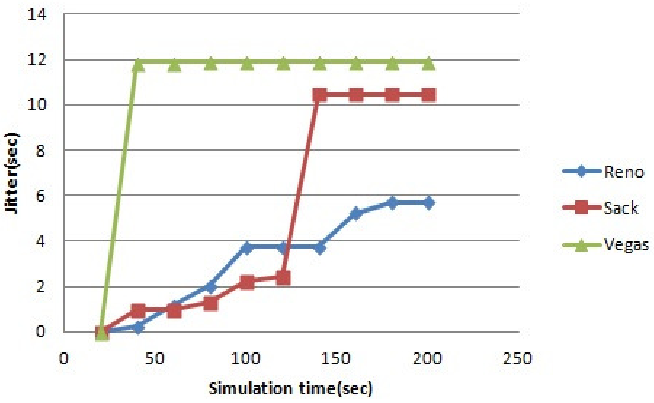

Figure 10. Jitter at Pause time-0sec, speed-5m/sec

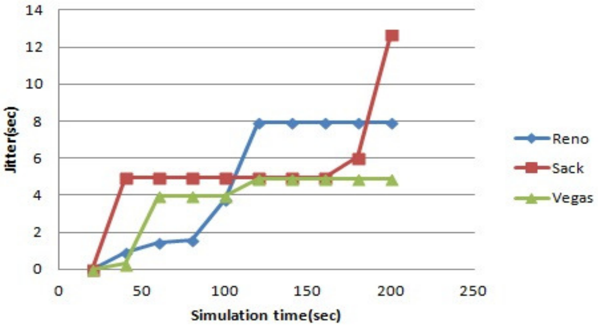

Figure 11. Jitter at Pause time-0sec, speed-10m/sec

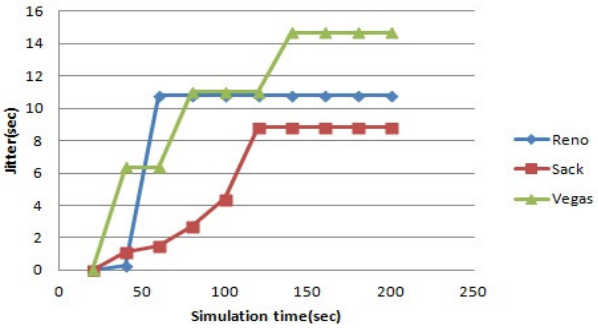

Figure 12. Jitter at Pause time-0sec, speed-30m/sec

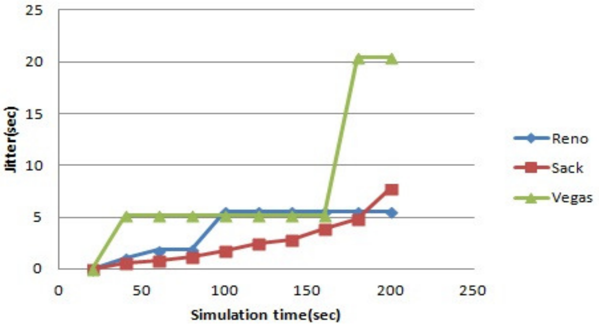

Figure 13. Jitter at Pause time-100sec, speed-5m/sec

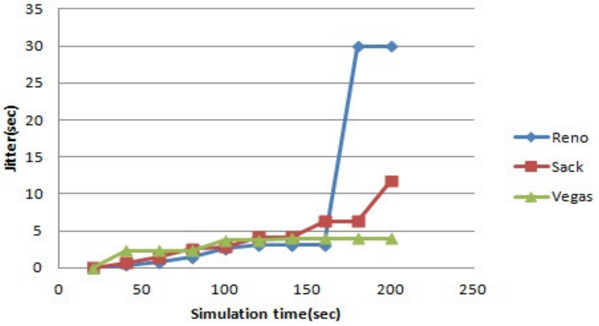

Figure 14. Jitter at Pause time-100sec, speed-10m/sec

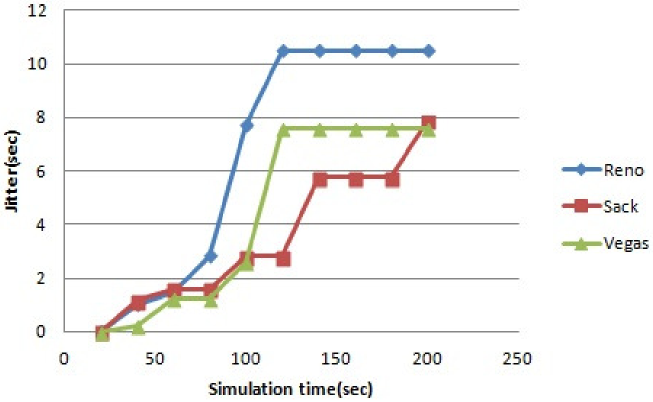

Figure15. Jitter at Pause time-100sec, speed-30m/sec

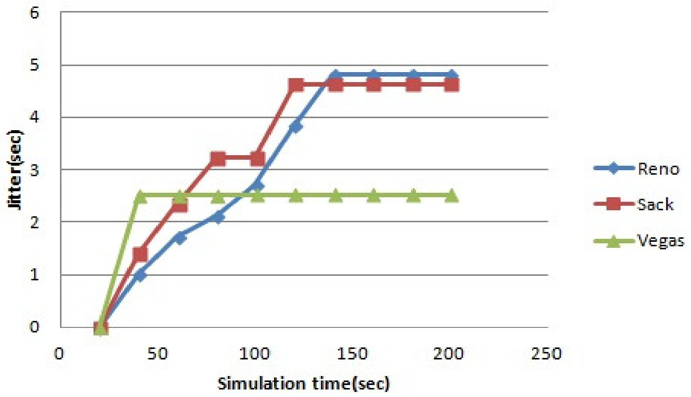

Figure 16. Jitter at Pause time-200sec, speed-5m/sec

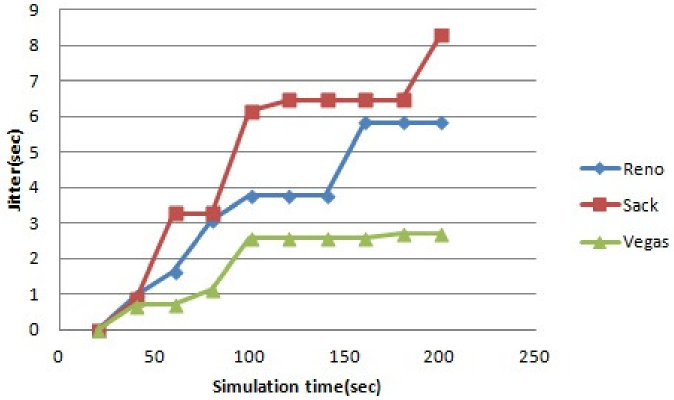

Figure 17. Jitter at Pause time-200sec, speed-10m/sec

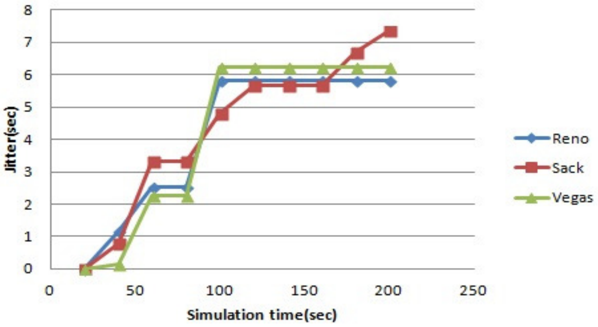

Figure18. Jitter at Pause time-200sec, speed-30m/sec

4. Discussion

Table 1, shows the comparison of Reno, Sack and Vegas in terms of packet delivery ratio with variation in pause time. From the results it can be inferred that Vegas outperforms Reno and SACK. From Table 2, it is concluded that end-end delay is best for Vegas and it is high for Sack. In Table 3 and Table 4, normalized routing overhead and loss rate with variation in pause time are shown and results indicate that routing overhead is less for Sack,and loss rate is high for Sack.

At different speeds packet delivery ratio, end-end delay, normalized routing overhead, loss rate are measured and the results are shown in Tables 5,6,7,8. Here pause time and data rates are kept constant. From Table 5 it is concluded that packet delivery ratio is high for Vegas. Table 6 infers that when speed is varied the end-end delay is less for Vegas. Normalized routing overhead and loss rates are calculated by varying speeds and from the obtained results it is concluded that routing overhead is less for Reno and high for Vegas. Loss rate is high for Reno and less for Vegas, as shown in Table 7 and Table 8.

Different metrics are evaluated by keeping pause time, and speed as constant and varying data rate. Table 9 shows that packet delivery ratio is high for Vegas;end-to-end delay is high for Sack and less for Vegas as shown in Table 10. Table 11 shows the comparison of Reno, Sack and Vegas in terms of normalized routing overhead at different data rates. From the results obtained, it is concluded that routing overhead is high for Reno and less for Vegas. Loss rate is high for Sack and low for Vegas, as is presented in Table 12.

The graphs represent the throughput and jitter which are evaluated at different simulation times with variation in pause time and speed. Figure 1 represents the throughput evaluation of Reno, Sack and Vegas at pause time 0sec, and speed 5m/sec. Here Vegas has high throughput. At pause time 0sec, speed 10m/sec Reno shows best throughput than Sack, Vegas, and at pause time 0 sec, speed 30 m/sec Vegas exhibits high throughput. These are shown in Figure 2 and Figure 3. Jitter is evaluated with variation in pause time and speed. At pause time 0 sec, speed 5m/sec, jitter is high for Vegas as is shown in Figure 10. From Figure 11, it is concluded that jitter is high for Sackwhen evaluated at pause time 0 sec, speed 10m/sec. At pause time 0 sec, speed 30m/sec jitter is measured for Reno, Sack and Vegas. Among three implementations Sack is best and it is presented in Figure 12.

Figure 4 represents the evaluation of throughput for Reno, Sack and Vegas at pause time 100 sec, and speed 5m/sec. From the results obtained it is concluded that Vegas exhibits best throughput than Sack, and Reno. At the same parameters jitter is evaluated and Vegas has high jitter as is shown in Figure 13. By keeping pause time 100 sec as constant and by varying speed 10m/sec, throughput, and jitter are measured. Vegas has high throughput and jitter is best for Vegas as is shown in Figure 5 and Figure 14. At pause time 100 sec, speed 30m/sec,Sack achieves best throughput and Reno results in high jitter as is shown in Figure 6 and Figure 15.

Throughput and jitter are calculated at pause time 200 sec, speed 5 m/sec. Throughput and jitter are better for Vegas and are presented in Figure 7 and Figure 16. From Figure 8 and 17, it is concluded that Reno has better throughput. Sack has high jitter when speed is 10m/sec. When pause time is 200 sec and speed is 30m/sec throughput and jitter are calculated, and the results are presented in Figure 9 and Figure 18. From the graphs it is concluded that throughput is high for Sack, and jitter is high for Sack.

5. CONCLUSION

In this paper performance analysis of TCP Implementations Reno, Sack and Vegas are done in wireless environment. From the results obtained it is concluded that throughput is high for Vegas but at the same time jitter is also high for Vegas at pause time 0sec. At pause time 100 sec Vegas has high throughput, jitter is best for Sack. When pause time is 200 sec throughput is best for Vegas, jitter is less for Vegas.When speed is varied packet delivery ratio is high for Vegas, end-end delay is less and loss rate is also less for Vegas. At different data rates packet delivery ratio is best for Vegas. End-end delay is high for Sack and Vegas has least loss rate.

REFERENCES

[1] Dong-HeeKown, (2008) “Performance comparisons of two on-demand ad hoc routing protocols in dynamic rate shifting WLANs”, *IEEE International Conference*, DOI: 10.1109/ICC.2003.1204229, Vol.1, pp 512-516.

[2] J.Postel, (1981)"Transmission Control Protocol", RFC 793, Available at:www.ietf.org/rfc.

[3] SuhasWaghmare, AdityaParab, PankajNikose and S. J. Bhosale, (2011) “Comparative analysis of different TCP Variants in a wireless environment, *3rd International conference*,Kanyakumari (India), DOI: 10.1109/ICECTECH.2011.5941878, Vol. 4, pp 158-162.

[4] Ra’ed Y. Awdeh, (2007) “Compatibility of TCP Reno and TCP Vegas in wireless ad hoc networks”, *IET Communications*, DOI: 10.1049/IET.COM:20060592, Vol. 1, pp1187-1194.

[5] V. Jacobson, R. Braden, D. Borman, (1988) “TCP Extensions for Long-Delay Paths”, RFC 1072Available at : http://tools.ietf.org/html.

[6] Balveer Singh, (2013)"A Comparative Study of Different TCP Variants in Networks", *IJCIT-International Journal of Computer Trends and Technology*, Vol. 4, pp2962-2966.

[7] Hala A. Elaarag and Mostafa A. Bassiouni, (1999) “Transport control protocol for wireless connections”, IEEE 49thconference on Vehicular Technology,Houston(TX), DOI: 10.1109/VETEC.1999.778072, Vol. 1, pp337-341.

[8] SandeepSandhu, (2014)"Comparative Analysis of Different TCP Variants with AODV in Mobile Ad-Hoc Network", *IJCSIT-International Journal of Computer Science and Information Technologies,* Vol. 5, No. 5, pp6329-6332.

[9] NeerajBhargava, (2013)"Analysis of Different Congestion Avoidance Algorithms", IRACST – *International Journal of Computer Networks and Wireless Communications (IJCNWC),*Vol. 3, No. 1, pp32-36.

[10] Mohit P. Tahiliani, K. C. Shet, and T. G. Basavaraju, (2013) “Comparative study of High speed TCP Variants in Multi-Hop wireless networks”, *International Journal of Computer Theory and Engineering,*Vol. 5, No. 5, pp802-806.

[11] MadihaKazmi, AzraShamim, NasirWahab, and Fozia Anwar, (2014) “Comparison of TCP Reno, Tahoe, Sack, New Reno, Vegas in IP and MPLS networks under constant bit rate traffic”, *International Conference on Advanced Computational Technologies & Creative Media (ICACTCM),*Pattaya(Thailand), pp33-38.

[12] M.A. Rahman, A.H.Kabir, K. A. M. Lutfullah, Z.Hassan and M. R. Amin, (2008) “Fair comparisons of different TCP variants for future deployment of Networks”, *International Conference on Electronics, Computer and Communication (ICECC)*,University of Rajshahi, (Bangladesh), pp260-263.