International Journal of Technology Research and Management
ISSN (Online): 2348-9006
Vol 2 Issue 9 September 2015

An enhanced approach for transmission control protocol traffic management Mechanism for Wireless Network

Nitesh Mishra[1], Prof. Shaileena John[2]
Department of Electronics & Communication[1, 2]
niteshmish20@gmail.com[1], shaileenajohnsj@gmail.com[2]

Abstract *- TCP Congestion management is finish to finish management mechanism. that is enforced through dominant some important parameter amendment scrutiny to wireless networks, there are many various characteristics in wireless environments. In this paper Enhanced mechanism for tcp |transmission management protocol TCP protocol communications protocol corruption control is presented. It considers the influences to protocol sender's packet sending rate not solely by the congestion however conjointly by the corruption. The causation window are going to be calculated when every transmission, in line with range of corrupt data packet. So, there is less packet drop occurring in data transmission. The comparative study of recent protocol traffic control with alternative protocol variants is additionally presented with variation in node speeds, pause time and number of nodes in network. The Enhanced Mechanism is simply implemented with fewer overheads and may effectively improve and efficient little variances of out run in this mechanism is performed in QualNet 5.0.2 machine.*

Keywords*- TCP, Tahoe, Reno, Enhanced Reno, Wireless.*

1. INTRODUCTION

Transmission Control Protocol (TCP), is the predominant Internet protocol and carries approximately 90% of Internet traffic in today's heterogeneous wireless and wired networks. TCP is Widely used as a connection oriented transport layer protocol that provides reliable packet delivery over unreliable links. TCP does not depend on the underlying network layers and, hence, design of various TCP variants is based on the properties of wired networks. However, TCP congestion control algorithms may not perform well in heterogeneous networks. Wireless networks have higher bit error rates due to weather conditions, obstacles, and multipath Interferences, mobility of wireless end-devices, and signal attenuation and fading, which may lead to packet loss. Various TCP algorithms and techniques have been proposed to improve congestion and reduce the non-congestion related packet loss. TCP Tahoe, TCP Reno, TCP Enhanced Reno with Selective Acknowledgement (SACK), TCP Enhanced Reno, TCP Vegas, and TCP Binary Increase Congestion (BIC) are examples of proposed end to end solutions. Snoop-TCP is a link layer control approach while N-TCP split connection approaches. They are all proposed to improve network Performance. The end to end techniques are the most promising because they require changes only to the end systems rather than to the intermediate nodes. These end-to-end control approaches are used in today's deployed networks.

2. DESCRIPTION OF TCP VARIANTS

TCP Reno

This Reno retains the basic principle of Tahoe, such as slow starts and the coarse grain re-transmit timer. However it adds some intelligence over it so that lost packets are detected earlier and the pipeline is not emptied every time a packet is lost. Reno requires that we receive immediate acknowledgement whenever a segment is received. The logic behind this is that whenever we receive a duplicate acknowledgment, then his duplicate acknowledgment could have been received if the next segment in sequence expected, has been delayed in the network and the segments reached there out of order or else that the packet is lost. If we receive a number of duplicate acknowledgements then that means that sufficient time have passed and even if the segment had taken a longer path, it should have gotten to the receiver by now. There is a very high probability that it was lost. So, Reno suggests an algorithm called 'Fast Re-Transmit'. Whenever we receive 3 duplicate ACK's we take it as a sign that the segment was lost, so we retransmitted segment without waiting for timeout.

The basic algorithm is presented as under:

- Each time we receive 3 duplicate ACK's we take that to mean that the segment was lost and we re-transmit the segment immediately and enter ‘Fast-Recovery’.

- Sets ssthresh to half the current window size and also set CWND to the same value.

- For each duplicate ACK receive increase CWND by one. If the increase CWND is greater than the amount of data in the path then transmit a Enhanced segment else wait.

Problems

Reno performs very well over TCP when the packet losses are small. But when we have multiple packet losses in one window then RENO doesn’t perform too well. The reason is that it can only detect single packet losses. If there is multiple packet drops then the first info about the packet loss comes when we receive the duplicate ACK’s. But the information about the second packet which was lost will come only after the ACK for the retransmitted first segment reaches the sender after one RTT (Round-Trip Times). Another problem is that if the widow is very small when the loss occurs then we would never receive enough duplicate acknowledgements for a fast retransmit and we would have to wait for a coarse grained timeout. .Reno’s Fast Recovery algorithm is optimized for the case when a single packet is dropped from a window of data.

TCP Lite

TCP Lite is a service that provides a transport method that interrupts TCP in order to reduce the overhead involved in session management in which no data is transmitted or received. TCP Lite reduces or eliminates pure TCP protocol data units used in the setup and ACK while maintaining order, integrity, reliability and security of traditional TCP. TCP lite uses large window and protection against wrapped sequence numbers.

Problems

Lite performs over TCP same as Reno. But when window increases it have some problems to maintain them.

TCP Enhanced Reno

Enhanced RENO is a slight modification over TCP-RENO. It is able to detect multiple packet losses and thus is much more efficient that RENO in the event of multiple packet losses. Like Reno, Enhanced-Reno also enters into fast-retransmit when it receives multiple duplicate packets , however it differs from RENO in that it doesn’t exit fast-recovery until all the data which was out standing at the time it entered fast recovery is acknowledged. Thus it overcomes the problem faced by Reno of reducing the CWND multiples times. The fast-transmit phase is the same as in Reno. The difference in the fast recovery phase which allows for multiple re-transmissions in Enhanced-Reno. Whenever Enhanced-Reno enters fast recovery it notes the maximums segment which is outstanding.

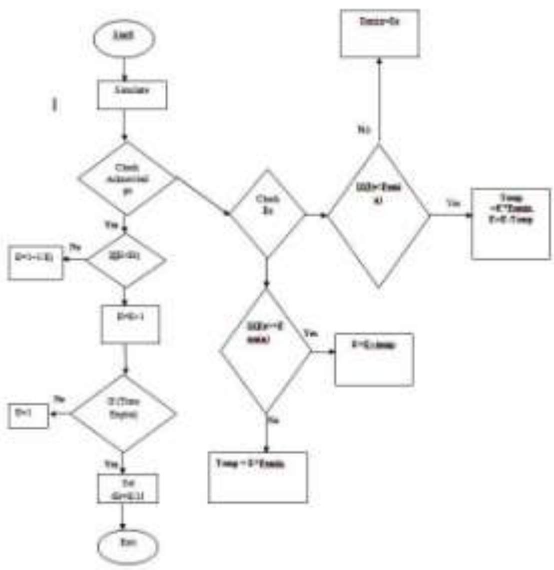
The fast-recovery phase proceeds as in Reno, however when a fresh ACK is received then there are two cases: If it ACK’s all the segments which were outstanding when we entered fast recovery then it exits fast recovery and sets CWND to ss thresh and continues congestion avoidance like Tahoe. If the ACK is a partial ACK then it deduces that the next segment in line was lost and it retransmits that segment and sets the number of duplicate ACKS received to zero. It exits Fast recovery when all the data in the window is acknowledged.

Problems

Enhanced-Reno suffers from the fact that it takes one RTT to detect each packet loss. When the ACK for the first retransmitted segment is received only then can we reduce which other segment was lost. Round-trip time until all of the lost packets from that window has been retransmitted.

3. PROPOSED WORK

Modern implementations of TCP contain four intertwined algorithms that have never been fully documented as standards wireless network: slow starts, congestion avoidance, fast retransmit, and fast recovery and we revise the algorithms for provides the source code for the implementation. TCP must implement slow start and congestion, but fast retransmit and fast recovery were implemented after the purpose of this document. ETCP (Enhanced Transport Control Protocol) is the Enhanced TCP congestion and corruption control mechanism for wireless network. It considers the influences to TCP sender’s packet sending rate not only by the congestion but also by the corruption. It is a good reference to apply the TCP to wireless networks.



Step 1:

Receipt of new ACK (non-repeated acknowledgment):

If (E < Et) set E = E++; slow-start phase.
Else set E = 1+1/E; congestion avoidance phase.

Step 2:
Receipt of NACK:
Record the NACK sequence number ,and then retransmit the “corrupted packet”.

Step 3:
Receipt of duplicate ACK:
Increment duplicate ACK count for segment being ACKed. When duplicate AC count exceed specified threshold value, retransmit “next expected ”.
if (Pe>= Pemin)Then set {temp = E*Pemin;E =E + temp;}

Step 4:
If the next expected packet is not a “corrupted packet”:
If (Pe<Pemin) Then set{Pemin = Pe;Temp =E *Pemin;E=E-temp;}Resume congestion avoidance using new window size once retransmission has been acknowledged.

Step 5:
Upon time expiry: Set Et = E/2
And then set E =1.
Recover the” missing segment” from the slow start phase.
Note that in above, E and Et denote the current congestion window size and the slow-start threshold, respectively.

4. COMPUTER SIMULATION

In this paper all the simulation work is performed in QualNet wireless network simulator version 4.0 [9]. Initially number of nodes are 50, Simulation time was taken 200 seconds and seed as 1. All the scenarios have been designed in 1500m x 1500m area. Mobility model used is Random Way Point (RWP). In this model a mobile node is initially placed in a random location in the simulation area, and then moved in a randomly chosen direction between at a random speed between [SpeedMin, SpeedMax]. The movement proceeds for a specific amount of time or distance, and the process is repeated a predetermined number of times. We choose Min speed = 5 m/s, Max speed = 30m/s, and pause time = 5s to 30s. All the simulation work was carried out using TCP variants (Reno, Lite, Tahoe) with DSR routing protocol .Network traffic is provided by using File Transfer Protocol (FTP) application. File Transfer Protocol (FTP) represents the File Transfer Protocol server and client. In wireless network which we have used have following values for different parameter:

A. Mobility model Random Way Point Minimum speed 0 mps

   Maximum speed 5mps, 10 mps, 15 mps, 20mps, 25 mps, and

   30 mps

   Pause time 5s, 10s, 15s, 20s, 25s, 30s. Simulation Time 200s

B. Terrain

   Coordination 1500 * 1500 m

   C. Connection

   FTP (File transfer protocol) : 41 (client) to 1(server) Item size 512(byte)

D. Radio/physical layer parameters:

   Radio type: 802.11b Radio
   Data rate: 2Mbps
   Packet reception model: Bit error rate (bpsk.ber)

E. MAC Protocol

   802.11

   F. Routing Protoco

   DSR (Dynamic Source Routing)

   G. Transport Protocol
   TCP Tahoe, TCP Reno, TCP Lite, TCP ETCP.

H. Number of Node

   50

I. Node Placement Random

F. Seed 1

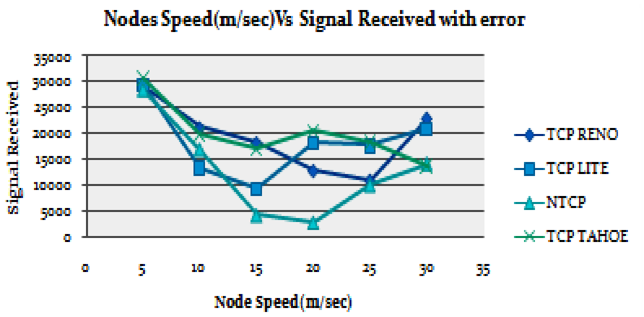
5. SIMULATION METHODOLOGY

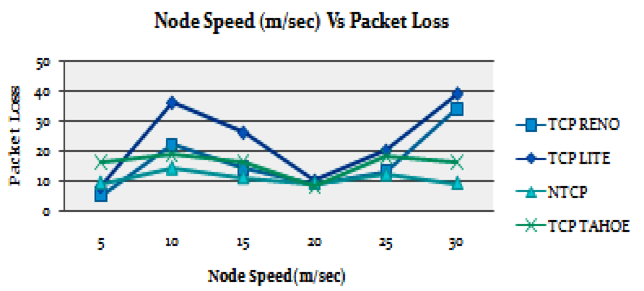
Parameter	Scenario1	Scenario2	Scenario3
Simulation Time	Constant	constant	constant
Node	constant	constant	change
Area	constant	constant	constant
Pause Time	constant	change	constant
TCP Protocol	change	change	change
Routing Protocol	constant	constant	constant
Node Speed (m/sec)	change	constant	constant

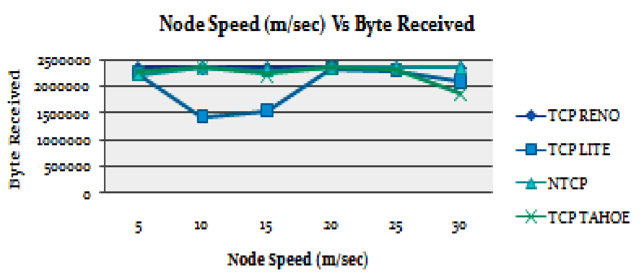
A Performance metrics used for this works are as follows:

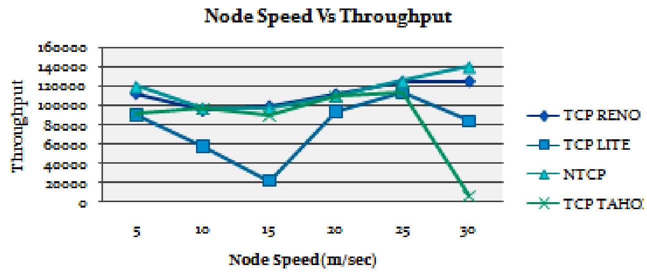
1. **Throughput** is the measure of the number of packets successfully transmitted to their final destination per unit time. It is the ratio between the numbers of sent packets vs. received packets.
2. **Signal Received with error** is the measure of signal received, but they have error. The error may be occurring due to noise or due to heavy traffic.
3. **Bytes received** are the measure of total packet received by server. The packets may be drop due to heavy traffic. So received packets may be vary according to traffic conditions.
4. **Packet loss** is the measure of total discarded packet due to corruption or due to packet drop. We can calculate it by subtracting total received packets by server in total sent packet by client.

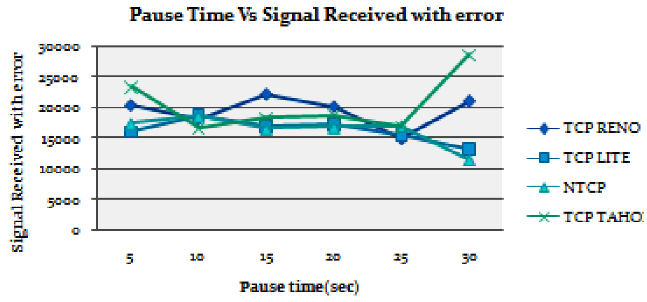
6. RESULT

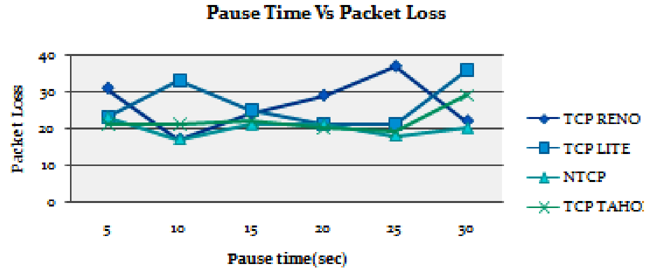


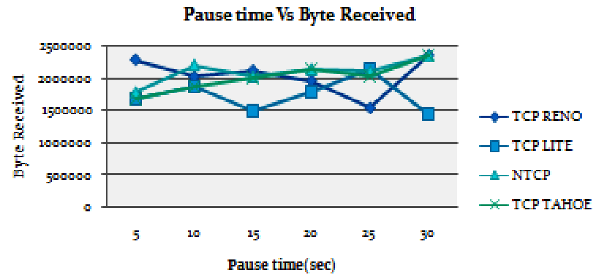


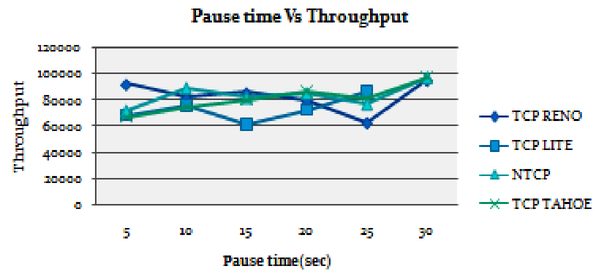


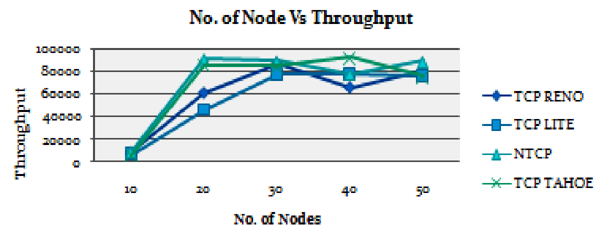












7. CONCLUSION

We proposed an Enhanced TCP (ETCP) with a Enhanced calculation for sending window and implemented it in a Mobile Ad-Hoc Network. Extensive simulation studies were undertaken to compare its performance with other standards TCP Reno, TCP Enhanced Reno and TCP Sack etc. over Wireless Network. The simulation results show that the performance of ETCP is better than other TCP variants. From result we conclude that the performance of ETCP is better in high density node because in this condition sender can get different paths through different nodes. Thus we conclude that, receiver can receive maximum data when we increase number of nodes in network. The existing TCP variants and its applicable algorithm are analyzed and describe about the protocol which one is better and suitable for packet and link utilization in the network congestion because the traditional TCP treat all packet losses due to the congestion, it does not treat from the link failure. The most of protocol shows better uses and many of them shows poor responsiveness to changing network conditions and network utilization.

REFERENCES

[1] Perkins, Ad Hoc Networking, Addison-Wesley 2008, ISBN 0201309769.
[2] C K Toh, Ad Hoc Mobile Wireless Networks, Prentice Hall Publishers, 2005.
[3] T.P. Pedersen, ―A Threshold Cryptosystem without a Trusted Party‖, In Proc. Of Eurocrypt'91, Lecture Notes in Computer Science, LNCS 547, Springer Verlag, pp.522-526, 1991
[4] Kazuo Takaragi, Kunihiko Miyazaki, Masashi Takahashi, A Threshold Digital Signature Issuing Scheme without Secret Communication.
[5] R. Vinod Kumar, Dr. R.S.D. Wahida Banu ―A Enhanced Approach For Load-balancing In EEAODV Protocol‖ IEEE-International Conference On Advances In Engineering, Science And Management (lCAESM -2012) March 30, 31, 2012
[6] Vasilios A. Siris and Despina Triantafyllidou ―Seamless Congestion Control over Wired and Wireless IEEE 802.11 Networks ―NETWORKING 2004, LNCS 3042, pp. 1470–1475, 2004. IFIP International Federation for Information Processing.
[7] K.Srinivasa Rao, R. Sudhistna Kumar, P. Venkatesh, R.V.Sivaram Naidu,and A.Ramesh ―Development of Energy Efficient and Reliable Congestion Control Protocol for Multicasting in Mobile Adhoc Networks compare with AODV Based on Receivers‖ International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9522 Vol. 2, Issue 2,Mar-Apr 2012, pp.531-534.
[8] Ms. Harshada Pingale, Ms. Ashwini Rakshe, Mr. S.A. Jain, D. Mr. S.R.Kokate ―A STUDY OF CONGESTION AWARE ADAPTIVE ROUTING PROTOCOLS IN MANET‖ International Journal of Advanced Technology & Engineering Research (IJATER), VOLUME 2, ISSUE 2, MAY 2012.
[9] Shital kumar Jain, Shrikant Kokate, Pranita Thakur, Shubhangi Takalkar ―A Study of Congestion Aware Adaptive Routing Protocols in MANET‖ Computer Engineering and Intelligent Systems ISSN 2222-1719 (Paper) ISSN 2222-2853 Vol 3, No.4, 2012
[10] Mr.S.A.Jain, Ms Aruna.A, Kadam ―A Study of Congestion Control Mechanism Using Link Failure Detection in MANET‖ International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9522 Vol. 2, Issue 1,Jan-Feb 2012, pp.1009-1012.
[11] Vishal Sharma, Amitoj Soni, Sodhi Gunjit Singh, Takshi Gupta ―AN EFFICIENT APPROACH FOR ROUTING IN WIRELESS AD HOC NETWORK USING ARTIFICIAL INTELLIGENCE‖ International Journal on Computer Science and Engineering (IJCSE), ISSN : 0975-3397 Vol. 4 No. 02 February 2012.
[12] Soundararajan, R.S. Bhuvaneswaran ―Adaptive Multi-Path Routing for Load Balancing in Mobile Ad Hoc Networks‖, Journal of Computer Science 8 (5): 548-555, 2012 ISSN 1549-3535 © 2012 Science Publications.
[13] Sanjeev Sharma, Mukesh Kumar Dhariwal ―An Improved Mechanism for Congestion Control in TCP for Ad Hoc Network‖ International Journal of Computer Applications (0975 – 8887) Volume 20– No.2, April 2011.