HCTL Open Int. J. of Technology Innovations and Research
HCTL Open IJTIR, Volume 1, January 2013
e-ISSN: 2321-1814
ISBN (Print): 978-1-62776-012-6

An Enhanced TCP Corruption Control Mechanism For Wireless Network

Preetam Suman and Amrit Suman
preetam.suman@ieee.org and amrit.it@gmail.com

Abstract

Mobile Ad Hoc networks are collections of mobile nodes, dynamically forming a temporary network without pre-existing network infrastructure or centralized administration. Transmission control protocol (TCP) provides connection oriented, reliable and end to end mechanism. Comparing to wire networks, there are many different characteristics in wireless environments. In this paper an improved mechanism for TCP corruption control is presented. It considers the influences sending rate to TCP sender's packet not only by the congestion but also by the corruption. The sending window size calculated after each transmission, based on number of corrupted packet. So, there is less packet drop in transmission. The comparative study of Enhanced TCP (ETCP) with other TCP variants is also presented on various parameters like mobility, size of network and the speed. The improved mechanism is implemented with fewer overheads and is effectively improve reliability with small variances of throughput and delay. Implementation and Simulation is performed in QualNet 4.0 simulator.

HCTL Open Int. J. of Technology Innovations and Research
HCTL Open IJTIR, Volume 1, January 2013
e-ISSN: 2321-1814
ISBN (Print): 978-1-62776-012-6

Keywords

Wireless Network, Congestion Control, Corruption Control, TCP Variants.

Introduction

Transmission control protocol (TCP) [1, 2] 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 efficiently in heterogeneous networks.

Wireless networks have higher bit error rates due to weather conditions, obstacles, multipath interferences, mobility of wireless end-devices, 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 [11], TCP Reno [13], TCP Reno with Selective Acknowledgement (SACK) [11], TCP NewReno [12], TCP Vegas [15], and TCP FACK [14] are examples of proposed end-to-end solutions. They are all proposed to improve network performance [3]-[9]. The end-to-end techniques are the most promising because they require changes only to the end systems instead to the intermediate nodes. These end-to-end control approaches are used in today's deployed networks. In Section 2, algorithm of ETCP is described. Description of simulated network is given in Section 3, while simulation scenarios and results are described in Section 4, simulation methodology is described in Section 5 conclusion and future work is explained with Section 6.

Corruption Loss Rate - The corruption loss rate defined as the packet loss rate because of corruption. Sending rate is not adjusted when the there is little corruption loss rate, but its required to decrease the sending rate rapidly to improve the reliability when the corruption loss rate becomes higher. Otherwise, there will be more lost packets due to corruption and more packets will be retransmission, responding the poor transmission reliability and more energy consumption of mobile hosts.

e-ISSN: 2321-1814
ISBN (Print): 978-1-62776-012-6

The corruption loss rate define as:

$$Pe = m/n \tag{1}$$

Where, Pe is the corruption loss rate defined in [10]. If TCP sender sends n packets in which m packets may be discarded due to weak wireless link.

The corruption loss rate decides by the bit error rate (BER) of wireless link layer and the length (Length) of data frame:

$$Pe = 1 - (1 - BER)Length \tag{2}$$

If the corruption loss rate Pe is higher than the certain lower limit P_{emin}, the sending rate will be decreased. Many factors decide the value of corruption loss rate lower limit P_{emin}, mainly include: the kind of application; the length of data frame; the bit error rate of data link layer; the bandwidth and the transmission delay of wireless network etc. Normally value of P_{emin} is 0.4.

Enhance TCP

ETCP (Enhanced Transport Control Protocol) is the improved 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.

Initial Window

The initial sending window is calculated by following formula:

```
Iw= min (4 * SMSS , Max (2 * SMSS , 4380 byte))
```

Slow-Start Algorithm

The slow start algorithm is used to start a connection of ETCP and the periods after the value of retransmission timer exceed the RTO (retransmission timeout). In the start of ETCP, the size of cwnd will be initialized to 1.

The slow start algorithm describes as below:

```
if (Receive ACKs && cwnd < ssthresh)
{
cwnd = cwnd++;
}
```

e-ISSN: 2321-1814
ISBN (Print): 978-1-62776-012-6

The slow start algorithm will be ended in two conditions. First, if the congestion window size reaches the slow start threshold size (ssthresh), the slow start will be ended and then congestion avoidance takes over. Second, if there lose any packet due to congestion or high packet loss rate due to corruption, the slow start also will be ended and then fast recovery takes over.

Congestion Avoidance Algorithm

If the congestion window size (cwnd) is less than or equal to the slow start threshold size (ssthresh), DW-TCP is in slow start; otherwise ETCP is performing congestion avoidance.
The congestion avoidance algorithm describes as below:

`if (Receive ACKs || (Receive Explicit Corruption Loss Notification && Corruption Loss Rate` $Pe < P_{emin}$`))`
`{`
`if (cwnd > ssthresh)`
`cwnd = 1+1/cwnd`
`else`
`cwnd++;`
`if (Receive Explicit Corruption Loss Notification)`
`Pe= (m/n);`
`If (Pe <` P_{emin}`)`
P_{emin} `= Pe;`
`temp=cwnd*` P_{emin}`;`
`cwnd=cwnd  temp;`
`}`

In the algorithm, **cwnd** is the congestion window size; **n** is total number of packets to be send; **m** is the number of lost packets due to wireless link corruption; **Pe** denotes the corruption loss rate.

Fast Retransmission and Fast Recovery

If the network congestion or heavy corruption, the fast recovery algorithm will be taken. When the network congestion, set ssthresh to one-half the flight size or double of SMSS (maximum segment size) window. `if (Congestion || Heavy Corruption)`
`{`
`if (Receive Same ACK 3 Times || Retransmission Timer Overtime) /* Congestion */`

HCTL Open Int. J. of Technology Innovations and Research
HCTL Open IJTIR, Volume 1, January 2013
e-ISSN: 2321-1814
ISBN (Print): 978-1-62776-012-6

```
{
Ssthresh = max(flightsize/2 , 2*SMSS);
// Flightsize are those data which have no acknowledged.
if (Retransmission Timer Overtime)
{
cwnd = 1; Exit and call slow-start;}
else /* Receive Same ACK 3 Time */
cwnd = ssthresh;
}
else if (Receive Explicit Corruption Loss Notification
&& Corruption Loss Rate Pe>=P_emin)
{
temp=cwnd* P_emin;
cwnd=cwnd+ temp;
};
if (Receive Explicit Loss Corruption Notification)
Pe= (m/n);
If (Pe < P_emin)
P_emin = Pe;
temp=cwnd*P_emin;
cwnd=cwnd - temp;
}
```

Computer Simulation

In this paper all the simulation work is performed in QualNet [11] wireless network simulator version 4.0. 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 $[Speed_{Min}, Speed_{Max}]$. The movement proceeds for a specific amount of time or distance, and the process is repeated a predetermined number of times. Nodes in network moves with speed of 5 m/s to 30m/s, and with pause time of 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

HCTL Open Int. J. of Technology Innovations and Research
HCTL Open IJTIR, Volume 1, January 2013
e-ISSN: 2321-1814
ISBN (Print): 978-1-62776-012-6

Transfer Protocol Server and Client.

Simulation Methodology

Parameters	Scenario 1	Scenario 2	Scenario 3
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

Table 1: *Simulation Scenarios*

Performance metrics used for this works are as follows:

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.

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.

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.

Packet loss is the measure of total discarded packet due to corruption or due to packet drop. It can be calculate by subtracting total received packets by server with total sent packet by client.

Results and Analysis

Fig 1, 2 & 3 shows signal received with error of different variants with variation in speed of node, pause time and number of node in network. It has to be seen that ETCP has less error signal than other TCP variants. Due to optimal

e-ISSN: 2321-1814
ISBN (Print): 978-1-62776-012-6

routing path between sender and receiver. When number of node in network increases than congestion also occurs, due to this reason, signals were distorted.

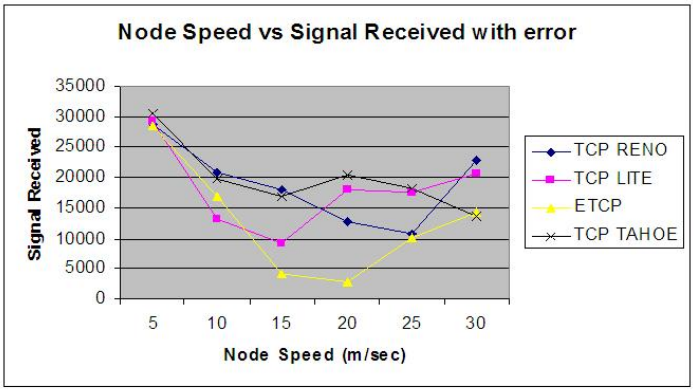

Figure 1

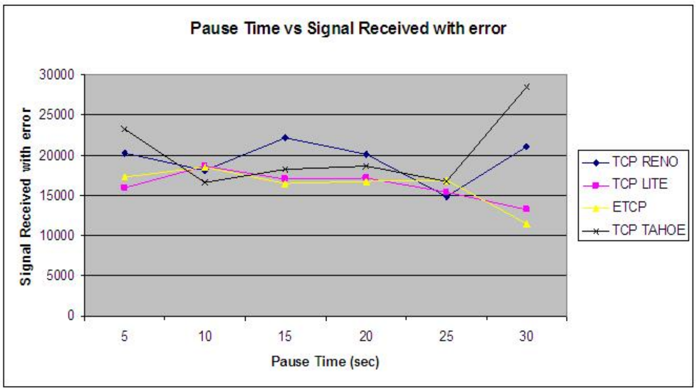

Figure 2

e-ISSN: 2321-1814
ISBN (Print): 978-1-62776-012-6

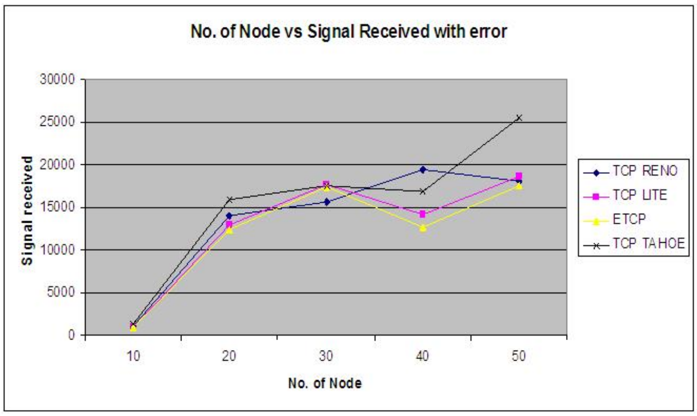

Figure 3

Fig 4, 5 & 6 shows Packet loss of different variants of TCP with variation in node speed, pause time and number of node. It has to be seen that ETCP has less packet loss than other TCP variants. It is due to proper sending window size, which is calculated according to number of discarded packets. So, receiver can receive those packets which are sent by sender.

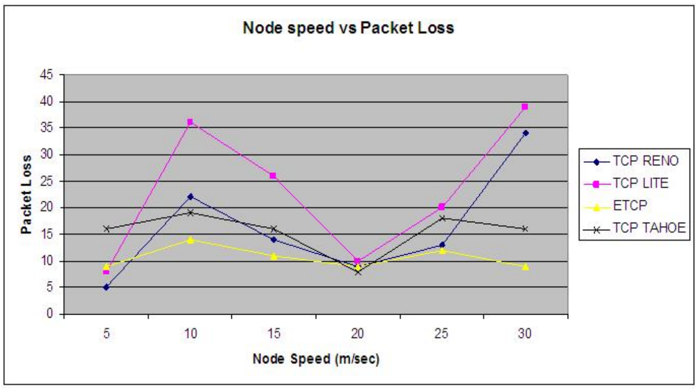

Figure 4

e-ISSN: 2321-1814
ISBN (Print): 978-1-62776-012-6

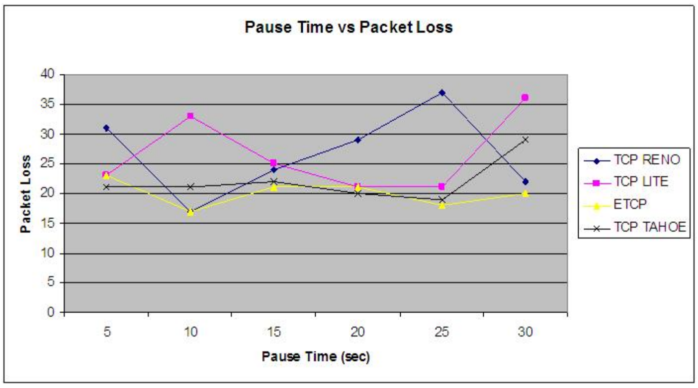

Figure 5

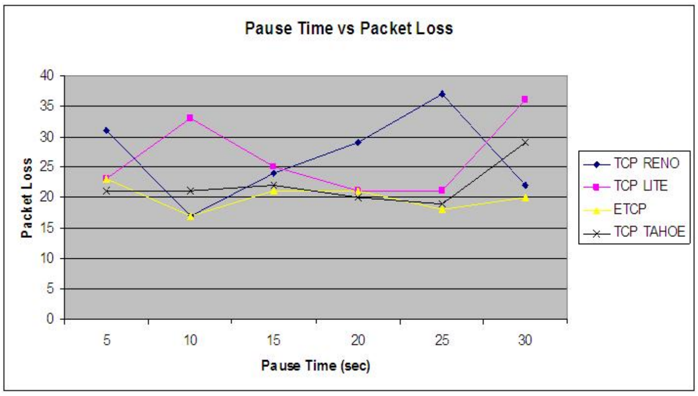

Figure 6

Fig 7, 8 & 9 shows the readings of total byte received for different TCP variants with variation in speed of node, pause time and number of node in network. It has to be seen that receiver can receive maximum byte at ETCP, but it is sometimes less due to congestion in network. Sometimes nodes can move faster, so nodes cannot get proper signal, due to this reason nodes cannot received packets.

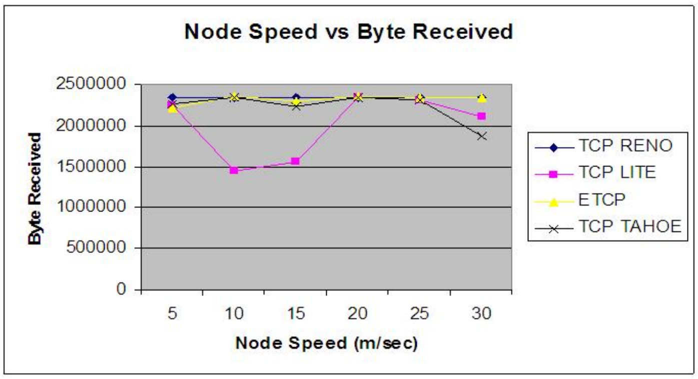

Figure 7

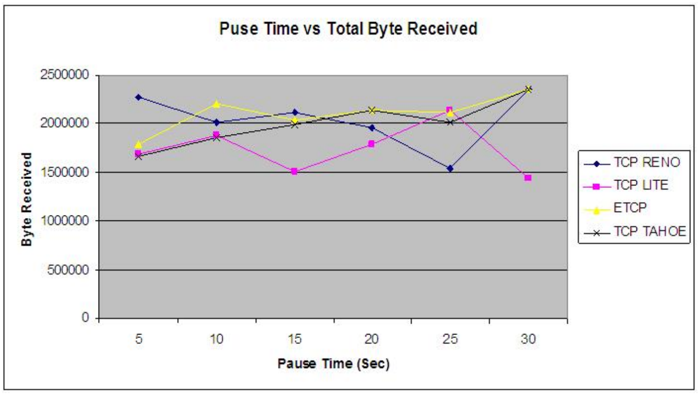

Figure 8

Fig 10, 11 & 12 shows throughput of different TCP variants with variation in node speed, pause time and number of node. Throughput is the ratio of numbers of sent packets with number of received packets. It has to be seen that Throughput of ETCP is better than other TCP variants. Receiver receive maximum packets and there are less packets were discarded, due to this reason throughput is better.

Conclusion

An enhanced TCP (ETCP) is proposed with a new calculation for sending window and implemented it in a Mobile Ad-hoc Network in QualNet 4.0. It

HCTL Open Int. J. of Technology Innovations and Research
HCTL Open IJTIR, Volume 1, January 2013
e-ISSN: 2321-1814
ISBN (Print): 978-1-62776-012-6

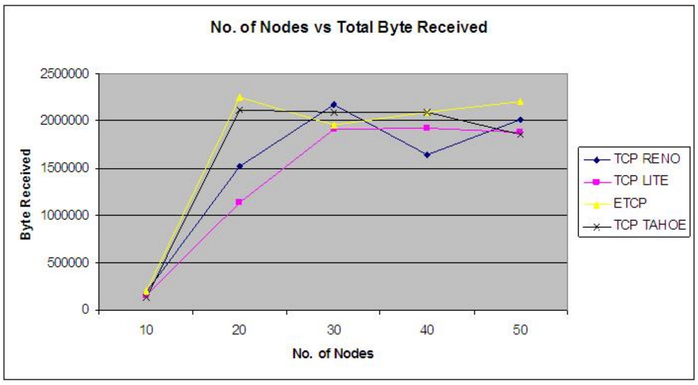

Figure 9

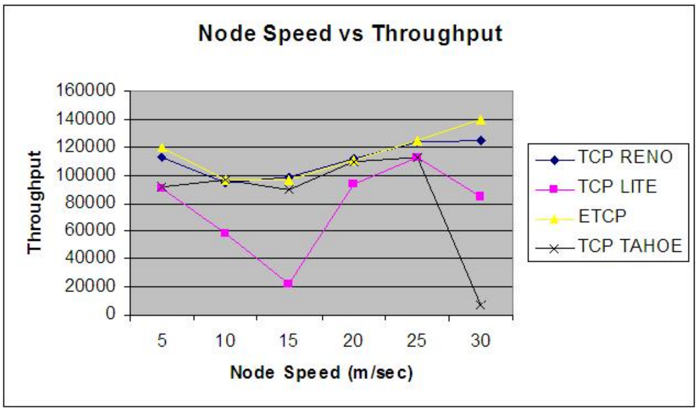

Figure 10

considers the influences sending rate to TCP sender's packet not only by the congestion but also by the corruption. The sending window for ETCP is calculated according to packet corrupted in network. Extensive simulation studies were undertaken to compare its performance with other standards TCP Reno, TCP Lite and TCP Tahoe over Ad-hoc Mobile Network. The simulation results show that the performance of ETCP is better than other than other TCP variants. The performance of TCP variants are analysed on various parameters like mobility, size of network and the speed. The comparison of TCP variants are performed using various performance metrics like signal received with error, total data packet received, total packet drop and throughput.

HCTL Open Int. J. of Technology Innovations and Research
HCTL Open IJTIR, Volume 1, January 2013
e-ISSN: 2321-1814
ISBN (Print): 978-1-62776-012-6

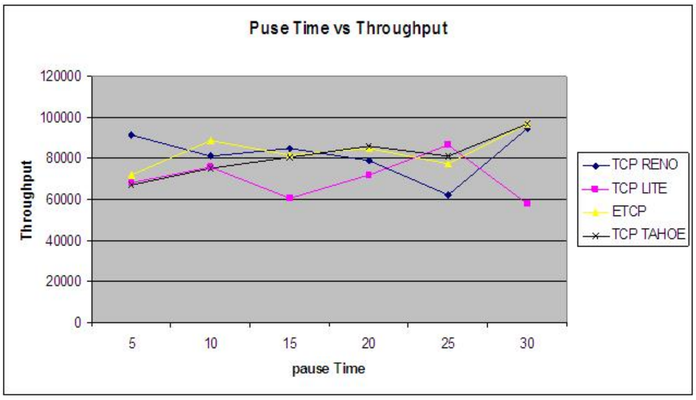

Figure 11

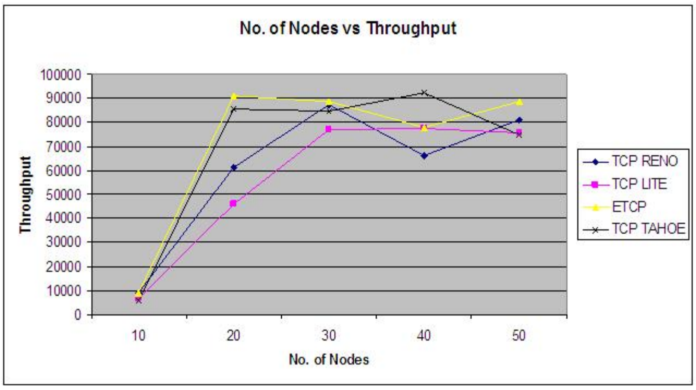

Figure 12

From results of implementation it has be seen that the performance of ETCP is better in high density node because in this condition sender can get different paths through different nodes.

Packets can be corrupted due to congestion in network or heavy traffic, due to this reason packet drop is fewer in less density network.

Signals also affected due to improper routing path in network, so maximum signal were distorted when nodes in network are moving with speed of 5m/sec.

Throughput depends on number of packet received and number of packets loss. It is better in high density network, because receiver can receive maximum packets when number of node in network is 50.

Future Work

The performance of ETCP can be measured in different wireless environments especially those with high error rates. It can also be plan to build a flexible and lightweight transport protocol for the wireless side of ETCP which can adapt to changes in the wireless environment and can support planned disconnections. Presentation layer services can also be built on top of ETCP which will allow mobile applications to dynamically choose a format for data transmitted over the wireless medium. Other work includes testing throughput intensive applications such as ftp and mosaic with ETCP.

References

[1] J. Postel, `Transmission control protocol`, RFC 793, Sept.1981.

[2] A. Gurtov and S. Floyd, `Modeling wireless links for transport protocols`, ACM SIGCOMM Comput. Commun. Rev., vol. 34, no. 2, pp. 85-96, Apr. 2004.

[3] G. Holland and N. Vaidya, `Analysis of TCP performance over mobile ad hoc networks` in Proc. ACM/IEEE Int. Conf. on Mobile Computing, Seattle, WA, USA, Sept. 1999, pp. 219-230.

[4] Y. Shang and M. Hadjitheodosiou, `TCP splitting protocol for broadband and aeronautical satellite network` in Proc. 23rd IEEE Digital Avionics Syst. Conf., Salt Lake City, UT, Oct. 2004, vol. 2.

[5] J. Zhu, S. Roy, and J. H. Kim, `Performance modeling of TCP enhancements in terrestrial-satellite hybrid networks` IEEE/ACM Trans. Netw., vol. 14, no. 4, pp. 753-766, Aug. 2006.

[6] C.-H. Ng, J. Chow, and Lj. Trajkovic, `Performance evaluation of the TCP over WLAN 802.11 with the Snoop performance enhancing proxy`, OPNETWORK 2002, Washington, DC, Aug. 2002.

[7] W. G. Zeng, M. Zhan, Z. Li, and Lj. Trajkovic, `Improving TCP performance with periodic disconnections over wireless links`, OPNETWORK 2003, Washington, DC, Aug. 2003.

[8] W. G. Zeng and Lj. Trajkovic, `TCP packet control for wireless networks` in Proc. IEEE Int. Conf. on Wireless and Mobile Computing, Networking and Communications (WiMob 2005), Montreal, Canada, Aug. 2005, vol. 2, pp. 196-203.

[9] M. Omueti and Lj. Trajkovic, `M-TCP+: using disconnection feedback to improve performance of TCP in wired/wireless networks` in Proc. SPECTS 2007, San Diego, CA, USA, July 2007, pp. 443-450.

[10] Xu Chang-Biao, Long Ke-Ping, Yang Shi-Zhong. `Corruption-based TCP rate adjustment in wireless networks` In: Chinese Journal of Computers, 2002, 25(4): pp.438-444.

[11] Scalable Network Technology, `QualNet4.0 simulator` tutorial and QualNet Forum, http://www.scalable-networks.com/forums/

[12] K.Fall, S.Floyd, `Simulation Based Comparison of Tahoe, Reno and SACK TCP`, ACM SIGCOMM Computer Communication Review, Volume 26 Issue 3, July 1996.

[13] S. Floyd, `The NewReno Modification to TCP's Fast Recovery Algorithm`, RFC 3782, April 2004.

[14] B. Qureshi, M. Othman, Member, IEEE, and N. A. W. Hamid, `Progress in Various TCP Variants`, February 2009.

[15] M. Mathis and J. Mahdavi, `Forward acknowledgment: refining TCP congestion control`, in Proceedings of ACM SIGCOMM, pp. 181-191, 1996.

[16] Lawrence S. Brakmo, Student Member, IEEE, and Larry L. Peterson, `TCP Vegas: End to End Congestion Avoidance on a Global Internet`, OCTOBER 1995.