International Journal Of Engineering Research & Management Technology

ISSN: 2348-4039

Email: editor@ijermt.org May- 2014 Volume 1, Issue 3 Website: ijermt.org

Comparative Performance of Different TCP Variants Over Routing Protocols In MANET Using NS2

Arwinder Singh, Amit Kumar
Department of Computer Science & Engineering
Shree Siddhivinayak Group Of Institutions,
Bilaspur, Yamuna Nagar, Haryana, India

ABSTRACT
Adhoc network is a collection of mobile nodes and wireless communication network is used to connect these mobile nodes. This type of network is known as MOBILE ADHOC NETWORK (MANET). TCP (Transmission Control Protocol) was designed to provide reliable end to end delivery of data over unreliable network. Traditionally TCP assumes that all the packet losses are due to congestion. Most TCP deployments have been carefully designed in the context of wired networks. Ignoring the properties of wireless Adhoc networks can lead to the implementations with poor performance. In order to adapt TCP to wireless network, improvements have been proposed in the literature to detect different types of losses. Indeed, in mobile or static Adhoc networks losses are not always due to network congestion, as it is in case of the wired networks. So, we presented some simulations analysis of TCP, TCP New Reno, TCP SACK, TCP Vegas, and TCP Westwood over DSDV, DSR and AODV in ns2.
Keywords: TCP, MANET, AODV, DSR and DSDV

I. INTRODUCTION

A wireless local-area network (LAN) uses radio waves to connect devices such as laptops to the Internet and to our business network and its applications. When we connect a laptop to a Wi-Fi hotspot at a cafe, hotel, airport lounge, or other public place, we are connecting to that business's wireless network. Situations like disaster, military settings have led to Adhoc networks. Adhoc network is a collection of mobile nodes and wireless communication network is used to connect these mobile nodes. This network is known as MOBILE ADHOC NETWORK (MANET) [1]. Each device in a MANET moves independently. MANET is an infrastructure less network with no fixed Base Station for communication. Intermediate mobile nodes act as router to deliver the packets between nodes. So, MANET is a highly dynamic network. There are various commonly used routing protocols in manet like DSDV (Destination Sequence Distance Vector), DSR (Dynamic Source Routing) and AODV (Adhoc On-demand Distance Vector). TCP is widely used transport layer protocol which is commonly used for data services. So, it is certainly used over Mobile Adhoc networks.

TCP (Transmission Control Protocol) was designed to provide reliable end to end delivery of data over unreliable network. Most of the TCP deployments have been carefully designed in the context of wired networks [5]. Ignoring the properties of wireless Adhoc networks can lead to the implementations with poor performance. In order to adapt TCP to wireless network, improvements have been proposed in the literature to help TCP to differentiate between the different types of losses. Indeed, in Mobile Adhoc Networks losses are not always due to network congestion, as it is in the wired networks. Different TCPs may perform differently over different routing protocols [2]. So, it is very important to understand various TCPs over different routing protocols. In this paper, we have done performance analysis of TCP Reno, TCP New Reno, TCP SACK, TCP Vegas and TCP Westwood over different routing protocols.[3]The rest of the paper is organised as follows. Section II presents different TCP variants. Section III presents an overview of the Adhoc routing protocols. Section IV describes the simulation methodology. In Section V, an analysis of the simulation results is presented. Section VI concludes the paper, finally Section VII suggests direction for future work.

II. TCP VARIANTS OVER MANET

In Adhoc network, packet losses are not always due to network congestion. When a packet is detected to be lost in wired network, either by Timeout or by multiple duplicate ACKs, TCP slows down the sending rate by adjusting the congestion window [11]. But Wireless networks suffer from several types of losses that are not related to congestion only [3], making TCP not adapted to this environment. A lot of optimizations and improvements have been proposed to improve TCP performance over wireless networks. In this section, we study TCP Reno, TCP New Reno, TCP SACK, TCP Vegas and TCP Westwood.

TCP Reno introduces a phase called fast recovery phase. If three duplicate ACKs are received, Reno will perform a fast retransmit, and enter a state called fast recovery where it will halve the congestion window and retransmits the lost packet that was signalled by three duplicate acknowledgements [7] [8], and waits for an acknowledgment of the entire transmit window. If there is no acknowledgment or if an ACK times out, TCP Reno experiences a timeout and enters the slow-start phase where it will increase the congestion window.[16]

TCP Vegas is a TCP implementation which is a modification of TCP Reno. To detect a packet loss, it does not require enough duplicate acknowledgements, and it also suggests another slow start algorithm known as modified slow start which prevents it from congesting the network [14]. It does not depend on packet loss to detect congestion. But it detects congestion before the packet losses occur [10]. It calculates an estimate of the RTT. Whenever a duplicate acknowledgement is received it checks to see if the (current time segment- transmission time)> RTT estimate; if it is then it immediately retransmits the segment without waiting for three duplicate acknowledgements or a timeout. If we talk about Congestion avoidance phase, it does not use any signal of loss of segment, that there is congestion. But it determines congestion by a decrease in sending rate as compared to the expected rate. So whenever the calculated rate is too far away from the expected rate then it increases transmissions to make use of the available bandwidth, and whenever the calculated rate comes too close to the expected value it decreases its transmission to prevent over saturating the bandwidth.[4]

TCP New-Reno is a modification of TCP Reno. It can detect multiple packet losses and hence is much more efficient than Reno in the event of multiple packet losses. When it receives multiple duplicate packets, it also enters into fast-retransmit like Reno, but it doesn't exit from fast-recovery like Reno, until all the data which was outstanding at the time it entered fast recovery is acknowledged. So, it overcomes the problem of reducing the congestion window multiples times. The fast-transmit phase works same as the fast transmit phase of Reno. But the difference is it allows multiple re-transmissions in New-Reno [6].

TCP Westwood (TCPW) is a TCP congestion window algorithm that makes improvements over the performance of TCP Reno in wired as well as wireless networks by using sender-side modification. TCP Westwood uses end-to-end bandwidth estimation to check the cause of packet loss whether due to congestion or wireless channel effect which is a major problem in TCP Reno. The basic idea is to continuously measure the TCP source the by monitoring the rate of returning ACKs [12]. This measurement is then used to calculate the congestion window and slow start threshold after three duplicate acknowledgments or after a timeout. The idea of this strategy is very simple: As in TCP Reno, which directly halves the congestion window after receiving three duplicate ACKs, TCP Westwood then selects a slow start threshold and a congestion window which is consistent with the effective bandwidth used at the time congestion is experienced. This mechanism is known as "faster recovery".After getting 3 duplicate ACKs: ssthresh= (BW*RTTmin)/Segment size.

if (cwin>ssthresh) cwin=ssthresh
After timeout : ssthresh= = (BW*RTTmin)/Segment size.
cwin=1. This mechanism is very effective over wireless links where losses are due to radio channel problems.[13]
We've noticed that this design is inefficient: as only packet 2 was missed, the server was required to retransmit packets 3 and 4 as well, because the client had no way to confirm that it had received those packets. This problem can be solved by introducing the selective acknowledgment (SACK) TCP option. SACK works like it allows the client to say "I have not received data 2, but I have received data segment 3 and 4" [9] [15].

III. OVERVIEW OF ADHOC ROUTING PROTOCOL

In this Section, we study different routing protocols in MANET. In DSDV (Destination Sequence Distance Vector) [1], each mobile node in the network keeps a routing table. Each of the routing table includes all available destinations and the number of hops to reach that destination. Each entry in the routing table has a sequence number. If a Link is present then sequence number will be even otherwise odd number will be used. This number is generated by the destination, and the sender node should have to send out the next update with this number. The Periodic transmissions of updated routing tables help to maintain the topology information in the network. If there is any new and significant change in the network then the updates will be sent out immediately to the neighbours. So, the routing information updates may be periodic or when any topology change occurs. DSDV protocol each mobile node in the network will send its routing table to its current neighbours. This is possible either by broadcasting or by multicasting. By the advertisements, the neighbouring nodes can know whether any change has occurred in the network due to the movements of nodes. The routing updates can be sent in two ways, one is called a ''full dump'' and another is ''incremental.' In full dump, the entire routing table is transmitted to the neighbours, when change occurs in the topology. But in case of incremental update only the entries that are updated due to changes are sent.

In AODV (Adhoc On-demand Distance Vector), It is an On-Demand routing Protocol. Each Node maintains only the next hop information of the route to destination. Destination sequence number is used to check the freshness of the route to destination. Periodic use of Beacons i.e. Hello packets used to check the presence of the neighbour. Each node uses a sequence number which is increased whenever the node observe a change in neighbour topology [17]. Each node maintains a routing table and the information is stored as

<Destination IP address, Destination Sequence number, Next hop address, hop count to destination>
If a source node wants to send data to destination node and if it doesn't have a route to it then the source node will prepare a route request message consist of
<Source IP address, Source Sequence Number, Destination IP Address, Destination Sequence number, Hop count value=0, Broadcast ID>
The Broadcast ID is incremented whenever a source node generates a new route request packet. A <Broadcast Id, Source sequence number> is used as a unique identifier for a route request packet.
Whenever an intermediate node receives a route request message then it makes a reverse path for the source node that is the source node entry in the routing table of this node will be:
<Source IP, Source Sequence number, next hop Address, Hop count to source node>

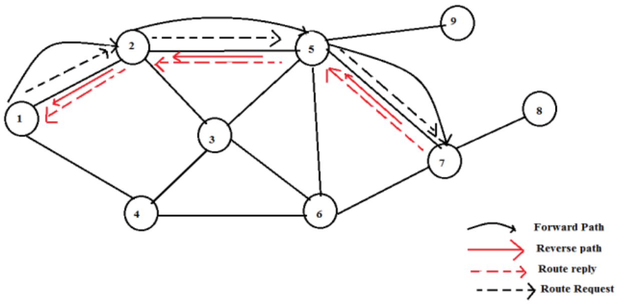

Fig.1: AODV Mechanism

If this intermediate node doesn't have route to destination then it will broadcast the route request packet to its neighbours. On the basis of Broadcast ID and source sequence number present in the route request packet, it will avoid the multiple transmission of duplicate route request packet [19]. The validity of a node is determined by comparing the destination sequence number recorded for the destination node in its table with the destination sequence number present in the route request packet. If the stored entry is higher than the value present in the packet, it means the node has a valid route to destination and so it will create a route reply packet. When an intermediate node receives a route reply packet then it also creates a forward path for the destination node. This forward path is used by the source node to transmit the data packets to destination.

In DSR (Dynamic Source Routing) [1], the basic idea of DSR is that, it uses the concept of source routing where the sender knows the complete hop-by hop route to the destination. In this protocol, all the mobile nodes are required to maintain route caches which contain the route to other nodes. The route cache is updated only when any new route is maintain/update for a particular entry in the route cache. The data packets carry the source route in the packet header. Routing in DSR is done in two phases: route discovery and route maintenance. Suppose there is a source node and wants to send a data packet to a destination, it first checks its route cache to determine whether its cache already contains any route to the destination or not. If there is already an entry for that destination, the source uses that route to send the packet. If not, then the source node broadcast a route request packet which includes the destination address, source address, and a unique request ID. Each intermediate node checks whether route is available or not. If the intermediate node does not know the route to destination, it adds its own address to route request packet and forwards the packet and to other nodes eventually this reaches the destination. The node processes the route request packet only if it is not previously processed that packet. A route reply is generated by the destination or by any of the intermediate nodes which knows the route to destination. Another Phase is Route Maintenance which is done by using the route error packet (RERR) and acknowledgements. Route error packets are generated by a node if there is any Link break occurs or any other error in the route. When a route error packet is received by node, the hop in error is removed from the route cache.

IV. SIMULATION METHODOLOGY

In this Paper, we have used simulations to study the performance of different TCP variants over three Adhoc routing protocols. We have carried out the simulations in Network Simulator (NS-2) from Lawrence Berkeley National Laboratory (LBNL) with extensions from MONARCH project at Carnegie Mellon University. [3]

At the physical layer, the extended NS employs a radio propagation model supporting propagation delay, Omni-directional antennas, and a shared media network interface. The IEEE 802.11 Medium Access Protocol is employed at the Link Layer level.

For performance analysis, we present the simulation of TCP Reno,TCP New Reno, TCP Vegas, TCP Westwood and TCP Sack over AODV, DSR, DSDV in MANET. The Simulation [18] shows 25 nodes and data transfer between these nodes.It also shows the X-graphs of Throughput in simulation. The implementation is performed in NS2 and analysis is presented using X-graphs. Different TCPs perform well over different routing protocols.

We have employed six kinds of TCP variants over three different Routing protocols. TCP variants used are traditional TCP, TCP Reno, TCP New Reno, TCP Vegas, TCP SACK and TCP Westwood. Every TCP variant has different properties due to which they perform differently over different routing protocols. We have used a Throughput metric to compare all TCP variant over AODV, DSDV and DSR.

V. SIMULATION RESULTS

In Fig. 2, 3 and 4 we show the comparison of different TCP variant over AODV, DSDV and DSR respectively with the throughput Vs time graph.

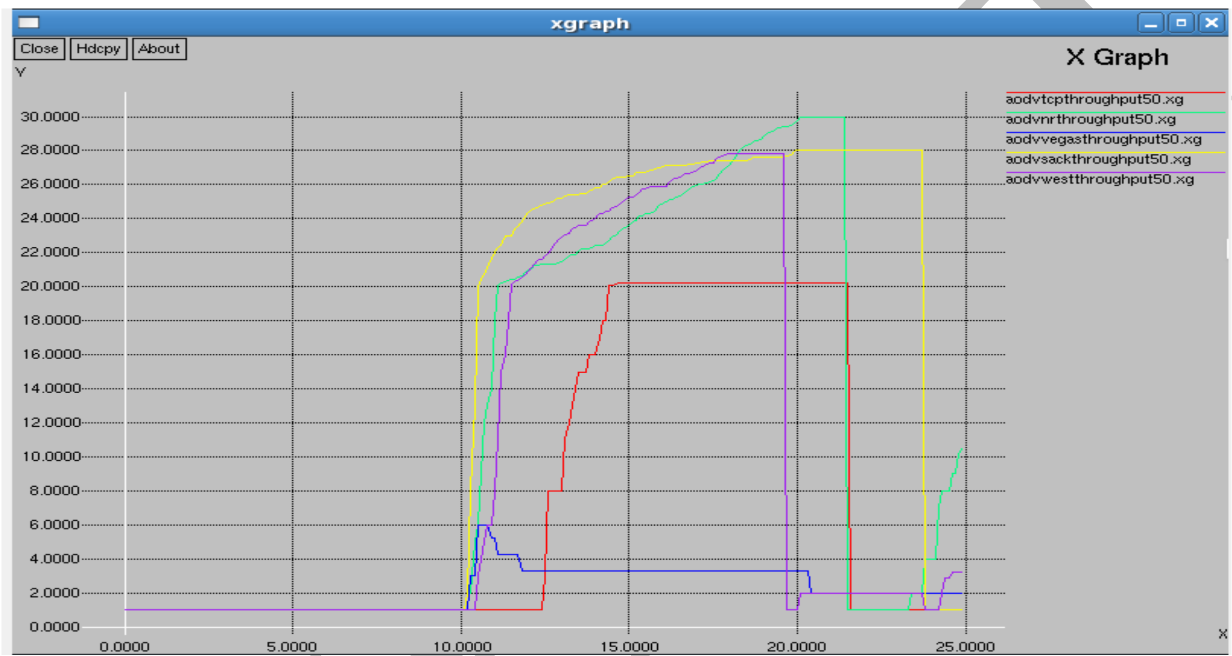

Fig. 2: Throughput Vs time graph for different TCP variants over AODV

Fig. 2 shows the simulation analysis and comparison of throughput among traditional TCP, TCP New Reno, TCP Vegas, TCP SACK, and TCP Westwood over AODV routing protocol. TCP New Reno performs very well in AODV.

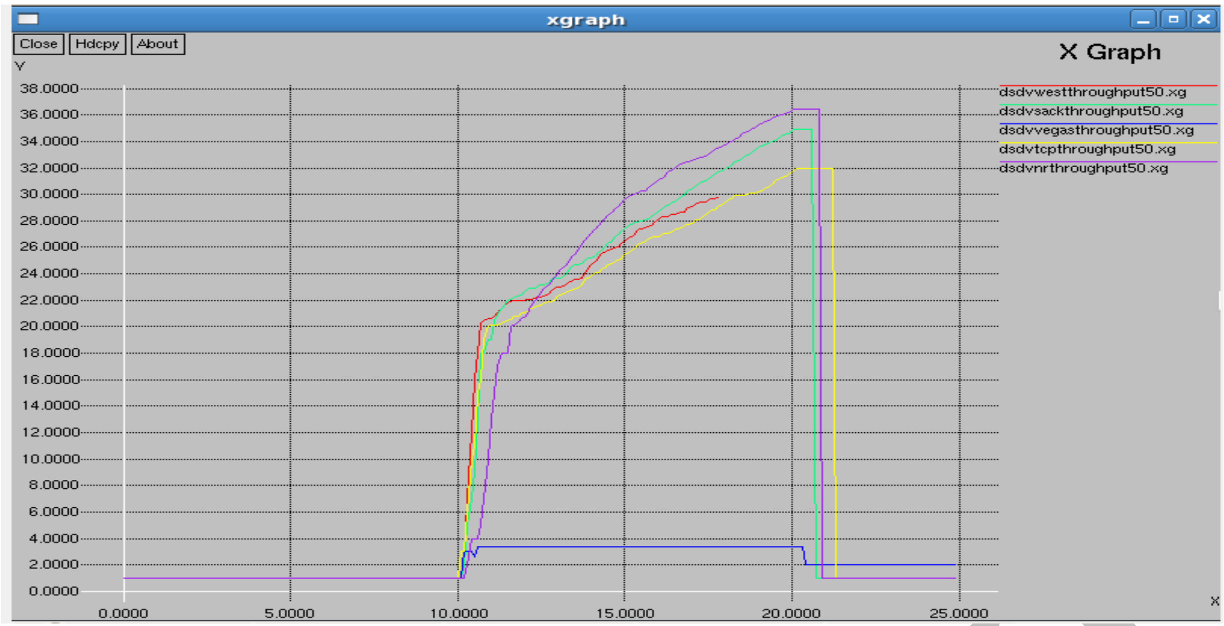

Fig. 3: Throughput Vs time graph for different TCP variants over DSDV

Fig. 3 shows the simulation analysis and comparison of throughput among TCP, TCP New Reno, TCP Vegas, TCP SACK, and TCP Westwood over DSDV routing protocol. TCP Westwood performs very well in DSDV.

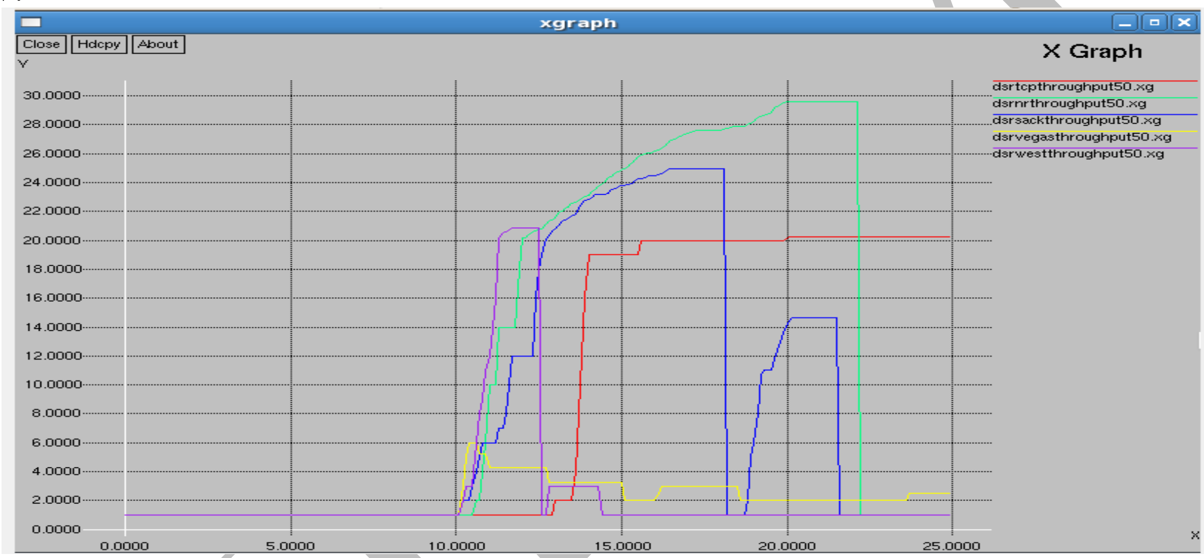

Fig. 4: Throughput Vs time graph for different TCP variants over DSR

Fig. 4 shows the simulation analysis and comparison of throughput among traditional TCP, TCP New Reno, TCP Vegas, TCP Sack, and TCP Westwood over DSR routing protocol. TCP Westwood performs very well in DSR.

VI. CONCLUSION

Traditional TCP assumes that all the packet loss in network is due to congestion.But this not always valid in Adhoc wireless network. So various improvment in TCP developed. In this Paper, we studied and compared different TCP variants over different routing protocols. In this Paper,we present the simulation of traditional TCP, TCP New Reno, TCP Vegas, TCP Westwood and TCP Sack over routing protocols like AODV, DSR, DSDV in MANET. The simulation shows the 50 nodes and data transfer between these nodes. It also shows the X-graphs of Throughput in simulation. The implementation is performed in NS2 and analysis is presented using X-graphs. Different TCP variants perform well over different routing protocols. Like TCP westwood performs well in DSDV and DSR but TCP New Reno performs well in AODV routing protocol.

VII. FUTURE WORK

In this Paper, we have compared TCP Reno, New Reno, Vegas, SACK and Westwood over three different routing protocols that are AODV, DSR and DSDV. Different TCP variants perform well over different routing protocols. In future we will try to find such TCP variants which will perform best with any wired or wireless routing protocol.

REFERENCES

1. C. Siva Ram Murthy and Manoj, B.S. Second Edition, Low price Edition, Pearson Education, 2007.Adhoc Wireless Networks, Architectures and Protocols.
2. Dong kyun Kim , Juan-Carlos Cano and P. Manzoni, C-K. Toh," A comparison of the performance of TCP-Reno and TCP-Vegas over MANETs", 1-4244-0398-7/06/$20.00 ©2006 IEEE
3. Foez ahmed, Sateesh Kumar Pradhan, Nayeema Islam, and Sumon Kumar Debnath," Performance Evaluation of TCP over Mobile Adhoc Networks" , (IJCSIS) International Journal of Computer Science and Information Security,Vol. 7, No. 1, 2010.
4. Xiang Chen, Hongqiang Zhai, Jianfeng Wang, and Yuguang Fang ,"TCP performance over mobile ad hoc networks",CAN. J. ELECT. COMPUT. ENG., VOL. 29, NO. 1/2, JANUARY/APRIL 2004
5. GAVIN HOLLAND NITIN VAIDYA," Analysis of TCP Performance over Mobile Ad Hoc Networks", Wireless Networks 8, 275–288, 2002 ㄥ□ 2002 Kluwer Academic Publishers. Manufactured in the Netherlands
6. Harpreet Singh Chawla, M. I. H. Ansari, Ashish Kumar, Prashant Singh Yadav," A Survey of TCP over Mobile ADHOC Networks", International Journal of Scientific & Technology Research Volume 1, Issue 4, May 2012 ISSN 2277-8616
7. Bogdan Moraru Flavius Copaciu Gabriel Lazar Virgil Dobrota," Practical Analysis of TCP Implementations: Tahoe, Reno, NewReno"
8. Ashish Ahuja, Sulabh Agarwal, Jatinder Pal Singh, Rajeev Shorey," Performance of TCP over Different Routing Protocols in Mobile Adhoc Networks,"0-7803-571 8-3/00/$10.00 *02000* IEEE.
9. Laxmi Subedi, Mohamadreza Najiminaini, and Ljiljana Trajković," Performance Evaluation of TCP Tahoe, Reno, Reno with SACK, and NewReno Using OPNET Modeler"
10. AHMAD AL HANBALI, EITAN ALTMAN, AND PHILIPPE NAIN, INRIA SOPHIA ANTIPOLIS FRANCE," A SURVEY OF TCP OVER AD HOC NETWORKS ",1553-877X IEEE Communications Surveys & Tutorials • Third Quarter 2005.
11. Chaoyue Xiong, Jaegeol Yim, Jason Leigh and Tadao Murata, "Energy-Efficient Method to Improve TCP Performance for MANETs"
12. Ahmad Al Hanbali, Eitan Altman, Philippe Nain," A Survey of TCP over Ad Hoc Network.
13. Mario Gerla, M. Y. Sanadidi, Ren Wang, and Andrea Zanella "TCP Westwood: Congestion Window Control Using Bandwidth Estimation"
14. L.S. Brakmo, L.L. Peterson, "TCP Vegas: End to End Congestion Avoidance on a Global Internet", IEEE Journal on Selected Areas in Communication, vol. 13[1995], (1465-1490).
15. David X. Wei, Pei Cao, "NS-2 TCP-Linux: An NS-2 TCP Implementation with Congestion Control Algorithms from Linux"
16. C. Casetti, M. Gerla, S. Mascolo, M. Sansadidi, and R. Wang. TCP Westwood: end-to-end congestion control for wired/wireless Networks. Wireless Networks Journal, 8:467–479, 2002.
17. Avinka Baweja , Kushagra Agrawal, Sanjeev Rana, Rajneesh Kumar Gujral," Network Layer Attacks and its Countermeasure in Mobile Ad hoc Networks" International Conference on Telecommunication and Networks (TEL-NET 2013).
18. NETWORK SIMULATOR (ns2). http://www.isi.edu/nsnam/ns/.
19. Charles Perkins, Elizabeth Royer, and Samir Das., "Adhoc on demand distance vector (AODV) routing", July 2003.