International Journal of Advanced Computer Research (ISSN (print): 2249-7277 ISSN (online): 2277-7970)
Volume-3 Number-3 Issue-12 September-2013

Performance Analysis of Some TCP Variants for Routing over Mobile Ad hoc Networks

Ibrahim K. Tabash[1], M. Salim Beg[2], Nesar Ahmad[3]

Abstract

TCP was designed for wired networks and the sender assumes that packet loss is an indicator of network congestion, but this assumption may not apply to Mobile Ad hoc Networks (MANETs). In Mobile Ad Hoc networks, performance of the standard TCP is significantly degraded due to characteristics of MANET such as route failures due to node mobility and link errors. In this paper, the authors investigate the effects of node's mobility on the performance of TCP variants such as Reno and Vegas. Reno views the packet loss as signal of network congestion, while Vegas uses the difference in the expected and actual throughput rates as network congestion indicator. Simulation results from the implementation of different static and dynamic scenarios have been obtained. Different routing protocols such as Ad hoc On-Demand Distance Vector (AODV) and Destination Sequence Distance Vector (DSDV) have been investigated to obtain the performance of TCP variants in this paper.

Keywords

TCP, MANET, TCP Reno, TCP Vegas, AODV, DSDV.

1. Introduction

Mobile Ad hoc Networks (MANETs) are a collection of mobile nodes forming a dynamic autonomous network. Nodes communicate with each other without the intervention of centralized access points or base stations. In such a network, each node acts both as a router and as a host. A MANET has several advantages over traditional wireless networks, including ease of deployment, speed of deployment, and decreased dependence on a fixed infrastructure.

Ibrahim K. Tabash, Aligarh Muslim University (AMU), Aligarh-202002, UP India.
M. Salim Beg, Aligarh Muslim University (AMU), Aligarh-202002, UP India.
Nesar Ahmad, Aligarh Muslim University (AMU), Aligarh-202002, UP India.

Typical applications of MANETs include personal communication with laptops and PDAs, group communication at conferences and presentations, communication in military, between moving vehicles and in emergency situations. Though MANETs are becoming extremely popular with the advent of various types of mobile devices; rapidly changing connectivity, network partitions, higher error rates, security threats, frequent collision probability, bandwidth and power constraints together pose new problems in designing protocols.

This paper focuses on the issues related to reliability of data transport over MANETs; Research on efficient transport protocols for MANETs is one of the most active topics in MANETs community [1-4]. TCP is affected by node mobility and link errors. There are several proposals that suggest either a new transport protocol or enhancements to the traditional transmission control protocol (TCP) to work efficiently in MANETs [2].

The rest of this paper is organized as follows. A brief introduction to TCP protocol operations is given in section II. In section III, an overview of routing protocols is presented. Section IV presents the simulation environment and topologies. Section V gives discussion and analysis. Finally, summary and conclusion of the paper are given in section VI.

2. Overview of TCP operations

The Transmission Control Protocol (TCP) [2] is a reliable, end-to-end, connection-oriented transport layer protocol that provides a byte-stream based service. TCP implements flow control by means of sliding window algorithms Reno and Vegas [4, 5], which make use of two phases, the SS (slow start) and CA (Congestion Avoidance) algorithm to adjust the window size. The following paragraphs give a brief primer on TCP Reno and Vegas mechanisms with its characteristics.

TCP Reno in slow start phase increases the congestion window by one MSS (Maximum Segment Size) on receiving an acknowledgment (ACK) packet which indicates a successful reception of a data packet by the receiver, and induces packet losses to estimate the available bandwidth in the network. The congestion avoidance phase is adopted by TCP Reno, in which for each arrival of an ACK, TCP increases the congestion window by a fraction of MSS.

TCP Vegas uses the difference between expected and actual throughput rates to estimate the available bandwidth in the network. The idea is that when the network is not congested, the actual throughput rate will be close to the expected throughput rate, otherwise, the actual rate will be smaller than the expected rate [6]. TCP Vegas, using this difference in throughput rates, estimates the congestion level in the network and updates the congestion window size accordingly. Note that this difference in the throughput rates can be easily translated into the difference between the congestion window size and the number of acknowledged packets during the Round Trip Time RTT, using the equation:

$$Diff = (Expected - Actual) * BaseRTT$$

Where *Expected* is the expected rate, *Actual* is the actual rate, and *baseRTT* is the minimum round trip time. The following steps show the details of the algorithm:

Step-1: First, the source computes the expected throughput rate $pected = \frac{CWND}{BaseRTT}$, where *CWND* is the current window size and *BaseRTT* is the minimum round Trip Time.
Step-2: Second, the source estimates the current throughput rate by using the actual round trip time according to
$Actual = \frac{CWND}{RTT}$, where *RTT* is the actual round trip time of a packet.
Step-3: The source, using the expected and actual flow rates, computes the estimated backlog in the queue from:

$$Diff = (Expected - Actual) * BaseRTT$$

Step-4: Based on *Diff*, the source updates its window size as follows:

$$CWND = \begin{cases} CWND + 1 & if\ Diff < \alpha \\ CWND - 1 & if\ Diff > \beta \\ CWND & otherwise, \end{cases}$$

Where α and β minimum and maximum bandwidth thresholds.

3. Routing in MANETS

Routing protocols in MANET are categorized as: proactive and reactive routing protocols. In the following subsections a brief description for each of them is given.

3.1 Proactive (Table-Driven) Routing Protocols

In this category, routing protocols such as Destination Sequence Distance Vector (DSDV) [7] attempt to maintain consistent and up-to-date routing information from each node to every other node in the network. This kind of approach has the property of lower latency and higher overhead. In DSDV the routes to all destinations are readily available at every node at all times. Here messages are passed between nodes to maintain their routing table. The routing table of each node consists of the shortest paths to all destinations from it. Each routing table entry in DSDV is tagged with a sequence number which is provided by the destination node. This is used to avoid the count-to-infinity problem associated with distance-vector protocols. The messages used in DSDV to maintain routing table are:

- Periodic update messages
- Triggered update messages

The periodic update messages are those in which the whole routing table is transmitted to all neighbours of a node, at regular intervals of time. Triggered update messages are transmitted when there is any change in network topology. In triggered update message only significant changes in routing table are transmitted between the nodes.

3.2 Reactive (On-Demand) Routing Protocols

In this category, routing protocols such as Ad hoc On-Demand Distance Vector (AODV) [8] source-initiated on-demand routing which creates routes only when desired by the source node. When a node requires a route to destination, it initiates a route discovery process within the network. In general, on-demand routing protocols are characterized by higher latency and lower overhead. AODV includes loop freedom and that link breakages cause immediate notifications to be sent to the affected set of nodes. Additionally, AODV has support for multicast routing and avoids the Bellman Ford "counting to infinity" problem. The use of destination sequence numbers guarantees that a route is fresh.

AODV uses different messages to discover and maintain links. Whenever a node wants to try and find a route to another node, it broadcasts a Route

Request (RREQ) to all its neighbours. The RREQ propagates through the network until it reaches the destination or a node with a fresh enough route to the destination, then the route is made available by unicasting a RREP back to the source. The algorithm uses hello messages (a special RREP) that are broadcasted periodically to the immediate neighbours. These hello messages are local advertisements for the continued presence of the node and neighbours using routes through the broadcasting node will continue to mark the routes as valid. If hello messages stop coming from a particular node, the neighbour can assume that the node has moved away and mark that link to the node as broken and notify the affected set of nodes by sending a link failure notification (a special RREP) to that set of nodes.

4. Simulations and experiments

This section presents the simulation environment setting to compare TCP variants (Reno, Vegas) in MANETs over DSDV and AODV routing protocols. First, consider a chain topology, then consider a grid topology and finally, evaluate the impact of nodes' mobility. All simulations in this paper have been carried out using NS-2 [9]. Both static scenarios (e.g. chain topology and grid topology with no mobility) as well as dynamic scenario with mobile nodes have been investigated in this work through simulation.

4.1 Chain Topology

A chain topology with 1, 2, 4, 6, 8 and 10 hop counts between the source and destination nodes is considered here in the simulations. Fig. 1 shows the topology with 4 hop count used in the simulation. Experiments in this scenario ran over AODV and DSDV respectively as routing protocols, and IEEE 802.11 as MAC layer protocol. The packet size is set to 1460 byte and one single connection is generated by FTP applications and starts at time 5 s and ends at time 399.5 s over TCP Reno and Vegas respectively. Simulation time is 400 s and the transmission range is set to 100 meter and the distance between the nodes is such that only adjacent nodes were within the transmission range of each other. In all experiments node *S* was the sender and *D* the destination.

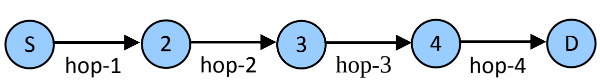

Fig 1: Chain topology with 4 hops

4.2 Grid Topology

The grid topology, considered 21 nodes in grids, where the distance between the horizontal and vertical adjacent nodes is 100 meter; we set up six FTP connections, each of which is 4-hops long. Each FTP flow starts at time 5 s and ends at time 399.5 s. As in the case of chain topology discussed above, experiments ran over AODV and DSDV respectively. IEEE 802.11 as MAC Layer protocol. The packet size is 1460 byte; Fig. 2 shows the topology used in the simulation and simulation time 400 s.

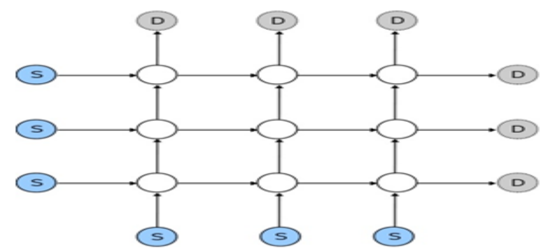

Fig 2: Grid topology

4.3 Mobile Scenario

The considered mobile network consisted of 50 nodes moving over a 1000 x 1000 meter. Table 1 shows the parameters and their values for mobile scenario.

Table 1: Parameters for the mobile scenario

No.	Parameters	Values
1	Area	1000 X 1000 m
2	Transmission range	250 m
3	Pause time	20 s
4	Mobility	20 m/s
5	Simulation time	400 s
6	Mac layer	IEEE 802.11
7	Application Layer	FTP
8	Transport layer	TCP Reno, Vegas
9	No. of connection	5 flows
10	No. of nodes	50
11	Routing protocols	AODV, DSDV

5. Results analysis and discussion

The work presented in this paper is an extension of an earlier work by the present authors in which only the throughput was considered as a performance metric [10]. This paper however has a much wider scope as it incorporates substantial changes to the simulation parameters and the performance metrics.

The results presented in this paper are obtained through simulation using NS-2.

This study considers the following as the performance metrics:

Throughput: is the average number of bytes successfully received by the final destination per unit time.

Fairness index: is how well the network delivers packets from the source to the destination. A network should demonstrate some degree of fairness to each traffic flow. The network should share the available bandwidth equally among competing flows. We use the following equation to compute the fairness index [11] to assign a fairness index to a set of throughputs $X = (x_1, x_2, \ldots, x_n)$.

$$f(X) = \left(\sum_{i=1}^{n} x_i \right)^2 \Bigg/ n \sum_{i=1}^{n} x_i^2$$

Since throughput is a non-negative value, the fairness index will always lie between 0 and 1. If all users receive equal throughput, the fairness index is 1.

Packet loss percentage: is the ratio of total bytes retransmitted to the total bytes transmitted expressed as a percentage and is also an important metric to be considered in MANETs. As a reliable transport protocol recovers the lost packets by several retransmissions, when the loss is high, the packet retransmissions consume a considerable amount of battery power. Due to this, the node may drain off its battery faster, thereby reducing the network lifetime.

Packet delay: is a measure of the total round-trip time (RTT) associated with packet. RTT is the length of time it takes for a packet to propagate through the link plus the length of time it takes for an acknowledgment of that packet to be received.

We discuss the performance of TCP protocols (Reno/Vegas) with other routing protocols (AODV/DSDV). In a chain topology, a single TCP (Reno/Vegas) connection has been established between a chosen pair of sender and receiver nodes and the throughput has been measured over the lifetime of the connection. The throughput is used as the performance metric in this chain topology.

Fig. 3 shows the measured throughput obtained as a function of number of hops. It shows that the throughput decreases rapidly when the number of hops is increased from 1, and then stabilizes once the number of hops becomes large. This happens due to the characteristics of MAC 802.11. The bi-directional nature of TCP (Data, ACK) results in significant increase in channel contention among neighbouring nodes. For example, as path length increases, more nodes must contend for channel access in order to forward both data and ACK packets. The increased channel contention results in MAC-Layer ACK timeouts, causing packet retransmissions. Further, to avoid or reduce packet collisions at the receiving node, the MAC-Layer 802.11 transmits a request-to-send (RTS) and waits for clear-to-send (CTS) message from each of its neighbours. Thus, the transmitter cannot send data until it receives a CTS message from its neighbours. If the CTS is not received by some time, *t*, the sender station times out and must re-enter contention phase, resulting in significant delay and packet losses [12].

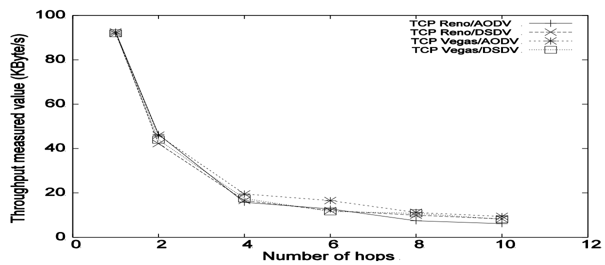

Fig 3: Measured throughputs for chain topology as a function of number of hops

Figs. 4, 5, and 6 show the throughput average, fairness index and packet delay average of TCP (Reno/Vegas) over AODV and DSDV respectively for the Grid topology. In the grid topology Figs. 4 and 5 show there are not significant differences between TCP Reno over AODV or DSDV. This was expected since in this topology problems related to link-layer contention are managed efficiently by the 802.11 MAC protocol. All nodes are within the transmission range of each other and can thus coordinate their transmissions efficiently. In TCP Vegas the throughput was degraded over AODV. The primary reason for this trend is due to inaccurate estimation of the *baseRTT* [6]. Since TCP Vegas uses *baseRTT* as an estimate of the propagation delay of route, its performance is sensitive to the accuracy of *baseRTT*. Fig. 6 shows a major difference in term of packet delay average between Reno and Vegas. This is due to the differences between Reno congestion control operations and Vegas congestion control operations.

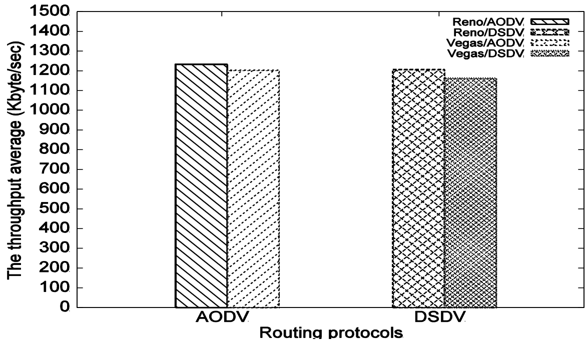

Fig 4: Throughput average for grid topology

In mobile scenario, Figs. 7, 8, 9 and 10 show the throughput average, fairness index, packet loss average and packet delay average over AODV and DSDV respectively for the mobile topology. These figures show TCP performance over AODV much higher than DSDV. This is because DSDV is dependent on periodic broadcasts and it needs some time to converge before a route can be used. This convergence time can probably be considered negligible in a static topology (chain and grid) topology, where the topology is not changing so frequently. In mobile topology on the other hand, where the topology is expected to be very dynamic, this convergence time will probably mean a lot of dropped packets before a valid route is detected [13]. The periodic broadcasts also add a large amount of overhead into the network. Since DSDV is dependent on its periodic updates, its ability to deal with a dynamic topology is very poor. Its ability to detect fast changes is poor, and it takes a lot of time to converge.

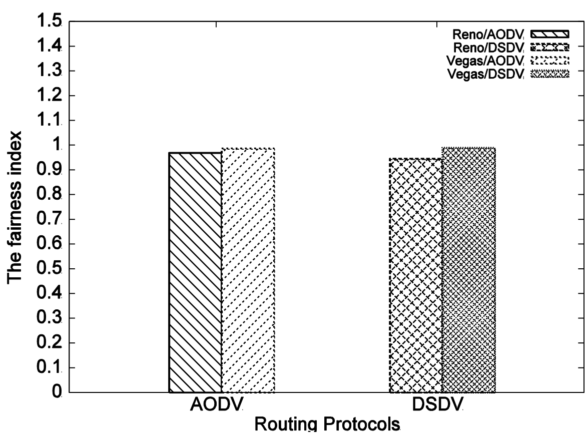

Fig 5: Fairness index for grid topology

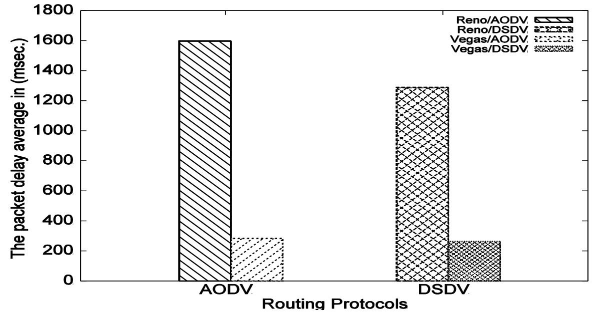

Fig 6: Packet delay average for grid topology

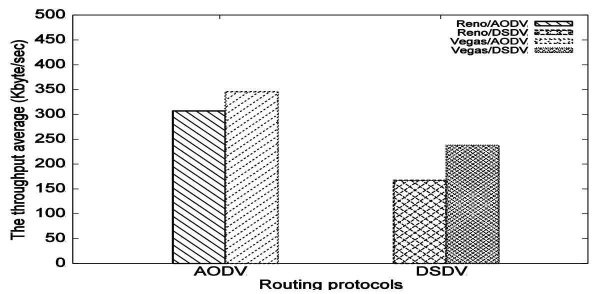

Fig 7: Throughput average for mobile scenario

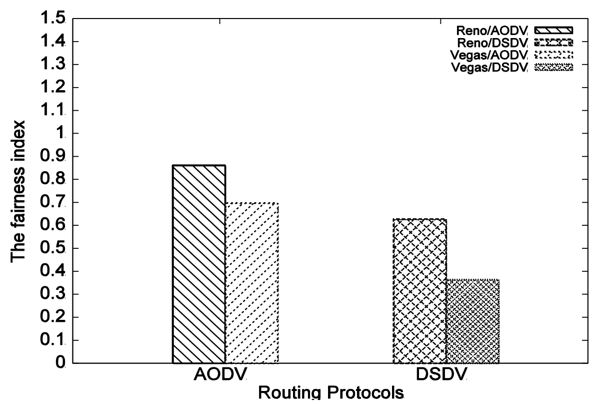

Fig 8: Fairness index for mobile scenario

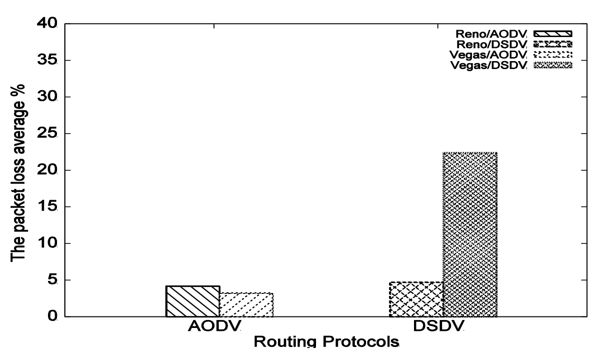

Fig 9: Packet loss average for mobile scenario

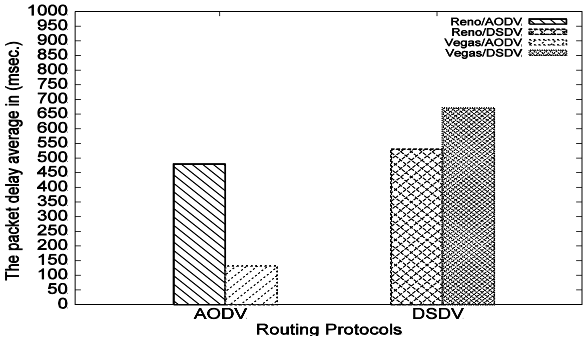

Fig 10: Packet delay average for mobile scenario

6. Conclusions

This paper has investigated and analysed the performance of two different TCP variants (Reno and Vegas) using two routing protocols (AODV and DSDV) in two environments static and dynamic topologies for MANETs. Where in both static and dynamic environments have been studied. Through simulation, it has been found that TCP performance decreases significantly when node movement causes link failures, due to TCP's inability to recognize the difference between link failure and congestion. Also, the simulation results show that the TCP Reno has better performance than TCP Vegas; this is because of TCP Vegas characteristics and bandwidth estimation mechanism. TCP performance over AODV has been found to be better than DSDV for the dynamic case i.e. mobile scenario.

References

[1] G. Anastasi, Emilio A., Marco C. and A. Passarella, 2009 Design and performance Evaluation of a Transport Protocol for Ad hoc Networks. The Computer Journal, vol. 52, no.(2), pp. 187-209.

[2] Christian L, Scheuermann B. Martin M. 2007 A survey on Congestion Control for Mobile Ad-Hoc Networks. Wiley Wireless Communications and Mobile Computing, vol. 7, no (5), pp. 655-676.

[3] B. Venkata Ramana, B. S. Manoj, and C. Siva Ram Murthy 2005 Learning-TCP: A Novel Learning Automata Based Reliable Transport Protocol for Ad hoc Wireless Networks. In Proc. of IEEE Broadnets, pp. 484 - 493.

[4] Zhenghua Fu, Xiaoqiao Meng and Songwu Lu 2002 How Bad TCP Can perform In Mobile Ad hoc Networks. In Proc. 7th ISCC'02.

[5] Foes Ahmad Sateesh K.P., N. Islam, S.K. Dehnath 2010 Performance Evaluation of TCP over Mobile Ad hoc Networks. IJCSIS, vol. 7, no. 1, pp. 140-146.

[6] Jeonghoon M., Richard J. La, Venkat A. and W. Jean, "Analysis and Comparison of TCP Reno and Vegas," in *Proc. IEEE INFOCOM '99*, 1999, pages 1556-- 1563, March 1999.

[7] C. Perkins and P. Bhagwat 1994 Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers. ACM SIGCOMM'94 Conference on Communications Architectures, Protocols and Applications, pp. 234-244.

[8] S. D. C. Perkins, E. Belding-Royer 2003 Ad hoc On-Demand Distance Vector Routing," RFC 3561.

[9] K. Fall and K. V (editors) 2001 Ns notes and documentation. The VINT Project, www.isi.edu/nsnam/ns/ns-documentation.html.

[10] I. K. Tabash, Nesar Ahmad, M. Salim Beg 2010 Performance Evaluation of TCP Reno and Vegas over different routing protocols for MANETs. In Proc. 4th IEEE International Symposium on Advanced Networks and Telecommunication Systems, Mumbai, Dec. 16-18 2010, pp. 82-84, Also in IEEE Explore.

[11] R. Jain 1991 The Art of Computer System Performance Analysis. Wiley, New York.

[12] Dmitri D, Perkins, Herman H. 2002 investigating the performance of TCP vin Mobile Ad Hoc Networks. Computer Communication, vol. 25, pp. l132-1139.

[13] Tony Larsson, Nicklas Hedman 1998 Routing Protocols in Wireless Ad hoc Networks A simulation Study M. Eng. thesis, LULEÅ University of Technology.