

Analyzing the TCP Performance on Mobile Ad-Hoc Networks

Pin-Chuan Liu^{ab}, Da-You Chen^c, Chih-Lin Hu^c, Wei-Cheng Sun^a, Jen-Hwa Lee^d, Chung-Kuang Chou^e,
Wei-Kuan Shih^b

^a Industrial Technology Research Institute (ITRI), Taiwan, R.O.C.

^b Department of Computer Science, National Tsing Hua University, Taiwan, R.O.C.

^c Department of Comm. Engineering, National Central University, Taiwan, R.O.C.

^d Department of Computer Science and Information Engineering, National Cheng Kung University, Taiwan, R.O.C.

^e Department of Electrical Engineering, National Taiwan University, Taiwan, R.O.C.

{flash, dmvector}@itri.org.tw, clhu@ce.ncu.edu.tw, {ro87498, mignsogo, leejenhwa1095}@gmail.com,
wshih@cs.nthu.edu.tw

Abstract—Transmission Control Protocol (TCP) is a connection-oriented protocol for reliable end-to-end data delivery over IP networks. Many TCP variants assume that packet loss occurs due to network congestion when wired networks have very low bit error rate. TCP reacts to unreliable network conditions by slowing down the data flow with congestion window adjustment. However, the nature of frequent noise in wireless networks heavily affects the conditions of packet loss, bit error and spurious packets. In mobile ad-hoc networks, multi-hop path forwarding further exacerbates the packet loss and throughput performance. Thus, this paper aims to analyze the TCP performance in wireless, and mobile ad hoc environments. This study first examines the effects of TCP throughput under different scales of data block sizes and TCP window sizes, respectively. This study also examines the packet delivery rate, hop counts, and flow counts with two practical routing protocols, AODV and OLSR, in mobile ad-hoc network environments. Experimental results show the sensitivities of TCP performance to various measure parameters in 802.11 wireless and ad-hoc networks.

Keywords—TCP throughput, performance measure, 802.11, wireless network, ad-hoc network.

I. INTRODUCTION

In seven-layer OSI model of computer networking, Transmission Control Protocol (TCP) [1] is the most well-known protocol of the transport layer which provides data delivery service at an intermediate level between an application program and the Internet Protocol (IP). Whenever an application program desires to send data to opposite end-host, the application program sends a large chunk of data to TCP and let TCP handle the IP data delivery. TCP breaks the data into pieces and issues a series of requests to IP. The IP delivers pieces of data, called packets. A packet consists of an IP header followed by a body. The IP header describes destination of this packet and the body contains the data received from TCP.

Due to unpredictable network situations, the IP packets can be lost, duplicated, or delivered out of order. TCP is a connection-oriented protocol which provides a reliable data delivery service between two network hosts. TCP deals with the network problems by requesting retransmission of lost

packets, rearranging out-of-order packets, and even running congestion control mechanisms to reduce the occurrence of network congestion. Whenever the end host has finally received the original chunk of data, it passes data to the application program. Notice that TCP has been optimized for wired networks. Since the bit error rate is very low in wired network, many TCP variants assume that packet loss occurs during network congestion. When TCP detects packet losses, the congestion window size is dramatically decreased to slow down the data delivery to avoid or control the congestion inside the network.

However, the nature of frequent noise in wireless networks heavily affects the conditions of packet loss, bit error and spurious packets. TCP slows down the sending rate as decreasing the congestion windows size in response to any wireless packet loss. This causes the bandwidth of radio link to be underutilized. Unfortunately, this circumstance becomes worse in mobile ad-hoc networks (MANETs), while the packets are routed over different multi-hop paths. Multi-hop forwarding can augment the packet loss probability and influence TCP performance to a significant extent.

This paper addresses the problems arising from the usage of TCP in wireless and mobile ad hoc network environments. To understand the TCP performance in such environments, we first examine the effects of TCP throughput in different scales of data block sizes and TCP window sizes, respectively. We also examine the packet delivery rate, hop counts, and flow counts with two practical routing protocols, i.e., Ad hoc On-Demand Distance Vector (AODV) and Optimized Link State Routing (OLSR), in mobile ad-hoc network environments. Experimental results by NS-2 simulator show the sensitivities of TCP performance to various measure parameters in 802.11 wireless and ad-hoc networks.

The rest of this paper is organized as follows. Section II mentions related works. Section III addresses the issues of delivering TCP flows over mobile ad-hoc wireless networks. Section IV presents TCP throughput with different data block sizes and TCP window sizes, and in addition examines the packet delivery rate, hop counts, and the number of

established data flows, using two practical routing protocols, AODV and OLSR. Conclusions are given in Section V.

II. REFERENCE

TCP is designed to provide a reliable end-to-end data delivery over unreliable networks. Hanbali et al. [2] figured many challenges when using TCP over ad-hoc networks. TCP is unable to distinguish between packet losses induced by network congestion and wireless packet losses. Recent research has proposed several proposals to solve these issues. Dyer et al. [3] examined the performance of the TCP protocol for bulk-data transfers in mobile ad-hoc networks. The authors proposed and evaluated the effectiveness of a heuristic scheme called fixed-RTO. With this scheme, a TCP sender can determine if a retransmission timeout is resulted by network congestion or temporary route loss.

Fu et al. [4] studied the TCP performance over multi-hop wireless network based on the IEEE 802.11 protocols. Their results showed that 802.11 MAC cannot perfectly handle signal interference of general multi-hop topologies. The throughput of TCP improves significantly if TCP operates around certain window that has the highest channel reuse. However, TCP always operates with a window larger than the optimal one, and the throughput is reduced. To solve this problem, they proposed a link-layer active queue management algorithm, called LRED, to stabilize the size of TCP window. Nahm et al. [5] proposed the fractional window increment scheme to limit the TCP aggressiveness. Their results showed that the scheme enhances 90% of TCP performance in some chain-like topologies.

There are two main categories of routing protocols in MANETs: proactive routing and reactive routing protocols. The proactive routing protocols, such as OLSR (e.g. [6]-[7]), need to maintain a global table by exchanging routing table periodically. The reactive protocols, such as AODV [8] and DSR[9], do not initiate the route discovery process until a route request occurs. In general, reactive protocols have higher latency than proactive protocols, but have lower overheads since they do not require to perform regular routing table exchanges.

In [10]-[12], authors compared the performance of AODV and OLSR with different conditions, such as node mobility, network density, and traffic loads. Laouiti et al. [13] considered on difference metric, such as control traffic overheads, packet delivery ratio, packet delay, route length, collision ratio and throughput. In [14], the authors compared AODV and OLSR in static multi-hop ad-hoc networks with difference TCP window size. The results in [15] showed that AODV protocol performs better in the networks with static number of source and destination pairs. The OLSR protocol is more efficient in networks with high density and highly sporadic traffic.

Most of previous studies do not consider the negative effect of the routing protocol when measure TCP performance over multi-hop wireless network. In this paper, we will investigate the influence of delivering TCP flows over mobile ad-hoc networks by using AODV and OLSR routing protocols.

III. TCP PERFORMANCE OVER MOBILE AD-HOC NETWORKS

TCP provides a reliable, ordered delivery of a stream of data from an application program on one end-host to another application program on another end-host. TCP uses a number of mechanisms to avoid network congestions, and achieves high performance. These mechanisms control the rate of data entering the network and maintain high network utilization. However, these mechanisms often fall short of their goals in wireless networks, since their designs were dedicated or optimized for wired networks without reference to any properties of wireless data transmission. To address these issues, the following subsections first briefly describe the TCP mechanisms first and then focus on delivering TCP flows over mobile ad-hoc networks.

A. Brief overview of TCP mechanisms

TCP provides reliable end-to-end data delivery service for upper layers. An application program on one end-host sends a larger chunk of data to TCP and requests TCP to deliver it to other application program on another end-host. TCP breaks the data into TCP segments and issues a series of requests to IP. Moreover, TCP includes the mechanisms of receiver flow control and network congestion control. These mechanisms control the rate of data entering the network and keep network utilization high.

1) TCP segment size: When an application program desires to send data to opposite end-host, the application program sends a large chunk of data to TCP and let TCP handle the data delivery. TCP breaks the data into TCP segments and issues a series of requests to IP. The IP delivers TCP segments and may fragmentize a large segment into short IP datagrams.

However, there are some issues that make the IP fragmentation undesirable. One is that CPU and memory overheads are increased to fragment an IP datagram. The overheads increase on sender, receiver and even routers on the routing path. The sender needs to create new fragment headers and copies the original datagram into the fragments. The receiver causes more overhead when reassembling the fragments because it must allocate memory for the arriving fragments and reassemble them into one TCP datagram after receiving all of the fragments. Moreover, reassembly is very inefficient on the routers. In practical, a router is not designed to hold packets and its job is to forward packets as fast as possible. Also a router can only allocate the largest buffer because it has no idea about the size of an original IP packet until the last fragment is received.

Another IP fragmentation issue is lost fragments handling. Once a fragment loses, the entire original IP datagram must be resent, and it will also be fragmented. In addition, firewalls and NAT boxes that filter or manipulate packets based on TCP information in the packet may also have trouble with IP fragments.

Moreover, wireless link layer also has the ability to divide long IP datagram into short packets. Hence, TCP uses a value called Maximum Segment Size (MSS) to limit the size of one segment. The methods of discovering MSS is beyond the scope of this paper. Later sections focus on the TCP performance on different TCP segment sizes.

2) TCP congestion control: TCP not only provides a connection-oriented, reliable communication channel to upper layers, but also provides the functions of receiver flow control and network congestion control. To manage the rate of data transmission and prevent network congestion, TCP maintains a congestion window to limit the number of unacknowledged packets. A slow-start mechanism is used to control the size of congestion window in sender. The congestion window starts from a base value 2 and doubles for every Round Trip Time (RTT) until it reaches a value which receiver advertised.

The standard implementations of TCP congestion control mechanism is optimized for the wired network. Since the bit error rate is very low in wired network, packet loss is mostly caused by buffer overflow and is thus taken as indication of congestion. Thus, the size of congestion window is dramatically decreased when TCP detects packet losses.

However, the nature of frequent noise in wireless networks heavily affects the conditions of packet loss, bit error and spurious packets. After decreasing the congestion windows size, due to wireless packet loss, TCP slows down the sending rate. This causes the bandwidth of radio link to be underutilized.

B. TCP Performance over wireless networks

This subsection addresses the issues of delivering TCP data on wireless networks. Firstly, we review the performance issues on single-hop wireless delivery. Secondly, we review the problems on delivering TCP data over multi-hop wireless networks. Finally, in the end of this subsection, the issues of sending TCP flows on mobile ad-hoc networks are addressed.

1) TCP performance over single hop wireless network: Distributed Coordination Function (DCF) scheme which has been widely used in wireless ad-hoc network is a MAC layer mechanism of the IEEE 802.11 based WLAN standard. The DCF scheme uses a Carrier Sense Multiple Access and Collision Avoidance (CSMA/CA) technique with binary exponential back-off algorithm to reduce packet collision probability on the wireless network.

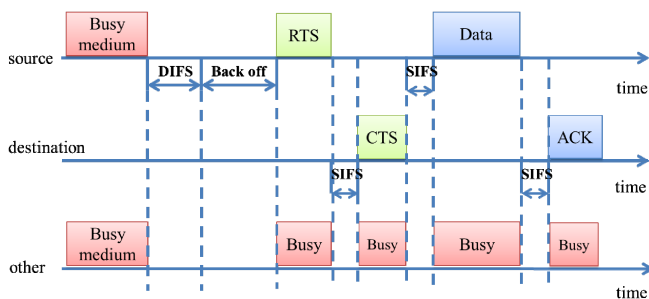


Figure 1. CSMA/CA with RTS/CTS

Figure 1. shows the CSMA/CA timing slot of DCF scheme in IEEE 802.11 wireless networks. DCF requires the source node which wishes to transmit to listen for the channel status for a DIFS interval. If the channel is found busy during the DIFS interval, the node runs the binary exponential back-off algorithm to wait. If the channel is free after the DIFS interval, the source node wishing to send data initiates the process by sending a Request to Send (RTS) frame. The destination node waits a Short Interframe Space (SIFS) time interval and sends a Clear to Send (CTS) frame. Then the source node begins sending data after waiting SIFS interval. Finally, the destination node feeds back an ACK message to confirm the delivery. With the DCF scheme, the collision of packets can be reduced in 802.11 networks but the channel might be not efficient utilized since there are many overhead time intervals in DIFS, Back-off, SIFS and RTS/CTS messages.

Let T_{DIFS} be the time interval of DIFS. T_{SIFS} be the time interval of SIFS. $T_{Backoff}$ be the back off time. T_{RTS} be the time interval of RTS frame. T_{CTS} be the time interval of CTS frame. T_{Data} be the time interval of transmitting data. T_{ACK} be the time interval of ACK frame. From Fig. 1, we observe that the channel utilization U can be defined as

$$U = T_{Data} / (T_{DIFS} + T_{Backoff} + 3 * T_{SIFS} + T_{Data} + T_{RTS} + T_{CTS} + T_{ACK}). \quad (1)$$

On (1), the channel utilization equals to the ratio of the time interval of transmitting packet to the time interval used on control messages. Obviously, the channel utilization shows positive correlation with the transmitted packet length.

Hence, the size of TCP segments also affects the throughput of IEEE 802.11 DCF scheme. If the TCP datagram is small, the channel utilization can be low. But if the TCP datagram is larger than the maximum allowable DCF packet size, the TCP datagram will be fragmented and the total data delivery throughput in wireless channel may not increase.

2) TCP performance over multiple hop wireless network: In multi-hop wireless network, the end-hosts are connected via wireless link, and there may be multiple hops as in an ad-hoc architecture.

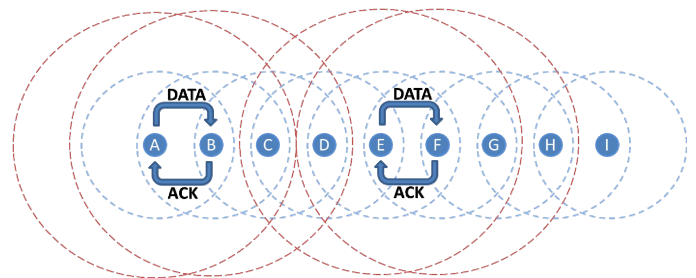


Figure 2. Multi-hop wireless transmission

In multi-hop wireless networks, the hidden node and exposed node problems affect the throughput of data delivery. We take Figure 2. as an example, in which the small circles around nodes indicate these nodes' transmission ranges. The carrier sensing ranges of node A, B, E and F are showed by

the large circles around respective nodes. Whenever node A sends data to node B, the sending node D becomes a hidden node and node C is the exposed node. So, the optimal transmission pairs are A-B and E-F only.

Although IEEE 802.11 uses RTS/CTS packets to partly overcome the multi-hop delivery problems, contention on wireless channel is still high and the throughput of multi-hop packets delivery is low. When TCP is used on multi-hop wireless network, the losses due to channel errors are mistaken for congestion losses and the sending rate is unnecessarily reduced. Moreover, RTT has much higher variability in the networks. This leads to frequent retransmission timeout in multi-hop wireless networks and causes performance degradation of TCP.

3) TCP performance over mobile ad-hoc networks: A mobile ad-hoc network is a decentralized and self-configuring wireless network. The network is ad-hoc because it does not have a fixed infrastructure or access points in the network. Instead, each node participates in routing and forwarding data for other nodes. The routing and forwarding decisions of each node are dynamically made by the routing protocol according to the wireless network connectivity.

This work intends to study the effect of ad-hoc routing protocols on TCP performance within MANETs. For this purpose, we consider two *reactive* and *proactive* ad-hoc routing protocols with different characteristics, respectively.

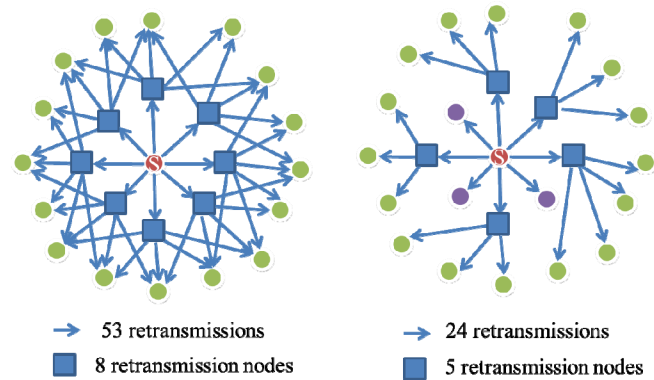


Figure 3. MPR-flooding in OLSR

The Optimized Link-State Routing (OLSR) is a proactive link-state routing protocol using periodic message exchange to update topological information in each node inside the network. Hence, the routing hops are immediately available when a node desires to send data. Link-state routing protocols such as OSPF and IS-IS elect a designated router on every link to perform flooding of topology information. OLSR is specially designed to operate in the wireless ad hoc networks. In wireless ad-hoc networks, packets can go out the same interface. OLSR protocol applies an optimized flooding mechanism, called Multipoint Relays (MPR) flooding. Figure 3 shows how the MPR-flooding is used to minimize the problem of receiving duplicate messages within a region.

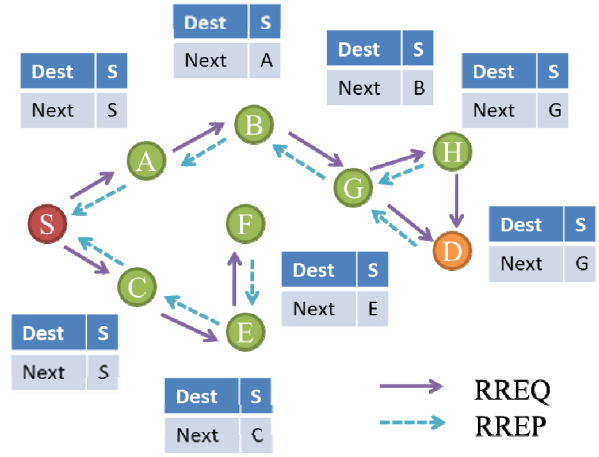


Figure 4. AODV path discovery process

The Ad Hoc On-Demand Distance Vector (AODV) routing protocol which improves from DSDV is a reactive routing protocol. AODV minimizes the number of required broadcasts by creating routes in an on-demand manner. When a source node desires to send data to other destination node, it needs to initiate a path discovery process to locate the other node. Figure 4 shows the path discovery process. A source node broadcasts a route request (RREQ) packet to its neighbors, which then forward the request to their neighbors, and so on, until the destination is located. Once the RREQ reaches the destination, the destination node responds a route reply (RREP) packet back to the source node. Hence, all the nodes participating at route discovery process will have the ability to update their routing tables accordingly.

Although two routing protocols both find the shortest path to the destination, the overheads of routing and average routing hop counts are different. TCP suffers from frequent route failures. In addition, TCP is unable to discriminate between packet loss due to route failures and network congestion. Moreover, the TCP congestion control mechanism may not work well if the congestion is caused by an excessive number of TCP connections allowed to compete for the same bottleneck.

IV. PERFORMANCE EVALUATIONS

In this section, we study the performance of TCP in terms of average throughput within single hop, multiple hop and mobile ad-hoc networks. Our simulations are conducted using Network Simulator version 2 (NS-2) [16] and UMOLSR patch. All nodes communicate with identical wireless radios using the standard MAC 802.11g that has a bandwidth of 54Mbps, a transmission range of 250 meters and 550 meters in carrier sensing range. In MANET context, the random waypoint mobility model is used. The simulation consisted of 80 nodes confined in an area of 3000x500 m². The simulation establishes 10 TCP connections (ftp traffic used with a packet size of 1500 bytes). The source-destination pairs for FTP sessions are assigned randomly. The simulation run time is set to 50 seconds, and the interval of TC and Hello message is set to 3 seconds and 1 second, respectively.

A. TCP performance over single hop network

In single hop wireless delivery, we study the effect under different TCP MSS and congestion window size settings. Figure 5 shows that a large MSS setting can increase TCP throughput, but the increment in throughput does not double when the MSS doubles. According to (1), the channel utilization is positively correlated with the transmitted packet length. If the TCP datagram is small, the channel utilization and the throughput are low. But if the TCP datagram is larger than the maximum packet size, the TCP datagram will be fragmented into multiple packets. The last packet may be small, and average utilization of wireless channel is not able to increase as MSS increases.

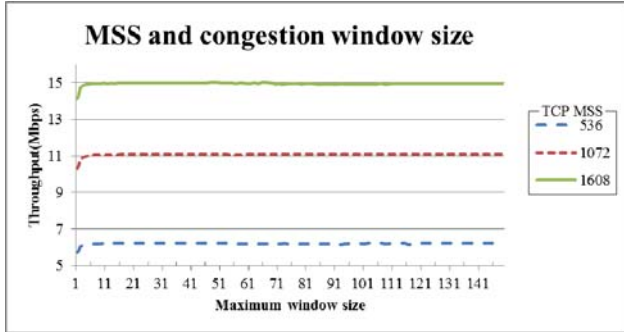


Figure 5. Different MSS and congestion window size

In Fig. 6, we verify the effect of changing SIFS. According to (1), there are 3 times of T_{SIFS} in denominator. We fix the MSS to 536 bytes. The result shows that the value of SIFS lightly affects the throughput. A smaller SIFS can result in higher throughput, but the MSS is the dominator.

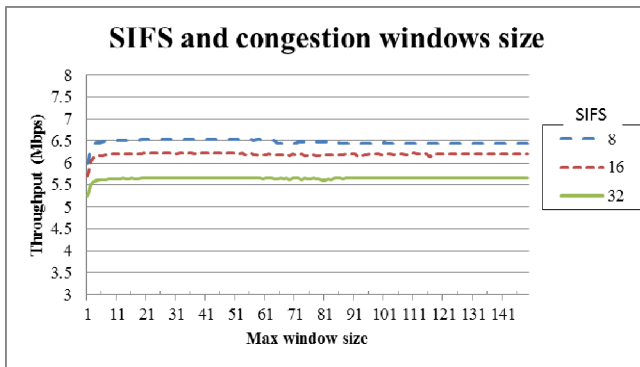


Figure 6. Different SIFS and congestion window size

B. TCP performance over multiple hop network

Multi-hop delivery can result in more contentions in wireless channels. This can be verified from Fig. 7. The throughput decreases exponentially with the number of relays in the wireless network. We fix the MSS to 536 bytes as the SIFS is 16us.

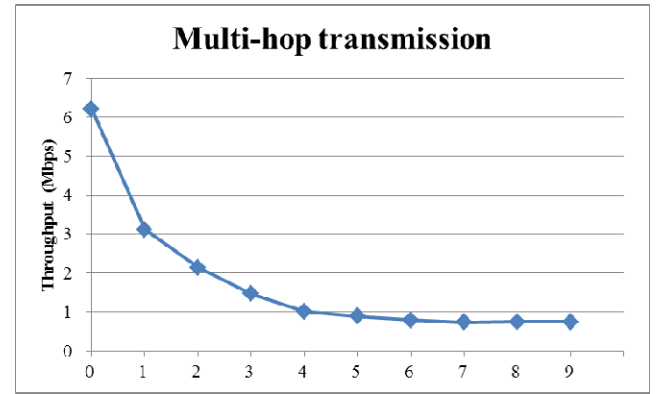


Figure 7. Multi-hop wireless transmission

C. TCP performance over mobile ad-hoc networks

In this work, we intend to study the effect of ad-hoc routing protocols on TCP performance within MANETs. We take two ad-hoc routing protocols: OLSR and AODV.

Figure 8 shows the throughput of using OLSR and AODV in different mobility. Although two routing protocols both find the shortest path to the destination, their average routing hop counts are different. The average hop counts of using OLSR and AODV are showed in Fig. 9. The OLSR can have lower hop count and higher throughput than AODV. In Fig. 10, we exam the control message overheads of these two routing protocols. The OLSR generates more overheads within higher mobility since it needs periodic message exchange to update topological information.

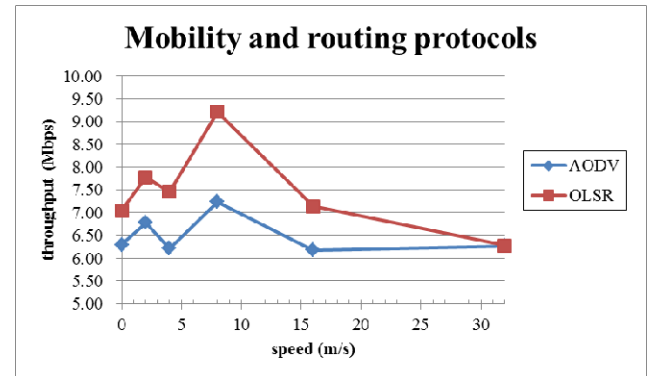


Figure 8. Different speed and routing protocols

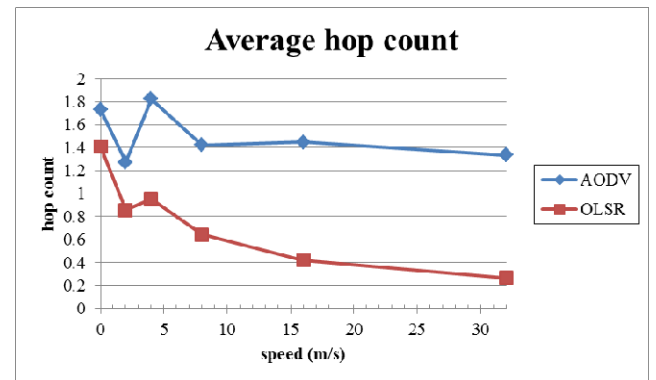


Figure 9. Average hop count

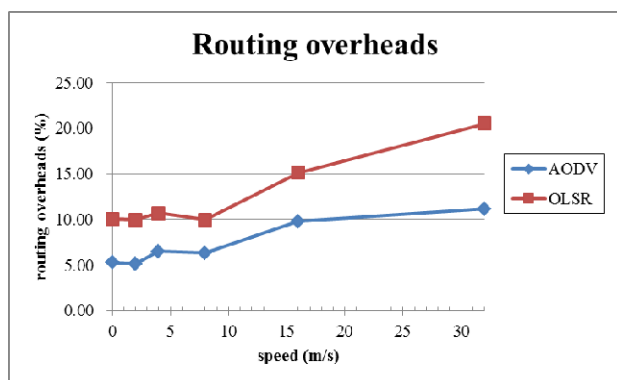


Figure 10. Routing overheads

In MANET, TCP suffers from frequent route failures. TCP is unable to distinguish between losses due to route failures and network congestion. In Fig. 11, we verify the packet delivery ratio of these two routing protocols. The OLSR shows highly delivery ratio even within high mobility.

In addition, the TCP congestion control mechanism may not work well if the congestion is caused by an excessive number of TCP connections allowed to compete for the same bottleneck. Figure 12 shows the average established flow number of using OLSR and AODV within different mobility. AODV has better behaviour of maintaining TCP connectivity while mobility is high. The OLSR which needs periodically update cannot catch up with the highly node mobility.

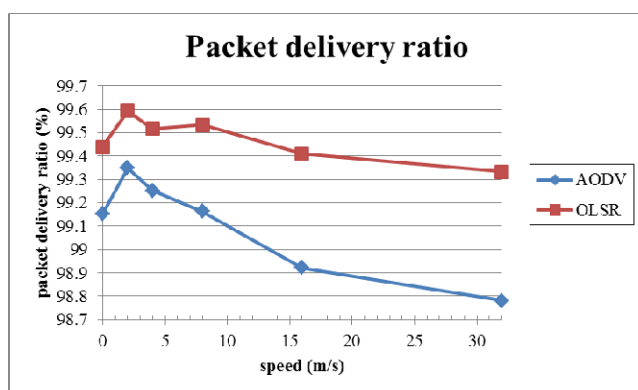


Figure 11. Packet delivery ratio

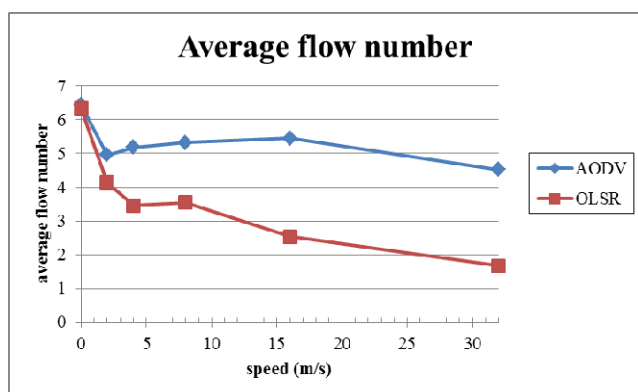


Figure 12. Average flow number

V. CONCLUSIONS

This study has described that dynamic nature of wireless and mobile ad-hoc networks highly influences TCP performance. AODV and OLSR ad-hoc routing protocols attain different effects on TCP performance. OLSR can result in most acceptable TCP performance in different mobility levels. However, it cannot catch up with the node mobility when the nodes move faster than the route update interval and then TCP connections will be broken. In contrast, AODV performs better on maintaining TCP connections although TCP congestion control mechanism does not work well if the congestion is caused by an excessive number of TCP connections allowed to compete for the same bottleneck. Therefore, experimental results show that the throughput of using AODV is lower than using OLSR.

REFERENCES

- [1] *Transmission Control Protocol*, RFC 793, Sep. 1981.
- [2] A. A. Hanbali, E. Altman, and P. Nain, "A survey of TCP over Ad Hoc Networks," *IEEE Communications Surveys & Tutorials*, vol. 7, pp. 22-36, 2005.
- [3] T. D. Dyer, and R. V. Boppana, "A comparison of TCP performance over three routing protocols for mobile ad hoc networks," in *Proc. of ACM Symposium on Mobile Ad Hoc Networking & Computing (MobiHoc)*, pp. 56-66, Oct. 2001.
- [4] Z. Fu, P. Zerfos, H. Luo, S. Lu, L. Zhang, and M. Gerla, "The Impact of Multihop Wireless Channel on TCP Throughput and Loss," in *Proc. of INFOCOM*, vol. 3, pp. 1744-1753, 2003.
- [5] K. Nahm, A. Helmy, and C.-C. J. Kuo, "TCP over multihop 802.11 networks: issues and performance enhancement," in *Proc. of 6th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '05)*, pp. 277-287, 2005.
- [6] *Optimized Link State Routing Protocol*, RFC-3626, Oct. 2003.
- [7] T. H. Clausen, G. Hansen, L. Christensen and G. Behrmann, "The optimized link state routing protocol - evaluation through experiments and simulation," in *Proc. of Wireless Personal Multimedia Communication*, Sep. 2001.
- [8] *Ad hoc On-Demand Distance Vector (AODV) Routing*, RFC 3561, Jul. 2003.
- [9] *The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4*, RFC 4728, Feb. 2007.
- [10] T. Clausen, P. Jacquet, and L. Viennot, "Comparative Study of Routing Protocols for Mobile Ad hoc Networks," in *Proc. of Med-Hoc-Net'02*, Sept. 2002.
- [11] A. Nilsson, "Performance Analysis of Traffic Load and Density in Ad Hoc Networks," in *Proc. of the 5th European Wireless*, Feb. 2004.
- [12] S. Gowrishankar, T. G. Basavaraju, M. Singh, and S. K. Sarkar, "Scenario based Performance Analysis of AODV and OLSR in Mobile Ad hoc Networks," in *Proc. of the 24th South East Asia Regional Computer Conference*, Nov. 2007.
- [13] A. Laouiti, P. Muhlethaler, A. Najid, and E. Plakoo, "Simulation results of the olsr routing protocol for wireless network," in *1st Mediterranean Ad-Hoc Networks workshop*, 2002.
- [14] G. Anastasi, E. Ancillotti, M. Conti, and A. Passarella, "Experimental Analysis of TCP Performance in Static Multi-hop Ad Hoc Networks," chapter 6 in *Mobile Ad Hoc Networks*, Nova Science Publisher, Sep. 2007.
- [15] A. Huhtonen, "Comparing aodv and olsr routing protocols," *Telecommunications Software and Multimedia*, 2004.
- [16] Network Simulator-NS-2, Available at www.isi.edu/nsnam/ns/