Improving TCP Fairness Flows in Wireless Ad Hoc Networks

Hatim Mohamad Tahir*, Abas Md Said, Alex W.M. Tan***
* School of Computing, Universiti Utara Malaysia, Kedah, Malaysia
** Department of Computer Information System, Universiti Teknologi PETRONAS, Perak, Malaysia

Keywords:

TCP Fairness
Wireless Ad Hoc Network
UDP

ABSTRACT

In scenario where wireless ad hoc networks are deployed, significant TCP unfairness in wireless ad hoc networks has been reported during the past several years. This unfairness results from the nature of the shared wireless medium and location dependency. However, this queue is not a FIFO queue. TCP flows sharing the queue have different, dynamically changing priorities determined by the topology and traffic patterns. Thus, they get different feedback in terms of packet loss rate and packet delay when congestion occurs. In this paper, we will use a normal FIFO scheduling scheme, which helps competing TCP connections to achieve fairness without much throughput loss. Simulation results show that our scheme successfully eliminates the extreme unfairness existing in above-mentioned scenarios.

Corresponding Author:

Hatim Mohamad Tahir
School of Computing
Universiti Utara Malaysia,
06010 UUM Sintok, Kedah, Malaysia.
Email: hatim@uum.edu.my

1. INTRODUCTION

A wireless ad hoc network is a decentralized type of wireless network without the use of any existing network infrastructure or centralized administration. The decentralized nature of wireless ad hoc networks makes them suitable for a variety of applications where central nodes can't be relied on, and may improve the scalability of wireless ad hoc networks compared to wireless managed networks. Minimal configuration and quick deployment make ad hoc networks can be expanded in the future such as university, military service since it doesn't need any fixed infrastructure to make a connection. The presence of dynamic and adaptive routing protocols enables ad hoc networks to be formed quickly. In this paper, we address TCP performance within a wireless ad hoc network. This has been an area of active research recently, and progress has been reported in several directions. Three different types of challenges are posed to TCP design by such networks. First, as the topology changes, the path is interrupted and TCP goes into repeated, exponentially increasing time-outs with severe performance impact. Efficient retransmission strategies have been proposed to overcome such problems [1]. The second problem has to do with the fact that TCP performance in a wireless ad hoc environment depends critically on the congestion window in use. If the window grows too large, there are too many packets (and ACKs) on the path, all competing for the same medium. Congestion builds up and causes "wastage" of the broadcast medium with consequent throughput degradation [2]. The third problem is significant TCP unfairness which has been revealed and reported through both simulations and test bed measurements recently [3], [4], [5].

This paper focuses on the third problem, namely, enhancing TCP fairness in ad hoc networks. Previous work on this topic mostly dealt with the underlying factors causing TCP unfairness. The rest of the paper is organized as follows: We review the literature in Section 2. Section 3 present the methodology whereas in section 4 is the simulation experiment. Simulation results and analysis of the simulation [6] are discuss in section 5 and Section 6 is the conclusion.

2. LITERATURE REVIEW

According to Postel [7], TCP which is the dominating end-to-end protocol on the internet today carry more than 90% of the total traffic. It provides a secure and reliable connection between two hosts in a multi-network environment appeared in numerous clones such as Newreno, Sack, and etc. All these are with different features and advantages but with maximal throughput as main objective. Recently, several researchers have studied TCP fairness in multi-hop wireless networks especially under the IEEE 802.11 MAC protocol.

Xu and Gerla [8] and He [9] proposed solutions to such problems that due to the interaction of the 802.11 MAC layer protocol, more precisely the hidden terminal problem and binary back off scheme with the TCP window mechanism and time out. This TCP 802.11 interaction was found to cause unfairness among competing TCP flows and in extreme cases "capture" of the channel by a few flows [10]. Li et al [11] has pointed out the fact that TCP performance in ad hoc multihop environment depends critically on the window in use. If the window grows too large, there are too many packets (and ACKs) on the path, all competing for the same medium. Congestion builds up and causes "wastage" of the broadcast medium and severe throughput degradation [12]. To address fairness at the MAC layer, several fair scheduling schemes have been proposed for general shared wireless channel environment. For example, Vaidya et al [13] presented a distributed fair scheduling algorithm for wireless LAN that emulates Self-Clocked Fair Queuing in a distributed manner and chooses a back off interval that is proportional to the end tag of the packet to be transmitted. While Nandagopal et al [14] proposed a general analytical framework that can translate any given fairness requirement into a matching back off scheme. These schemes address the fairness of MAC in general whereas here we try to eliminate the extreme unfairness among TCP flows in a broad class of ad hoc network environments. On the other hand, compared to the scheme proposed in this paper, these schemes are back off-based solutions, i.e. they try to achieve fairness by modifying the back off policy of MAC protocol. Broch et al [15] noted in the MONARCH project that use ad hoc extensions to do simulation to make comparisons about performance of multi-hop wireless ad hoc network routing protocols. In the simulation, Broch, Johnson and Maltz [16] admits the Dynamic Source Routing is used as the routing protocol for implementation of wireless ad hoc Network. Koksal et al [17] observed a same phenomenon that is a short-term unfairness is always significant even though UDP traffic was used in that paper. In this paper, however, we are more interested in the long-term unfairness of multiple TCP flows. This type of unfairness may starve some TCP flows, although the aggregate throughput may still be high. Bensaou, Wang and Ko [18] presented a general "fair scheduling based framework" to guarantee the fair share of the wireless media. A predefined fair share for each node is determined during the admission control. Then, each node will continuously monitor its currently achieved throughput. Based on this information, a fair index is calculated for each node and the BEB back off scheme is replaced by a new scheme based on the fair index. In fact, the fairness of the MAC layer has been an active area of research. The fairness of the MAC layer has an impact on network performance in general, regardless of the transport protocol used. However, few studies have addressed TCP and the MAC protocol interactions. In Gerla et al [19] and Yang [20] reported that significant TCP unfairness is still dominant over MACAW

3. METHODOLOGY

In our experiment, our simulations are considered to be using normal FIFO data with its designed scheme. The purpose of this experiment is to measure the performance of the TCP connection in a wireless ad hoc network that use to get the data from the scheme and testing the data delay and packet sequences. In our experiment, we just focus on wireless connection only. In addition, we add 2 new UDP agents, which are UDP0 and UDP1 for our simulation. It used 2 FTP connections. Problem occurred if we set the packet size in FTP connection. Therefore, we decided to use a UDP agent to set the packet size. By adding the UDP agent our simulation can work as well.

4. EXPERIMENT

In this experiment, we use the configuration and parameter to get the simulation. In Figure 1 below, we use seven nodes for our simulation. There are five wireless nodes as node 0, node 1, node 2, node 3, and node 4. While there are two agents which is node 5 represents as router and node 6 represents as a server. These five wireless nodes are space 200m away from its neighbors equally.

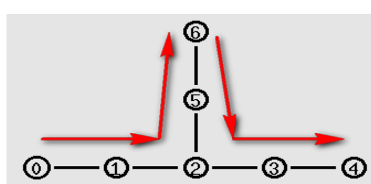

Figure 1. Scenario for simulation

Each node just can communicate with its immediate neighbors at this distance effectively. The TCP version that we use is the new Reno type. 2 FTP connections which are FTP0 and FTP1 are used as TCP traffics. Both FTP sources have unlimited backlogs to send. Besides that, we also use 2 UDP connections which are UDP0 and UDP1. In Figure 1 the FTP0 and UDP0 connections are from node 0 to node 6 (Server) while FTP1 and UDP1 connections are from node 6 (Server) to node 4. The setting of the simulation parameters are given in Table 1.

Table 1. Simulation Parameter

	NS-2 Setting	Our Experiment Setting
Wireless channel bandwidth	2 Mb/s	2 Mb/s
Wireless node interface queue limit	50 Packets	10 Packets
Version of TCP used	New Reno	New Reno
Nominal radio transmission range	250 m	-
Buffer management for wireless nodes	Drop Tail	Drop Tail
Queue limit for wireless nodes	50 Packets	-
Packet size	1024 Bytes	1024 Bytes
New UDP agent	-	2 (UDP0 & UDP1)
Delay between nodes	-	20 ms
Total number of nodes	-	7

Packets are transported among network nodes via network links. These packets are generated by network applications (data packet) or network protocols (control packet). An event scheduler is in charge of ordering all the packets (or events) by their arriving time and let the nodes, links, and agents to handle these packets in sequence. A full trace of each packet or their statistical information can be stored in a trace file, and used to generate graphs or animations. Sometimes there are problem which the network bandwidth is not used fairly among the nodes. Nodes near the gateway may overuse the bandwidth while nodes far away from the gateway scarcely share the bandwidth. Furthermore, if the destination node is located in the same ad hoc domain as the source, existing route selection methods result in high overhead or longer routes. When a node is created, it is automatically assigned an address and a default routing module. Nodes are the sources and destinations of the packets. In the script, $ns is the scheduler object and node is its method to create a node, $n0 and $nl hold the reference to the newly created node object. TCP often faces severe unfairness in this type of connection scenario, which forces some TCP flows to completely stop transferring any data despite all links being in good states.

The following script segment creates one link to connect these two nodes with 2M bandwidth and 20ms delay. A link receives packets from one end, compute the delay and transmission time, and send them to the other end after the time elapsed. If two or more packets are going through the same link, the later packet should wait in a queue until all previous packets are sent. If the number of packets waiting exceeds the queue limit, new packets sent to this link are dropped.

ns duplex-link $n0 $n1 2Mb 20ms Drop Tail

When the network topology is created, agents can be attached to nodes to generate packets and put them onto the network. There are agents in different network protocol layers. In the transportation layer, there are TCP and UDP agents. In our simulation we used two TCP agents and two UDP agent and put them on node $n0. When an agent is attached to a node, it sends and receive packet through this node. For our simulation, we use FTP and CBR agents to generate continued data streams. When an agent is attached to another agent, they send and receive data through this agent.

If nodes and links is the scene on the stage, agents are the actors; then the scheduler is the director. The scene and actors do not know how to perform unless the director tells them. The scheduler is a built-in object in NS-2 user can tell it when to do what. The script segment asks the scheduler to open the traffic generator at 0.1 seconds of the simulation and close it at 3.5 second. The script segment also instruct the scheduler the whole simulation will last for 3.9 seconds. Every time a packet is received by an agent or a link, a delay time representing the processing time or network delay is computed and actually the packet is handed to the scheduler as an event. The scheduler orders all the events by time and fires them one by one. In NS-2, the scheduler is non real-time by default. The timestamps are not physical time in the simulation but used to order the events. So a 3.9 second simulation may take 2.0 seconds or 1 hour, depending on the complexity of the topology and the number of events fired during the simulation.

5. RESULTS AND DISCUSSION

The performance parameters are as follows End-to-end delay refers to the time taken for a packet to be transmitted across a network from source to destination, Jitter is the fluctuation of end-to-end delay from one packet to the next packet of connection flow, Throughput is the ratio of the total amount of data which reaches the receiver from the sender to the time it takes for the receiver to receive the last packet. Lastly, Packet loss is the failure of one or more transmitted packets to arrive at their destination. As a rule of thumb derived from day-to-day practical experience, in general with TCP/IP protocols a packet loss below 0.1% (1 lost packet in every 1000 packets) can be tolerated; anything higher will have more or less impact (depending on circumstances) and needs to be addressed. In MANETs throughput is affected by various changes in topology, limited bandwidth and limited power. Unreliable communication is also one of the factors which adversely affect the throughput parameter.

a. End-to-End Delay

In Figure 2 gives the results of normal FIFO scheduling and demonstrates result for end-to-end delay.

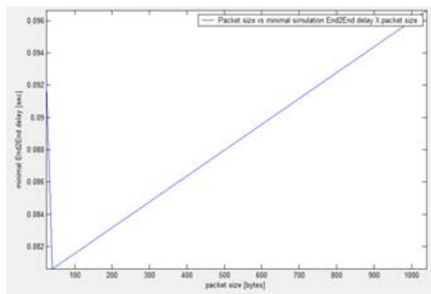

Figure 2. End-to-end Delay Versus Packet Size

The highest data transfer rate is 0.096 seconds. Best results attained at data transfer rate of 1000 bytes. Higher value of data transfer rate shows irregularities in terms of time measurement and unpredictability in terms of data delivery.

b. Jitter

The jitter may have been distributed as several bursts of higher jitter shown in Figure 3. Some of jitter peak is not been able to be measured properly and information is lost.

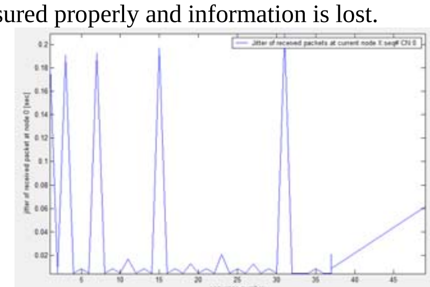

Figure 3. Jitter of Received Packet at node (sec) vs sequence number

It is because roughly 1/5 of the packets have a much higher jitter frequency distribution. From Figure 3, it is not possible to tell how the jitter was distributed over time. For example, the maximum simulation jitter was set to 0.002 seconds. The simulation jitter 0.004 seconds was lost because the maximum jitter frequency distribution was exceeded.

c. Throughput

In Figure 4 and 5 show how the throughput affects transaction performance.

d.

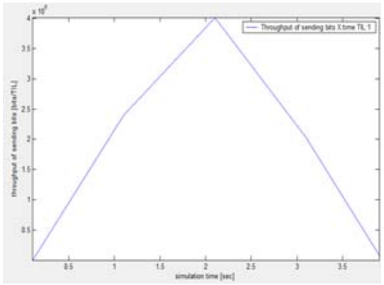

Figure 4. Throughput of sending bits vs time (sec)

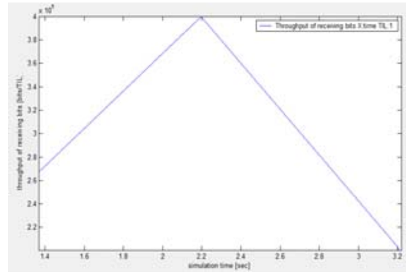

Figure 5. Throughput of receiving bit vs time (sec)

The highest throughput is about 4000000 bit on 2.15 seconds for the sending throughput while for the receiving throughput is around 2.2 seconds. Throughput is calculated by dividing the total content length (in bit) by response time (in sec). This data is useful to measure the user connection speed as well as the bandwidth usage of the server. In Figure 6 the highest throughput of dropping bit is about 180000 bit on 2.3 seconds. The throughput is constantly increased until it reaches at the above 180000 bit after that following by decreased.

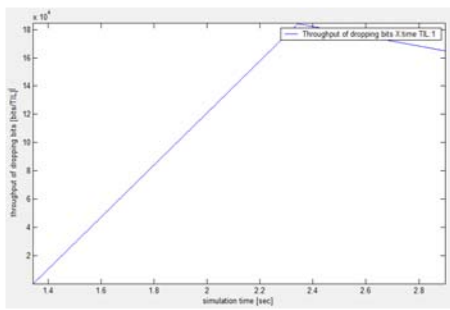

Figure 6. Throughput of dropping bit vs time (sec)

e. Packet Loss

Packet loss is where network traffic fails to reach its destination in a timely manner. Most commonly packets get dropped before the destination can be reached. There isn't too much packet loss in this wireless. Only 1.39 percent of packets that are not arrive in the destination.

6. CONCLUSION

TCP is a window-based acknowledgement-clocked flow control protocol. It uses additive-increase multiplicative decrease strategy for changing its window as a function of network conditions. Packets of a TCP connection are sent with increasing consecutive sequence numbers. In the simplest operation of TCP, at each arrival of a packet at the destination, an ACK is sent back to the source with the information of the next sequence number that is expected. The simulation results show that the scheme improves the fairness among TCP connections greatly.

ACKNOWLEDGEMENTS

We would like to express our gratitude to the Ministry of Science, Technology and Innovation (MOSTI), Government of Malaysia under e-Science Grant No. 01-01-07-SF0018 and Research & Innovation Management Center (RIMC), Universiti Utara Malaysia Grant number S/O 12121

REFERENCES

[1] G. Holland and N. H. Vaidya, "Analysis of TCP performance over mobile ad hoc networks," in *Proceedings of ACM MobiCom'99*, 1999.

[2] 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 *INFOCOM 2003. Twenty-second annual joint conference of the IEEE Computer and Communications. IEEE Societies*, 2003, vol. 3, pp. 1744–1753.

[3] K. Xu, S. Bae, S. Lee, and M. Gerla, “TCP behavior across multihop wireless networks and the wired internet,” *Proceedings of the 5th ACM international workshop on Wireless mobile multimedia - WOWMOM ’02*, p. 41, 2002.

[4] S. Xu and T. Saadawi, “Does the IEEE 802.11 MAC protocol work well in multihop wireless ad hoc networks?,” *Communications Magazine, IEEE*, vol. 39, no. 6, pp. 130–137, 2001.

[5] K. Tang and M. Gerla, “Fair sharing of MAC under TCP in wireless ad hoc networks,” in in *Multiaccess, Mobility and Teletraffic in Wireless Communications: Volume 4*, Springer, 1999, pp. 231–240.

[6] T. Issariyakul and E. Hossain, *An introduction to network simulator NS2*. Springer, 2012.

[7] J. Postel, “Transmission control protocol,” 1981.

[8] K. Xu, S. Bae, S. Lee, and M. Gerla, “TCP behavior across multihop wireless networks and the wired internet,” in *Proceedings of the 5th ACM international workshop on Wireless mobile multimedia*, 2002, pp. 41–48.

[9] W. He, Y. Xue, and K. Nahrstedt, “Delay Management in Wireless Ad Hoc Networks,” in in *Guide to Wireless Ad Hoc Networks*, Springer, 2009, pp. 311–331.

[10] A. Al Hanbali, E. Altman, and P. Nain, “A survey of TCP over ad hoc networks.,” *IEEE Communications Surveys and Tutorials*, vol. 7, no. 1–4, pp. 22–36, 2005.

[11] J. Li, C. Blake, D. De Couto, H. I. Lee, and R. Morris, “Capacity of wireless ad hoc networks,” in *Proc. ACM SIGMOBILE*, 2001, vol. 1.

[12] N. Xu, S. Rangwala, K. K. Chintalapudi, D. Ganesan, A. Broad, R. Govindan, and D. Estrin, “A wireless sensor network for structural monitoring,” in *Proceedings of the 2nd international conference on Embedded networked sensor systems*, 2004, pp. 13–24.

[13] N. Vaidya, A. Dugar, S. Gupta, and P. Bahl, “Distributed fair scheduling in a wireless LAN,” *Mobile Computing, IEEE Transactions on*, vol. 4, no. 6, pp. 616–629, 2005.

[14] T. Nandagopal, T.-E. Kim, X. Gao, and V. Bharghavan, “Achieving MAC layer fairness in wireless packet networks,” in *Proceedings of the 6th annual international conference on Mobile computing and networking*, 2000, pp. 87–98.

[15] J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, and J. Jetcheva, “A performance comparison of multi-hop wireless ad hoc network routing protocols,” in *Proceedings of the 4th annual ACM/IEEE international conference on Mobile computing and networking*, 1998, pp. 85–97.

[16] D. B. Johnson, D. A. Maltz, and J. Broch, “DSR: The dynamic source routing protocol for multi-hop wireless ad hoc networks,” *Ad hoc networking*, vol. 5, pp. 139–172, 2001.

[17] C. E. Koksal, H. Kassab, and H. Balakrishnan, “An analysis of short-term fairness in wireless media access protocols (poster session),” in *ACM SIGMETRICS Performance Evaluation Review*, 2000, vol. 28, no. 1, pp. 118–119.

[18] B. Bensaou, Y. Wang, and C. C. Ko, “Fair medium access in 802.11 based wireless ad-hoc networks,” in *Mobile and Ad Hoc Networking and Computing, 2000. MobiHOC. 2000 First Annual Workshop on*, 2000, pp. 99–106.

[19] M. Gerla, K. Tang, and R. Bagrodia, “TCP performance in wireless multi-hop networks,” in *Mobile Computing Systems and Applications, 1999. Proceedings. WMCSA’99. Second IEEE Workshop on*, 1999, pp. 41–50.

[20] L. Yang, W. K. G. Seah, and Q. Yin, “Improving fairness among TCP flows crossing wireless ad hoc and wired networks,” in *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*, 2003, pp. 57–63.

BIBLIOGRAPHY OF AUTHORS

	Hatim Mohamad Tahir obtained his BSc and MSc from the USA in 1984 and 1986 respectively. He is currently lecturing in Universiti Utara Malaysia. His research interest is in the area of Information and Network Security.
	Abas Md Said obtained his BSc and MSc from USA. He obtained his PhD in United Kingdom. Currently, he is lecturing at Universiti Teknologi Petronas. His research interest is in Information System.
	Alex W.M. Tan is pursuing for Bachelor of Information Technology with Honors at Universiti Utara Malaysia.