2006-06-01

The Impact of TCP Sliding Window on the Performance of IEEE 802.11 WLANs

Enrique Roques Gomez
Dublin Institute of Technology, egomez@cnri.dit.ie

Mark Davis
Dublin Institute of Technology, mark.davis@dit.ie

Recommended Citation

Roques, E. & Davis, M. (2006) The impact of TCP sliding window on the performance of IEEE 802.11 WLANs. *IEE Irish Signals and Systems Conference 2006* (ISSC 2006), Dublin, June 2006.

The Impact of TCP Sliding Window on the Performance of IEEE 802.11 WLANs

Enrique Roques Gomez and Mark Davis

Communication Network Research Institute
Focas Building,
Dublin Institute of Technology,
Kevin St .D8
IRELAND
E-mail:egomez@cnri.dit.ie *E-mail: mark.davis@dit.ie*

In this paper the TCP sliding window mechanism is experimentally investigated as one of the possible causes of the unfairness often observed on IEEE802.11 wireless LANs. We show how by appropriately sizing the sliding window it is possible to re-introduce fairness into the operation of the WLAN.

Keywords – WLAN, TCP performance, QoS, Sliding Window.

I INTRODUCTION

Today network traffic is dominated by data traffic sent via the TCP transport protocol. It has been shown that TCP behaves as a greedy protocol, which can have devastating effects on the quality of service (QoS) for multimedia applications [1]. In order to avoid this situation the new IEEE 802.11e MAC Enhancement standard [2] has recently been ratified which introduces tuning parameters at the MAC layer to support QoS provisioning through support for prioritised access to the WLAN medium. However, the TCP settings are not considered and they could have a significant impact on the fairness between TCP streams. In this paper we will present the results from a number of experimental scenarios, which concern the contention between TCP loads. There are quite a number of works [3,4] that investigate TCP unfairness through computer simulation and they attribute the unfairness to the interaction between the MAC and the TCP transport layer mechanisms. However, in this paper we will present experimental results that differ significantly from these simulated results in that they do not exhibit the gross differences among the throughputs of the TCP stations.

II TCP PERFORMANCE ON WLANS

We will consider two experimental scenarios here. In the first one, TCP upload flows only compete against each other and in the second TCP download flows are added to the network traffic load.

TCP is an example of a reliable transport protocol where flow and congestion mechanisms are employed at the transport layer which make use of acknowledgments to confirm the arrival of the data packets at the destination and also to determine a suitable data rate for the TCP sender to match the network capacity. In an infrastructure mode WLAN with only TCP upload streams, all the TCP acknowledgments (TCP ACKs) coming from the destination are queued in the access point (AP) buffer. The IEEE 802.11 is considered to a fair protocol in the sense that every contending station enjoys the same probability in gaining access to the medium in order to transmit its traffic load. Assuming that there are N competing stations, including the AP, then any station would win $1/N$ out of the total opportunities for accessing the wireless medium. Consequently, this situation leads to an asymmetry between the forward and the reverse path, which has been shown to be a cause of degradation on the TCP performance in wired networks [5]. In a scenario involving TCP upload streams only, just $1/N$ of the access opportunities can be gained by the AP for the $(N\text{-}1)/N$ TCP ACKs corresponding to the TCP senders, e.g. for every packet that AP receives it has to transmit a TCP ACK. Consequently, the larger the number of wireless stations streaming upload TCP stream, the greater the number of TCP ACKs that need to transmitted by the AP. Therefore, the AP needs to transmit more often than an individual station. Under heavy load conditions, there will be an insufficient availability of transmission opportunities at the AP to immediately transmit the TCP ACKs and they will be temporarily buffered at the AP.

This can have an impact on the TCP mechanism, such as congestion window growth and retransmission timers. It is also worth pointing out here that the TCP protocol was designed to work over wired networks where the reliability is much higher than in wireless networks. All these factors can lead to a situation of a large number of packet retransmissions, which comes from the timeouts caused by TCP ACKs dropped or stored for longer than permissible by the TCP timers in the AP or packet loss, or both. If TCP download streams are added to the WLAN, there will be a clear unfairness between the download and the upload flows as all the download flows must be sent to the destination through the AP. Therefore, just 1/*N* of the accessing opportunities would be shared by all the wired stations plus the TCP ACKs coming from the destination source of the wireless TCP senders. Conversely, the wireless stations would enjoy (*N*-1)/*N* of the radio medium bandwidth less the channel capacity allocated for the ACKs coming from the wireless TCP receivers.

III TESTBED CONFIGURATION

We set up an IEEE 802.11b wireless network operating in the infrastructure mode with the RTS/CTS mechanism disabled. In Figure 1 the hardware and software features are shown. The network is composed of a Cisco Series 1200 AP, the wired and wireless nodes are Dell desktop PCs running on Linux 2.6.9-1.667 kernel. The software tools used are tcpdump as packet sniffer and DBS [6] (Distributed Benchmark System) tool as TCP generator. Due to the requirements of the TCP generator and in order to maintain a time reference for the experiment all the nodes were synchronized via NTP protocol through the wired Ethernet interface. The wireless nodes were synchronized every minute. On the other hand, the wired stations were synchronized every 15 minutes in order to avoid adding unnecessary traffic in to the wired network as the TCP data packets and ACKs also travelling over it. Figure 2 shows the overall test-bed scenario.

Software:
–Operating System: Linux 2.6.9-1.667 kernel –TCP generator tool: Distributed Benchmark System (DBS) –Packet capturer: tcpdump –Synchronization: NTP protocol
Hardware
–Access Point: Cisco Series 1200 –PC: Dell GX 150,240,260 –Wireless Adapter: NETGEAR WAG511

Figure 1: Table of the Software and Hardware Features of the Test-Bed

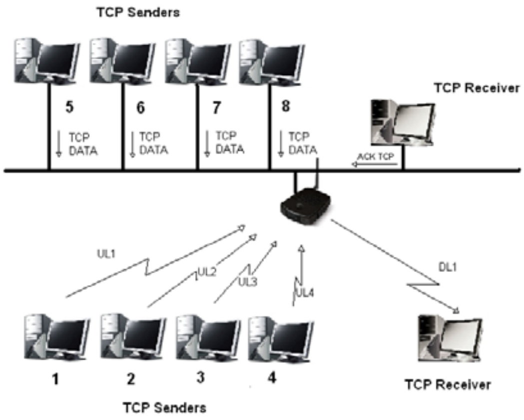

Figure 2: Test-Bed Scenario

IV MEASUREMENTS AND RESULTS

As outlined earlier, the main objective of this paper is to show the influence of the TCP contention window size on the TCP performance of WLAN networks. By controlling the size of the TCP congestion window, we show how fairness can be recovered.

Initially, a simple scenario where two TCP wireless stations compete to stream TCP packets to a receiver located on the wired side. Each node sends packets of 1500 bytes at a rate of 400 pps (as measured at the application layer) over a 50 minute test interval. The maximum buffer size of the TCP contention window is set at 32 KB. Furthermore, each wireless station always has a packet to send, i.e. the stations are operating under saturation conditions. Figure 3 shows a typical TCP plot where the TCP sequence number vs. the time is presented. Towards the end of

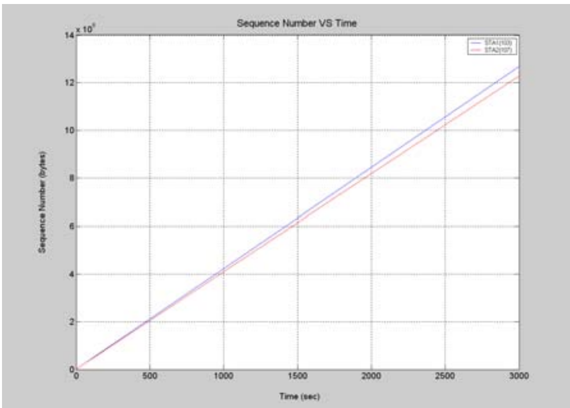

Figure 3: Sequence Number VS time for a 2 TCP Uploads. Both TCP Buffer Size are 32KB.

the experiment, one can observe a slight difference between the two stations. , However, it is unlikely that this difference arises from the TCP unfairness mechanism discussed earlier. Instead, it is more likely to be caused by differences in the radio propagation paths rather than by protocol issues.

However, after repeated experiments, we observed that at times one of the stations emerged as being dominant and at other times, the other station was dominant.
In the next experimental scenario, the value of the TCP contention window on station two (STA2) is increased to 1MB. On a wired network the expected result would be a clear increase in the throughput of STA1 as it can send more packets in a stream in a Round Trip Time (RTT) and this is what is shown in Figure 4. This effect could be considered to be a pure TCP effect, however based on the supposed interaction between the MAC and the TCP layers, the enlargement of the sliding window on a wireless station could have a negative impact on the performance, in terms of greatly reduced throughput. Nevertheless, Figure 4 illustrates a normal TCP behaviour.

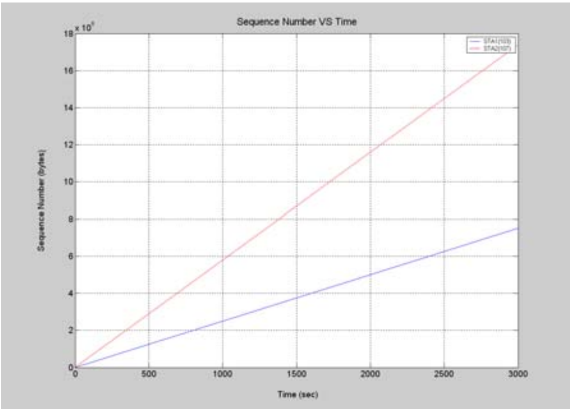

Figure 4: Sequence Number VS time for a 2 TCP Uploads. STA1 TCP Buffer Size equal to 32KB. STA2 TCP Buffer Size equal to 1MB .

Up to ten wireless stations were added and the experimental procedure was repeated. This new scenario differs from the first one in the number of stations, which have a greater probability to collide at the MAC layer. As a result of collisions it will take longer to send a TCP packet, which could provoke timeouts at the TCP layer and lead to retransmissions of the data packets. Nevertheless, the results in Figure 5 show the type of TCP performance that one would expect to find on a wired network.

Figure 6 shows the performance of 10 TCP uploads where the TCP Buffer size has been enlarged on the last 5 stations. As observed, the result matches the one for 2 stations scenario. It is also worth pointing out the similar fairness amongst both wireless and wired nodes.

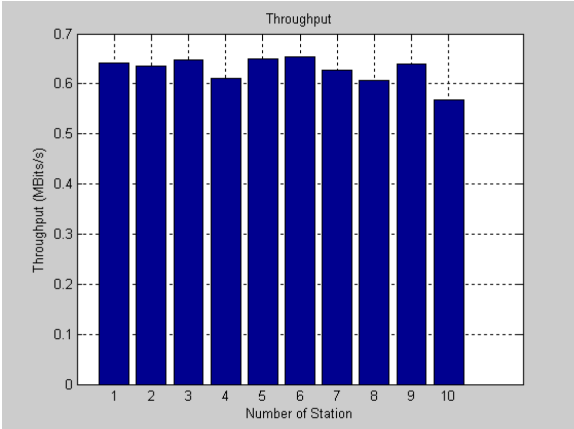

Figure 5: Performance of 10 TCP uploads under standard 802.11b operation. Same TCP sliding window size used in all stations (32KB).

For the packets captured, we observe that the maximum TCP contention window size allowed by the TCP receiver is 41110 bytes on those stations, which have the possibility of expanding their contention windows up to 1MB. The rest made full use of the buffer size allocated, e.g. 32KB.

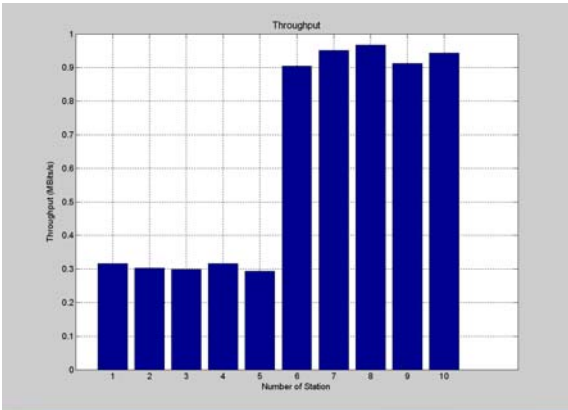

Figure 6: Performance of 10 TCP uploads under standard 802.11b operation, STA1-5 TCP Buffer Size equal to 32KB. STA6-10 TCP Buffer Size equal to 1MB

The next scenario involves 4 wired stations, which send their traffic load through the AP to a wireless TCP receiver, competing to gain access to the radio channel with 4 wireless stations doing the same but to a destination located on the wired network, As was previously explained, there is a clear unfairness between the two groups as the single gateway to the wireless medium for the wired stations is the AP. Therefore, they have to share the same transmission opportunities rather than as a single wireless station. Figure 7 shows the average throughput of each station as measured over the 50 minutes of experiment. As expected the wireless nodes enjoy a larger allocation of bandwidth. Figure 7 illustrates a similar fairness amongst both wireless and wired nodes.

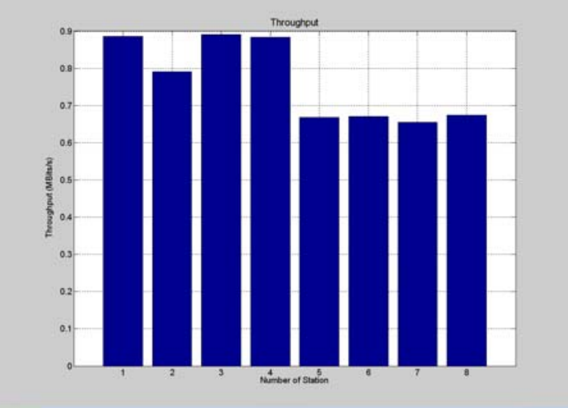

Figure 7: Performance of 4 TCP uploads (STAs 1-4) and 4 TCP downloads (STAs 5-8) under standard 802.11b operation. Same TCP sliding window size used in all stations (32KB).

The next step is to investigate what happens by enlarging the TCP sliding window size up to 1MB in the wired network. Figure 8 shows how the situation has been inverted. Thus, it is observe the significant impact of the TCP buffer size on a infrastructure mode WLAN. Once again, the maximum achieved by the window is 41110B, therefore, by selecting a value between 32-41KB we expect to achieve fairness between the wireless and the wired side of the network.

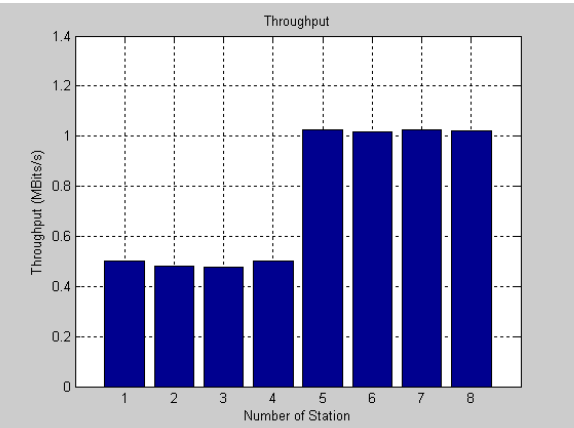

Figure 8: Performance of 4 TCP uploads (STAs 1-4) and 4 TCP downloads (STAs 5-8) under standard 802.11b operation, STA1-4 TCP Buffer Size equal to 32KB. STA5-8 TCP Buffer Size equal to 1MB

V CONCLUSIONS

Throughout our experiments we have investigated the impact of the TCP sliding window size on a IEEE 802.11b wireless network operating in the infrastructure mode. We can conclude that TCP buffer size has a major impact on the unfairness among the different TCP stations. However, it could also be observed how the TCP buffer size, instead of being a cause of unfairness, can be used as a tool to achieve unfairness between the TCP wireless and wired stations. Unlike computer simulations, the results presented in this paper show fairness between wireless stations under the same TCP settings. However, the original TCP transport protocol is a fairly old protocol at this stage that may be implemented in a number of different versions, which have been shown to be incompatible in terms of fairness [7]. This could be the case here in this experimental setup and might explain why the unfairness predicted by computer simulation was not observed here.

VI ACKNOWLEDGEMENTS

The authors gratefully acknowledge the support of Science Foundation Ireland, grant 03/IN3/1396, under the National Development Plan

VII REFERENCES

[1] Peter Cliford, Ken Duffy, John Foy, Doug Leith, David Malone, " Improved performance of voice and data using the 802.11e EDCP" Hamilton Institute,NUI Maynooth, Ireland.
[2] Stefan Mangold, Sunghyun Choi, Peter May, Ole Klein, Guido Hiertz, Lothar Stibor,"IEEE 802.11e Wireless LAN for Quality of Service", Proc. European Wireless '2002, Florence, Italy, February 2002.
[3] D.J. Leith, P. Clifford, "Using the 802.11e EDCF to Achieve TCP Upload Faieness over WLAN links", Hamilton Institute, NUI Maynooth.
[4] A.Detti, E. Graziosi, V.Minichiello, S. Salsano and V. Sangregorio,"TCP fairness issues in IEEE 802.11 based access networks", paper submitted for publication.
[5] DBS (Distributed Benchmark System): http://www.ai3.net/products/dbs/
[6] Jeonghoon Mo, Richard J. La_, Venkat Anantharam, and Jean Walrand "Analysis and Comparison of TCP Reno and Vegas", Department of Electrical Engineering and Computer Sciences University of California at Berkeley, July 13, 1998
[7] H. Balakrishnan, V. Padmanabhan, "How Network Asymmetry Affects TCP". IEEE Communications Magazine, April 2001, pp60-67.