Quantitative Assessment of IP Service Quality in 802.11b and DOCSIS networks

Thuy T.T. Nguyen, Grenville J. Armitage

Centre for Advanced Internet Architectures
Swinburne University of Technology
Melbourne, Australia
{tnguyen,garmitage}@swin.edu.au

Abstract – This paper experimentally identifies usage scenarios that trigger IP performance limitations in two common Internet access technologies: DOCSIS cable networks and 802.11b wireless local area networks. We use commercial, standards-compliant implementations of each link technology and demonstrate that data transfers from a remote server to a wireless- or cable-attached client can create substantial latency spikes (upwards of 100ms) on the shared wireless or cable link segment, despite each technology's generous downstream link bandwidth. These spikes have a significant impact on delay-sensitive applications (such as voice over IP, online games or interactive streaming video) sharing the link. We also observe the negative impact of 802.11b's CSMA/CA on end-to-end TCP performance in the presence of low bandwidth, non-reactive traffic, and DOCSIS request/grant cycle on maximum DOCSIS upstream and downstream bandwidths. We illustrate the former point by calculating the performance degradation of an 802.11b link shared by online game players. Our results should motivate future work on optimised media access algorithms for 802.11b and DOCSIS.

I. Introduction

The most popular broadband Internet access technologies today include DOCSIS (Data Over Cable Service Interface Specification) cable [1] and 802.11b wireless networks. A large range of applications is driving uptake – higher speed web downloads, real-time content streaming, interactive voice and video services, and online multiplayer gaming. However, the service quality implications of mixed interactive and non-interactive applications have not been fully explored.

This paper reports on an experimental study using standards-compliant, commercial DOCSIS and 802.11b equipment in a lab environment. Our IP over DOCSIS and wireless testbed [2] allowed us to explore interactions between TCP flow control behavior, upstream (US) and downstream (DS) rate limits and end-to-end latencies experienced by applications sharing a DOCSIS link. We also emulate and study the IP service quality for multiple wireless clients sharing the same Access Point (AP) at a hot spot (such as an Internet Café or at the airport). The results of our study will be useful in motivating and guiding future work on priority queuing and packet scheduling systems in these networks.

We demonstrate that a capped link from the Cable Modem Termination System (CMTS) to a Cable Modem (CM) and a limited link capacity between an Access Point (AP) and a wireless client can introduce substantial delay to other traffic sharing the DOCSIS link or the wireless medium over end-to-end path. Delay-sensitive applications sharing the network resources, such as VoIP or interactive voice, video or online game traffic, are affected by the increase in per-hop latency.

One of the major sources of performance degradation of an 802.11b network has its origins in the network's CSMA/CA link sharing and access mechanism. We are providing a lower bound on how much capacity might be stolen, and in fact it is quite possible in the multi-host scenarios that TCP throughput(s) will degrade worse. We show that low-rate, non-reactive packet flows to and from one client can 'steal' significant capacity from concurrent TCP flows to other clients. For example, a flow of 64 byte 'ping' ICMP packets at 250 packets per second (roughly 128Kbps) degrades a concurrent TCP flow's throughput by up to 50% (from 4Mbps to ~2Mbps). An approximation of 20 Half-Life or 10 Quake III players with total game traffic of less than 1Mbps would degrade concurrent TCP throughput by at least ~ 3.5 Mbps. This has implications for the use of UDP-based audio and video conferencing and game applications at 802.11b hotspots or in enterprise networks.

Similarly, the DOCSIS media access request/grant cycle also strongly affects the maximum DS and US transmission rates, and hence potential TCP performance across a DOCSIS link, regardless of the actual bandwidth limits assigned to the DS and US channels.

The rest of the paper is organized as follows. Section II provides a background on 802.11b and DOCSIS networks. Section III outlines details of our experimental setup, and presents analysis of the findings and theoretical verification of our results. Section IV demonstrates the interaction of networked games (Half-Life and Quake III) and 802.11b network performance. Section V briefly discusses some related work and our contributions. The paper is concluded in section VI with some discussions of implications for mixing interactive and non-interactive traffic.

ISBN 0-646-44190-6

Australian Telecommunications Networks & Applications Conference 2004 (ATNAC2004), Sydney, Australia, December 8-10 2004

II. Background

A. 802.11b networks

The IEEE's 802.11b specifications define the physical layer and media access control (MAC) sublayer for communications across a shared, wireless local area network at up to 11Mbps. At the physical layer, IEEE 802.11b radios operates at 2.45 GHz and use direct sequence spread spectrum (DSSS) transmission. At the MAC sublayer 802.11b uses carrier sense multiple access with collision avoidance (CSMA/CA) [3].

802.11b operates in either Ad hoc mode or Infrastructure mode. In ad hoc mode wireless clients communicate directly with each other. In infrastructure mode, wireless clients communicate with a wired network (an enterprise LAN or Internet connection) or other clients via an Access Point (AP). Infrastructure mode networks consist of APs, wireless clients (computing devices with 802.11b-based network interfaces) and a wired network. An AP acts as an Ethernet bridge between wireless clients and the wired network. Our study focuses on 802.11b in infrastructure mode.

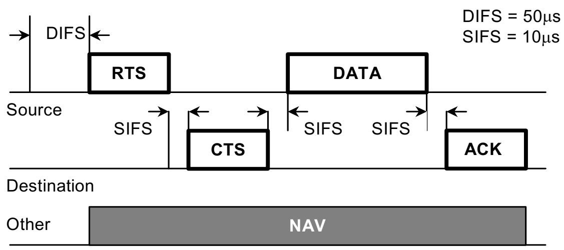

Fig. 1. RTS/CTS and data transactions[1].

802.11b transmission medium is half-duplex. It uses Positive Acknowledgement (ACK) of every transmission and a Virtual Carrier Sense mechanism to reduce the probability of two clients colliding [5][6].

A client wanting to transmit senses the medium and defers if the medium is busy. The client transmits when the medium is free for a specified time (the Distributed Inter Frame Space, DIFS). First, a Request to Send (RTS) control packet is transmitted, carrying the source, destination, and duration of the desired transaction (a data packet and the corresponding ACK). The receiver responds (if the medium is free) with a Clear to Send (CTS) control packet which include the same duration information. Upon receiving a CTS, the sender waits for a Short Interframe Space (SIFS) then sends the data packet. The receiver checks the received packet for errors, waits for a SIFS and sends an ACK packet. Receipt of the ACK indicates to the sender that no collision occurred. The sender retransmits the data frame until it gets acknowledged (and throws the data frame away if not ACKed after a number of unsuccessful retransmissions).

All stations receiving either the RTS and/or the CTS will set their Virtual Carrier Sense indicator (called Network Allocation Vector, NAV), for the duration. Would-be senders use the status of their current NAV, in conjunction with physical carrier sensing, to decide if the medium is likely to be in use at any given time. Fig. 1 shows the transactions between two wireless clients and the NAV status of a third, neighboring node [5].

1. 802.11b encapsulation

In addition to the payload data, the MAC frame encapsulation process adds 42 additional bytes of overhead. The 802.11 MAC header adds 30 bytes of data for various control and management functions, error detection, and addressing and a trailing 4 byte Frame Check Sequence (FCS). LLC/SNAP encapsulation adds another 8 bytes [7].

A PLCP (Physical Layer Convergence Protocol) header and a PLCP preamble is prepended to every frame before it is transmitted. These headers are transmitted at 1Mbps. The PLCP preamble may be either a "long" preamble of 18bytes, or a "short" preamble of 9 bytes. Long preamble is the default setting on most devices so we based our theoretical calculations on a long preamble that takes 192µs to transmit [8].

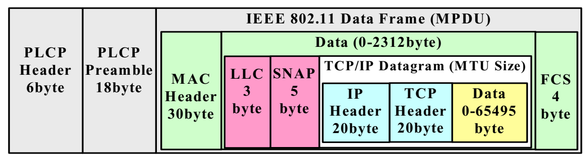

Fig.2. 802.11frame encapsulation.

2. 802.11b timeline

The MAC frame is transmitted as a series of 8-bit symbols at maximum 1.375 million symbols per second. From this we can estimate the transaction time for a hypothetical TCP stack that requires one TCP ACK for every TCP Data packet. A 1500 byte TCP/IP data packet thus generates 1542 symbols in the MAC data frame, while the TCP ACK frame generates 82 symbols. The 802.11b ACK is 14 bytes long, and the RTS and CFS packets are 20byte and 14byte respectively. Based on this, we can calculate the overall transaction time as shown in Table 1[8].

Table 1
TCP Transaction Time

	1500-byte MTU TCP Data (µs)	TCP ACK (µs)
DIFS & SIFS	50 + 10*3 = 80	50 + 10*3 = 80
RTS & CTS	192*2 + (20 +14)/0.125 = 656	192*2 + (20 +14)/0.125 = 656
802.11 Data	192 + 1542/(1.375) = 1,313.4	192 + 82/(1.375) = 251.6
802.11 ACK	192 + 14/(1.375) = 203	192 + 14/(1.375) = 203
Frame exchange total	2252.4	1190.6
Total Transaction	**3443**	

B. DOCSIS networks

While communication between an AP and a wireless

[1] Back off time scheme [4] is not considered here to simplify the analysis. Our calculations only provide a lower bound on TCP throughputs degradation in multi-base stations scenarios.

ISBN 0-646-44190-6

Australian Telecommunications Networks & Applications Conference 2004 (ATNAC2004), Sydney, Australia, December 8-10 2004

client over the 802.11b network is established over a half-duplex medium, using CSMA/CA and positive ACK scheme, communication between a CM and the CMTS in a DOCSIS network is full-duplex, with configurable downstream and upstream maximum link bandwidths.

Communication between the CM and the CMTS relies on a reservation scheme, which is normally called the request/grant cycle. A CMTS periodically sends MAC allocation and management messages (MAPs) to all CMs on the network, defining the transmission availability of channels for specific periods of time. The MAP message transmission interval can be dynamic or fixed (our Cisco CMTS allowed dynamic intervals between 100usec and 2ms, or a fixed interval defaulting to 2ms). Before sending data upstream the CM must ask permission from the CMTS for a time slot to transmit. The shared request time slots in broadcast MAP messages allow the CM sending a request time for US transmission. Upon receiving the CM's request, the CMTS grants time slots according to slot availability and queues the grant message for transmission back to the CM during the next MAP transmission time. The maximum burst size of data that a CM can send upstream per MAP opportunity is limited as specified in the SID for the CM. We use the default US max-burst size of 1600 bytes. This back and forth communication mandated by the DOCSIS protocol produces additional latency into the network's performance [9].

III. Test Setups & Findings

A. Impact of Downstream being a bandwidth bottleneck

From [9] we already demonstrated that the presence of traffic in the DS direction causes a dramatic increase in RTT from approximately 13ms when the link is

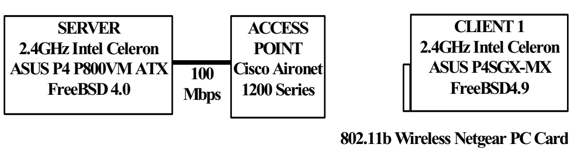

Fig.3 – 802.11b Bottleneck Effects Test Setup.

essentially idle to over 100ms when the link is loaded, at the point where the server's offered load begins to exceed the DS rate cap.

We now run similar test with the 802.11b network and examine the relationship between overall TCP performance and rate caps between the server and the AP. The test configuration is shown in Fig.3

We used FreeBSD for our client and server hosts. We repeatedly ran nttcp [10] from server to client with different link MTUs, and gathered round trip time (RTT) estimates before, during and after each run using ICMP ping (one per second) from client to server. Each trial transferred 8Mbyte (using the nttcp default of 2048 4Kbyte buffers) three times with TCP client window of 32Kbyte.

Our experimental results for this test are similar with what we had seen with the DOCSIS network. The presence of traffic in the server to client direction also causes a significant increase in RTT (with the actual increase depending noticeably on MTU). An idle link shows 2.6ms RTT. During the nttcp transfer phase the RTT jumps to just over 120ms at an MTU of 512 bytes, 70ms with 1000 byte MTU, 67ms with 1200 byte MTU

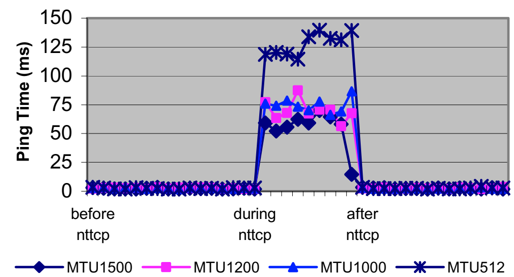

Fig. 4. Ping Time before, during and after nttcp transfer.

and 55ms with 1500 byte MTU respectively (Fig. 4).

To characterize this increase in RTT as a function of offered load we artificially throttled the server to client data rate at the server using FreeBSD's kernel-resident 'dummynet' module. We set dummynet's internal queue limit to 62Kbytes and applied bandwidth limits between

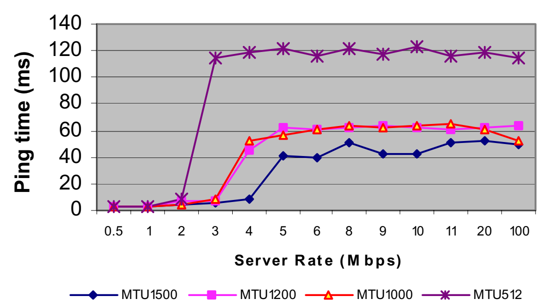

Fig.5. Ping Time during nttcp transfer for all MTUs.

500Kbps and 100Mbps (the natural rate of the server to AP link). TCP ACKs and all ICMP packets were not rate limited.

We repeat the test with different MTU sizes of 1500, 1200, 1000 and 512 bytes, with the maximum TCP window sizes of 32Kbyte. Fig.5 shows the average RTT during an nttcp transfer as a function of server-side rate limit. For each MTU the server rate is varied from 500Kbps to 100Mbps.

The RTT increases dramatically at the point where the server's offered load begins to exceed the wireless link's maximum rate for the specific MTU, at ~2Mbps for MTU 512, ~3Mbps for MTU 1200 and 1000, and ~4Mbps for MTU 1500 (taking into the account the 802.11b MAC overheads discussed in Section II).

To sum up, despite the relatively high downstream capacity of both DOCSIS and 802.11b networks it is clear that potential exists for some rather undesirable end-to-end behaviours. The spike in RTT over the

ISBN 0-646-44190-6

downstream links affects all traffic that sharing the particular CM in a DOCSIS network and an AP in a wireless network. This also has significant real-world implications. For example, consider an ISP hosting local content servers on their 100Mbps or 1Gbps backbone and

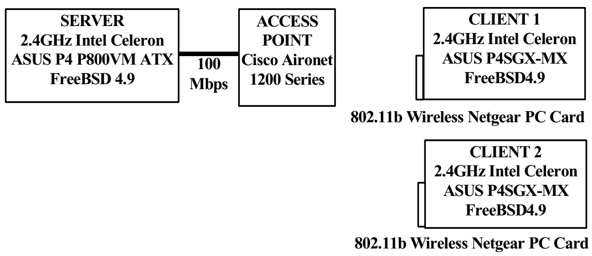

Fig. 6.Impacts of CSMA/CA scheme Test setup.

encouraging their directly attached 'broadband' customers to download locally rather than from distant servers. Such customers are likely to discover their RTT to other parts of the Internet jumping up by over 100ms or 50ms during the local content transfer phase. Although probably not noticeable if the customer isn't doing anything else at the time, the RTT jump will be highly disruptive if the customer site was attempting concurrent interactive Voice over IP or online gaming.

It is also clear that configuring the optimal window size based on the RTT of an idle DOCSIS or 802.11b link would provide a highly sub-optimal result. While most customers are unlikely to be manually tweaking their operating system's TCP window sizes, those who do are quite likely to complain to their ISP's helpdesk. It is worth knowing about the RTT jump if only to help inform such customers about what they should expect and why it is normal.

B. *Impact of 802.11b's CSMA/CA scheme*

In this test we examined TCP performance from one wireless client while a low-rate, non-reactive traffic flow (in our case a flow of ICMP 'ping' packets) from another wireless client competes for resources on the 802.11b link (Fig. 6).

Our server and both clients ran FreeBSD 4.9. Repeated runs of nttcp were used to measure TCP performance between client 1 and the server. Each trial

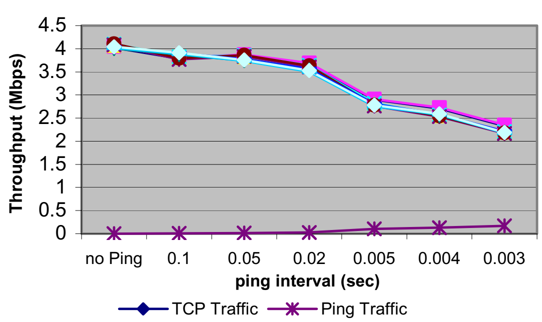

Fig. 7. nttcp throughput vs. Ping rate

transferred 8Mbyte (using the nttcp default of 2048 4Kbyte buffers) three times. We injected interfering traffic over the wireless link by concurrently "pinging" the Server from Client2, using ping interval from 100ms down to 3ms.

The resulting degradation in TCP throughput was quite dramatic, particularly given the relatively low bandwidth of the competing ICMP traffic flow (Fig. 7). When the ping interval was 3ms (roughly 171Kbps given 64 byte ICMP packets) the TCP throughput dropped by 50% (from 4Mbps to approximately 2Mbps). Different curves in the figure represent the results of different trials run.

An explanation for this observed behavior could be found by closer analysis of the 802.11b frame transmission protocol. We know from TABLE 1 that a TCP transaction requires 3443µs. The time taken to complete one ping transaction (an ICMP echo request and ICMP echo reply) is calculated in TABLE 2 to be ~2416.2µs for a 64 byte ping packet. If we treat every ping transaction as a lost opportunity for transmitting TCP data, then we can predict the TCP degradation fairly well. For example, assume we sending one ping every 4ms, i.e. 250 packets per second or roughly 128Kbps. The total time taken by 802.11b link to handle these transactions would be 250*2416.2µs = 604.05ms. During that time, 604.05ms/3443µs = 175.4 TCP transactions could have occurred if there had been no competing ping traffic. With a 1500byte MTU nttcp would lose ~2.05Mbps.

TABLE 2

PING TRANSCTION TIME

	64-byte Echo Request & Reply (µs)	**128-byte Echo Request & Reply (µs)**	**256-byte Echo Request & Reply (µs)**
DIFS + RTS +CTS + SIFS	736	736	736
802.11 Data	192 + (64+42)/(1.375) = 269.1	192 + (128+42)/(1.375) = 315.6	192 + (256+42)/(1.375) = 408.7
802.11 ACK	192 + 14/(1.375) = 203	192 + 14/(1.375) = 203	192 + 14/(1.375) = 203
Frame exchange total	1208.1	1254.6	1347.7
Total Transaction	1208.1*2 = **2416.2**	1254.6*2= **2509.2**	1347.7*2 = **2695.4**

Fig. 8 shows the nttcp throughput seen at client 1 (the continuous line) and the effective nttcp throughput 'stolen' by the ping flow from client 2 (the dotted line). The sum of these rates matches that achieved by nttcp in the absence of competing traffic. (The predicted total drifts higher than 4Mbps due to our simplified calculation of equivalent TCP throughput was stolen by the competing ICMP traffic).

Fig. 9 shows the nttcp results when the interfering ping packets were 128 and 256 bytes long (thin solid lines), the predicted 'stolen' capacity due to the ping traffic (dotted lines), and the sum total (thick solid lines). These results indicate that an 802.11b's link shared link capacity can be substantially degraded with only modest level of competing traffic (e.g. 100 to 200 packets per second). At such low rates the very act of transmitting the

Australian Telecommunications Networks & Applications Conference 2004 (ATNAC2004), Sydney, Australia, December 8-10 2004

ISBN 0-646-44190-6

Australian Telecommunications Networks & Applications Conference 2004 (ATNAC2004), Sydney, Australia, December 8-10 2004

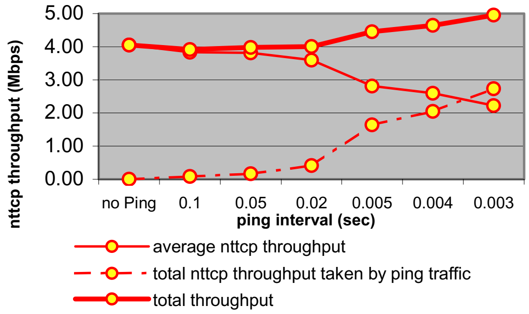

Fig. 8. nttcp throughput of TCP traffic and taken by ping traffic.

ICMP packets was more significant then their size in 'stealing' capacity from other clients on the link.

We have related our experimental observations back to a theoretical model of performance estimation. From this basis we can work out the expected maximum throughput that the AP can provide under different circumstances. For example, if we were only sending ICMP ping packets the 802.11b link could handle no more than 413 pings per second (10^6/2416.2µs). If the traffic was solely TCP with 1500 byte TCP/IP data

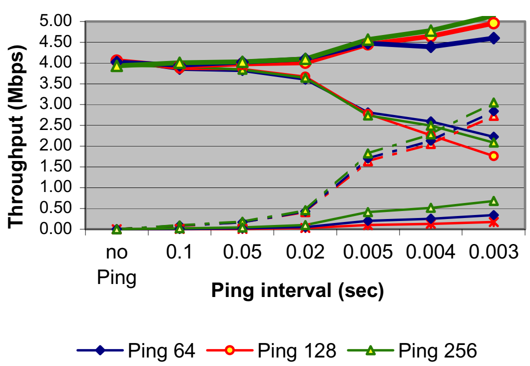

Fig. 9. Results for ping packet sizes of 64, 128 and 256 bytes.

packets we would be limited to 290TCP transactions per second (10^6/3443µs).

C. Impacts of MAPs request/grant cycle in a DOCSIS network

From [9] we had shown that both DS and US throughputs are constrained by the MAPS request/grant cycle. An estimation of 256kbps of the DS throughput would be used purely for MAPs transmission. With a default MAP interval of 2ms, it would lead to maximum of 3Mbps for upstream bandwidth and a limit of 500 PPS in the downstream direction, regardless of the actual bits/second limit assigned to the DS or US channels.

IV. 802.11B AND NETWORKED GAMES

One scenario that could easily lead to unexpected

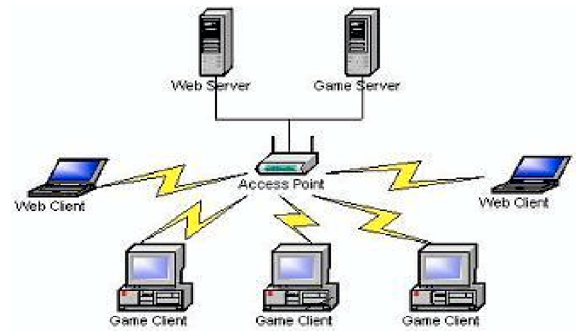

Fig. 10 - 802.11b wireless network scenario.

performance degradation is where a number of 802.11b-enabled game clients cluster around an 802.11b hot-spot, or utilize an 802.11b enterprise network as a backbone for a LAN-party. To estimate the impact of highly interactive game client traffic on an 802.11b network we combine previously published results for Quake III and Half-Life traffic ([11][12]) with our earlier analysis of the 802.11b media access protocol.

A. Quake III traffic

For Quake III the packet transmission rate from the server to an individual client is almost constant, independent of all parameters (maps, number of players, client hardware), at one packet every 50ms, hence the packet per second (PPS) rate of 20 packets/sec. The packet lengths, however, are strongly dependent on the number of players participating in the game and the map they play [11]. Results from [11] had also shown that the former is the dominant parameter governing the packet length distribution, and the later parameter could be ignored. The packet length distribution of the 2-player game is a lognormal with mean ~79.340 and standard deviation ~0.245. With an additional client, the mean packet length increases by an average of 13 bytes, and the packet length distribution becomes more normal as the number of clients increases. For the purpose of this analysis, we use only the mean values. If N is the number of players participating in the game and $f_{(N)}$ is the packet length as a function of N, we have:

$$f_{(N)} = 79.340 + (N-2)*13 = 13N + 53.34 \text{ with } N >= 2$$

For the client to server traffic, the packet length distribution is independent of all observed parameters, and therefore the same for each client participating in the game, which is a normal with mean ~64.151 and standard deviation of 3.203. The packet transmission rate, on the other hand, is dependent on the map played and the client graphic card. In this paper we choose to analyze the most demanding scenario discussed in [11] (namely clients with modern graphics cards), where packets are transmitted to the server every 10.75ms (~93 PPS).

With the traffic profile above, we estimate the time taken per second for a client playing Quake III game over the 802.11b network in TABLE 3.

ISBN 0-646-44190-6

Australian Telecommunications Networks & Applications Conference 2004 (ATNAC2004), Sydney, Australia, December 8-10 2004

TABLE 3

AN ESTIMATION OF 802.11B NETWORK RESOURCES CONSUMED BY QUAKE III TRAFFIC IN TERMS OF TRANSACTION TIME PER SECOND PER CLIENT

Quake III Traffic	Client to Server (µs)	Server to 1 Client (µs)
DIFS + RTS +CTS + SIFS	736	736
802.11 Data	192 + (64.151+42)/1.375 = 269.2	192 + ((13N+53.34)+42)/1.375 = 9.45N + 261.3
802.11 ACK	203	203
Frame exchange total	1028.2	9.45N + 1200.3
Packets Per Second Rate	93	20
Total Transaction Time (per sec) for N clients	**(189N + 119,628.6)N**	

Based on the results in TABLE 3 we calculated how much effective TCP capacity would be lost to (or perhaps

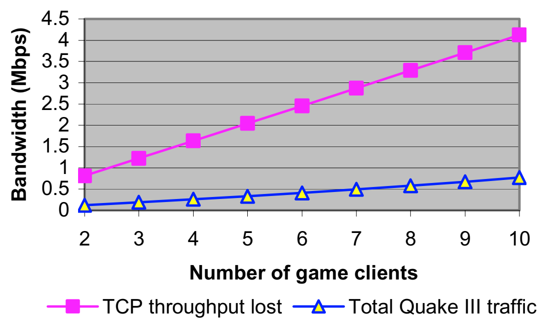

Fig.11. Estimated equivalent TCP capacity consumed by Quake III traffic.

'stolen' by) a group of two to eleven game players play with each other sharing an 802.11b access point (the number of players per game is equal to the number of game clients sharing the 802.11b access point).

Fig.11 shows the nominal average bandwidth consumed by the aggregate client-server and server-client traffic as a function of number of game clients (given the packet rate and average packet size). It also shows the effective capacity reduction ('TCP throughput lost') caused by carrying the game traffic as a function of number of game clients (based on the number of media accesses per second that remain available to other traffic flowing through an access point). For example, although the actual bandwidth requirement of 10 Quake III players is less than 1Mbps, they would "steal" roughly 4Mbps of potential TCP throughput respectively on an 802.11b network.

B. Half-Life traffic

Similarly, we analyze the effects of Half_Life traffic on 802.11b performance based on the traffic profile given in [12]. For server to client traffic, the packet inter-arrival time used for this analysis is at an average of 60ms, i.e. 16 PPS. The packet length is dependent on the map played. We do the analysis for four different maps: ChilDM, Odyssey, Rats3 and Xflight, which have the packet length distribution as a lognormal with mean 202.9, 154.1, 129.6 and 109.7 respectively.

Fifty percent of client to server traffic has an inter-arrival time of 33ms while the other 50% are sent every 50ms. For the purpose of this analysis, we use the average of PPS rate of 25 packets/sec. Client packet length distributions are independent of client computer hardware, number of players or maps, as lognormal with mean of 72.3 bytes.

TABLE 4 estimates the time taken per second for a user playing Half-Life game over the 802.11b network.

TABLE 4

AN ESTIMATION OF 802.11B NETWORK RESOURCES CONSUMED BY HALF-LIFE TRAFFIC IN TERMS OF TRANSACTION TIME PER SECOND PER CLIENT

Half-Life Traffic	Client to Server (µs)	Server to 1 Client (µs)			
		ChilDM	Odyssey	Rats3	Xflight
DIFS + RTS +CTS + SIFS	736	736	736	736	736
802.11 Data	192+ (72.3+42)/ 1.375	192 + (202.9+42)/ 1.375	192 + (154.1+42)/ 1.375	192 + (129.6+42)/ 1.375	192 + (109.7+42)/ 1.375
802.11 ACK	203	203	203	203	203
Frame exchange total	1214.1	1309.1	1273.6	1255.8	1241.3
Packets Per Second Rate	25	16	16	16	16
Total Transaction Time (per sec) for N clients		**51298.1*N**	**50730.1*N**	**50445.3*N**	**50213.3*N**

Based on the results in TABLE 4 we calculated how much effective TCP capacity would be lost to (or perhaps 'stolen' by) a group of one to twenty game players

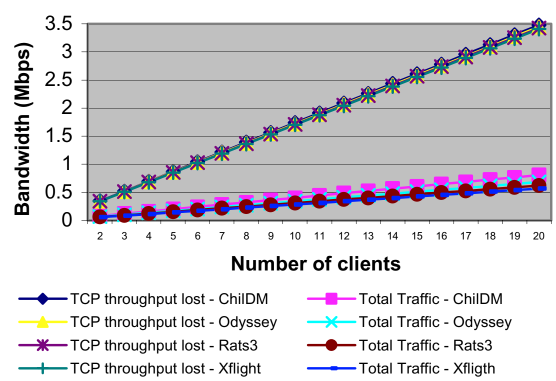

Fig. 12. Estimated equivalent TCP capacity consumed by Half-Life traffic.

sharing an 802.11b access point. Fig. 12 shows the nominal average bandwidth consumed by the aggregate

ISBN 0-646-44190-6

client-server and server-client traffic as a function of number of game clients. It also shows the effective capacity reduction ('TCP throughput lost') caused by carrying the game traffic as a function of number of game clients (based on the number of media accesses per second that remain available to other traffic flowing through an access point) for four different game maps. As showing in the figure, although the actual bandwidth requirement of 20 Half-Life players is less than 1Mbps, they would "steal" roughly 2Mbps to 2.5Mbps of potential TCP throughput respectively on an 802.11b network.

V. Related Work and Our Contributions

A number of papers have studied the performance of 802.11b network. Effects of a wireless client who used lower bit rate on other mobile hosts sharing the link were studied in [13]. The situation considered was when a host was far away from an AP and hence was subject to signal fading and interference, which caused its bit rate to degrade from 11Mbps to 5.5, 2 or 1 Mbps. In such a case, for other hosts sharing the AP, although they were transmitting at 11Mbps, would degrade to a rate of lower than 1Mbps due to the CSMA/CA channel access method. [14] investigated the short-term unfairness of the CSMA/CA as implemented in the WaveLAN network. [15] characterized the expected performance of the standard's ad hoc and infrastructure 802.11b networks. Its simulation models incorporate the effect of burst errors, offered load, packet size, RTS threshold and fragmentation threshold on network throughput and delay.

In this paper, we measure and characterize the negative impact of 802.11b's CSMA/CA on end-to-end TCP performance in the face of low bandwidth, non-reactive traffic. We provide a lower bound TCP throughput degradation that low-rate, non-reactive packet flows to and from one client can 'steal' significant capacity from concurrent TCP flows to other clients. (In fact with the existence of collisions, it is quite possible that TCP throughput(s) will degrade much worse). This has implications for the use of UDP-based audio and video conferencing and game applications at 802.11b hotspots or in enterprise networks. Other parameters such as bottleneck bandwidth and MTU sizes are also considered and investigated in our work.

By looking at both DOCSIS and 802.11b networks in this paper, we had shown the similarity in these networks' reaction to the bottleneck link problem. We also pointed out the negative impacts of the channel access schemes of both networks.

While most other papers use simulation [15][16]or Markov chains [14] for their analysis, our major contribution is the use of direct trials on commercial equipment, rather than relying on simulations that (of necessity) do not always properly implement all aspects of the respective protocols. Our experimental results also quantitatively verify the effect of non-reactive traffic on data traffic sharing the wireless network resource as mentioned in [15].

VI Conclusions

Internet and intranet services are being deployed around the world using DOCSIS-based cable and 802.11b-based wireless LAN links, promising higher speeds and better overall performance to consumers. However, we have experimentally demonstrated that end-to-end service over commercial implementations of these link technologies exhibits a few non-obvious characteristics.

For example, round trip time (RTT) spikes seen over a loaded DOCSIS or 802.11b link has significant implications for ISPs who wish to concurrently host local content and yet support interactive applications such as voice, video and online games through their AP(s) and CM(s). This is highly relevant to ISPs who encourage their clients to use local, well-connected content servers (either explicitly, or e.g. by transparently forcing client web browsing through a local caching proxy). All clients sharing the AP or the CM will find their RTT to other parts of the Internet jumping by over 50ms or 100ms respectively while someone is performing local content transfer– rather disruptive to other clients who may be engaged in online game play or teleconferencing at the time.

We have also observed that the CSMA/CA scheme in 802.11b networks significantly affect how TCP performs in the presence of non-reactive flows from other interactive applications. For example, a UDP-based IP telephony application, video conferencing application or online game can 'steal' far more capacity than would be predicted from the application's average packet rates and packet sizes. We illustrated this by modeling two representative online games - Half-Life and Quake 3 - played through an 802.11b access point. Under a typical usage scenario, 20 people playing Half-Life or 10 people playing Quake 3 would 'steal' at least 3.5 Mbps of potential 802.11b link capacity even though each game's actual aggregate bandwidth requirement is below 1 Mbps.

Our experiments illuminate a number of factors that wireless and DOCSIS network operators should consider when deploying these networks' services to customers with heterogeneous applications and requirements. We expect our work will motivate and focus future work on priority queuing and packet scheduling system in both the DS and US directions of DOCSIS systems and 802.11b wireless links, and especially in the request/grant cycles and the CSMA/CA schemes of these access networks.

Our future work will carry out the actual experiments with the game scenario discussed in the paper, also take into account the impacts of other factors, such as the backoff time in the CSMA/CA scheme, collision rate, packet size and transmission probability.

Acknowledgment

We thank Cisco Systems Australia for supporting our work at the Centre for Advanced Internet Architectures through equipment donations and post-graduate student stipends.

We would like to thank the anonymous reviewers for their evaluation feedback to improve the paper and our

ISBN 0-646-44190-6

future work.

References

[1]. CableLabs, "Data-Over-Cable Service Interface Specifications Radio Frequency Interface Specification SP –RFIv1.1-I01-990311," 1999.

[2]. Centre for Advanced Internet Arcnitectures, "Broadband Access Research Testbed," http://www.caia.swin.edu.au/bart (as of July 2004).

[3]. "IEEE 802.11b Wireless Networking Overview," Microsoft Technet, March 2002 http://www.microsoft.com/technet/community/columns/cableguy/cg0302.mspx (as of May 2004).

[4]. Bianchi, G 2000, 'Performance Analysis of the IEEE 802.11 Distributed Coordination Function,' IEEE Journal on Selected Areas in Communications, vol. 18, pp. 535-547.

[5]. Brenner,P., "A Technical Tutorial on the IEEE 802.11 Protocol," Spread Spectrum Scene Online, 1997, http://www.sss-mag.com/pdf/802_11tut.pdf (as of May 2004).

[6]. Campbell, T.A, "IEEE 802.11 Wireless LAN," http://www.columbia.edu/itc/ee/e6951/2002spring/LectureNotes/lecture3_4.pdf (as of May 2004).

[7]. Leira, J. "Throughput," UNINETT, March 2003 http://www.uninett.no/wlan/throughput.html (as of May 2004).

[8]. Gast,M. " When Is 54 Not Equal to 54? A Look at 802.11a, b, and g Throughput," O'Reilly Wireless Devcenter, Aug 2003 http://www.oreillynet.com/pub/a/wireless/2003/08/08/wireless_throughput.html (as of May 2004).

[9]. T.T.T.Nguyen, G.Armitage "Experimentally derived interactions between TCP traffic and service quality over DOCSIS cable links," (accepted for publication) Global Internet and Next Generation Networks Symposium, IEEE Globecomm 2004, Texas, USA, November 2004.

[10]. http://www.leo.org/~elmar/nttcp/ (as of May 2004).

[11]. T.Lang, P.Branch, G.Armitage "A Synthetic Traffic Model for Quake 3," ACM SIGCHI ACE2004, Singapore, June 2004.

[12]. T.Lang, G.Armitage, P.Branch, H-Y.Choo. "A Synthetic Traffic Model for Half-Life," Australian Telecommunications Networks & Applications Conference 2003 (ATNAC 2003), Melbourne, Australia, December 2003.

[13]. M. Heusse, F. Rousseau, G. Berger-Sabbatel, and A. Duda, "Performance Anomaly of 802.11b," in Proceedings of IEEE INFOCOM 2003, San Francisco, 2003.

[14]. C. Koksal, H. Kassab, and H. Balakrishnan, "An Analysis of Short-Term Fairness in Wireless Media Access Protocols," in Proceedings of ACM SIGMETRICS, 2000.

[15]. B. P. Crow et al., "Investigation of the IEEE 802.11 Medium Access Control (MAC)," in Proceedings of INFOCOM 1997, 1997.

[16]. M. Natkaniec et al., "Analysis of Backoff mechanism in IEEE 802.11 standard," in Proceeding of ISCC 2000, 2000.

[17]. S. G. Xu and T. Saadawi, Revealing the problems with 802.11 medium access control protocol in multihop wireless ad hoc networks, Computer Network 38 (2002) pp 531-548.

Australian Telecommunications Networks & Applications Conference 2004 (ATNAC2004), Sydney, Australia, December 8-10 2004

ISBN 0-646-44190-6