Network Measurements of a Wireless Classroom Network

Carey Williamson Nuha Kamaluddeen
Department of Computer Science, University of Calgary
Email: {carey,kamaludd}@cpsc.ucalgary.ca

Abstract

The advent of mobile computers and wireless networks enables the deployment of wireless Web servers and clients in short-lived ad hoc network environments, such as classroom area networks. This paper studies wireless Web performance in a classroom environment. Experiments are conducted on an ad hoc IEEE 802.11b wireless LAN, using a wireless-enabled Apache Web server, several wireless clients, and a wireless network analyzer. The experiments focus on the HTTP performance and network throughput achievable in the wireless classroom environment.

1 Introduction

Two exciting and highly popular Internet technologies are the World Wide Web and wireless networks. The Web has made the Internet available to the masses, and wireless technologies have revolutionalized networks, by freeing users from the constraints of physical wires.

A natural step in the wireless Internet evolution is the convergence of these technologies to form the "wireless Web": the wireless classroom, the wireless campus, the wireless office, and the wireless home [5, 8, 14, 15]. Interestingly, the same technology that allows Web clients to be mobile (i.e., wireless network cards) also enables the deployment of wireless Web servers.

Wireless Web servers play a valuable role in *short-lived networks*. A short-lived (or *portable*) network is created in an *ad hoc* fashion at a particular location in response to some event. The network operates for a short time period (minutes to hours), before being disassembled, moved, and reconstituted elsewhere. Examples of deployment scenarios for short-lived networks are sporting events, press conferences, conventions and trade shows, disaster recovery sites, and classroom area networks [4]. In many of these contexts, an ad hoc wireless network, with a wireless Web server as an information repository, provides a suitable solution [3, 4, 9].

This paper studies the feasibility of wireless Web servers in classroom area networks. The paper reports measurements from wireless Web server usage in a classroom environment at the University of Calgary in February 2003. We use an Apache Web server running on a laptop computer with an IEEE 802.11b wireless LAN interface to study in-classroom Web performance for wireless Web clients. All mobile computers are configured in ad hoc mode, since no existing network infrastructure is assumed. The clients download course content from the wireless Web server. A wireless network analyzer is used to collect and analyze traces from the experiments, with traffic analysis spanning from the Medium Access Control (MAC) layer to HTTP at the application layer.

Our experiments focus on the HTTP performance and network throughput achievable in an ad hoc wireless network environment, and the impacts of factors such as Web object size, persistent

HTTP connections [10], user behaviour, Web browser type, and TCP behaviours [13]. The results show satisfactory performance in the wireless classroom network, with throughput ranging from 1-5 Mbps, depending on the nature of the Web workload. The bandwidth is shared fairly amongst TCP connections, and user response times are acceptable. Persistent HTTP connections improve the performance for mobile clients accessing content from a wireless Web server.

2 Experimental Setup

Our network traffic measurement experiments are conducted on an IEEE 802.11b wireless LAN in a classroom at the University of Calgary. The configuration, shown in Figure 1, consists of about 8 mobile clients and one Web server. In addition, we use a wireless network analyzer to monitor the wireless channel.

The Web server machine is a Compaq Evo Notebook N600c running RedHat Linux 7.3 and X windows, using a 1.2 GHz Mobile Intel Pentium III. Most of the client machines are similarly-configured Compaq Evo notebooks, but running the Windows 2000 operating system rather than Linux. Two Compaq iPAQ PDA clients running Windows CE were also used.

Each wireless device (server and clients) has a Cisco Aironet 350 Series Adapter for access to the IEEE 802.11b wireless LAN. The wireless cards are configured to operate in ad hoc mode. For simplicity, we do not consider node mobility, multihop, or ad hoc routing issues in our experiments.

The Web server in our experiments is an Apache HTTP server (Version 1.3.23). This version is a process-based implementation of Apache, which is a flexible and powerful HTTP/1.1-compliant Web server [1, 11, 12]. Apache is currently widely deployed on the Internet, used by approximately 60% of all Web sites.

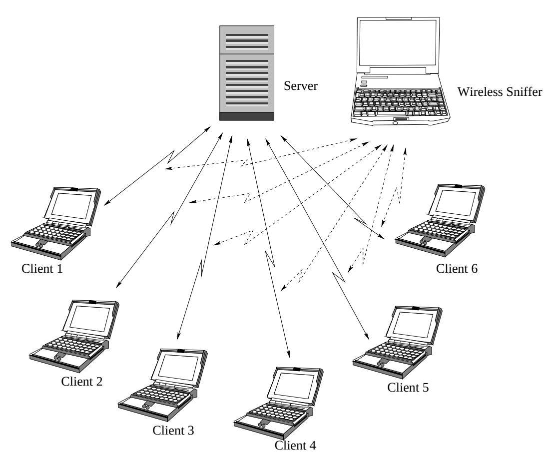


Figure 1: Experimental Setup for Wireless Classroom Network Traffic Measurements

Network traffic measurements are collected using a wireless network analyzer. Decoding of the captured traces enables protocol analysis at the MAC, IP, TCP, and HTTP layers. These traces are used to assess wireless channel contention, TCP protocol behaviours, and HTTP transaction performance.

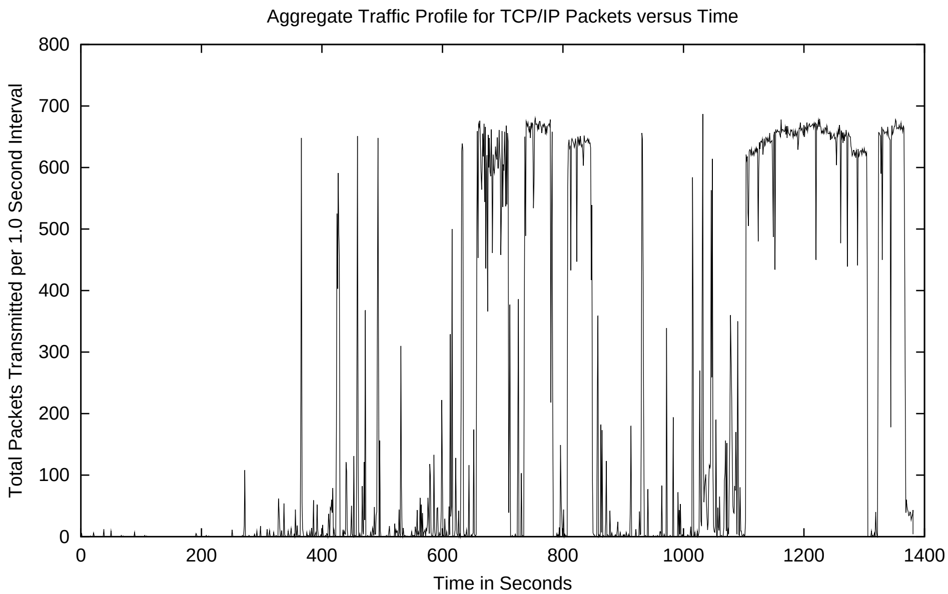


Figure 2: TCP/IP Packet Counts versus Time for Aggregate Traffic from Wireless Classroom Network (February 2003, 13 users accessing Web-based course content for 23 minutes)

3 Overview of Results

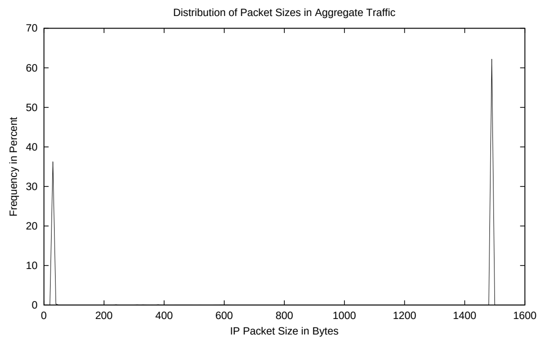
Our network traffic measurements were collected during a graduate-level networking course taught at the University of Calgary in January 2003. The course was taught in a "legacy classroom" environment that had neither wired nor wireless Internet access. Since much of the course content was available on-line (www.cpsc.ucalgary.ca/˜carey/CPSC601.38/archive/2003/), we created a mirrored copy of the course content and made it available in the classroom using a wireless Web server. The prototype was tested live in the classroom in February 2003 (during the course modules on wireless networking and network traffic measurement). Students were provided with wireless laptops and PDAs for use in the classroom at this time.

Figure 2 shows selected network traffic measurements from the classroom environment. Following a brief description of the experimental setup, the 13 students (sharing 6 laptops and 2 PDAs) were allowed to download course content, review prior lecture notes, and begin preliminary work on a course assignment (involving a large trace file). The graphs in Figure 2 show 23 minutes of wireless network activity in the classroom.

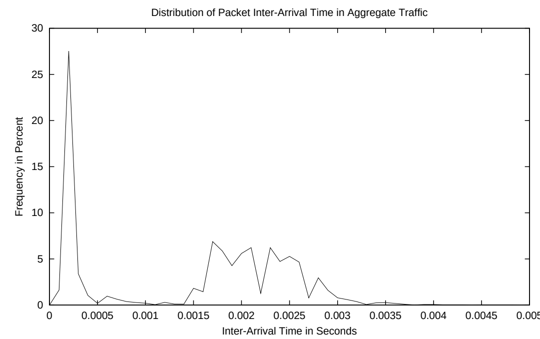
Figure 2 shows the total number of TCP/IP packets transmitted on the wireless LAN per one-second interval during the trace. The traffic is bursty, with a high peak-to-mean ratio. The peak traffic rate approaches 700 packets per second. The peak data rate achieved is approximately 5.0 Mbps. This user-level throughput is typical for an IEEE 802.11b WLAN.

Figure 3(a) shows the frequency distribution of the IP packet sizes observed. The distribution has two main peaks: one at 1500 bytes for full-size TCP/IP packets, and one at 40 bytes for TCP acknowledgements (ACKs). The peak for ACKs is lower because of TCP Delayed-ACKs, which typically result in one TCP ACK sent for every two TCP data packets received.

Figure 3(b) shows the distribution of the packet inter-arrival times on the WLAN. The tall peak



(a) Packet Size Distribution



(b) Packet Inter-Arrival Time Distribution

Figure 3: Characterization of Aggregate TCP/IP Packet Traffic

on the left reflects the inter-arrival times between a TCP ACK and the next TCP data packet. The broader hump represents the typical time spacing between TCP data packets. There is significant dispersion to this distribution because of the nature of the CSMA/CA MAC protocol in IEEE 802.11b.

Figure 4 illustrates the per-client activity for the six busiest Web clients. Clearly, the bursty aggregate traffic arises from the highly bursty behaviours of the individual clients. A single client is able to obtain most of the WLAN capacity when needed (e.g., Client 3 at time 760 seconds), while sharing the WLAN capacity if other clients are active (e.g., Clients 2, 4, and 6 around time 1200 seconds).

4 Detailed Results

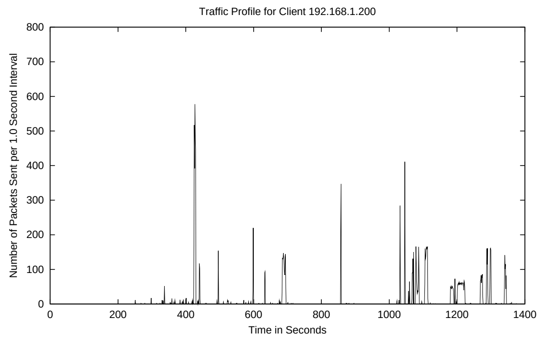
This section presents a more detailed multi-layer protocol analysis of the network traffic measurements collected from the wireless classroom network. The analyses focus on protocol behaviours for TCP (e.g., connection duration, byte volume, throughput), and HTTP (e.g., response time, object sizes, HTTP requests per TCP connection).

4.1 TCP Analysis Results

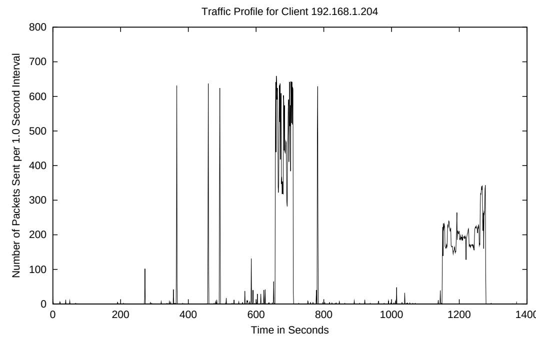
The network traffic measurement data collected from the wireless classroom environment were stored in a trace file for off-line analysis. The trace file provides information about each TCP/IP packet exchanged between the server and the clients. The trace was collected for 1400 seconds (about 23 minutes).

Connection-level analysis was performed on this data set, to understand the TCP protocol behaviours. A C program called `tcpconnparse` [6] was used for processing the trace file. This program takes a TCP trace file as input and produces summary information about the TCP connections observed in the given trace. This summary includes start time, end time, duration, total packets exchanged, and total data bytes exchanged for each TCP connection.

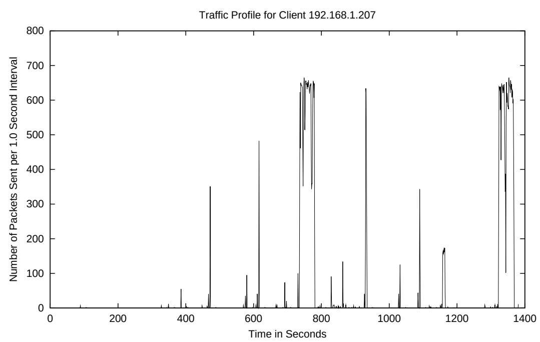
Processing the trace file from our measurements identified 262 TCP connections. More information about these connections is given in Table 1. As indicated in the table, 244 connections were considered for further analysis, while the other 18 connections were discarded.



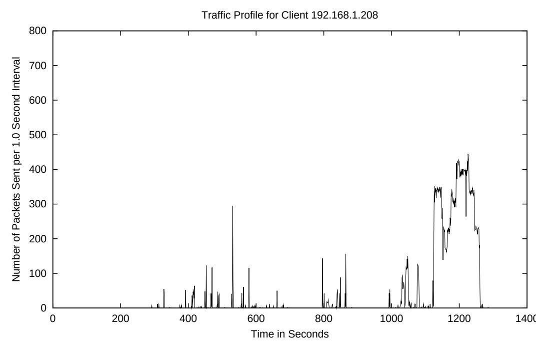
(a) Client 1



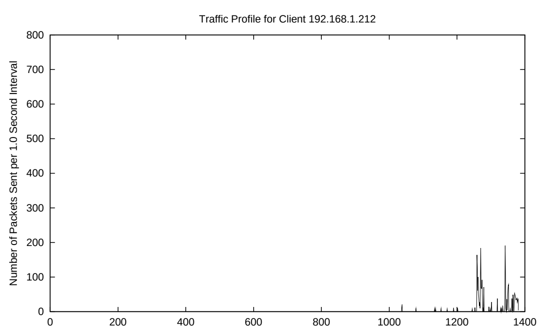
(b) Client 2



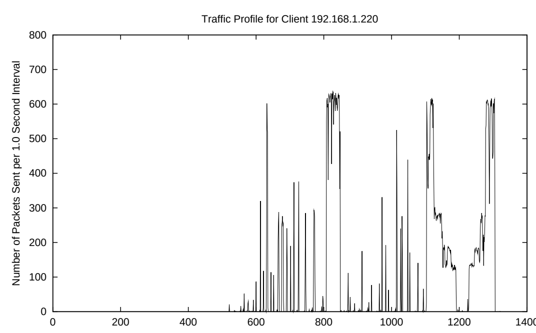
(c) Client 3



(d) Client 4



(e) Client 5



(f) Client 6

Figure 4: Per-Client Traffic Measurements from Wireless Classroom Area Network

Table 1: TCP Connection-Level Analysis of Wireless Classroom Measurements

Description of Connection Status	Number of Conns	Percent of Total Conns	Analysis Decision
TCP SYN and TCP FIN	55	21.0%	Used (normal)
TCP SYN but TCP RST	189	72.1%	Used (RST replaced by FIN)
TCP SYN but no SYN ACK	15	5.7%	Ignored
TCP SYN but no FIN or RST	3	1.2%	Ignored

Table 1 shows that only 21% of the TCP connections were "normal" TCP connections. A normal TCP connection starts with a three-way SYN (Synchronize) handshake to negotiate a connection, and ends with a three-way FIN (Finish) handshake to close a connection gracefully. However, the Microsoft Internet Explorer (IE) browser often uses a TCP RST (Reset) to terminate a connection [2], rather than the proper TCP FIN. Most of the students in our classroom used the IE browser, which explains the results that we observed. With a simple change to our TCP connection analysis technique, we were able to include the TCP RST connections in our analysis. There were 18 other TCP connections that had to be ignored: 15 of these were TCP SYN connection attempts that never received a response from the server (and thus no TCP connection was established), while 3 other connections were still in progress when the trace collection ended.

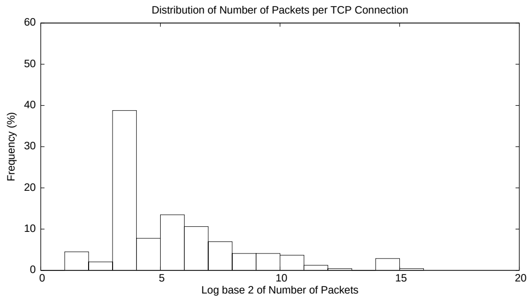
Figure 5 presents a graphical summary of TCP connection-level behaviours. Figure 5(a) shows the distribution of the number of packets sent on each of the 244 TCP connections studied. Figure 5(b) shows the number of data bytes exchanged on each TCP connection. Figure 5(c) shows the distribution of TCP connection duration in seconds, while Figure 5(d) shows the distribution of TCP connection throughput.

The results in Figure 5 indicate some of the TCP-level characteristics of Web browsing activities. About 56% of the connections lasted less than 10 seconds, with only a few lasting longer than 3 minutes. The number of bytes transferred per connection ranged up to 49 MB, indicating that some users were extremely active on the classroom network. However, 92% of the connections transferred less than 250 KB. Similarly, the number of packets per TCP connection ranged up to 50,000, but 92% of the connections exchanged fewer than 2000 packets. The single connection with the most bytes transferred contributed 19% of the total byte traffic in the network, while the other 99.5% of the connections contributed 81% of the byte load. This result suggests that Web workloads might be heavy-tailed [7].

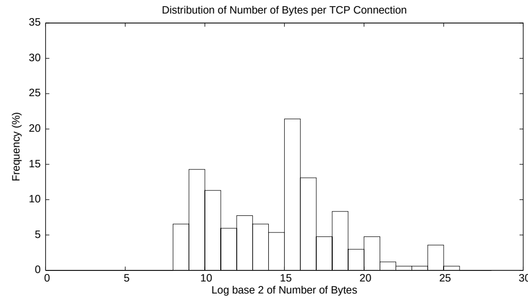
Figure 5(d) shows the distribution of TCP connection throughput. The throughput was calculated as the total number of data bytes transferred divided by the transfer time of each connection. The transfer time is defined as the connection duration minus the idle time (user think time) evident on each connection. Figure 5(d) shows that the throughput ranges from 0 to 4.8 Mbps. About half (55%) of the connections had relatively low throughput, less than 100 Kbps.

4.2 HTTP Analysis Results

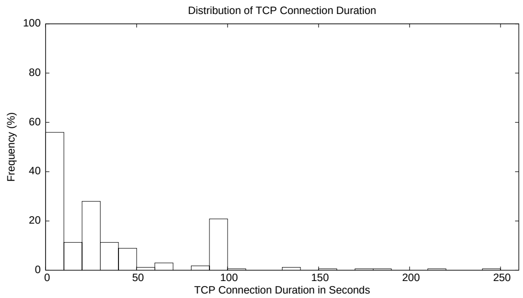
The relatively small number of TCP connections in our 23-minute data set is due in part to the use of persistent HTTP conections [10], which are widely supported by most Web browsers and Web servers today. A persistent connection allows a Web client to download multiple HTTP objects using a single TCP connection, rather than setting up a separate TCP connection for each object.



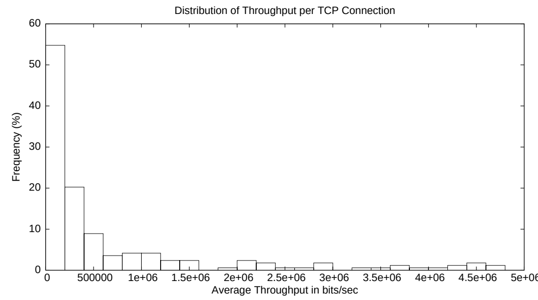
(a) Packets per TCP Connection



(b) Bytes per TCP Connection



(c) TCP Connection Duration



(d) TCP Connection Throughput

Figure 5: TCP Connection-Level Analysis

The remaining analyses in this paper study the HTTP-level behaviour for the TCP connections.

Figure 6 shows the distribution of the number of HTTP requests sent per TCP connection. For this analysis, every new data-carrying packet sent by the client on the same TCP connection was considered as a new HTTP request.

Figure 6 shows a mix of persistent and non-persistent HTTP connection behaviour, perhaps reflecting the Web browsers and/or the Web site content used in the experiment. Some TCP connections sent almost 100 HTTP requests during the trace. However, more than half (60%) of the TCP connections sent only 1 request. The significant tail to the distribution again suggests heavy-tailed or power-law behaviours by Web users.

Figure 7 provides a more detailed analysis of HTTP transfers. Figure 7(a) shows the distribution of the number of packets per HTTP transfer. Over half of the transfers require at most 2 TCP packets, and very few transfers require more than 30 packets. The largest HTTP transfer size observed is about 50,000 packets. Figure 7(b) shows the corresponding distribution for the number of bytes in each HTTP transfer. The vast majority of HTTP objects transferred are 2 KB or smaller, though transfers as large as 24 MB are observed. Figure 7(c) shows the distribution of HTTP transfer duration. This distribution is heavily skewed to the low end: more than 90% of the transfers complete in 10 seconds or less. Finally, Figure 7(d) shows the distribution of throughput observed per HTTP transfer. Comparing this distribution to that in Figure 5(d), we observe a significantly higher average throughput for HTTP transfers than for TCP connections. Many HTTP transfers achieve a throughput of 2-4 Mbps or more.

There are two main reasons for the higher throughputs observed for HTTP transfers. First, persistent connections are highly effective for Web transactions from a wireless Web server. Because the TCP connection handshaking is amortized over multiple HTTP transfers, the individual HTTP

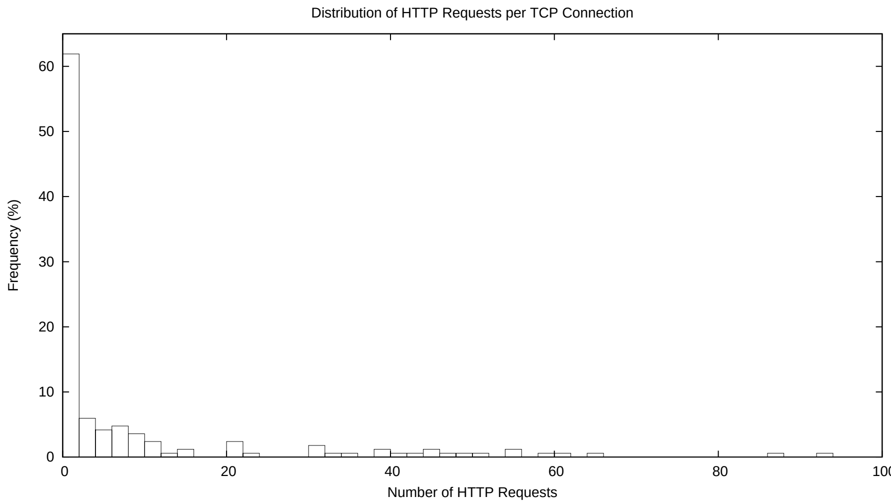
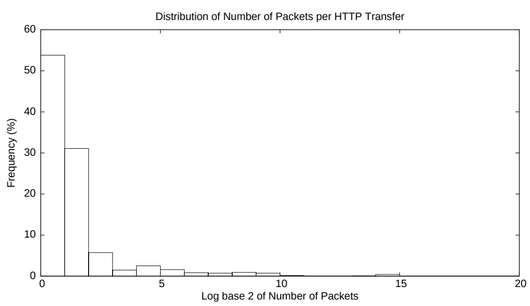
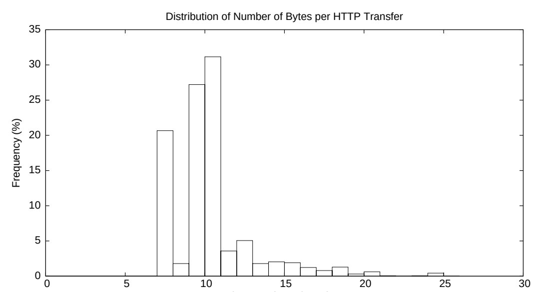


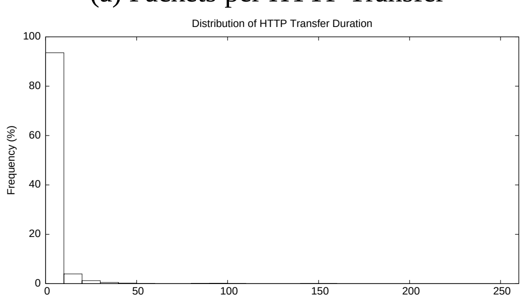
Figure 6: HTTP-Level Analysis of Persistent Connection Behaviour



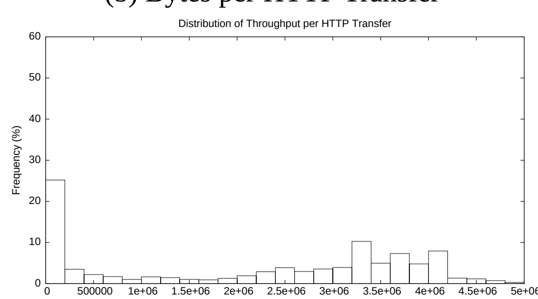
(a) Packets per HTTP Transfer



(b) Bytes per HTTP Transfer



(c) HTTP Transfer Duration



(d) HTTP Transfer Throughput

Figure 7: HTTP-Level Analysis

transfers are unencumbered by TCP control packet overhead. The HTTP transfers can also exploit the larger TCP flow control (cwnd) window sizes established by earlier transfers. Second, the persistent TCP connections themselves have a lot of idle time, due to user think time and the persistent connection timeout. Both of these effects can extend the TCP connection duration significantly. It is thus no surprise that the average TCP connection throughput is much lower than the throughput for individual HTTP transfers.

5 Summary and Conclusions

This paper studied the performance of a wireless Web server in a classroom area network. Measurements were conducted on an IEEE 802.11b wireless LAN, using a wireless-enabled Apache Web server, several wireless client laptops, and a wireless network analyzer. Application-layer and network-layer measurements are used to measure performance, focusing on HTTP and TCP protocol behaviour as a function of user behaviour, Web transfer sizes, and HTTP features.

The results indicate satisfactory performance for a wireless Web server in a classroom area network. These results are consistent with our prior benchmarking work [3, 9] showing that a wireless Web server can provide up to 110 transactions per second for 1 KB objects with non-persistent HTTP before the wireless LAN bottleneck becomes a problem. Throughputs in the classroom range from 1-5 Mbps depending on the sizes of the Web objects downloaded. The bandwidth is shared fairly amongst TCP connections, and user response time is acceptable. The use of persistent HTTP connections amortizes the overheads of TCP and the IEEE 802.11b MAC protocol, improving HTTP performance.

Our ongoing work is studying multimedia (audio and video) streaming performance for wireless media servers in classroom network environments. We believe that the portable networks paradigm offers a flexible and cost-effective solution for education and entertainment applications.

Acknowledgements

The wireless LAN traffic measurements presented in this paper were collected by Guangwei Bai and Kehinde Oladosu. Financial support for this work was provided by iCORE (Informatics Circle of Research Excellence) in the Province of Alberta.

References

[1] M. Arlitt and C. Williamson, "Understanding Web Server Configuration Issues" *Software: Practice and Experience*, Vol. 34, No. 2, pp. 163-186, February 2004.

[2] M. Arlitt and C. Williamson, "Analysis of TCP Reset Behaviour on the Internet", submitted for publication, November 2003.

[3] G. Bai, K. Oladosu, and C. Williamson, "Performance Issues for Wireless Web Servers", *Proceedings of the International Workshop on Mobile, Wireless, and Ad Hoc Networks* (MWAN), Las Vegas, NV, June 2004.

[4] G. Bai and C. Williamson, “Simulation Evaluation of Wireless Web Performance in an IEEE 802.11b Classroom Area Network”, *Proceedings of the Third International Workshop on Wireless Local Networks* (WLN 2003), Bonn, Germany, pp. 663-672, October 2003.

[5] A. Balachandran, G. Voelker, P. Bahl, and P. Rangan, “Characterizing User Behavior and Network Performance in a Public Wireless LAN”, *Proceedings of ACM SIGMETRICS*, Marina del Rey, CA, pp. 195-205, June 2002.

[6] R. Epsilon, J. Ke, and C. Williamson, “Analysis of ISP IP/ATM Network Traffic Measurements”, *ACM Performance Evaluation Review*, Vol. 27, No. 2, pp. 15-24, September 1999.

[7] F. Hernandez-Campos, K. Jeffay, and F. Donelson-Smith, “Tracking the Evolution of Web Traffic: 1995-2003”, *Proceedings of the 11th IEEE/ACM MASCOTS Conference*, Orlando, FL, October, 2003.

[8] D. Kotz and K. Essein, “Analysis of a Campus-Wide Wireless Network”, *Proceedings of ACM MOBICOM*, Atlanta, GA, September 2002.

[9] K. Oladosu, *Performance and Robustness Testing of Wireless Web Servers*, M.Sc. Thesis, Department of Computer Science, University of Calgary, August 2003.

[10] V. Padmanabhan and J. Mogul, “Improving HTTP Latency”, *Computer Networks and ISDN Systems*, Vol. 28, pp. 25-35, December 1995.

[11] RFC 1945: “Hypertext Transfer Protocol – HTTP/1.0”, http://www.ietf.org/rfc/rfc1945.txt

[12] RFC 2616: “Hypertext Transfer Protocol – HTTP/1.1”, http://www.ietf.org/rfc/rfc2616.txt

[13] W.R. Stevens, *TCP/IP Illustrated, Volume 1: The Protocols*, Addison-Wesley, 1994.

[14] D. Tang and M. Baker, “Analysis of a Metropolitan-Area Wireless Network”, *Proceedings of ACM MOBICOM*, Seattle, WA, pp. 13-23, August 1999.

[15] D. Tang and M. Baker, “Analysis of a Local-Area Wireless Network”, *Proceedings of ACM MOBICOM*, Boston, MA, pp. 1-10, August 2000.

Biographical Information

Dr. Carey Williamson is an iCORE Professor in the Department of Computer Science at the University of Calgary, specializing in *Broadband Wireless Networks, Protocols, Applications, and Performance*. His research interests include Internet protocols, wireless networks, network traffic measurement, workload characterization, network simulation, and Web server performance. He is an Adjunct Scientist at TRLabs Calgary.

Nuha Kamaluddeen is an M.Sc. candidate in the Department of Computer Science at the University of Calgary, supervised by Dr. Lisa Higham. Her research interests include distributed systems, wireless networks, and network reliability. Most of the network traffic analyses described in this paper were done as a course project in CPSC 641 (Performance Issues in High Speed Networks) at the University of Calgary.