A SURVEY OF NETWORK PERFORMANCE TOOLS FOR COMPUTER NETWORKING CLASSES

S. Zeadally E. Yaprak[1] Y. Li X. Che
Department of Computer Science
Wayne State University
Detroit, MI 48202

Abstract

Real traffic measurements and analysis over wide area networks is a challenge for many computer science and engineering educators. This is mainly because there are various networking tools available for measurements and testing over wide area networks and many of them have constraints, which limit their use to only a select group of users. As a result, the collection of wide area traffic data and their analyses becomes a difficult task, particularly in cases where the test site lacks collaboration support with other sites with which there is desire for traffic measurements to be used in undergraduate/graduate computer networking classes. In this work, we review a selection of some of the tools that can be used for wide area traffic measurements. Our survey highlights the features and characteristics of each tool as well as their limitations. We also demonstrate the use of one of the tools we used in some preliminary wide area traffic tests on bandwidth, packet loss, delay, and jitter, and present some initial results and observations. We hope this review will enable educators to make appropriate decisions on the selection of a networking tool that is best suited to meet their teaching goals in the area of computer networking.

INTRODUCTION

Computer networking is an important subject area in the computer science, engineering, and engineering technology curricula. This fast evolving field requires instructors to always be in touch and understand the latest technological advancements made in the field. Traditionally, computer networking has been taught primarily based on theoretical material. The reasons were because hardware or software were expensive and were unaffordable for institutions [11][12]. Over the last few years, however, we have witnessed significant progress in making this possible. The continued decreasing costs of computer hardware along with the availability of freeware programs readily downloadable from the Internet are now making it possible to provide *hands-on* experience to students that were otherwise difficult and costly to achieve many years ago. It is becoming increasingly important to incorporate hands-on training in the teaching of computer networking [1][2]. This has become a necessity because of the plethora of networking devices and technologies available today. In order to better understand how networks and their components work, practical training along with the teaching of theoretical concepts in this subject area is required. This will enable a better understanding of the computer networking hardware and software technologies.

The main contribution of this work is its applicability to improving hands-on experience in the computer networking curriculum. In many other universities and colleges, computer networking is often taught by the engineering, engineering technology, and the computer science departments and this is the case at our institution. One of the main goals of this work is to contribute to the learning objectives set out for Net-Centric Computing in the **C**omputing Curricula 2001 project [3]. As pointed out in the Computing Curricula 2001 project, students are likely to take a greater interest in the subject if they have hands-on experience with real data and real systems rather than some highly simplified and abstract simulations. This will also satisfy Accreditation Board of Engineering and Technology (ABET) Technology Criteria 2000 (TC2000) Program criteria for Computer Engineering Technology program outcomes. Students will be able to work with “live-data” and demonstrate hands-on competence on local and wide area networks in the building, testing, and operating computer systems and associated hardware systems.

[1] Division of Engineering Technology, College of Engineering.

The Internet is playing a central role in society today. It has undergone enormous growth in the last few years. However, we have little understanding of the complex interactions taking place in this global network [8]. Tracking the network performance, analysis of network maps over sufficiently long periods of time depicting the status of the networks will help students better understand how the Internet evolves. Wide area traffic testing, measurements, and analyses contribute to a better understanding of the behavior and dynamics of Wide Area Networks (WAN) such as the Internet. In particular, they give further insight into traffic patterns, bottleneck paths, network performance, network availability, and the occurrence of network faults. The use of networking tools will help simplify and enhance the understanding of important concepts while supplementing existing laboratory setups in colleges [6][9][10]. We plan to use network performance tools surveyed in this work in computer networking classes where the collection and analysis of Internet traffic data can be used in laboratory project assignments given to students. This kind of activity will involve data collection and synthesis, empirical modeling and evaluation of alternative design models. All of these are important concepts that can best be understood by experimentation with real networks and actual data. In this regard, the free availability of several tools discussed in this paper makes an important contribution to pedagogy in this area.

Various networking performance tools have been developed and implemented over the last few years that allow different types of tests to be performed on the Internet. Our main motivation behind this survey arose from our desire to find tools that will provide students with a better understanding of Internet traffic behavior. As part of our networking curricula, we were specifically interested in the capacities that are being delivered by the Internet2 backbone available today. This will enable the investigation of end-to-end application performance over the Internet using performance metrics such as throughput, latency, and packet loss. To achieve these goals, we selected several sites across the United States for our test measurements. Our choice of these sites was mainly determined by geographical locations, which cover different parts of the country. We quickly discovered that despite the crave for performance results over Internet2, the search for partners willing to invest time, efforts, and resources to make that happen was difficult. We initially experimented with the "ping" command for latency measurements such as round-trip times. However, we had to explore other tools for other measurements involving throughput, packet loss, and jitter. We ran into major difficulties at this stage because several tools (which we present later in the paper) have severe limitations on their use. The most severe limitation we faced was the need for execution of some programs at the other test site for the duration of the test. This was difficult to achieve in practice since many system administrators at the remote site were either reluctant to have test traffic entering their sites or they did not have the manpower to oversee the test (at different times of the day or week). As a result, it became difficult to perform these wide area test measurements. This led us to search for a network performance tool that does not have the constraints mentioned earlier. Consequently, we set out to perform a search of some of the tools available to achieve these objectives. We present in the next section the survey of WAN performance tools and the limitations on the usage of each of them when in use in networking curricula. This survey of tools will be useful to network educators in computer networking, as it will help them choose the tool most suitable to meet their teaching goals as well as being aware of the constraints and limitations associated with each tool.

SURVEY OF NETWORK PERFORMANCE TOOLS

In this section, we focus on WAN performance tools that have been developed over the last few years. The table below is not a complete collection of all the tools available. However, we have made every attempt to include most of the well-known tools.

2002	aslookup, AlertCenter, Alertra, AlertSite, Analyse-it, BestFit, Chariot, CommView, elkMonitor, Etherpeek, Fidelia, Finisar, Fpinger, GDChart, HipLinkXS, ipMonitor, LANExplorer, MGEN, Netarx, NetDetector, NetGeo, NEPM, NetReality, PageREnterprise, PastMon, Pathprobe, remstats, RIPmon, RFT, RUDE, Silverback, SmokePing, Snuffle, Telchemy, UDPmon, WebAttack,
2001	AdventNet SNMP API, Alchemy Network Monitor, Anasil analyzer, Argent, Autobuf, Bing, Clink, Cycletraders, DSLReports, Firehose, GeoBoy, PacketBoy, Internet Control Portal, Internet Periscope, ISDNwatch, Metrica/NPR, Mon, NetPredict, NetSaint, NetTest, Nettimer, Net-One-1,

	Pathrate, RouteView, sFlow, Shunra, Third Watch, Traceping, Trellian, TowerView, WCAT, What's Up Gold, WS_FTP, Zinger,
2000	Analyzer, bbftp, Big Brother, Bronc, Cricket, EdgeScape, Ethereal, gen send/gen recv, GSIFTP, Gtrace, Holistix, InMon, NcFTP, Natas, NetAlly, Network Simulator, Ntop, PingGraph, PingPlotter, Pipechar, RRD, Sniffer, Snoop, StatScope, Synack, View2000, VisualPulse, WinPcap, WU-FTPD, WWW performance monitoring, Xplot
1999	Cheops, Ganymede, hping2, Iperf, JetMon, MeasureNet, MatLab, MTR, NeoTrace, Netflow, NetLogger, Network health, NextPoint, Nmap, Pchar, Qcheck, Quallaby, SAA, SafeTP, Sniffit, SNMP from UCSD, Sting, ResponseNetworks, Tcpshow, Tcptrace WinTDS,
1998	NetOps, Triticom, Maple, PV-Wave, S-Plus, VisualRoute
1997	INS Net Perf Mgmt survey, tcpspray, Mapnet, Keynote, prtraceroute clflowd flstats, fping, tcpdpriv, NetMedic Pathchar, CAIDA Measurement Tool Taxonomy, bprobe & cprobe
1996	mrtg, NetNow, NetraMet, Network Probe Daemon, SNIF, InterMapper, Lachesis, Optimal Networks, Digex

Table 1. Network Performance Tools developed between 1996 and 2002.

The tools listed in Table 1 have been developed by individuals, commercial companies, non-profit organizations, and government organizations. Some of them are free, open source, while some of them are not. Different tools are utilized for collecting different sets of performance metrics. Due to space limitations, we do not cover every tool listed in the Table 1. Instead, we will focus on those tools, which allow experimentation on end-to-end performance measurements to be used in networking laboratory setup. We organize the taxonomy of tools based on criteria which we believe will best help decide whether to adopt such a tool or not. For each of the selected tool, we identify the following criteria: the mode of operation (such as path characterization, end-to-end throughput), operating system platform supported, licensing issues (whether free and open source), the performance metrics that are measured, the testing mechanism (active or passive; active testing makes network administrators more reluctant to conduct such tests), the test mode used (client/server, one-way-metric, two-way-metric, whether test can be run without human intervention, whether we need remote peer's cooperation - this characteristic becomes essential in cases where it is difficult to find a partner willing to participate in the tests), ease of use (such as the availability of a Graphical User Interface (GUI)), potential benefit(s) and limitation(s). The table below describes some of the performance tools we have selected in our survey.

NAME	TYPE	OS	LICENSE	PERFORMANCE METRICS	GUI	TEST MODE	LIMITATION	BENEFIT
Iperf *dast.nlanr.net/Projects/Iperf/*	End-to-end bandwidth /throughput	All major OS	Free Open Source	Bandwidth, delay jitter, packet loss	Yes	Active, TCP/UDP	C/S mode, 2 ends operation	
Pchar *www.employees.org/~bmah/Software/pchar*	Path Characterization	Major Unix and Linux	Free Open Source	Bandwidth, Throughput, Latency, Packet loss	No	Active and passive, UDP; ICMP		No need for remote attendance
Chariot *www.netiq.com/products/chr/default.asp*	Application performance	Windows	Commercial	Throughput, Jitter, Delay, Packet loss	Yes	Active, TCP/UDP		VoIP performance evaluation, IPv6 support
Mapnet *www.caida.org/Tools/Mapnet/Backbones*	Mapping	Java	Free	Infrastructure of multiple backbone providers	Yes			

NAME	TYPE	OS	LICENSE	PERFORMANCE METRICS	GUI	TEST MODE	LIMITATION	BENEFIT
Traceping *Slacvx.slac.Stanford.edu:8097/www/traceping_description.html*	One-way availability/ Latency tests	VAX/ VMS	Free	Packet loss	No	Active, ICMP		
SYNACK *www-iepm.slac.Stanford.edu/tools/synack*	Path characterization	Solaris, Linux	Free, Open source	Latency	No	Active, SYN request and SYN/ACK	C/S mode, 2 ends operation	
Sting *www.cs.washington.edu/homes/savage/sting*	Path characterization	FreeBSD, Linux	Free, Open source	Packet loss	No	Active,TCP		No need for remote attendance
Smoke-Ping *people.ee.ethz.ch/~oetiker/webtools/smokeping/*	Path characterization	All Unix	Free, Open source	Latency and packet loss	Yes	Active, ICMP		Long term database
UDPmon and TCP Multi-stream *www.hep.man.ac.uk/~rich/net*	Path characterization	All Unix	Free, Open source	Latency and Bandwidth	No	Active, TCP/UDP		Throughput/ Bandwidth between two nodes in a route.
Pathchar *ftp.ee.lbl.gov/pathchar*	Path characterization	FreeBSD, Solaris, Linux	Free, Open source	Bandwidth, Throughput, Latency, Packet loss	Yes	Active/ Passive, UDP; ICMP		No need for remote attendance
Clink *Rocky.Wellesley.edu/downey/clink*	Path characterization	All Unix	Free, Open Source	Bandwidth, Throughput, Latency, Packet loss	No	Active/ Passive, UDP; ICMP		
Nettest *www-itg.lbl.gov/nettest/*	Path characterization	FreeBSD, Linux, Unix, MAC OS	Free, Open source	Latency, Burst Size, UDP throughput, TCP throughput	No	Active, UDP, TCP, ICMP	Needs to run at both ends	Capable of fully using the high-speed networks, e.g., saturating 1Gbps local network from a single x86 platform
Nettimer *Mosquitonet.stanford.edu/~laik/projects/nettimer/*	End-to-end bandwidth/ throughput	All Unix	Free, Open source	Bottleneck link Bandwidth	No	Active/ Passive, UDP/TCP/ ICMP	libdpcap and libkl component	
Qcheck *www.qcheck.net*	End-to-end bandwidth/ throughput	Windows	Free	Latency, Throughput	Yes	Active, UDP, TCP, ICMP	For throughput, maximum one megabyte	
NetGeo *www.caida.org/tools/utilities/netgeo*	Mapping	Web	Free, but has some restriction	Map IP addresses, domain names to geographical locations	Yes	Whois		A database and collection of Perl scripts, could be integrated with other tools.

NAME	TYPE	OS	LICENSE	PERFORMANCE METRICS	GUI	TEST MODE	LIMITATION	BENEFIT
RUDE & CRUDE *www.atm.tut.fi/rude*	Traffic generator	Linux, Unix	Free, Open source		No	Active, UDP		UDP traffic generator
Auto-tuned NcFTP *www.ncftp.com*	FTP	Unix	Free,Open Source	TCP Goodput	No	Active, TCP	Kernel must support TCP Large windows (beyond 64KB)	Auto tuning ability
BBFTP *Doc.in2p3.fr/bbftp*	FTP	All major Unix	Free, Open source	TCP Goodput	No	Active TCP	Compression takes large system resources. Kernel must support large TCP window, not suitable for LAN and small file transfer.	Good for transfer of very large file (e.g 2GB) across WAN
Bing *Spengler.econ.duke.edu/~ferizs/bing.txt*	Path characterization	Linux, Unix, OSF/1	Free, Open source	Bandwidth	No	Active ICMP	Not good for High speed link too far away, or Frame Relay, ATM,Satellite, X.25	
BPROBE and CPROBE *cs-people.bu.edu/carter/tools/Tools.html*	path characterization	SGI IRIX	Free, Open source	Available Bandwidth, Bottleneck Bandwidth	No	Active, ICMP	Restrict to SGI/IRIX	
MRTG *www.ee.ethz.ch/~oetiker/webtools/mrtg/mrtg.html*	Traffic monitor	Windows NT, Unix	Free, Open source	Traffic load	Yes	SNMP		Visual representations of the traffic
Pathprobe *www.psc.edu/~web100/pathprobe*	Path Characterization	Linux/ Unix	Free, Open source	Latency, MSS, Bandwidth/Throughput, Burst Size, Max Window	No	Active, TCP/UDP	Web100 kernel needed	Works hop-by-hop tests to determine if the paths along the way are capable of supporting the desired end-to-end target bandwidth between the sender and receiver.
Fping *www.fping.com*	One-way availability/ Latency tests	Linux/ Unix	Free,Open Source	Latency	No	Active, ICMP		Optimized to ping a large number of hosts in parallel
PasTmon *www.pastmon.org*	Application performance	Linux/ Unix	Free,Open Source	Response time, Segment size, TCP window size	Yes	Passive, packet capture,	libpcap required	
gen_send, gen_recv *www.citi.umich.edu/projects/qbone/generator.html*	Traffic generator	Linux/ Unix	Free, Open source	Bandwidth, Packet loss	No	Active, UDP		Good for testing the quality of service functionality

NAME	TYPE	OS	LICENSE	PERFORMANCE METRICS	GUI	TEST MODE	LIMITATION	BENEFIT
Gtrace *www.caida.org/tools/visualization/gtrace*	Forward Path Probe	Solaris/ Linux/ FreeBSD	Free, Open source	Node name, IP location, Latency	Yes	Active, UDP, ICMP		Graphical front-end to traceroute
MGEN *Manimac.itd.nrl.navy.mil/MGEN*	Traffic generator	Linux/ Unix	Free, Open source	Packet Loss, Delay, Delay jitter	Yes	Active, UDP		Support RSVP, Multicast
MTR *www.bitwizard.nl/mtr*	Path Characterization	Linux/ Unix	Free, Open Source	Packet loss, Delay, Delay jitter	Yes	Active, ICMP		

Table 2. Selected network performance tools classified by type, operating system platform supported, licensing requirements, performance metrics measured, test mode, benefit(s) and their limitation(s). The associated web link with each tool is also given.

MEASUREMENT PROCEDURES

Internet2 is a consortium being led by 194 universities (as of April 2002) working in partnership with industry and government to develop and deploy advanced network applications and technologies to accelerate the creation of tomorrow's Internet. The primary goals of Internet2 are to create a leading edge network capability for the national research and education community and to enable revolutionary Internet applications while ensuring the rapid transfer of network services and applications to the broader Internet community. Universities are the main players of Internet2. Instructors and students at Internet2 universities are able to explore important networking concepts and capabilities beyond today's Internet as they use these tools.

All the measurements tests given in this section were performed in the High-Speed Networking laboratory at Wayne State University (which in April 1999 became one of the earliest members of Internet2) on an UltraSparc 10 workstation connected by a 100 Mbits/s link to an Ethernet switch in the Department of Computer Science. The University connects to Internet2 by the Merit Network, Inc. with an OC3 (155 Mbits/s) link.

Although this paper does not provide extensive measurements and data traffic analyses for wide area networks, we nevertheless believe it would be useful to illustrate some performance results collected from preliminary tests we have conducted so far using one of the tools in our survey. We have selected ***pchar*** [7] as the network performance tool for our wide area traffic measurements. The reasons for this choice stem primarily from its free and open source, as well as the passive mode support which does not require any remote program in execution while a test measurement is in progress (a requirement we found hard to achieve in the real world as explained earlier since very few administrators were willing to let test traffic into their sites during the measurements). In addition, ***pchar*** also provides a good set of performance metrics (such as throughput, round-trip latency, packet loss) we find interesting to investigate that will provide insight into the capabilities of the current Internet infrastructure and its link reliability.

For our tests, we selected eight different Internet2 sites covering the different parts of the United States: Caltech, Colorado State University (ColoState), Florida State University (FSU), Harvard, Ohio State University, University of Pennsylvania (Upenn), University of California at San Diego (UCSD), University of Michigan (UoM), and University of Oregon (Uoregon).

PRELIMINARY TEST RESULTS

Table 3 gives the RTT results to the sites chosen for measurements taken over a week period. We observe that RTTs to the West Coast are about twice those obtained for the South or East coasts. It is also worth noting from the results in Table 3 that the percentage of time during the one-week period that the same routing path is chosen tends to be high for the Midwest and the East coast. For the West coast and Southern sites tested, there are more variations in the frequency of paths taken to the destination.

Network Path Performance from Wayne State University (Detroit) (Data collection period: 5/12/2002 - 5/18/2002)						
Destination	**RTT (milliseconds)**				**Routing**	
	Minimum	Mean	Maximum	STD	Primary Route	Path Changes
UoM	3.00	7.64	2448.00	33.15	100.00%	1
Ohio State	22.00	24.99	45.00	1.80	98.73%	5
Upenn	38.00	40.26	121.00	2.02	97.89%	8
Harvard	35.00	36.92	212.00	2.64	98.53%	4
ColoState	39.00	47.25	342.00	11.48	94.94%	5
FSU	33.00	40.68	2059.00	31.76	94.56%	9
Uoregon	68.00	78.18	287.00	7.90	94.64%	8
Caltech	72.00	79.16	163.00	5.78	94.85%	7
UCSD	73.00	79.67	285.00	6.26	95.06%	4

Table 3. RTT and Path changes to different Internet2 sites.

We conducted bandwidth and packet loss measurements using ***pchar*** for the above sites. However, due to space limitations, we present only a subset of the results obtained in the Table 4. It is worthwhile noting that the bandwidth results give the *available bandwidth* of each link between hops. The available bandwidth parameter is a crucial parameter in capacity provisioning, routing and traffic engineering, QoS management, streaming applications, server selection, and in several other areas. Each link can transmit data at a rate called the *link capacity* [5]. Two metrics associated with a network path are the end-to-end *capacity* and the *available bandwidth* [4].

We report the bandwidth capacity available at each hop to gain insight into the various speeds at which Internet2 is running and to explore the availability of high-capacity links available among Internet2 sites. This enables us to better understand how much performance traffic typical receives on the current Internet2 infrastructure. We make the following observations based on the results obtained from Table 4: the traffic data used in our tests was using Internet2 infrastructure as demonstrated by the paths through Abilene and Calren2 and others. We obtained available bandwidth ranging from 20 Mbits/s to 480 Mbits/s (on Calren2 – the California research network). Frequently, on some links we experience available bandwidth of several hundreds Mbits/s which demonstrates that the capacity for high-speed Internet 2 access is in place.

Another point to note is that the last mile is always the bottleneck. This clearly demonstrates that although very high-speed Internet2 connectivity has been brought to University campuses across the country, the last mile is still not running at high-speed (most probably running at typical Ethernet speeds and the connections on campus also probably compete with commodity Internet traffic). This is why the last mile is only around 20 Mbits/s. To reap the full end-to-end speed of Internet2, the last mile has to be upgraded to high-speed links to deliver high end-to-end throughput to end-users. We also note from the results that the packet loss was quite high (almost close to 1%). Further investigation revealed that Solaris (the operating used on our end) implement rate control on some Internet Control Message Protocol (ICMP) packets they generate, typically limiting the rate to at most two packets per second to destination for security reasons. By increasing the gap between packets to slightly over 0.5 seconds (using the –g option) makes ***pchar*** slightly slower for intermediate hops. We confirmed this by repeating one test to the Caltech site and we obtained a much lower packet loss. The Solaris operating system was responsible rather than the ***pchar*** software itself. We recommend that future packet loss tests on Solaris platforms to use the –g option.

Bandwidth and Packet Loss from Wayne State (szsun1.cs.wayne.edu) (Detroit, Michigan)

Destination	**Hop**	**Bandwidth (Mbits/s) (average of several tests)**	**Packet Loss (average of several tests)**

Caltech	0	szsun1.cs.wayne.edu	17.891	1.49%
	1	cs-gw17.cs.wayne.edu	68.722	0.79%
	2	router-05net.cc.wayne.edu	20.163	1.11%
	3	gw-wsu.cc.wayne.edu	36.582	1.34%
	4	at-0-3-0x6.aa1.mich.net	87.810	0.97%
	5	abilene-clev.mich.net	76.891	1.25%
	6	ipls-clev.abilene.ucaid.edu	84.446	0.95%
	7	kscy-ipls.abilene.ucaid.edu	32.838	1.06%
	8	dnvr-kscy.abilene.ucaid.edu	--	1.25%
	9	snva-dnvr.abilene.ucaid.edu	35.238	0.93%
	10	losa-snva.abilene.ucaid.edu	62.497	0.97%
	11	USC--abilene.ATM.calren2.net	26.905	0.97%
	12	QAnh--USC.POS.calren2.net	36.073	1.00%
	13	UCI--QANH.POS.calren2.net	23.908	1.04%
	14	UCR--UCI.POS.calren2.net	--	1.04%
	15	CIT--UCR.POS.calren2.net	232.550	0.84%
	16	Caltech-CalREN.caltech.edu	17.483	0.88%
	17	SteeleB-RSM.ilan.caltech.edu	23.070	0.84%
	18	amp-monitor.caltech.edu	17.483	
University of Michigan	0	szsun1.cs.wayne.edu	16.478	0.82%
	1	cs-gw17.cs.wayne.edu	88.505	0.82%
	2	router-05net.cc.wayne.edu	15.625	0.66%
	3	gw-wsu.cc.wayne.edu	--	0.50%
	4	198.108.23.5	61.135	0.52%
	5	a-arbl-merit1.c-arbl.umnet.umich.edu	10.113	0.45%
	6	NLAR-AMP.umnet.umich.edu	10.113	
University of Oregon	0	szsun1.cs.wayne.edu	18.916	1.68%
	1	cs-gw17.cs.wayne.edu	204.207	1.09%
	2	router-05net.cc.wayne.edu	20.111	1.31%
	3	gw-wsu.cc.wayne.edu	225.490	1.29%
	4	at-0-3-0x6.aa1.mich.net	113.006	1.06%
	5	abilene-clev.mich.net	74.123	1.49%
	6	ipls-clev.abilene.ucaid.edu	--	1.02%
	7	kscy-ipls.abilene.ucaid.edu	66.338	1.22%
	8	dnvr-kscy.abilene.ucaid.edu	39.648	1.02%
	9	pos-6-3.core0.eug.Oregon-gigapop.net	69.585	1.25%
	10	ge-0-0-0.car0.eug.Oregon-gigapop.net	229.744	1.11%
	11	nlanr-amp.Oregon-gigapop.net.163.32.198.in-addr.arpa	18.916	

Table 4. Available link capacity and packet loss from Wayne State University (Detroit) to remote Internet2 sites.

It is worthwhile stressing that the thrust of this work is not about performance measurements and analysis of wide area Internet2 traffic. Such work is beyond the scope of this paper. In this work, the main goal was to present a survey of tools that educators can easily use to decide which tool will best suit their particular needs and prevent them from going through the same experience and difficulties we went through during the early stages of finding an appropriate performance tool. This avoids duplicating work by others searching for WAN performance tools to perform similar research or educational activities.

CONCLUSIONS

In this paper, we present a taxonomy of network performance tools for wide area traffic measurements and data analyses. These tools allow students to experiment with "live network data". Exploiting freeware networking tools will be affordable for institutions and will allow additional laboratory setups involving large-scale networks. This will ultimately allow students to better understand the behavior of Internet traffic, topology, reliability, and other characteristics. In addition, using one of the tools, we have presented initial results based on some preliminary measurements we have obtained during our tests.

We hope that this survey of tools will be useful to educators interested in supplementing existing laboratory setups and generating laboratory teaching assignments while satisfying ABET 2000 Computer Engineering Technology Program criteria and Computing Curricula 2001 Project.

REFERENCES

[1] M. Beers, "Infusing Technology in Teacher Education: Incorporating the Bigger Picture", in Proc. of the Sixth Annual SUNY PACT Conference on Instructional Technology, (CIT 97), 1997.

[2] D. Hoang, "A Lab-based Computer Networking Course at the University of Sydney", in Proc. of ACM SIGCSE Australian Conference on Computer Science Education, July 1996.

[3] IEEE-CS and ACM: Computing Curricula 2001 project. http://computer.org/education/cc2001/final

[4] R. Jain and C. Dovrolis, "End-to-end Available Bandwidth: Measurement, Methodology, Dynamics, and Relation with TCP Throughput", in Proc. of ACM SIGCOMM 2002, PA, August 2002.

[5] K. Lai and M. Baker, "Measuring Link Bandwidths Using a Deterministic Model of Packet Delay", in Proc. of ACM SIGCOMM 2000, August 2000.

[6] L. Lankewicz, "Resources for teaching computer networks", in Proc. of 29th ACM SIGCSE Symposium on Computer Science Education, GA, 1998.

[7] B. A. Mah, "Pchar: A Tool for Measuring Internet Path Characteristics", http://www.employees.org/~bmah/Software/pchar/, February 1999.

[8] W. Matthews and L. Cottrell, "The PingER Project: Active Internet Performance Monitoring for the HENP Community", IEEE Communications Magazine on Network Traffic Measurements and Experiments, May 2000.

[9] D. Nelson and Y. Ng, "Teaching Computer Networking Using Open Source Software", in Proc. of 5th Annual SIGCSE/SIGCUE ITiCSE Conference on Innovation and Technology in Computer Science Education, Helsinki, Finland, 2000.

[10] J. Rickman, "Enhancing the Computer Networking Curriculum", in Proc. of 6th Annual SIGCSE/SIGCUE ITiCSE Conference on Innovation and Technology in Computer Science Education, Canterbury, UK, June 2001.

[11] P. Tymann, "VNET: a tool for teaching computer networking to undergraduates", ACM SIGCSE Bulletin, Vol. 23, No. 1, pages 21-24, March 1991.

[12] C. McDonald, "Teaching Computer Networking using Simulation", ACM SIGCSE Bulletin, Vol. 27, No. 1, 1995.