Measurement of Packet Processing Time of an Internet Host using Asynchronous Packet Capture at the Data-Link Layer

Khondaker M. Salehin and Roberto Rojas-Cessa
Networking Research Laboratory, ECE Department
New Jersey Institute of Technology, Newark, NJ 07102-1982
Email: {kms29, rojas}@njit.edu

***Abstract*—As transmission speeds increase faster than processing speeds, the packet processing time (PPT) of a host (i.e., workstation) is becoming more significant in the measurement of different network parameters, in which packet processing by the host is involved. The PPT of a host is the time elapsed between the arrival of a packet at the data-link layer and the time the packet is processed at the application layer of the TCP/IP protocol stack (RFCs 2679 and 2681). In this paper, we propose a methodology to measure the PPT of a host using Internet Control Message Protocol (ICMP) packet and a specialized packet-capture card that does not require synchronization between the host under test and the packet-capture card. We tested the proposed methodology on two hosts with different specifications. The experimental results show that the proposed methodology consistently measures PPT.**

***Index Terms*—Active measurement, packet processing time, clock synchronization, local area network.**

I. Introduction

Packet processing time (PPT) of a host (i.e., workstation) is the time elapsed between the arrival of a packet in the host's input queue of the Network Interface Card, NIC, (i.e., the data-link layer of the TCP/IP protocol stack) and the time the packet is processed at the application layer [1], [2]. As link rates increase faster than processing speeds [3]-[5], the role of PPT becomes more important in the measurement of different network parameters.

One-way delay (OWD) in a local area network (LAN) is an example of a parameter that PPT can significantly impact [6]. Figure 1 illustrates the OWD of a packet P over an end-to-end path, between two end hosts, the source (src) and the destination (dst) hosts. The figure shows the different layers of the TCP/IP protocol stack that P traverses at both end hosts, as defined in RFC 2679 [1]. The transmission time (t_t) and propagation time (t_p) of P take place at the physical layer, the queuing delay (t_q) takes place at the network layer, and the time stamping of the packet creation at src (PPT_{src}) and packet receiving at dst (PPT_{dst}) take place at the application layer of the end hosts. The actual OWD experienced by P from src to dst is:

$$OWD = PPT_{src} + t_t + t_q + t_p + PPT_{dst}$$

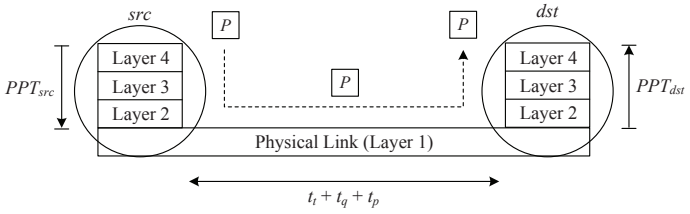


Fig. 1. End-to-end one-way delay (OWD) of packet P between two directly connected hosts.

However, because of the low transmission rates of legacy systems, PPT has been considered so far negligible (i.e., $PPT_{src} = PPT_{dst} \simeq 0$). As data rates increase, the contribution of PPT increases.

The error in the measurement of OWD in high speed LANs can be large if PPTs are neglected. For example, the measurement of OWD of 1500- and 40-byte packets between the end hosts with $PPT_{src} = PPT_{dst} = 2$ μs and an average level of queuing delay, $t_q = 40$ μs [7], on a 100-Mb/s link would have an error of 2.5 and 8.5%, respectively. In these calculations, error = $|\frac{(OWD-OWD')}{OWD}| \times 100$ %, $OWD = OWD' + PPT_{src} + PPT_{dst}$, $OWD' = t_t + t_q + t_p$, and $t_p = 0.5$ μs, considering the maximum transmission length (100 m) of a Fast-Ethernet cable [8]. This error increases up to 52% when queuing delay is relieved ($t_q \simeq 0$ μs [7], [9]-[11]) for a 40-byte packet, which constitutes 50% or more of the IP traffic [12], [13]. In a similar scenario, the error of OWD on a 1-Gb/s link can be up to 14% (as $t_p = 25$ μs for a 5-km optical cable in Gigabit Ethernet [8]). Therefore, PPT must be considered for an accurate measurement of OWD in LAN.

Similarly, knowledge of the PPT of servers used in financial-trading datacenters can increase customer confidence as OWD is estimated with high accuracy [14], [15].

In a wide area network (WAN), high-resolution OWD measurement can be used to increase accuracy in IP geolocation [16]-[18]. In IP geolocation, each microsecond of propagation time varies the estimated geographic distance by 300 m between two end hosts connected over optical links. PPT is also an important parameter in the measurement of link capacity and available bandwidth on high speed networks [19], [20]. For example, in the measurement schemes based on packet-pair structure [21]-[26], 1 μs of PPT can incur 8% error on a 1-Gb/s link if 1500-byte packets are used. This error increases as the packet length decreases.

The measurement of the PPT of a host can be complex because the host must record the time a packet arrives at the data-link layer and the time the application layer processes the packet. However, time stamping at the data-link layer is not readily available in popular and deployed NICs. PPT measurement can be performed by placing a specialized packet-capture card in the same subnet where the host under test is located. Furthermore, existing packet-capture cards have a time stamping resolution in the nanosecond range [27], [28], and their use require time synchronization [29] with the host's clock. This is difficult to accomplish as operating systems of a host can provide up to microsecond resolution. In this paper, we present a methodology to measure the PPT of a host using a specialized packet-capture card in the same subnet. The proposed approach does not require clock synchronization between the host under test and the packet-capture card. We present an experimental evaluation of the proposed methodology, and the outcomes show consistent and measurable results.

The remainder of the paper is organized as follows: Section II discusses the basic architecture of a host and its NIC operations. Section III introduces the proposed methodology to measure PPT. Section IV shows the experimental results of PPTs measured on two different hosts. Section V discusses related work. Section VI concludes our discussion.

II. Determining Factors of PPT in Hosts

In this paper, we consider the time stamping of a packet transmission as the packet processing event. This latency is determined by the properties of the central processing unit (CPU), bus speeds, NIC driver, and system-call latencies of the operating system of the host [4], [30], [31]. Therefore, the architecture of a host and its NIC operations for sending and receiving a packet have a major impact towards PPT. In the following discussion, we present the basic architecture of a host and the operations of a NIC.

A. Host Architecture

A host architecture has basically one or more CPUs, where each can have one or multiple processing cores, a chipset to operate in conjunction with the CPUs, main memory (blocks), and NICs. These different subsystems are interconnected by means of buses, a front-shared bus to connect the CPU and the chipset, a memory bus to connect the main memory and the chipset, and a Peripheral Component Interconnect (PCI) bus to interconnect the chipset with the NICs. Figure 2 shows this simplified architecture.

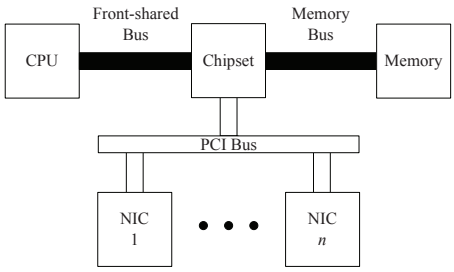


Fig. 2. Top-level architecture of a general-purpose host.

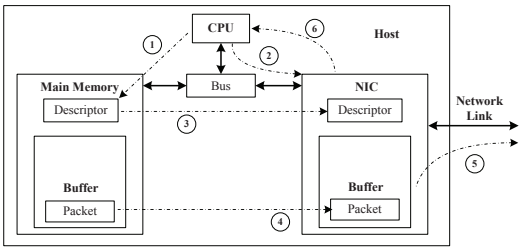
B. NIC Operations

Figure 3 illustrates the two basic operations of the NIC: packet transmission and packet receiving processes [30], [31]. Figure 3(a) shows that for transmitting a packet, the host initially creates buffer descriptors in the main memory containing the location, both address and length, of the packet (Step 1) and informs about this event to the NIC (Step 2). The NIC then copies the packet to its local buffer through two Direct Memory Access (DMA) transfers, one for the packet descriptors and the other for the packet itself (Steps 3 and 4). The NIC sends the packet out to the network (Step 5) and finishes the transmission process by interrupting the CPU (Step 6). According to Figure 3(b), when a packet arrives in the NIC buffer from the network (Step 1), the NIC initiates the receiving process by copying the packet into a pre-allocated buffer at the main memory along with the packet's descriptors through two DMA transfers (Steps 2 and 3). The NIC finishes the receiving process by sending an interrupt to the CPU (Step 4).

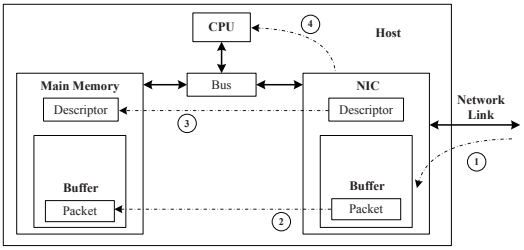
III. Proposed PPT Measurement Method

Figure 4 shows the proposed method and experimental setup to measure PPT of dst. The setup consists of a source host (src) directly connected to a destination host (dst) through an Ethernet link. A sniffer (*hsf*), a workstation equipped with a two-port packet-capture card (Endace DAG 7.5G2 card [27]), captures the packets transmitted between src and dst by connecting its two ports to the Ethernet link using a custom-built wire tap. In this experimental setup, the propagation times of the sniffed packets are considered negligible because the distance between src and dst is 2 m.

To measure the PPT, src sends an ICMP echo request packet (Q) to trigger an ICMP echo reply packet (R) at dst. *hsf* captures the exchanged ICMP echo packets and time stamps them at the data-link layer. dst also time stamps the exchanged packets, however, at the application layer.



(a) Transmission process



(b) Receiving process

Fig. 3. Basic NIC operations: (a) transmission process and (b) receiving process.

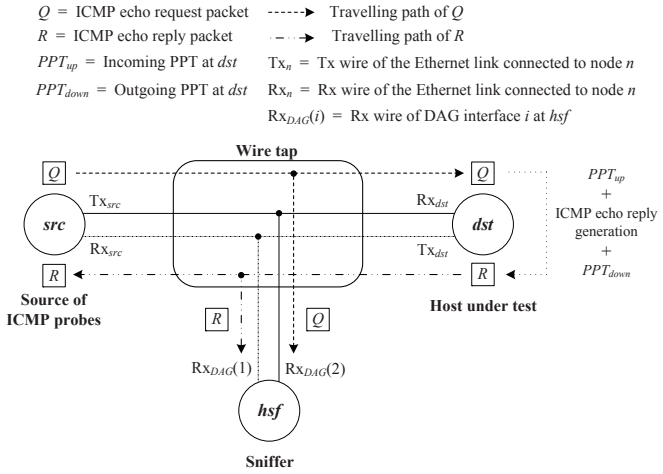


Fig. 4. Experimental setup to measure PPT of dst.

Figure 5 presents the time line of the events that take place between the exchange of ICMP echo packets Q and R at dst. Here, the gap measured by hsf, $t_7 - t_1$, includes the receiving time of Q, $t_2 - t_1$, the PPT experienced by Q on its travel up through the TCP/IP protocol stack, $t_3 - t_2$ or PPT_{up}, the time taken by dst to generate R, $t_5 - t_3$, and the PPT experienced by R on its travel down through the TCP/IP protocol stack, $t_7 - t_5$ or PPT_{down}, at dst. The gap measured by dst at the application layer, $t_5 - t_3$, includes the actual time needed to generate R, $t_4 - t_3$, plus the system-call latency of the operating system for time stamping R, $t_5 - t_4$. The gaps $t_2 - t_1$ and $t_5 - t_3$ are subtracted from $t_7 - t_1$, which is equal to $2PPT$ at dst if $PPT_{up} = PPT_{down}$. The assumption of equal PPTs for both the incoming Q and outgoing R at dst may not always be the case, but we consider a scenario where there is no other traffic passing through the host. Moreover, the travel path of a packet between the data-link and application layers is the same, as discussed in Section II-B.

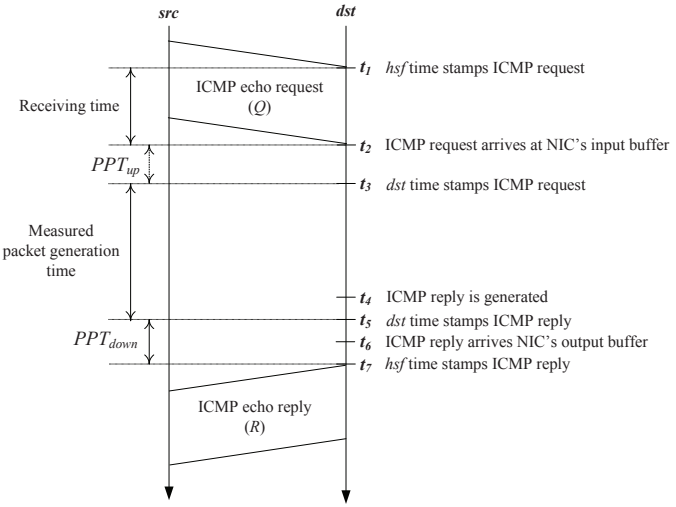


Fig. 5. Timeline of ICMP echo request and echo reply packets at dst.

IV. Experimental Results

We measured PPT of two hosts, a Dell Dimension 3000 (D3000) and a Dell Inspiron I531S (I531S) workstations, to evaluate the proposed methodology. The specifications of the workstations are shown in Table I. We performed PPT measurements on these two workstations with their NICs running at 10 and 100 Mb/s to investigate the effect of interface speed on PPT besides the workstations' specifications. We tested each workstation using 500 ICMP echo packets and repeated the test 10 times. Each ICMP echo packet consists of the default frame length of 110 bytes.

TABLE I
Host specifications

	Dell Dimension 3000	Dell Inspiron 531S
Name	D3000	I531S
CPU (speed)	Intel Pentium 4 (3 GHz)	AMD Athlon 64 X2 (1 GHz)
RAM	512 MB	1024 MB
RAM speed (data width)	400 MHz (64 bits)	667 MHz (64 bits)
PCI bus speed	266 MB/s	133 MB/s
Linux kernel version	2.6.18	2.6.18
NIC driver	Intel Corp. 82562EZ	nVidia Corp. MCP61

Table II shows the summary of the measured PPTs (PPT column) and their standard deviations (std column) on the D3000 and I531S workstations. Table II shows that the PPTs of the D3000 workstation are 21 μs and 14 μs under 10-Mb/s and 100-Mb/s transmission speeds, respectively. For the I531S workstation, the PPTs for those transmission speeds are 16 μs and 7 μs, respectively. The standard deviations of the measured

TABLE II
SUMMARY OF PPT MEASUREMENT

dst	Link capacity (Mb/s)	$t_7 - t_1$ (μs)	$t_2 - t_1$ (μs)	$t_5 - t_3$ (μs)	$2 \times PPT$ (μs)	PPT (μs)	std (μs)
D3000	10	151	88	22	41	21	0.31
D3000	100	57	9	22	27	14	0.42
I531S	10	161	88	41	32	16	0.31
I531S	100	63	9	41	13	7	0.42

PPTs on both workstations under each transmission speed is smaller than 1 μs.

Figure 6 shows the distributions of samples of ICMP packet generation times ($t_5 - t_3$) of each workstation under 10- and 100-Mb/s transmission speeds for 500 ICMP packets. Figures 6(a) and 6(b) show that the mean packet generation time measured on the D3000 workstation under 10-Mb/s and 100-Mb/s transmission speeds are 22 μs and 23 μs, respectively. According to Figures 6(c) and 6(d), the mean packet generation times measured on the I531S workstation for those transmission speeds are 40 μs and 41 μs, respectively. Similar packet generation times on both transmission speeds on each workstation, as shown in Figure 6 and Table II (the $t_5 - t_3$ column), suggest that the packet generation time of a host is independent of its transmission speed. However, the D3000 workstation has smaller packet generation time (i.e., about 20 μs smaller) of the two workstations as it has higher CPU and bus speeds, according to Table I. This phenomenon infers that the packet generation time depends on the host's specifications.

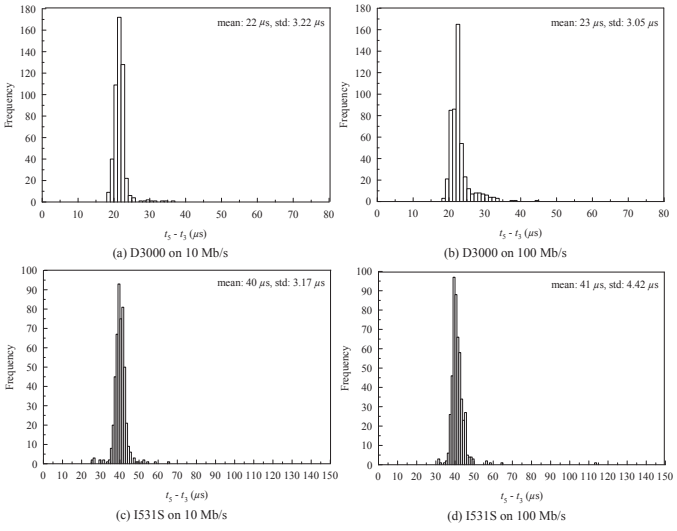


Fig. 6. ICMP packet generation time ($t_5 - t_3$) on the workstations: D3000 with a NIC running at (a) 10 Mb/s and (b) 100 Mb/s, and I531S with a NIC running at (c) 10 Mb/s and (d) 100 Mb/s.

Figure 7 shows sampled distributions of the time stamping intervals between Q and R ($t_7 - t_1$), measured by hsf. Figures 7(a) and 7(b) show that the mean of $t_7 - t_1$ measured on the D3000 workstation under 10-Mb/s and 100-Mb/s transmission speeds are 150 μs and 59 μs, respectively. Here, the mean value measured on 10 Mb/s is larger than that on 100 Mb/s because $t_7 - t_1$ includes the receiving time of Q, as shown in Figure 5, which is inversely proportional to the transmission speed of dst. Figures 7(c) and 7(d) show that the mean of $t_7 - t_1$ measured on the I531S workstation are 160 μs and 63 μs under 10-Mb/s and 100-Mb/s transmission speeds, respectively.

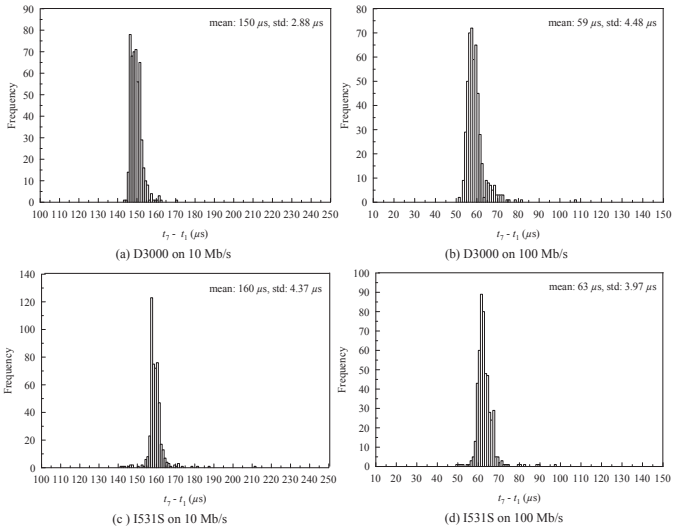


Fig. 7. Time stamping interval between Q and R ($t_7 - t_1$) recorded by hst on the workstations: D3000 with a NIC running at (a) 10 Mb/s and (b) 100 Mb/s, and I531S with a NIC running at (c) 10 Mb/s and (d) 100 Mb/s.

V. RELATED WORK

There is no existing scheme to measure the PPT of an end host to the best of our knowledge, but the measurement of PPT of a router has been considered of interest. A previous work measured PPT of hardware routers in an end-to-end path by instrumenting their input and output links with packet-capture card given that the method has physical access to the routers under test [7]. Here, the PPT of a router is defined as the time interval between the departure of a packet from the ingress queue of the input link and the arrival of the same at the egress queue of the output link of the router; therefore, the actual value is equivalent to two times PPT plus the packet-switching latency through the router's switching fabric.

Another study extended the above mentioned method to measure PPT of software routers by instrumenting the routers with dedicated software processes (i.e., kernel functions) that capture the ongoing traffic between the input and output links, both at the data and application layers [32].

Beside PPT of routers, a scheme, called fast-path/slow-path discriminator (*fsd*), was proposed to measure packet generation time of routers using ICMP packets [33]. In *fsd*, the source host sends two different types of probing packets, a direct probe and a hop-limited probe, toward the destination host

of a multiple-hop path, consisting of n nodes, for estimating OWD between the end hosts to measure the packet generation time of routers, e.g., node i, where $2 \leq i \leq n-1$, in the path. The direct probe is a specially crafted ICMP echo reply packet with a Time-to-live (TTL) value to enable reaching the destination host through node i, which is the router under test. The hop-limited probe is a specially crafted ICMP echo reply packet spoofed with the destination's IP address as its source address and a TTL value that expires at node i. The hop-limited probe forces node i to generate an ICMP Time Exceeded (TE) packet and sends it to the destination (because of the spoofed source address) so that the OWD of the end-to-end path can be measured. Because the OWD measured by the hop-limited packet is overestimated by the packet generation time of TE at node i, the packet generation time of node i is estimated from the difference between the OWDs of the direct and hop-limited probes over the path. Unlike the above mentioned methods for measuring PPT of routers, *fsd* does not require instrumentation and physical access to the routers for measuring their packet processing times.

VI. Conclusions

We proposed a methodology to measure the PPT of a host (i.e., workstation) using a specialized packet-capture card in a LAN setup. To measure PPT, we send an ICMP echo request packet to trigger an ICMP echo reply packet at the host under test, and collect the time stamps at the data-link and application layers using the clocks of the packet-capture card and the host, respectively. The methodology does not require clock synchronization between the host and the packet-capture card. We tested the proposed methodology on two different hosts connected to the network with an interface running at 10 and 100 Mb/s. The experimental results show that our methodology can measure PPT of the hosts consistently, and without requiring clock synchronization.

References

[1] G. Almes, S. Kalidindi, and M. Zekauskas. RFC 2679 – A one-way delay metric for IPPM. [Online]. Available: http://www.ietf.org/rfc/rfc2679.txt.

[2] —. RFC 2681 – A round-trip delay metric for IPPM. [Online]. Available: http://www.ietf.org/rfc/rfc2681.txt.

[3] N. McKeown. High performance routers – Talk at IEE, London UK. October 18th, 2001. [Online]. Available: http://tiny-tera.stanford.edu/∼nickm/talks/index.html.

[4] G. Jin and B. Tierney, "System capability effects on algorithms for network bandwidth measurements," in Proc. of *ACM IMC*, 27-29, FL, USA, 2003, pp. 27–83.

[5] R. Prasad, M. Jain, and C. Dovrolis, "Effects of interrupt coalescence on network measurements," in Proc. of *PAM*, France, 2004, pp. 247-256.

[6] A. Hernandez and E. Magana, "One-way delay measurement and characterization," in Proc. of *ICNS*, Athens, Greece, 2007, p. 114.

[7] K. Papagiannaki, S. Moon, C. Fraleigh, P. Thiran, and C. Diot, "Measurement and analysis of single-hop delay on an IP backbone network," *IEEE JSAC*, vol. 21, no. 6, pp. 908-921, 2003.

[8] B. Forouzan, *TCP/IP Protocol Suite*, ch. 3. NY, USA: McGraw Hill, 2010.

[9] M. Garetto and D. Towsley, "Modeling, simulation and measurements of queuing delay under long-tail Internet traffic," in Proc. of *ACM SIGMETRICS*, CA, USA, 2003, pp. 1-11.

[10] A. Downey, "Using pathchar to estimate Internet link characteristics," in Proc. of *ACM SIGCOMM*, MA, USA, 1999, pp. 241-250.

[11] S. Zander and S. J. Murdoch, "An improved clock-skew measurement technique for revealing hidden services," in Proc. of *USENIX Security'08*, CA, USA, 2008, pp. 211-225.

[12] Caida. Packet size distribution comparison between Internet links in 19 98 and 2008. [Online]. Available: http://www.caida.org/research/traffic-analysis/pkt_siz_distribution/graphs.xml.

[13] R. Sinha, C. Papadopoulos, and J. Heidemann, "Internet packet size distributions: Some observations," USC/Information Sciences Institute, Tech. Rep. ISI-TR-2007-643, May 2007. [Online]. Available: http://www.isi.edu/ johnh/PAPERS/Sinha07a.html.

[14] M. Lee, N. Duffield, and R. Kompella, "Not all microseconds are equal: Fine-grained per-flow measurementswith reference latency interpolation," in Proc. of *ACM SIGCOMM*, Dehli, India, 2010, pp. 27-38.

[15] Wall street's quest to process data at the speed of light. [Online]. Available: http://www.informationweek.com/news/199200297?pgno=1.

[16] V. Padmanabhan and L. Subramanian, "An investigation of geographic mapping techniques for Internet hosts," in Proc. of *ACM SIGCOMM*, CA, USA, 2001, pp. 173-185.

[17] E. Katz-Bassett, J. John, A. Krishnamurthy, T. Anderson, and Y. Chawathe, "Towards IP geolocation using delay and topology measurements," in Proc. of *ACM IMC*, NY, USA, 2006, pp. 71-84.

[18] B. Gueye, A. Ziviani, M. Crovella, and S. Fdida, "Constraint-based geolocation of Internet hosts," *IEEE/ACM ToN*, vol. 14, no. 6, pp. 1219-1232, 2006.

[19] Internet2 network: Networking for the future. [Online]. Available: http://www.internet2.edu/network/.

[20] S. Leinen. What flows in a reserach and education network? [Online]. Available: http://pam2009.kaist.ac.kr/presentation/switchflows.pdf.

[21] R. Carter and M. Crovella, "Measuring bottleneck link speed in packet switched networks," *Performance Evaluation*, vol. 27 and 28, pp. 297-318, 1996.

[22] V. Paxson, "Measurements and analysis of end-to-end Internet dynamics," Ph.D. dissertation, University of California, Berkeley, Stanford, California, 1997.

[23] C. Dovrolis, P. Ramanathan, and D. Moore, "Packet dispersion techniques and capacity estimation," *IEEE/ACM ToN*, vol. 12, no. 6, pp. 963-977, 2004.

[24] K. Salehin and R. Rojas-Cessa, "A combined methodology for measurement of available bandwidth and link capacity in wired packet networks," *IET Commun.*, vol. 4, no. 2, pp. 240-252, 2010.

[25] —, "Scheme to measure relative clock skew of two Internet hosts based on end-link capacity," *IET Electron. Lett.*, vol. 48, no. 20, pp. 1282-1284, 2012.

[26] —, "Active scheme to measure throughput of wireless access link in hybrid wired-wireless network," *IEEE Wireless Commun. Lett.*, vol. 1, no. 6, pp. 645-648, 2012.

[27] Endace DAG 7.5G2 datasheet. [Online]. Available: http://www.endace.com/assets/files/resources/END_Datasheet_DAG7.5G2_5.0.pdf.

[28] I. Csabai, A. Fekete, P. Haga, B. Hullar, G. Kurucz, S. Laki et al., "ETOMIC advanced network monitoring system for future Internet experimentation," *TridentCom*, Berlin, Germany, 2010, pp. 243-254.

[29] D. Mills, J. Martin, J. Burbank, and W. Kasch. RFC 1305 - Network time protocol version 4: Protocol and algorithms specification. [Online]. Available: http://www.ietf.org/rfc/rfc5905.txt.

[30] P. Willman, H. Kim, S. Rixner, and V. Pai, "An efficient programmable 10 gigabit ethernet network interface card," in Proc. of *IEEE HPCA-11*, CA, USA, 2005, pp. 96-107.

[31] K. Ramakrishnan, "Performance Considerations in Designing Network Interfaces," *IEEE JSAC*, vol. 11, no. 2, pp. 203-219, 1993.

[32] L. Angrisani, G. Ventre, L. Peluso, and A. Tedesco, "Measurement of processing and queuing delays introduced by an open-source router in a single-hop network," *IEEE TIM*, vol. 55, no. 4, pp. 1065-1076, 2006.

[33] R. Govindan and V. Paxson, "Estimating Router ICMP Generation Delays," in Proc. *PAM*, CO, USA, 2002, pp. 1-8.