Copyright Warning & Restrictions

The copyright law of the United States (Title 17, United States Code) governs the making of photocopies or other reproductions of copyrighted material.

Under certain conditions specified in the law, libraries and archives are authorized to furnish a photocopy or other reproduction. One of these specified conditions is that the photocopy or reproduction is not to be "used for any purpose other than private study, scholarship, or research." If a, user makes a request for, or later uses, a photocopy or reproduction for purposes in excess of "fair use" that user may be liable for copyright infringement,

This institution reserves the right to refuse to accept a copying order if, in its judgment, fulfillment of the order would involve violation of copyright law.

Please Note: The author retains the copyright while the New Jersey Institute of Technology reserves the right to distribute this thesis or dissertation

Printing note: If you do not wish to print this page, then select "Pages from: first page # to: last page #" on the print dialog screen

ABSTRACT

END-TO-END NETWORK MEASUREMENT FOR WIRED AND WIRELESS NETWORKS

by
Khondaker Musfakus Salehin

A number of Internet applications require accurate characterization of different network parameters. There is, therefore, a need to determine whether the networks that carry packets for those applications comply with the requirements. Network measurement provides the means to perform this determination. This work focuses on schemes for measuring several parameters both in wired and wireless environments.

The need of measuring packet processing time (PPT) of the end hosts (i.e., workstations) in a wired network is unveiled. PPT is defined as the time elapsed between the arrival of a packet at the data-link layer and the time when the packet is time stamped at the application layer of the TCP/IP protocol stack of an end host, as defined by RFCs 2679 and 2681. The role of PPT becomes important to increase accuracy in the measurement of different network parameters (e.g., one way delay) as data rates increase.

Two schemes to measure the PPT of an end host have been proposed. The first scheme is designed to measure PPT of an end host using Internet Control Message Protocol (ICMP) packets and a specialized packet-capture card, given that the node is physically accessible. The second proposed scheme measures the PPT of a remote end host connected over a multiple-hop path based on the estimation of the capacity of the link connected to the end host using a packet-pair structure, called compound probe. Experimental evaluations in both testbed and Internet environments of the proposed schemes are presented.

A solution for measuring clock skew between two end hosts connected over a wired end-to-end path through the estimation of link capacity is proposed. Unlike existing schemes, the proposed solution is simple and does not require complex processing of the sampled data because the compound probe can also be used to detect data samples affected by noise (i.e., cross traffic) over the path. Simulation results of the proposed scheme are presented.

A scheme to measure the throughput of an IEEE 802.11 wireless access link in a hybrid wired-wireless path is introduced. Throughput estimation of a wireless access link is challenging and it can be affected by the cross traffic on the wired links and bottleneck-link location of the path. The proposed scheme measures the throughput of the download wireless link with high accuracy and without being affected by the cross traffic on the wired links and bottleneck-link location of a hybrid wired-wireless path. Experimental evaluations of the proposed scheme in a testbed environment are presented.

END-TO-END NETWORK MEASUREMENT FOR WIRED AND WIRELESS NETWORKS

by
Khondaker Musfakus Salehin

A Dissertation
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy in Electrical Engineering

Department of Electrical and Computer Engineering

April 2013

APPROVAL PAGE

END-TO-END NETWORK MEASUREMENT
FOR WIRED AND WIRELESS NETWORKS

Khondaker Musfakus Salehin

Dr. Roberto Rojas-Cessa, Dissertation Advisor — Date
Associate Professor of Electrical and Computer Engineering, NJIT

Dr. Nirwan Ansari, Committee Member — Date
Professor of Electrical and Computer Engineering, NJIT

Dr. Sotirios G. Ziavras, Committee Member — Date
Professor of Electrical and Computer Engineering, NJIT

Dr. Abdallah Khreishah, Committee Member — Date
Assistant Professor of Electrical and Computer Engineering, NJIT

Dr. Aleksandar Kolarov, Committee Member — Date
Senior Scientist, Applied Communication Sciences

BIOGRAPHICAL SKETCH

Author: Khondaker Musfakus Salehin

Degree: Doctor of Philosophy

Date: April 2013

Undergraduate and Graduate Education:

- Master of Science in Telecommunications Engineering,
  New Jersey Institute of Technology, Newark, NJ, USA, Dec. 2007
- Bachelor of Science in Computer Science and Engineering,
  Ahsanullah University of Science and Technology, Dhaka, Bangladesh, Nov. 2004

Major: Electrical Engineering

Publications:

K. Salehin, and R. Rojas-Cessa, "Measurement of Packet Processing Time of an Internet Host using Asynchronous Packet Capture at the Data-Link Layer," to appear in *Proc. of IEEE International Conference on Communications*, Budapest, Hungary, pp. 1-5, 2013

K. Salehin, and R. Rojas-Cessa, "Scheme to Measure Throughput and Available Bandwidth of Download Wireless Access Link in a Wired-Wireless Network," in *Proc. of Passive and Active Measurement Conference*, Hong Kong, pp. 1-2, 2013 (Poster)

K. Salehin, and R. Rojas-Cessa, "Packet-pair sizing for controlling packet dispersion on wired heterogeneous networks," in *Proc. of IEEE International Conference on Computing, Networking and Communication*, CA, USA, pp. 1-5, 2013

K. Salehin, and R. Rojas-Cessa, "Active scheme to measure throughput of wireless access link in hybrid wired-wireless network," *IEEE Wireless Communications Letters*, vol. 1, no. 6, pp. 645-648, 2012

K. Salehin, and R. Rojas-Cessa, "Scheme to measure relative clock skew of two Internet hosts based on end-link capacity," *IET Electronics Letters*, vol. 48, no. 20, pp. 1282-1284, 2012

R. Rojas-Cessa, K. Salehin, and K. Egoh “Experimental Performance Evaluation of a Virtual Router,” in *Proc. of IEEE Sarnoff Symposium*, NJ, USA, pp. 1-5, 2012

R. Rojas-Cessa, K. Salehin, and K. Egoh “Experimental Performance Evaluation of a Virtual Router,” in *Proc. of IEEE Workshop on Local and Metropolitan Area Networks*, NC, USA, pp. 1-2, 2011 (Poster)

K. Salehin, and R. Rojas-Cessa, “Ternary Search Based Scheme to Measure Link Available Bandwidth in Wired Netowrks,” in *Proc. of IEEE Global Communications Conference*, FL, USA, pp. 1-5, 2010

K. Salehin, and R. Rojas-Cessa, “Schemes to Measure Available Bandwidth and Link Capacity with Ternary Search and Compound Probe for Packet Networks,” in *Proc. of IEEE Workshop on Local and Metropolitan Area Networks*, NJ, USA, pp. 1-5, 2010

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

K. Salehin, R. Rojas-Cessa, C. Lin, Z. Dong, and T. Kijkanjanarat, “Scheme to Measure Packet Processing Time of a Remote Host through Estimation of End-Link Capacity,” major revision, submitted to *IEEE Transactions on Computers*

K. Salehin and R. Rojas-Cessa, “Scheme to Measure Throughput and Available Bandwidth of Download Wireless Access Link in a Wired-Wireless Network,” under review at *IEEE Global Communications Conference*, 2013

To my parents: Md. Mosharraf Hossain Khondaker and Rahima Begum

ACKNOWLEDGMENT

I would like to thank Professor Roberto Rojas-Cessa for giving me the opportunity to work under his tutelage. It was him who taught me how to do research with meticulous care. He was a continuing source of motivation and support for me both in academic and personal matters. I enjoyed working with him.

I would like to thank Professor Nirwan Ansari, Professor Sotirios Ziavras, Professor Abdallah Khreishah, and Professor Aleksandar Kolarov for serving on my dissertation committee. Their insightful advice gave me the direction to improve my work.

Special thanks go to all members of Networking Research Laboratory, who I have come across with over the years, for being wonderful colleagues and friends. Among them, Chuan-bi Lin and Ziqian Dong deserve appreciation for helping me in the Internet experiments in Chapter 3.

I admire Professor Muhammad M. Haque of McNeese State University for being a father figure far away from home.

Finally, I want to extend my gratitude to my parents, Md. Mosharraf Hossain Khondaker and Rahima Begum, elder sister, Laila Farzana, and grandmother for providing unconditional love and inspiration. My father provided all the good things in life. My mother not only took me to school when I was little but also sat with me in the classroom for about a month to increase my determination. I dedicate this dissertation to both of them for their unremitting patience and sacrifice.

TABLE OF CONTENTS

TABLE OF CONTENTS
(Continued)

Chapter **Page**

TABLE OF CONTENTS
(Continued)

Chapter **Page**

LIST OF TABLES

Table **Page**

LIST OF FIGURES

Figure **Page**

LIST OF FIGURES
(Continued)

Figure **Page**

LIST OF FIGURES
(Continued)

Figure **Page**

CHAPTER 1

INTRODUCTION

Internet is the primary domain of communication in today's world. Wide Internet infrastructure and prevalent use of different digital devices prompted the development of various applications since the last decade. Popularity of the Internet is still on the rise with the recent advent of social networks, where users spend a large amount of their time for personal and commercial interactions. To keep up with the increasing popularity and demands, information-carrying capacity of the Internet is expanding in both wired and wireless environments. Characterization of the Internet is also gaining more importance to efficiently utilize network resources, deploy new services, and ensure better user experience.

The functioning of the Internet can be described by various network parameters. Measurement of these parameters is required to characterize the Internet to define its qualitative performance and user experience. Measurement of any network parameter in the Internet, both in wired and wireless environments, is complex due to its large and distributed infrastructure, and dynamic nature. Yet, high accuracy and precision in the measurement of different network parameters (e.g., delay, link capacity, throughput, clock skew) are necessary for optimal use of the Internet.

Existing methods available to measure different network parameters are broadly categorized into two broad spectrums: passive measurement and active measurement [1]. Passive measurement involves estimating network parameters using the data packets of existing traffic flow(s) in a network path. This approach requires capturing of the

data packets from the traffic flow(s) and it is accomplished by instrumenting the point of measurement interest (e.g., an intermediate node) in a network path with specialized hardware [2, 3]. Even though passive measurement is an attractive approach because it does not perturb the user traffic by injecting additional probing packets into the network, its applicability is limited due to lack of universal administrative access to the Internet. Active measurement has attracted significant interest to alleviate this problem. In active measurement, network parameters are measured by actively sending probing packets from a source node to a destination node of a network path without requiring additional administration access along the path. This approach, however, is susceptible to noise (i.e., the user traffic or cross traffic) in the path under interest, and that has the potential to negate the quality (accuracy and precision) of measurement.

This dissertation focuses on active measurement for estimating three different network parameters in wired and wireless networks. Both high quality and wide applicability of measurement were considered as the motivating factor behind this work.

Packet processing time (PPT) is defined as the time elapsed between the arrival of a packet at the data-link layer and the time when the packet is time stamped at the application layer of the TCP/IP protocol stack of an end host (i.e., workstation), as defined by FRCS 2679 and 2681. It is shown that measurement of PPT is important as the transmission speed of Internet is increasing. The knowledge of PPT can improve the measurement accuracy of different other network parameters (e.g., one way delay, OWD, of an end-to-end path) and related applications (e.g., IP geolocation) in the Internet. Considering the above stated significance, an active scheme using Internet Control Protocol (ICMP) packet and a specialized packet-capture hardware is introduced to measure PPT in the local areal network (LAN). This scheme does not require synchronization between the end host and

the packet-capture card for estimating PPT. Three workstations with different specifications have been used to experimentally evaluate the scheme under three different (i.e., 10, 100, and 1000 Mb/s) transmission speeds. The stability of the scheme is demonstrated in the experimental evaluation.

To measure PPT in the wide area network (WAN) or in the Internet, another active scheme is proposed based on the estimation of the capacity (i.e., the transmission speed) of the link directly connected to the host under measurement, or end link, in a multiple-hop path. Estimation of end-link capacity is performed using a packet pair-structure, called compound probe. An analytical model for sizing the packets of the compound probe for ensuring robustness against the cross traffic of an end-to-end path is derived and then it is verified through numerical evaluations. This scheme is tested on a controlled testbed and in the Internet using different path configurations with 10-Mb/s and 100-Mb/s end links. The accuracy of the scheme is compared with that of the above scheme proposed for LAN environment, as mentioned above.

A discussion on the effect of clock resolution (i.e., the smallest instance of time that can be realized by a timing device) and link heterogeneity in the Internet for measuring PPT using existing clocks in the commodity workstations is also presented. The consistent accuracy of the proposed scheme confirms its applicability in the Internet.

Various clock issues (e.g., clock offset and clock skew) between a pair of end hosts involved in OWD estimation in wired network can degrade the accuracy of the measurement. A solution for measuring relative clock skew between two end hosts connected over a multiple-hop path in a wired environment is explored. Existing schemes for measuring clock skew are complex and their accuracy is prone to cross traffic of the measurement path [4]. The proposed scheme uses compound probes to accurately the

measure relative clock skew between two end hosts using a simple statistical processing of the data samples collected from capacity measurements of the links directly connected to the end hosts. The accuracy of the scheme is tested under high cross-traffic loads on an end-to-end path in simulation environment. The simulation results validates the high accuracy of the scheme.

Throughput of wireless access link (i.e., end link) is an important qualitative parameter in a hybrid wired-wireless path, an end-to-end path consisting of multiple wired links and a wireless end link. Access methodology of the wireless link based on the IEEE 802.11 protocol standard [5] is not the same as the access methodology used in the IEEE 802.2 protocol standard [6], defined for wired network. A novel throughput-measurement scheme is proposed to accurately characterize an IEEE 802.11 based wireless access link in a hybrid wired-wireless path. The proposed scheme uses compound probe to measure throughput of wireless access link without requiring the bottleneck link of the hybrid path on the wireless access link unlike existing schemes [7]. The scheme is evaluated in a controlled testbed environment using IEEE 802.11b/g links under high cross-traffic loads on the wired links to show its robustness against the cross-traffic loads on the wired link of the hybrid wired-wireless path. The accuracy of the experimental results shows that the proposed scheme outperforms the scheme considered as the state-of-art.

CHAPTER 2

PACKET PROCESSING TIME: SINGLE-HOP CASE

2.1 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 [8, 9]. As link rates increase faster than processing speeds [10, 11, 12, 13], the role of PPT becomes more important in the measurement of different network parameters.

One-way delay (OWD) in a LAN is an example of a parameter that PPT can significantly impact [14]. Figure 2.1 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 [8]. 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:

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

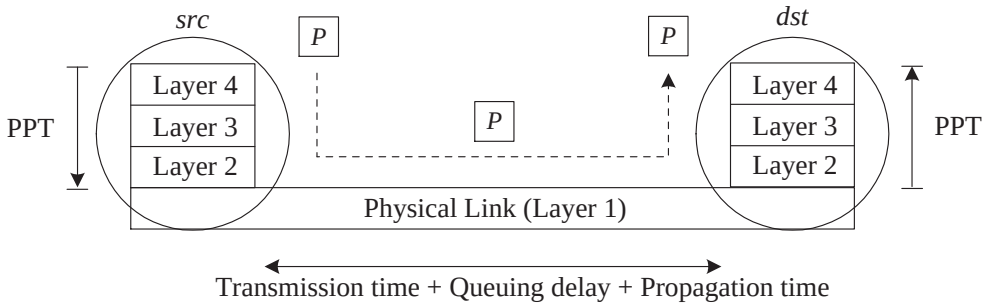


Figure 2.1 End-to-end one-way delay (OWD) of packet P between two directly connected hosts.

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 [15], 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 [16]. This error increases up to 52% when queuing delay is relieved ($t_q \simeq 0$ μs [15], [17],[18]) for a 40-byte packet, which constitutes 50% or more of the IP traffic [19, 20]. 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 [16]). 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 [21, 22].

In WAN, high-resolution OWD measurement can be used to increase accuracy in IP geolocation [23, 24, 25, 26]. In IP geolocation, each microsecond of propagation time varies the estimated geographic distance by 200 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 [27], [28]. For example, in the measurement schemes based on packet-pair structure [29, 30, 31, 32, 7, 33], 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 effect of PPT on accuracy of the network measurement tools is an open problem [34, 35].

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 (here, the time stamping performed at the application layer is considered to be the packet-processing event). However, time stamping at the data-link layer is not readily available in popular and deployed NICs [3]. PPT measurement can be performed by placing a specialized packet-capture card in the same subnet where the host under test is located. Existing packet-capture cards have a time stamping resolution in the nanosecond range [36], [2], and their use require time synchronization [37] with the host's clock. This is difficult to accomplish since operating systems of a host can provide up to microsecond resolution [38]. In this chapter, a scheme to measure the PPT of a host using a specialized packet-capture card in the same subnet is proposed with requiring clock synchronization between the host under test and the packet-capture card. An experimental evaluation of the scheme is presented and the outcomes show consistent and measurable results.

The remainder of the chapter is organized as follows: Section 2.2 discusses the basic architecture of a host and its NIC operations. Section 2.3 introduces the proposed scheme to measure PPT in LAN setup. Section 2.4 shows the experimental results of PPTs measured on three different hosts. Section 2.5 presents the existing schemes for PPT measurement. Section 2.6 concludes the discussion.

2.2 Determining Factors of PPT in Hosts

The packet-processing event is considered to be the time stamping of a packet transmission at an end host. 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 [11, 39, 40]. 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, the basic architecture of a host and the operations of a NIC is presented.

2.2.1 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 2.2 shows this simplified architecture.

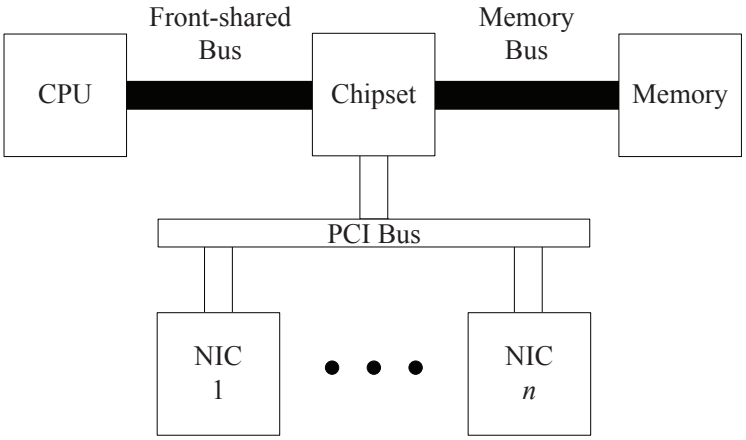
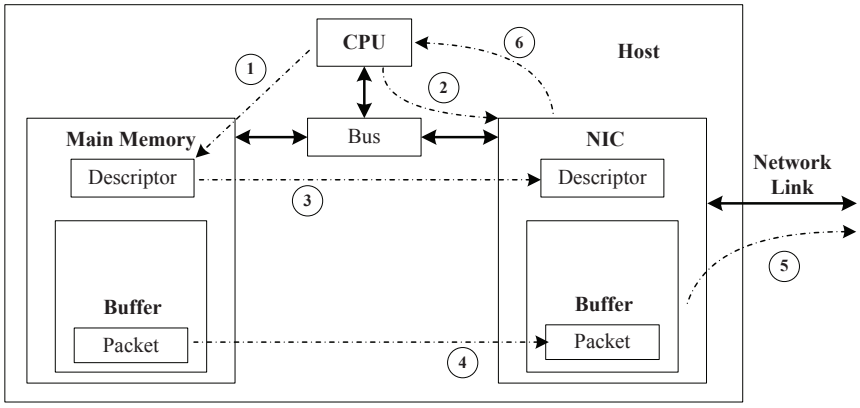


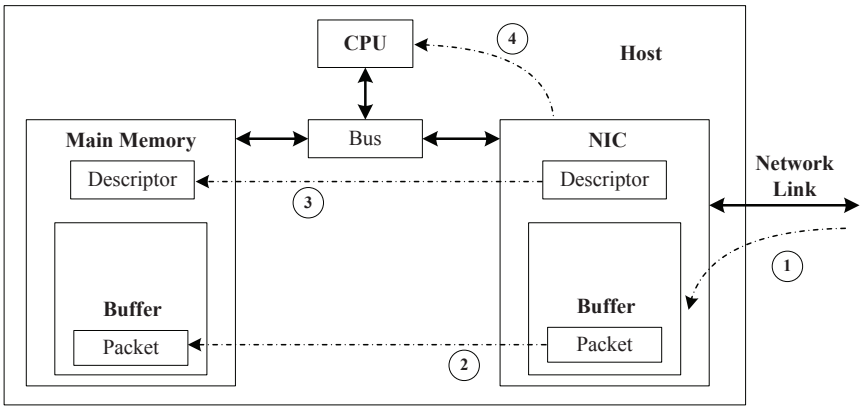
Figure 2.2 Top-level architecture of a general-purpose host.

2.2.2 NIC Operations

Figure 2.3 illustrates the two basic operations of the NIC: packet transmission and packet receiving processes [39, 40]. Figure 2.3(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 2.3(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).



(a) Transmission process



(b) Receiving process

(b) Receiving process

Figure 2.3 Basic NIC operations: (a) transmission process and (b) receiving process.

2.3 Proposed PPT Measurement Method

Figure 2.4 shows the proposed scheme and measurement 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 [36]), 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.

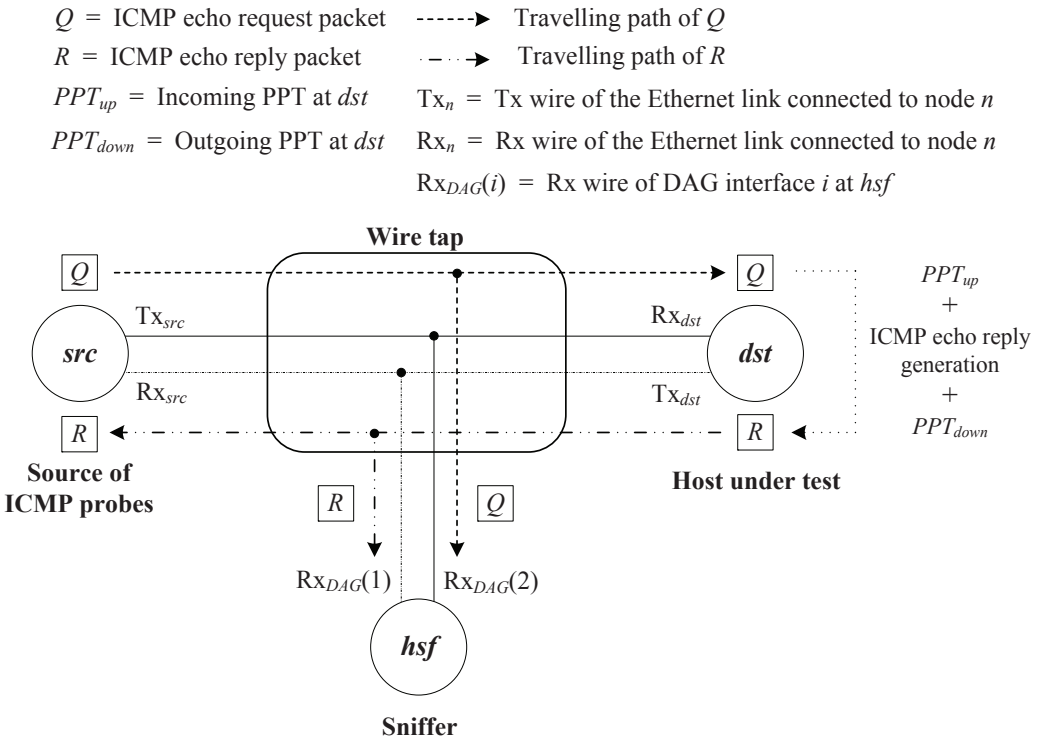


Figure 2.4 Experimental setup to measure PPT of dst.

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.

Figure 2.5 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 Figure 2.5 considers 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 2.2.2.

2.4 Experimental Results

PPT of three different hosts, a Dell Dimension 3000 (D3000), a Dell Inspiron I531S (I531S), and a Dell Optiplex 790 (DO790) workstations, were measured to evaluate the proposed scheme. The specifications of the workstations are shown in Table 2.1. PPT measurements on these three workstations was performed with their NICs running at 10, 100, and 100 Mb/s to investigate the effect of interface speed on PPT besides the workstations' specifications.

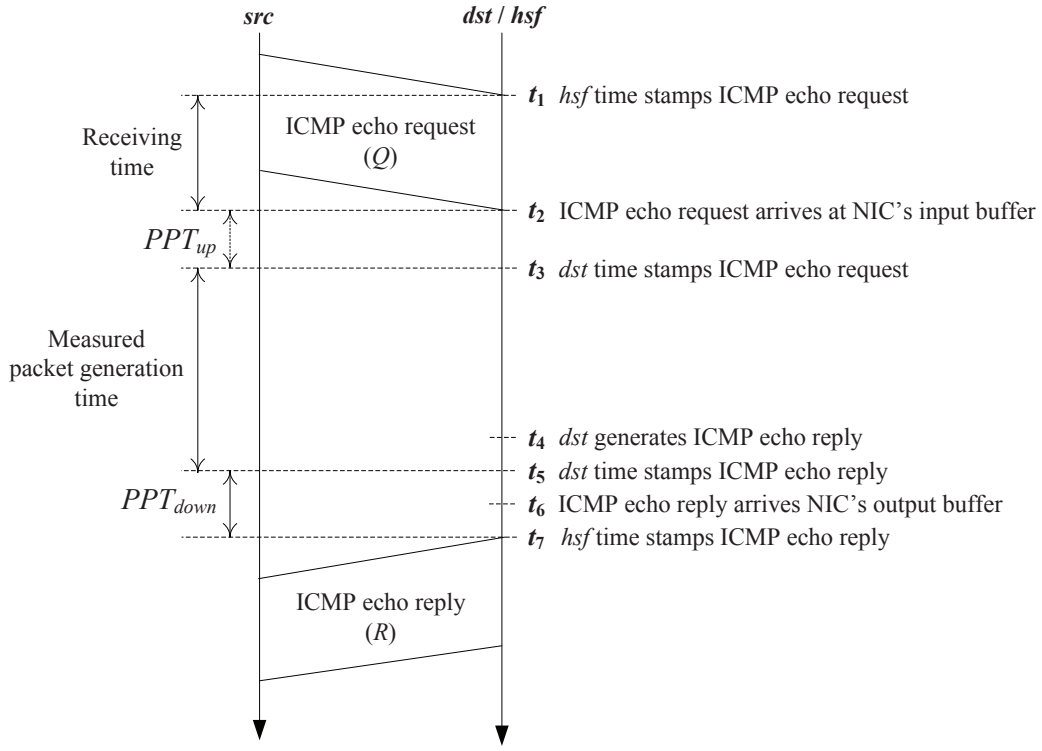


Figure 2.5 Timeline of ICMP echo request and echo reply packets at dst.

Table 2.1 Workstation Specifications

	Dell Dimension 3000	Dell Inspiron I531S	Dell Optiplex 790
Name	D3000	I531S	DO790
CPU (speed)	Intel Pentium 4 (3 GHz)	AMD Athlon 64 X2 (1 GHz)	Intel Core i3 (3.3 GHz)
RAM	512 MB	1024 MB	8148 MB
RAM speed (data width)	400 MHz (64 bits)	667 MHz (64 bits)	1333 MHz (64 bits)
PCI bus speed	266 MB/s	133 MB/s	4 GB/s
NIC speed	10/100 Mb/s	10/100/1000 Mb/s	10/100/1000 Mb/s
Linux kernel version	2.6.18	2.6.18	2.6.35

2.4.1 Measured PPTs

PPT measurements on the D3000 and I531S workstations were performed using their integrated Fast-Ethernet NICs: Intel Corp. 82562EZ and nVidia Corp. MCP61. PPTs of the above mentioned workstations under 10- and 100-Mb/s transmission speeds were measured 10 times, using 500 ICMP echo packets each time. Each ICMP echo packet consists of a frame length of 110 bytes.

Table 2.2 shows the summary of the measured PPTs (the PPT column) and the standard deviations (the std column) of the D3000 and I531S workstations. Table 2.2 shows that the PPTs of the D3000 workstation are 21 and 14 μs under 10- and 100-Mb/s transmission speeds, respectively. For the I531S workstation, these values are 16 and 7 μs, respectively. The standard deviations of the measured PPTs on both workstations are smaller than 1 μs.

Table 2.2 Measured PPTs using Integrated NICs

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

Another set of measurement on the I531S and DO790 workstations were performed under 10-, 100-, and 1000-Mb/s transmission speeds using an extended Gigabit-Ethernet NIC, Marvell Tech. 88E8053 PCI-E, and the same number of ICMP echo packets, as

used in the previous set of experiments. Table 2.3 shows the summary of the measured PPTs (the PPT column) and the standard deviations (the std column) of the tested workstations. The PPTs of the I531S workstation with the extended NIC are 71 μs under 10-Mb/s transmission speed, and 63 μs under 100- and 1000-Mb/s transmission speeds. The measured PPTs on the DO790 workstation are 99, 91, and 90 μs under 10-, 100-, and 1000-Mb/s transmission speeds, respectively. The standard deviations of the measured PPTs on both workstations, under each transmission speed, are smaller than 1 μs.

Table 2.3 Measured PPTs using Extended NICs

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)
I531S	10	265	88	35	142	71	0.42
I531S	100	170	9	35	126	63	0.48
I531S	1000	162	1	35	126	63	0.70
DO790	10	301	88	17	197	99	0.52
DO790	100	208	9	17	182	91	0.16
DO790	1000	198	1	16	180	90	0.72

Tables 2.2 and 2.3 show that the PPTs measured on each workstation using the integrated and extended NICs under 10-Mb/s transmission speed is about 8 μs larger than that under 100-Mb/s transmission speed. This variation in the measured PPTs for these two speeds is due to the minimum idle time period required after receiving a packet at the NIC, called Interframe Gap (IFG), as defined by the Ethernet standard [41]. For example, IFG under 10- and 100-Mb/s transmission speeds are 9.6 and 0.96 μs, respectively; therefore,

there is a difference of 8.64 μs in IFG, which is close to the above stated variation in the PPTs measured under these two speeds. The measured PPTs on the I531S and DO790 workstations under 100- and 1000-Mb/s transmission speeds are similar because the IFG under the latter speed is 0.096 μs.

The PPTs measured on the I531S workstation using the integrated and extended NICs show that the type of NIC of a workstation plays a major role in determining the PPT of a host, in addition to the transmission speed, which is discussed in [39]. As Tables 2.2 and 2.3 show, the PPTs measured on the I531S workstation using the extended NIC is around 50 μs larger than that using the integrated NIC.

2.4.2 Quality of Intermediate Variables

1) ICMP Packet Generation Time: Figure 2.6 shows the sampled distributions of the ICMP packet generation times ($t_5 - t_3$) of each workstation under 10- and 100-Mb/s transmission speeds for 500 ICMP packets using the integrated NIC. Figures 2.6(a) and 2.6(b) show that the mean packet generation time measured on the D3000 workstation under 10- and 100-Mb/s transmission speeds are 22 and 23 μs, respectively. According to Figures 2.6(c) and 2.6(d), the mean packet generation times measured on the I531S workstation for those transmission speeds are 40 and 41 μs, respectively.

Figures 2.7 and 2.8 present the sampled distributions of $t_5 - t_3$ measured on the I531S and DO790 workstations using the extended NIC. According to Figures 2.7(a) - 2.7(c), the mean packet generation time on the I531S workstation are 35, 35, and 34 μs under 10-, 100-, and 1000-Mb/s of transmission speeds, respectively. In case of the D0790 workstation, these values are 17, 17, and 16 μs for 10-, 100-, and 1000-Mb/s transmission speeds, as shown in Figures 2.8(a) - 2.8(c).

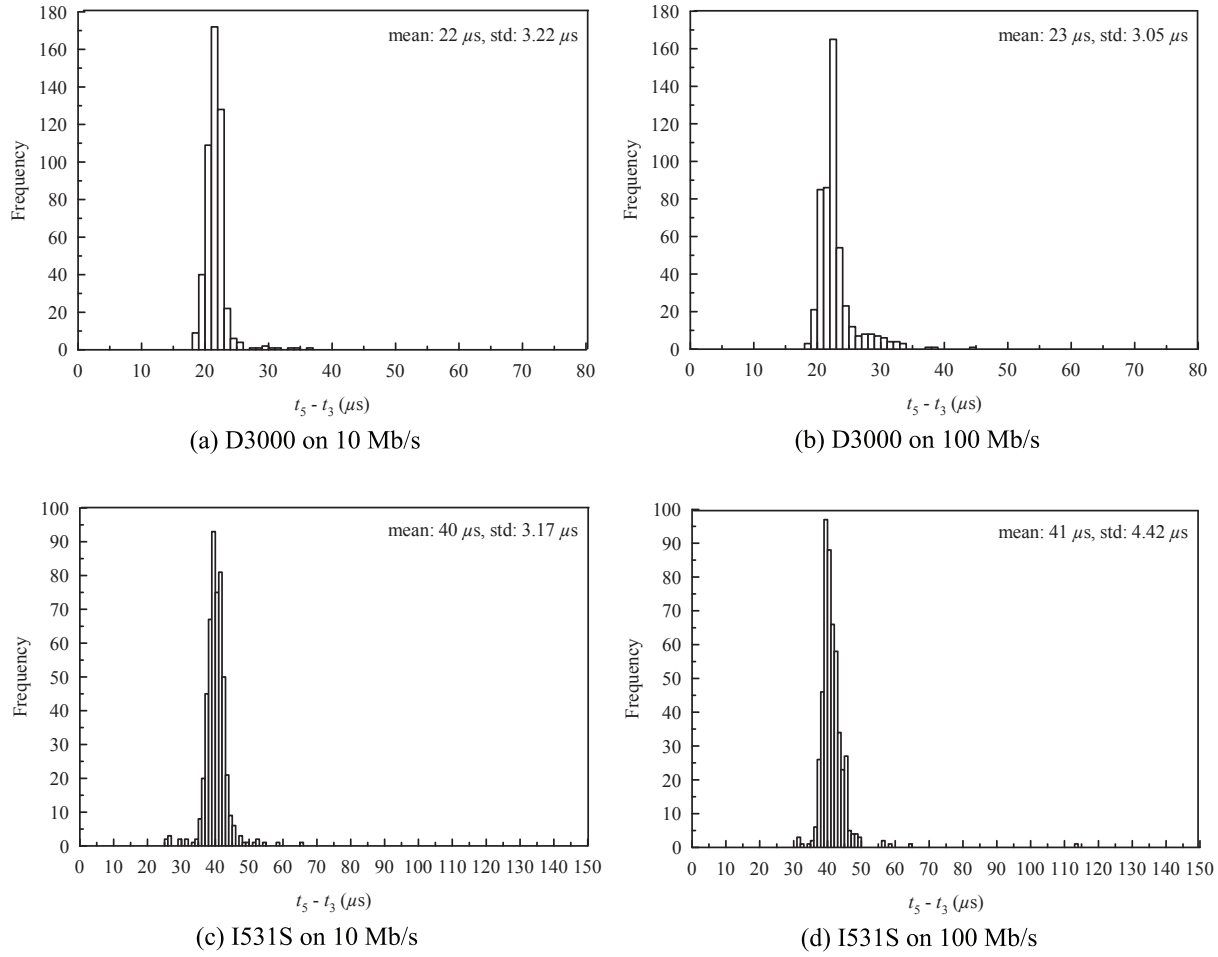
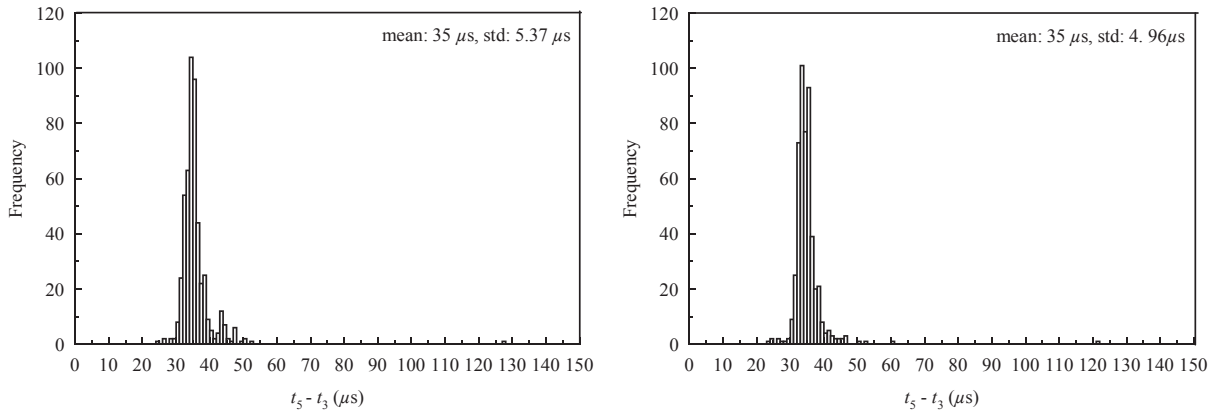
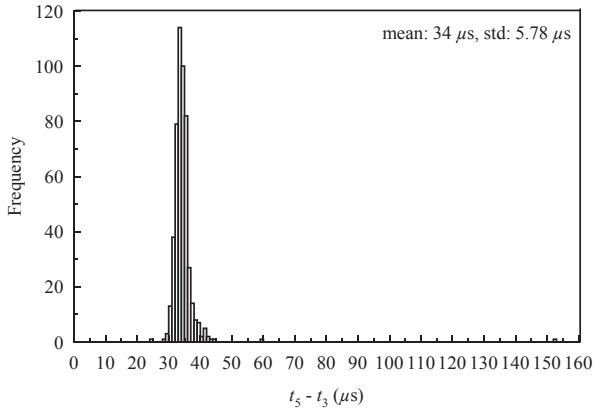


Figure 2.6 ICMP packet generation time ($t_5 - t_3$) on the workstations with an integrated NIC: D3000 under (a) 10 Mb/s and (b) 100 Mb/s, and I531S under (c) 10 Mb/s and (d) 100 Mb/s.



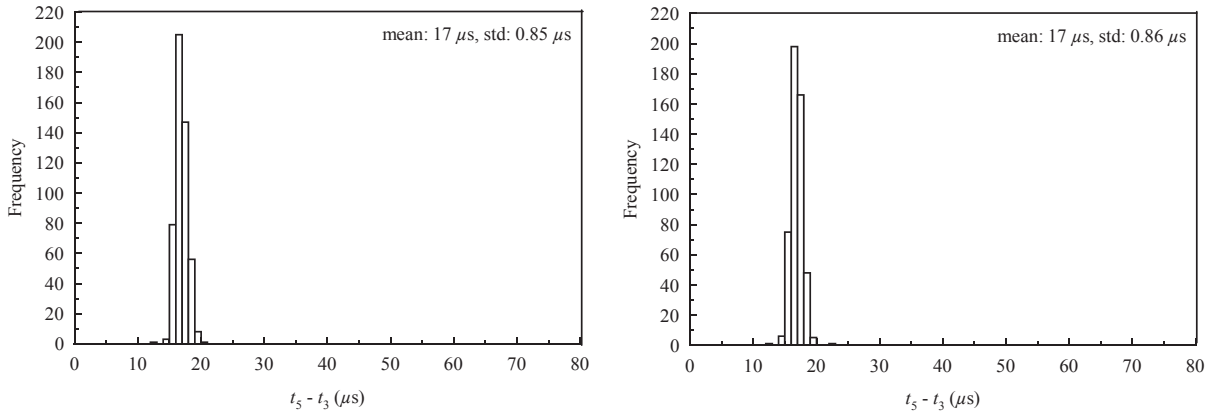
(a) I531S under 10 Mb/s

(b) I531S under 100 Mb/s



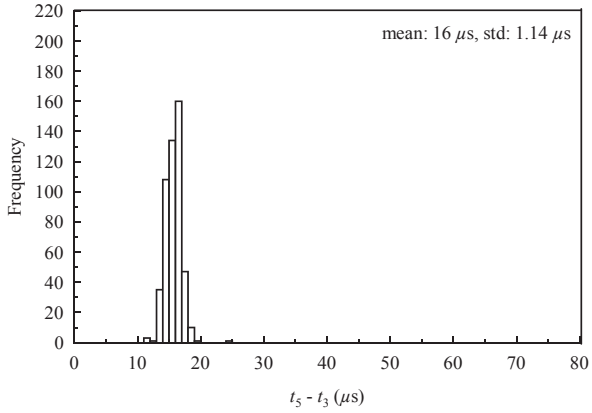
(c) I531S under 1000 Mb/s

Figure 2.7 ICMP packet generation time ($t_5 - t_3$) on the I531S workstation with an extended NIC: under (a) 10 Mb/s, (b) 100 Mb/s, and (c) 1000 Mb/s.



(a) DO790 under 10 Mb/s

(b) DO790 under 100 Mb/s



(c) DO790 under 1000 Mb/s

Figure 2.8 ICMP packet generation time ($t_5 - t_3$) on the DO790 workstation with an extended NIC: under (a) 10 Mb/s, (b) 100 Mb/s, and (c) 1000 Mb/s.

The packet generation times in Tables 2.2 and 2.3 and Figures 2.6 - 2.8 show that the DO790 workstation has the smallest packet generation time (i.e., 17 μs) among all workstations irrespective of the transmission speed because this workstation has the highest CPU, RAM, and bus speeds, as shown in Table 2.1. On the other hand, the I531S workstation has different packet generation times (22 and 35 μs) when the PPT is measured using integrated and extended NICs. These values infer that the packet generation time dependents both on the host's specifications and the type of NIC rather than the transmission speed.

2) DAG Time Stamping: Figure 2.9 shows the sampled distributions of the time stamping intervals between Q and R ($t_7 - t_1$), measured by hsf in the integrated NIC based experiments. Figures 2.9(a) and 2.9(b) show that the mean of $t_7 - t_1$ measured on the D3000 workstation under 10- and 100-Mb/s transmission speeds are 150 and 59 μs, respectively. 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 2.5, which is inversely proportional to the transmission speed of dst. Figures 2.9(c) and 2.9(d) show that the mean of $t_7 - t_1$ measured on the I531S workstation are 160 and 63 μs under 10- and 100-Mb/s transmission speeds, respectively.

Figures 2.10 and 2.11 present the sampled distributions measured on the I531S and DO790 workstations using extended NIC. According to Figures 2.10(a) - 2.10(c), the mean of $t_7 - t_1$ on the I531S workstation under 10-, 100-, and 1000-Mb/s transmission speeds are 266, 170, and 164 μs, respectively. The respective values on the DO790 workstation are 302, 207, and 195 μs, as shown in Figures 2.11(a) - 2.11(c).

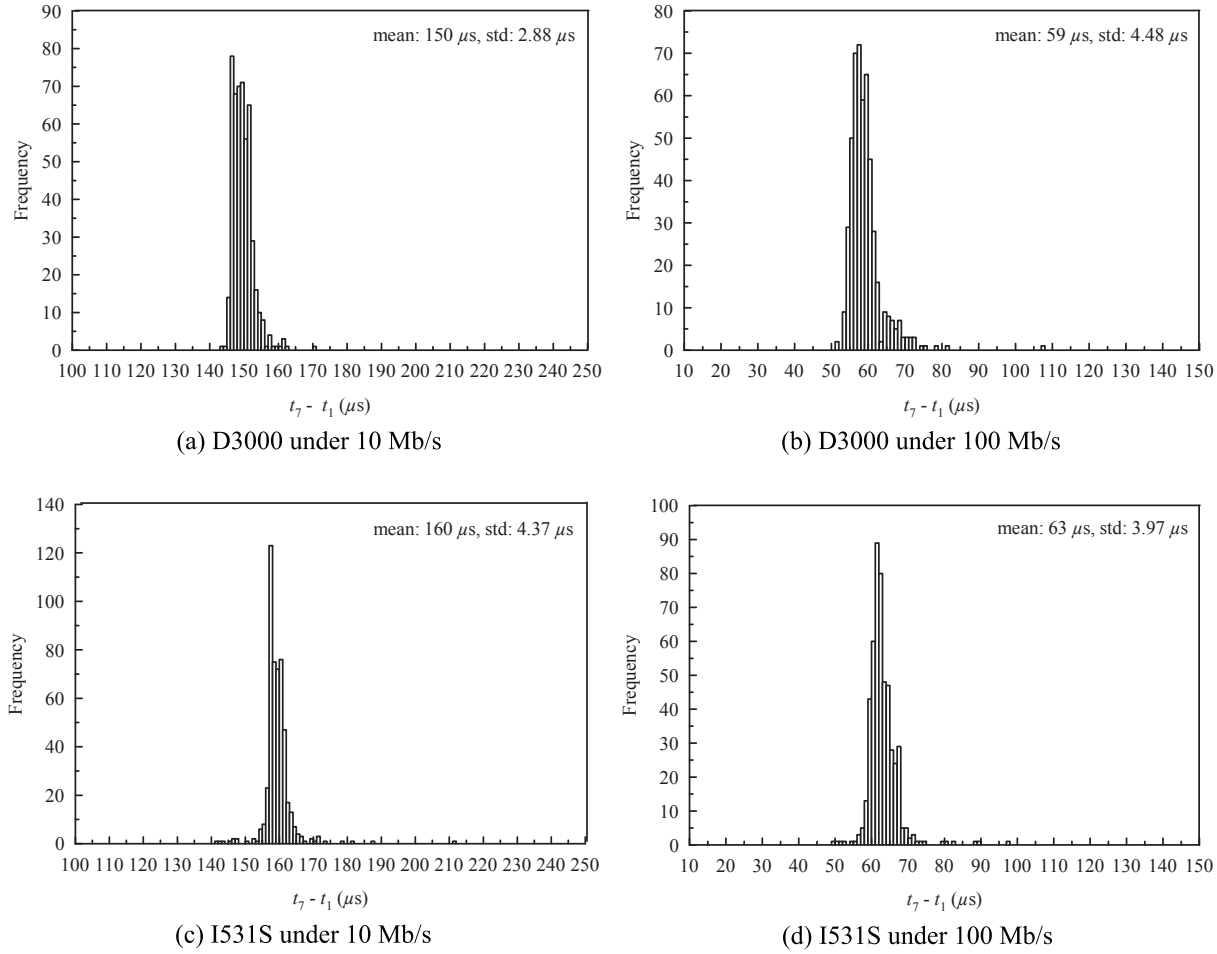
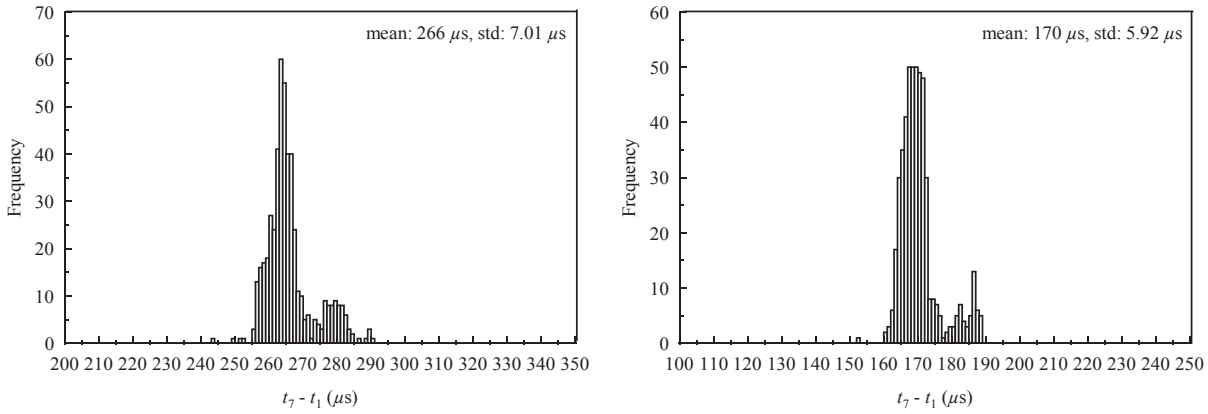


Figure 2.9 Time stamping interval between Q and R ($t_7 - t_1$) recorded by hst using DAG card on the workstations with an integrated NIC: D3000 under (a) 10 Mb/s and (b) 100 Mb/s, and I531S under (c) 10 Mb/s and (d) 100 Mb/s.

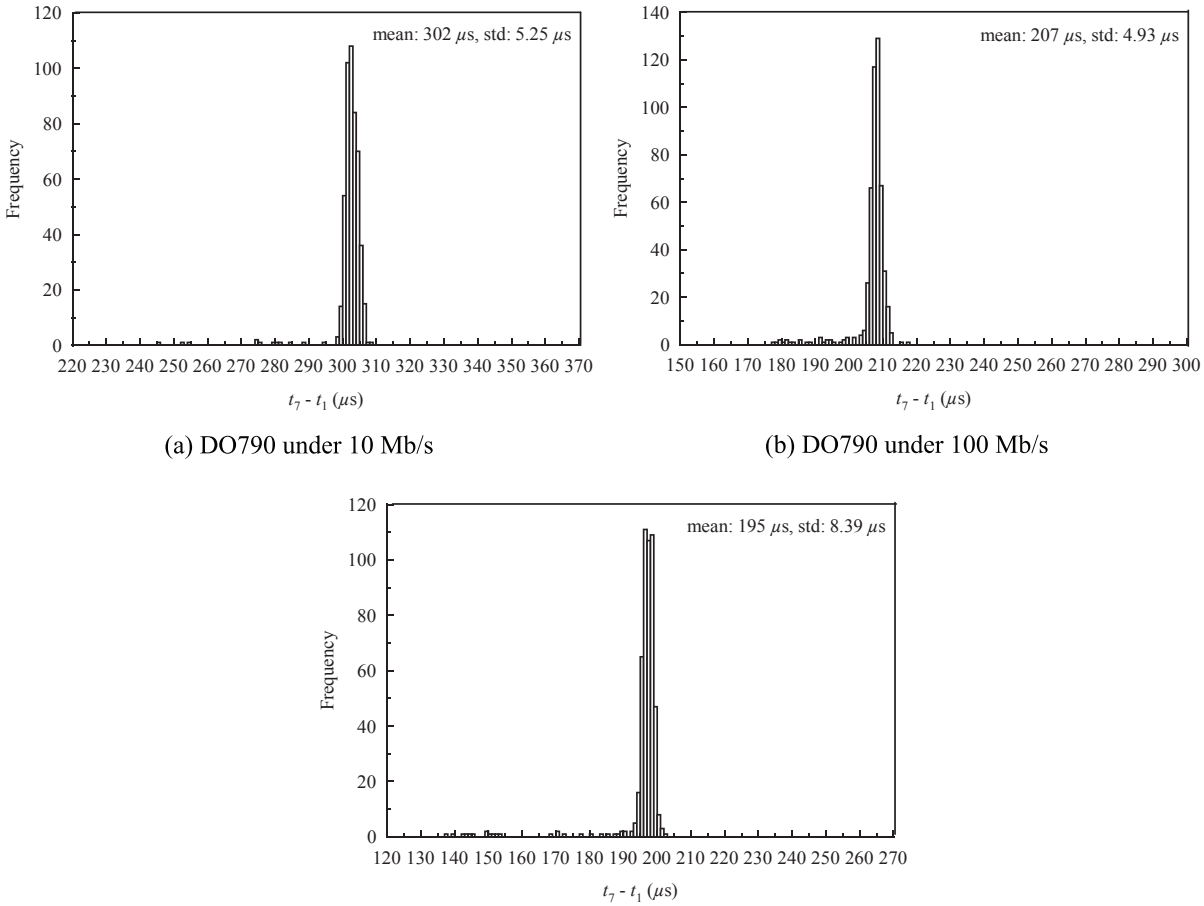


(a) I531S under 10 Mb/s

(b) I531S under 100 Mb/s

(c) I531S under 1000 Mb/s

Figure 2.10 Time stamping interval between Q and R ($t_7 - t_1$) recorded by hst using DAG card on the I531S workstation with an extended NIC under (a) 10 Mb/s, (b) 100 Mb/s, and (c) 1000 Mb/s.



(a) DO790 under 10 Mb/s

(b) DO790 under 100 Mb/s

(c) DO790 under 1000 Mb/s

Figure 2.11 Time stamping interval between Q and R ($t_7 - t_1$) recorded by hst using DAG card on the DO790 workstation with an extended NIC under (a) 10 Mb/s, (b) 100 Mb/s, and (c) 1000 Mb/s.

2.5 Related Work

There is no existing scheme to measure the PPT of an end host to the best of the 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 [15]. 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 [42].

A scheme, called fast-path/slow-path discriminator (*fsd*), was proposed to measure packet generation time of routers using ICMP packets [43]. 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) large enough 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, which is sent 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.

2.6 Conclusions

A scheme to measure the PPT of a host (i.e., workstation) using a specialized packet-capture card in a LAN setup was proposed. To measure PPT, the proposed scheme sends 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 scheme does not require clock synchronization between the host and the packet-capture card. The scheme was experimentally tested on two different hosts connected to the network with an interface running at 10, 100, and 1000 Mb/s. The experimental results show that the proposed methodology can measure PPT of the hosts consistently, and without requiring clock synchronization.

CHAPTER 3

PACKET PROCESSING TIME: MULTIPLE-HOP CASE

3.1 Introduction

An end host under test for PPT measurement may be remotely located and access to the host is only allowed through the Internet over a multiple-hop path. This issue raises the following question: is it possible to measure the PPT of remote end hosts through intranet of large LANs or Internet? As a response to this question, an active scheme to measure the PPT of remote end hosts is proposed. The proposed approach consists of the following three components: 1) An active probing scheme to measure the time an end host takes to time stamp a packet. The scheme is based on the estimation of the capacity of the link directly connected to the receiving end host, called the *end link*, by using *compound probes* (see Figure 3.1(a)), each comprising a heading packet (P_h) and a trailing packet (P_t) [44, 45, 7, 33], sent from the source to the receiving end host, which is the host under test. This compound probe is used to measure the intra-probe gap of each compound probe (i.e., the gap between the last bit of P_h and the last bit of P_t) without deploying any specialized hardware. 2) A methodology to remove the sampled intra-probe gaps affected by the network traffic and other network phenomena. 3) A methodology to obtain the PPT of the host under test from the gaps measured from the compound probes.

The remainder of the chapter is organized as follows: Section 3.2 introduces the proposed scheme to measure PPT of a remote end host. Section 3.3, presents the experimental results and accuracy of the proposed scheme obtained from a testbed and the Internet measurements. Section 3.4 discussed about the quality and values of

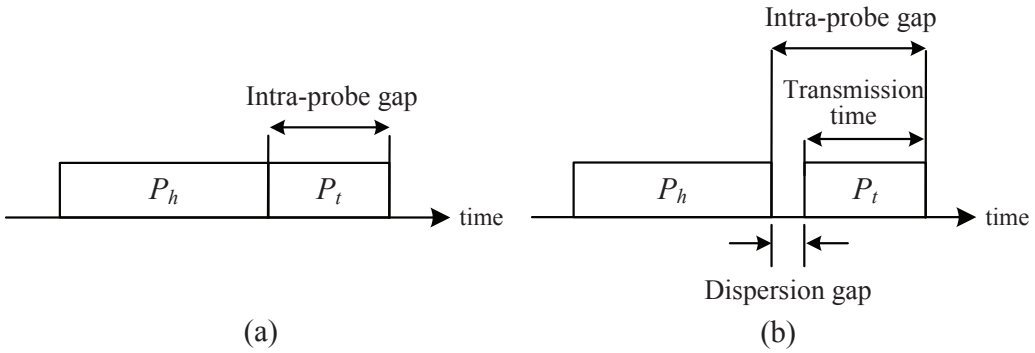


Figure 3.1 Compound-probe structure comprising of a heading (P_h) and a trailing (P_t) packets a) without and b) with a dispersion gap.

the intermediate variables obtained in the PPT experiments. Section 3.7 discusses the parameters that limit the accuracy of the proposed scheme. Section 3.5 introduces an analytical model for the sizing of the compound probe. Section 3.6 presents the existing schemes for link-capacity measurement. Section 3.8 concludes the discussion.

3.2 Scheme for PPT Measurement

In this section, a scheme to measure the PPT of a remote end host connected over a multiple-hop path. The proposed scheme is based on the measurement of intra-probe gaps of two compound probes where the P_ts have different sizes, $s_t = \{s_1, s_2\}$ to estimate the capacity of the remote-end link, c_n, as illustrated in Figure 3.2. The estimated link capacity is used to determine the transmission times of each P_t, which is considered as the expected transmission time. The expected transmission time is compared to the transmission time obtained by time stamping packets at the application layer (through Wireshark [46], for example), and the discrepancy, ΔPPT, is then used to determine PPT at the remote end host.

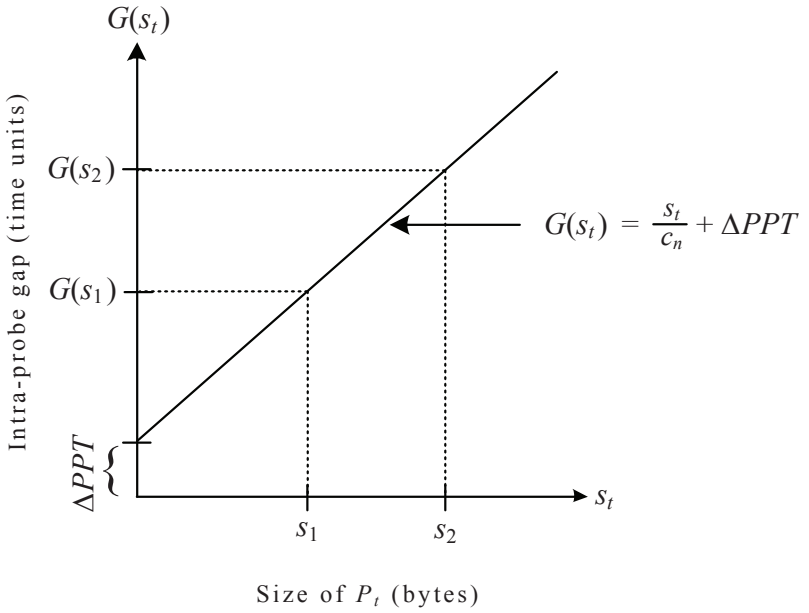


Figure 3.2 Linear relationship between intra-probe gap and trailing-packet size in the estimation of end-link capacity.

As Figure 3.2 shows, the offset ΔPPT provides information about how much delay (which is determined by the CPU and bus speeds, the NIC driver, and the system-call latency [40, 11, 39]) a packet experiences before being time-stamped by the receiving end host. Here, the intra-probe gaps, $G(s_1)$ and $G(s_2)$, are in (linear) function of the transmission times, including the difference of the PPTs of P_h and P_t (i.e., PPT_h and PPT_t, respectively) or

$$G(s_t) = \frac{s_t}{c_n} + \Delta PPT \tag{3.1}$$

where $\Delta PPT = PPT_h - PPT_t$. ΔPPT is defined by the intersection of the straight line, which has a slope of $\frac{1}{c_n}$, and the y axis (i.e., $s_t = 0$).

For generality, PPT_h is considered equal to or different from PPT_t. Figure 3.3 shows the case where PPT_h and PPT_t are different, where TS_h and TS_t are the time stamps for P_h and P_t, respectively, assigned at the application layer.

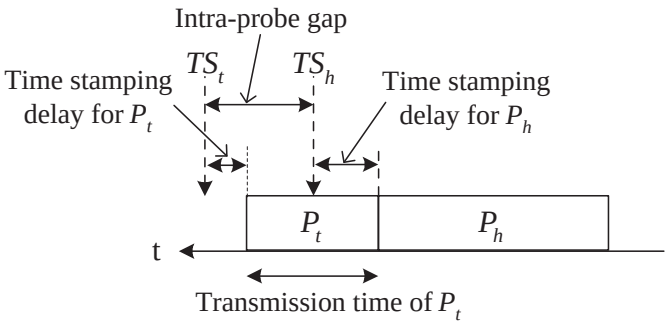


Figure 3.3 Variation in time stamping of P_h and P_t by the application layer.

3.2.1 Measurement Scheme

The proposed methodology to measure PPT is divided into two phases: 1) Estimation of the average ΔPPT, ΔPPT_{avg}, from the measured $G(s_t)$s for each compound probe, shown in Figure 3.5, and 2) Estimation of PPT_h and PPT_t using the packet receiving model, which is shown in Figure 3.6. Considering the multiple-hop path between src and dst in Figure 3.4, the detailed steps of the two phases are presented below.

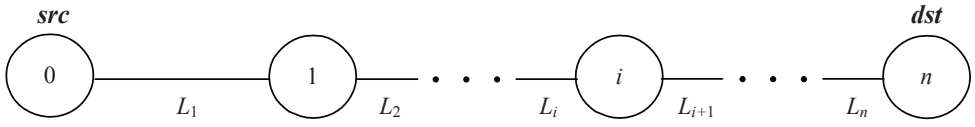


Figure 3.4 An n-hop end-to-end path.

Phase 1: Estimation of ΔPPT_{avg}.

1. Send a train of compound probes from src to dst using a P_h with $s_h = s_{max}$, where s_{max} is equal to the Maximum Transmission Unit (MTU) of the path, and a P_t with $s_1 < s_{max}$ such that the largest possible packet-size ratio, $\alpha = \frac{s_h}{s_t}$, in the compound probe is obtained. As an alternative, s_1 can be estimated if information about the capacity of each link along the path is known, or else, it can be determined by exploration of the path (as discussed in Section 3.5).

2. Send a train of compound probes with $s_h = s_{max}$ and $s_t = s_2$, where $s_1 < s_2 < s_{max}$ and s_2 is slightly larger than s_1. The minimum difference between s_1 and s_2 is determined by the resolution of the clock used in the measurement, or in incremental steps of $2Rc_n$ bits (or bytes), where R is the clock resolution (e.g., if $R = 1$ μs and c_n=100 Mb/s, the step is 25 bytes).

3. Filter out the affected intra-probe gaps and identify the upper and lower bounds, and the average of intra-probe gaps, $G_{low}(s_t)$, $G_{high}(s_t)$, and $G_{avg}(s_t)$, respectively, for s_1 and s_2.

4. Calculate the lower and upper bounds of the reciprocal of the end-link capacity (i.e., slope, see Figure 3.5) using the upper and lower bounds of the measured intra-probe gaps:

$$m_{low} = \frac{G_{high}(s_2) - G_{low}(s_1)}{s_2 - s_1} \tag{3.2}$$

and

$$m_{high} = \frac{G_{low}(s_2) - G_{high}(s_1)}{s_2 - s_1}. \tag{3.3}$$

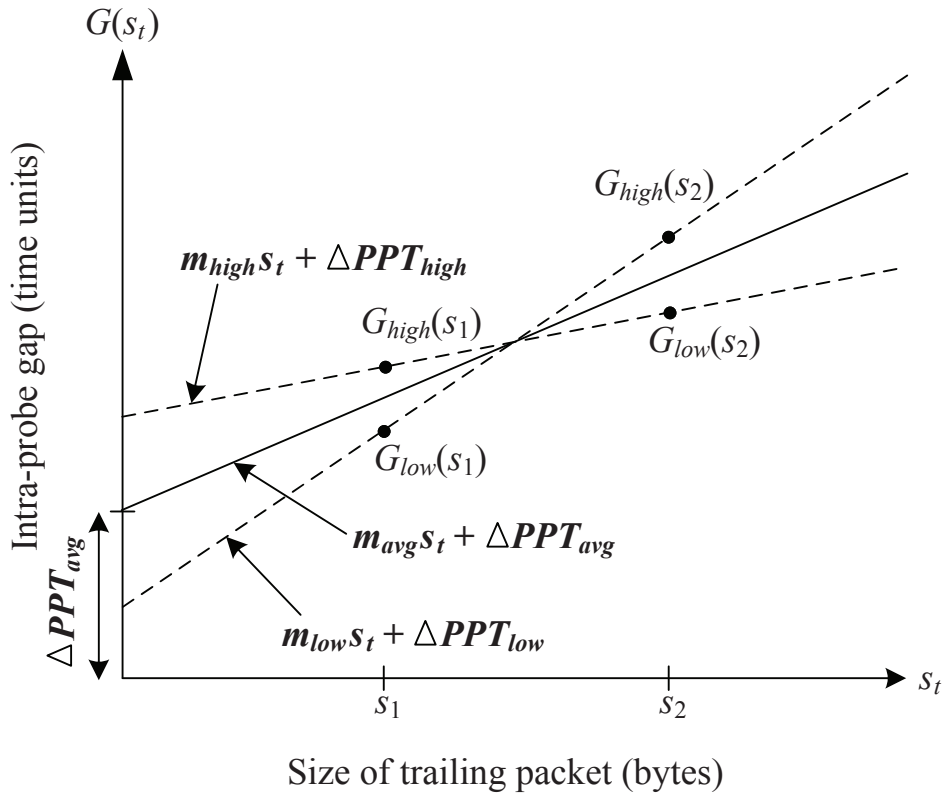


Figure 3.5 Estimation of ΔPPT_{avg} using the lower and upper bounds of intra-probe gaps.

5. Determine the average of the reciprocal of the end-link capacity (m_{avg}):

$$m_{avg} = \frac{m_{low} + m_{high}}{2}. \tag{3.4}$$

6. Calculate the expected intra-probe gaps **($\hat{t}_t$)** of the probing train(s) using m_{avg}:

$$\hat{t}_t(s_1) = m_{avg}s_1 + \gamma \tag{3.5}$$

and

$$\hat{t}_t(s_2) = m_{avg}s_2 + \gamma. \tag{3.6}$$

where γ is the IFG on the end link, if an Ethernet link is connected to the receiving end host.

7. Estimate ΔPPT_{avg} at dst using the difference between $G_{avg}(s_t)$ and $\hat{t}_t(s_t)$ obtained from Steps 3 and 6, respectively:

$$\Delta PPT_{avg} = G_{avg}(s_1) - \hat{t}_t(s_1) \tag{3.7}$$

Phase 2: Estimation of PPTs. The relationships among the transmission times, intra-probe gaps, and PPTs of P_h and P_t are shown in the timing diagram of Figure 3.6 where the receiving buffer at dst uses a first-in-first-out (FIFO) queuing model. The figure shows the timing of the two compound probes with respect to P_h and P_t. $P_t(s_1)$ and $P_t(s_2)$ denote the trailing packets of sizes s_1 and s_2, respectively. Because P_hs in each compound probe have the same length and arrive before P_t, the timing of P_h of both compound probes is overlapped in the figure.

From this model, PPT_ts are determined as:

$$PPT_t(s_1) = PPT_h + G_{avg}(s_1) - t_t(s_1) \tag{3.8}$$

and

$$PPT_t(s_2) = G_{avg}(s_2) - G_{avg}(s_1) - \epsilon, \tag{3.9}$$

where ϵ is the interval between the time stamping of the $P_t(s_1)$ and the arrival time of the $P_t(s_2)$ at the data-link layer.

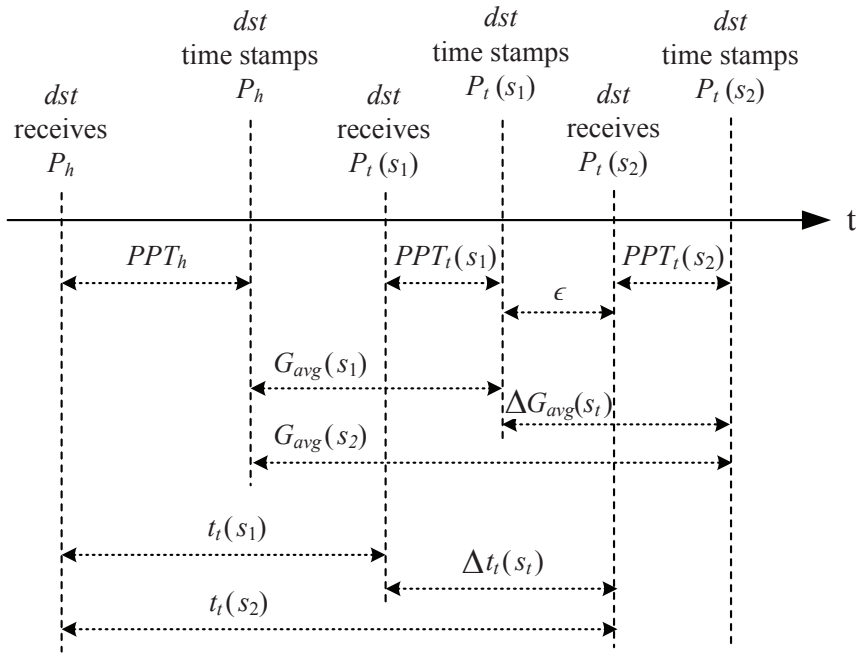


Figure 3.6 Packet-receiving model of heading and trailing packets at dst.

On the other hand, if the intra-probe gaps have zero dispersion gaps and the time-stamping latency at the application layer is smaller than the time to transfer a minimum packet size (e.g., 64 bytes for Ethernet), Equation 3.4 complies with:

$$m_{avg} = \frac{\Delta G_{avg}(s_t)}{s_2 - s_1} = \frac{\Delta t_t(s_t)}{s_2 - s_1} \tag{3.10}$$

or

$$\Delta G_{avg}(s_t) = \Delta t_t(s_t)$$

which shows that PPT is independent of s_t and the difference of the expected transmission times, $\Delta t_t(s_t)$, or the difference of the intra-probe gaps, $\Delta G_{avg}(s_t)$, does not affect Equation 3.4.

According to Figure 3.6, the magnitude of PPT_ts at dst is defined by ϵ, $\Delta t_t(s_t)$, and $\Delta G_{avg}(s_t)$:

$$\Delta t_t(s_t) = PPT_t(s_1) + \epsilon \tag{3.11}$$

and

$$\Delta G_{avg}(s_t) = PPT_t(s_2) + \epsilon \tag{3.12}$$

which lead to equal PPTs for both trailing packets, or

$$PPT_t(s_1) = PPT_t(s_2) = PPT_t \tag{3.13}$$

Equation 3.13 indicates that the magnitude of the two PPT_ts, depends on ϵ, and that the largest PPT_t is found when $\epsilon = 0$, or

$$PPT_t = G_{avg}(s_2) - G_{avg}(s_1) = \Delta G_{avg}(s_t) \tag{3.14}$$

and PPT_h is:

$$PPT_h = PPT_t(s_t) - \Delta PPT. \tag{3.15}$$

3.2.2 Keeping a Zero-Dispersion Gap in a Compound Probe

In the compound probe, P_h and P_t are used to identify the beginning and the end of P_t as time-stamping process records the transmission time of P_t. To determine the transmission time of P_t on the end link, the dispersion gap between P_h and P_t is required to be zero at the time of arrival in the end host. Figure 3.1 shows a compound probe a) without and b) with a dispersion gap. The dispersion gap is defined by the separation between the last bit of P_h and the first bit of P_t.

Gap dispersion (i.e., the increment of the dispersion gap) in a compound probe can occur because of the following two events: 1) one or more cross-traffic packets are inserted between P_h and P_t, or 2) if the packet-size ratio (α) is smaller than the link-capacity ratio of a network node i, $cr_i = \frac{c_{i+1}}{c_i}$, where c_i and c_{i+1} are the capacities of the input link, L_i, and output link, L_{i+1}, respectively, of node i. Because event 1 can occur in a network node where cross traffic flows and it is out of an end-user's reach, the main focus is on on event 2. In a network node, if the transmission time of P_h on the output link is smaller than the transmission time of P_t on the input link, the compound probe experiences dispersion [44],[47, 48]. Therefore, the packet-size ratio between P_h and P_t to keep a zero-dispersion gap or to avoid dispersion in node i must follow:

$$\alpha \geq \frac{c_{i+1}}{c_i}. \tag{3.16}$$

Note that the zero-dispersion gap requirement might be achieved even if the dispersion gap becomes not zero along the path but it is reduced to zero before reaching the end link (e.g., due to a link-capacity ratio smaller than 1). But the aim is keeping a zero-dispersion gap along the path, using a suitable α if link capacities are known. The condition in Equation 3.16 is extended for an n-hop path in Section 3.5.

3.2.3 Filtering of Affected Gaps

The intra-probe gap of a compound probe can be affected by cross-traffic. This affected (i.e., dispersed) intra-probe gap adds errors to the measurement of the end-link capacity. Therefore, a filtering scheme to detect and remove the affected gaps from the collected samples is introduced. In addition to the effect of cross traffic, packet-processing jitter

(i.e., the variations in the receiving-packet process at the end host [12]) may also add errors to the measured intra-probe gaps. To remove those errors, the scheme first identifies the smallest and the most frequent intra-probe gaps in a sampled set to determine the level of packet-processing jitter, and it then calculates the standard deviation of the sampled set to find a range of acceptable (i.e., unaffected by cross traffic) intra-probe gaps. The use of the most-frequent data element has been considered in link-capacity measurement [7], [48, 29]. The following steps filter the affected intra-probe gaps in a sampled set:

1. Identify the smallest intra-probe gap, $G_{small}(s_t)$, of the sampled set.

2. Determine the frequencies of intra-probe gaps (i.e., histogram) in the sampled set.

3. Select the smallest intra-probe gap with the highest frequency $G_{peak}(s_t)$.

4. Estimate the intra-probe gap variations, or packet-processing jitter, as $J = G_{peak}(s_t) - G_{small}(s_t)$, and discard all data elements in the sampled set that are greater than $G_{peak} + J$.

5. Calculate $G_{avg}(s_t)$ and standard deviation σ of the new sampled set.

6. Determine the lower and the upper bounds of the intra-probe gaps as

$$G_{low}(s_t) = G_{avg}(s_t) - \sigma \tag{3.17}$$

and

$$G_{high}(s_t) = G_{avg}(s_t) + \sigma, \tag{3.18}$$

respectively.

The range of the intra-probe gaps $[G_{low}(s_t),\ G_{high}(s_t)]$ depends on the packet-processing jitter (i.e., variation of PPT).

3.3 Experimental Results

The proposed scheme is tested on a controlled testbed, under different network conditions, and in the Internet using the D3000 and I531S workstations. The specifications of these two workstations were presented in Chapter 2. In these experiments, integrated NICs of the above stated workstations have been used to verify the proposed scheme.

3.3.1 Experimental Measurement on a Controlled Testbed

The testbed (Figure 3.7) was implemented using four different configurations, listed in Table 3.1. The testbed configurations, each denoted as Cx-r, where x is the index of each of the four considered configurations and r is the capacity of the end link of configuration x, are C1-10, C2-100, C3-100, and C4-100. These configurations provide variation of the end-link capacities, link-capacity ratios, and location of the narrow link (i.e., the smallest link capacity of the path).

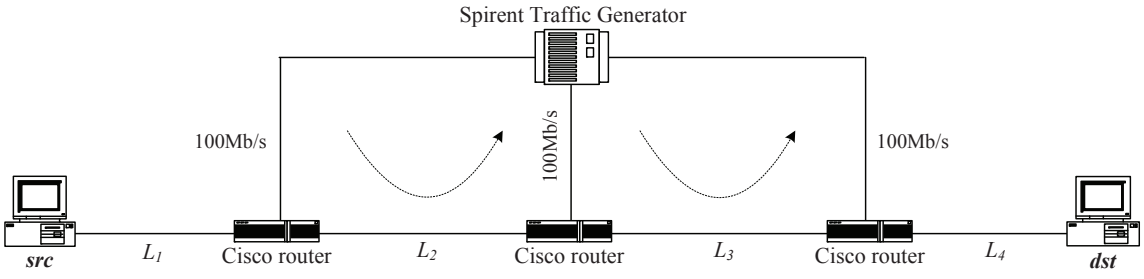


Figure 3.7 Testbed setup.

Table 3.1 Path Configurations on the Testbed

Path	Link capacity (c_i) (Mb/s)				Link-capacity ratio ($cr_i = \frac{c_{i+1}}{c_i}$)			Packet-size ratio ($\alpha = \frac{s_b}{s_t}$)	
	c_1	c_2	c_3	c_4	cr_1	cr_2	cr_3	Calculated value	Evaluated value
C1-10	100	155	100	10	1.55	0.645	0.1	1	1
C2-100	10	155	10	100	15.5	0.064	10	10	10
C3-100	100	10	155	100	0.1	15.5	0.645	6.49	6.67
C4-100	10	10	155	100	1	15.5	0.645	6.49	6.67

The testbed consists of one Cisco 3600 router, two Cisco 7200 routers, a Spirent Smartbits 6000C traffic generator [49], a sender workstation src, and a receiver workstation dst. The D3000 and I531S workstations were used as dst or systems under test. The proposed scheme was implemented as an application on a Linux system, which provides a clock with 1-μs resolution [50] for time stamping (using the pcap library [51]).

On the testbed, symmetrical cross-traffic loads between 0 and 90 Mb/s, with steps of 10 Mb/s, were generated on the second (L_2) and third (L_3) links of the testbed path (indicated by the dotted-line arrows in Figure 3.7). No cross-traffic load was applied to the end links (L_1 and L_4) to avoid having extra CPU load on dst that could bias the PPT measurement. The packet size of each Constant-Bit-Rate (CBR) [52] cross-traffic flow was set between 64 and 128 bytes. The packet sizes are used to generate different levels of traffic loads. It is considered that traffic models with different distributions (e.g., Pareto and Exponential [53]) might not differ significantly from CBR traffic at these high loads.

For the compound probes, s_h was determined by the testbed path's MTU of 1448 bytes of User Datagram Protocol (UDP) payload plus 54 bytes of encapsulation over the Ethernet links (the Ethernet encapsulation includes a 12-byte of preamble, start of frame delimiter, SFD, and frame check sequence, FCS) or a total frame length of 1502 bytes. Two values for s_t, $s_1 = 87$ bytes and $s_2 = 112$ bytes so that these s_ts set large packet-size ratios, $\alpha_1 = \frac{1502 \text{ bytes}}{87 \text{ bytes}} = 17.26$ and $\alpha_2 = \frac{1502 \text{ bytes}}{112 \text{ bytes}} = 13.41$, which are within the lower bounds of the required packet-size ratios on each testbed path configurations, as mentioned in $Calculated\ value$ column of Table 3.1. Further details on the lower bound value of the packet-size ratio are discussed in Section 3.5.3). The time stamps of the probe packets at dst were obtained using Wireshark. Each configuration and workstation was tested using 500 compound probes. Each test was repeated 10 times.

Table 3.2 shows PPT_h and PPT_t of each workstation measured in the testbed experiments (Section 3.4.1 shows the values of all intermediate variables that were used to calculate the PPTs on the testbed). Table 3.2 also shows the average PPT of the measured PPT_h and PPT_t ($\frac{PPT_h+PPT_t}{2}$). According to the table, PPT_{avg}s of the D3000 and I531S workstations with 10-Mb/s end link (on C1-10) are 20 and 21 μs, respectively. The PPT_{avg}s of the D3000 workstation with 100-Mb/s end link are 2, 4, and 3 μs on C2-100, C3-100, and C4-100, respectively. Here, the variation in the PPT_{avg} is attributed to the 1-μs clock resolution of the Linux system. For the I531S workstations with 100-Mb/s end link, PPT_{avg} is 3 μs.

The last column of Table 3.2 shows the errors of the average PPTs measured on the testbed paths in reference to the actual PPTs (PPT_{actual}) of each workstation, where error = $|\frac{(PPT_{actual}-PPT_{avg})}{PPT_{actual}}| \times 100\%$. The actual PPT of each workstation is measured using a specialized packet-capture hardware, Endace DAG 7.5G2 card [36], as discussed in

Table 3.2 Summary of PPTs of Testbed Experiments

dst	Path	Packet processing time				Error
		Actual value	Measured values			
		PPT_{actual} (μs)	PPT_h (μs)	PPT_t (μs)	PPT_{avg} (μs)	(%)
D3000	C1-10	21	21	19	20	5
D3000	C2-100	14	3	1	2	86
D3000	C3-100	14	6	2	4	71
D3000	C4-100	14	5	1	3	79
I531S	C1-10	16	21	20	21	31
I531S	C2-100	7	3	2	3	57
I531S	C3-100	7	3	2	3	57
I531S	C4-100	7	3	2	3	57

Chapter 2. The PPTs of the D3000 and I531S workstations measured using specialized packet-capture card (see Table 2.2) are considered as the ground-truth values for error calculation. On average, the errors of the PPTs measured with the proposed scheme are 5% on a 10-Mb/s link and 79% on a 100-Mb/s link for the D3000 workstation. These values are 31 and 57%, respectively, for the I531S workstation. The results show that the errors for 10-Mb/s links are small. The errors on the 100-Mb/s links are large because of the 1-μs clock resolution of the Linux system.

Besides the above error calculation for the measured PPT, the zero-dispersion gaps in the compound probes has been tested successfully on the testbed configurations. For this,

an Endace DAG 7.5G2 card was used at dst to time stamp P_h and P_t at the data-link layer, and verified the arrival times at the data-link layer (the DAG card time stamps the packet capture locally and without recurring to host processing). This specialized card has a clock resolution of 7.5 ns.

3.3.2 Experimental Measurements over the Internet

The proposed scheme was tested on two Internet paths, an inter-state path in the U.S. and an international path between Taiwan and the U.S., between December 2010 and January 2011. The inter-state path was set from New York Institute of Technology (NYIT), New York, New York, to New Jersey Institute of Technology (NJIT), Newark, New Jersey, and the path is labeled as NYNJ. This path comprises 19 hops. The international path was set between Chaoyang University of Technology (CYUT), Taichung, Taiwan, and NJIT, and it is labeled as TWNJ. This path comprises 21 hops. The workstations at NYIT and CYUT were configured as src nodes, and the nodes at NJIT, the same workstations used in the testbed experiments, as dst nodes. As for the compound probes, s_t kept the same values of 87 and 112 bytes, as in the testbed experiments; however, s_h = 1512 bytes was used because this is a common and a large packet length in the Internet [19]. As in the testbed experiment, trains of 500 compound-probes were also used in each of the 10 measurements. The workstations at both ends were connected to either 10-Mb/s or 100-Mb/s links, and the capacities of the intermediate links along the Internet paths were unknown.

Table 3.3 shows PPT_h, PPT_t, and PPT_{avg} measured in the Internet experiments (Section 3.4.2 discusses the measured intra-probe gaps). The measured PPT_{avg}s of the D3000 and I531S workstations on both the NYNJ and TWNJ paths, with 10-Mb/s end links, are 24 and 21 μs, respectively. The measured PPT_{avg}s of the D3000 workstation on

the NJNY and TWNJ paths with the 100-Mb/s end link are 3 and 4 μs, respectively. The $PPT_{avg}s$ of the I531S workstation on these two Internet paths with the 100-Mb/s end link is 3 μs. The errors of the PPTs measured on the Internet paths in reference to the actual PPTs of the workstations are presented in the last column of Table 3.3. These errors are the same as those obtained on the testbed, except for the D3000 workstation on NYNJ-10 and TWNJ-10 where the error is 9% larger than that on C1-10. This difference between the measurement error on the Internet and testbed paths for the D3000 workstation is produced by the 1-μs clock resolution.

Table 3.3 Summary of PPTs of the Internet Experiments

dst	Path	Packet processing time				Error
		Actual value	Measured values			
		PPT_{actual} (μs)	PPT_h (μs)	PPT_t (μs)	PPT_{avg} (μs)	(%)
D3000	NYNJ-10	21	27	21	24	14
D3000	TWNJ-10	21	27	21	24	14
D3000	NYNJ-100	14	5	1	3	79
D3000	TWNJ-100	14	6	2	4	71
I531S	NYNJ-10	16	21	20	21	31
I531S	TWNJ-10	16	21	20	21	31
I531S	NYNJ-100	7	3	2	3	57
I531S	TWNJ-100	7	3	2	3	57

3.4 Quality of the Measured Variables

In this section, data samples of the measured intra-probe gaps and the summary of intermediates variables in testbed and Internet experiments, respectively, are presented to discuss the quality of the PPT measurements.

3.4.1 Testbed Experiments

1) Quality of Measured Intra-Probe Gaps at dst. Figure 3.8 shows samples of the distributions of the intra-probe gaps measured on C1-10 and C2-100 by dst without cross-traffic load. The theoretical intra-probe gaps (i.e., $\frac{s_t}{c_4} + \gamma$) for the 87- and 112-byte packets on C1-10 are 80 and 100 μs, respectively, and on C2-100 are 8 and 10 μs, respectively. The theoretical intra-probe gaps on each path are indicated by the solid and dashed vertical-lines in each graph of Figure 3.8. Each graph also shows the smallest intra-probe gaps with the highest frequency (G_{peak}) for both trailing-packet sizes measured by the D3000 and I531S workstations. Even though the G_{peak}s are smaller than the theoretical values in each graph, the distributions of the measured gaps show that the workstations are not affected by interrupt coalescence [12]. Otherwise, the measured intra-probe gaps for both trailing-packet sizes would have similar distributions, with a single G_{peak}.

2) Quality of Intra-probe Gap Measurements at dst under Cross-traffic Effect. Figure 3.9 shows the measured intra-probe gaps (without filtering the affected gaps) on C1-10 and C2-100 with 60% cross-traffic load. In this figure, the distributions of the intra-probe gaps are similar to those in Figure 3.8. However, Figures 3.9(b) and 3.9(d) show some outliers (i.e., large intra-probe gaps at the right-hand side of each graph) for the 87- and 112-byte

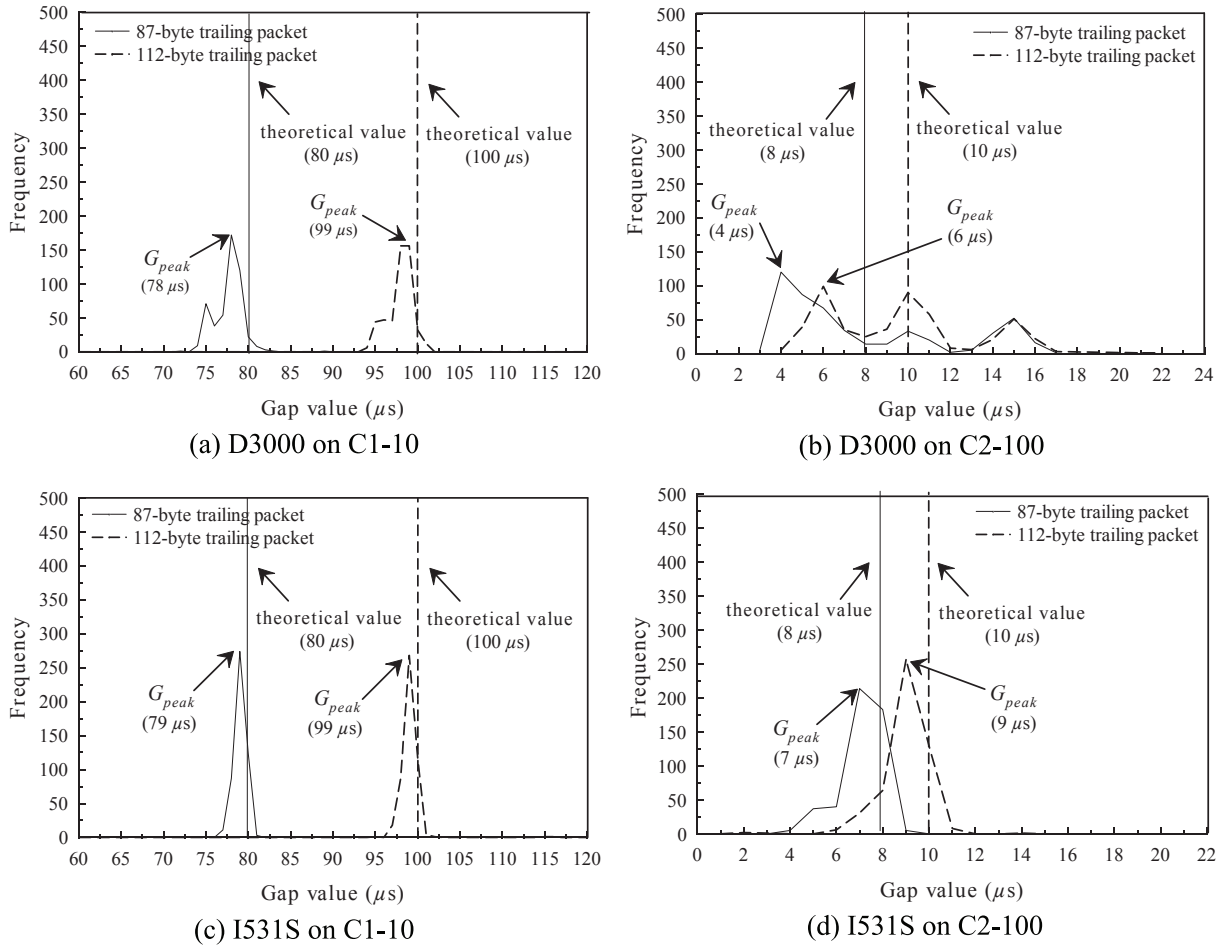


Figure 3.8 Distributions of intra-probe gaps, with no cross-traffic load in the network, measured by the D3000 workstation on: (a) C1-10 and (b) C2-100, and by the I531S workstation on: (c) C1-10 and (d) C2-100.

trailing packets as compared to those of Figures 3.8(b) and 3.8(d). These are gaps affected by cross traffic.

Figure 3.10 shows the distributions of the filtered intra-probe gaps of the measurement samples of Figure 3.9. This figure also shows G_{peak}, and the average intra-probe gap, G_{avg}, for both trailing-packet sizes. In each graph, the filtered gaps are clustered around G_{avg} with one standard deviation, according to Section 3.2.3. The distribution of the filtered gaps show that the proposed filtering scheme eliminates the outliers caused by the cross traffic.

4) Summary of Intra-probe Gap Measurements at dst. Table 3.4 shows the measured values of the intermediate variables, $G(s_t)_{avg}$, m_{avg}, $\hat{t}_t$, and ΔPPT_{avg}, which are used to calculate the PPTs of the D3000 and I531S workstations in the testbed experiments. According to the table, the measured ΔPPT_{avg} of the D3000 and I531S workstations on C1-10 are -2 and -1 μs (the negative sign means that $PPT_h > PPT_t$), respectively. The measured ΔPPT_{avg} of these workstations on C2-100, C3-100, and C4-100 are -2, -4, and -3 μs, respectively, for the D3000 workstation, and -1 μs for the I531S workstation.

The average slopes (m_{avg}) measured by the D3000 and I531S workstations on C1-10 are 0.78 and 0.8, respectively, and the actual slope (i.e., the expected slope) is 0.8, since c_4 = 10 Mb/s. In the cases of C2-100, C4-100, and C4-100, the actual slope is 0.08 (since c_4 = 100 Mb/s) and the values measured by the D3000 workstation are 0.06 on C2-100, and 0.08 on both C3-100 and C4-100. The slope measured by the I531S workstation is 0.08 for all path configurations. Table 3.4 shows that the proposed scheme measures the end-link capacity of each path with high accuracy.

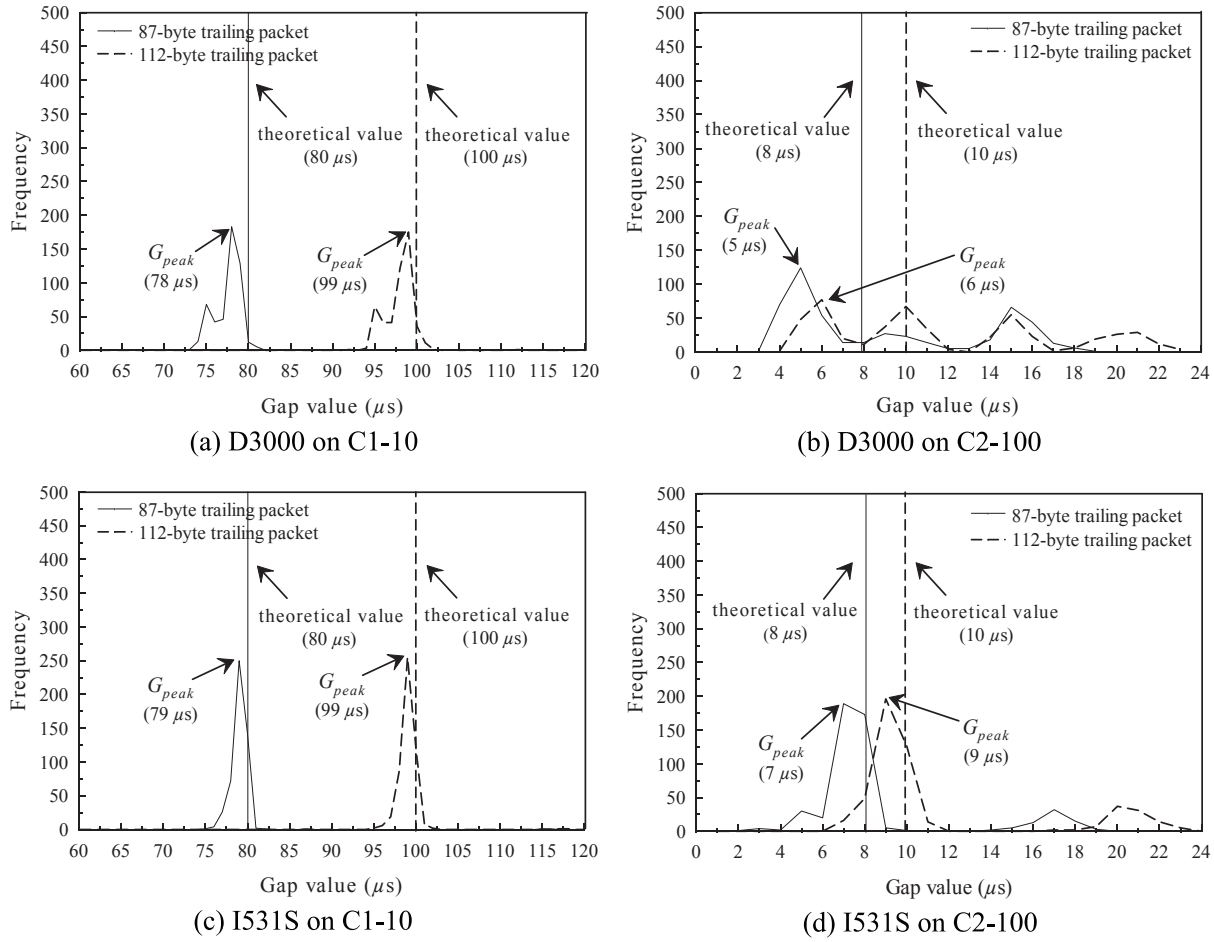


Figure 3.9 Distributions of intra-probe gaps under 60% cross-traffic load measured by the D3000 workstation on: (a) C1-10 and (b) C2-100, and by the I531S workstation on: (c) C1-10 and (d) C2-100.

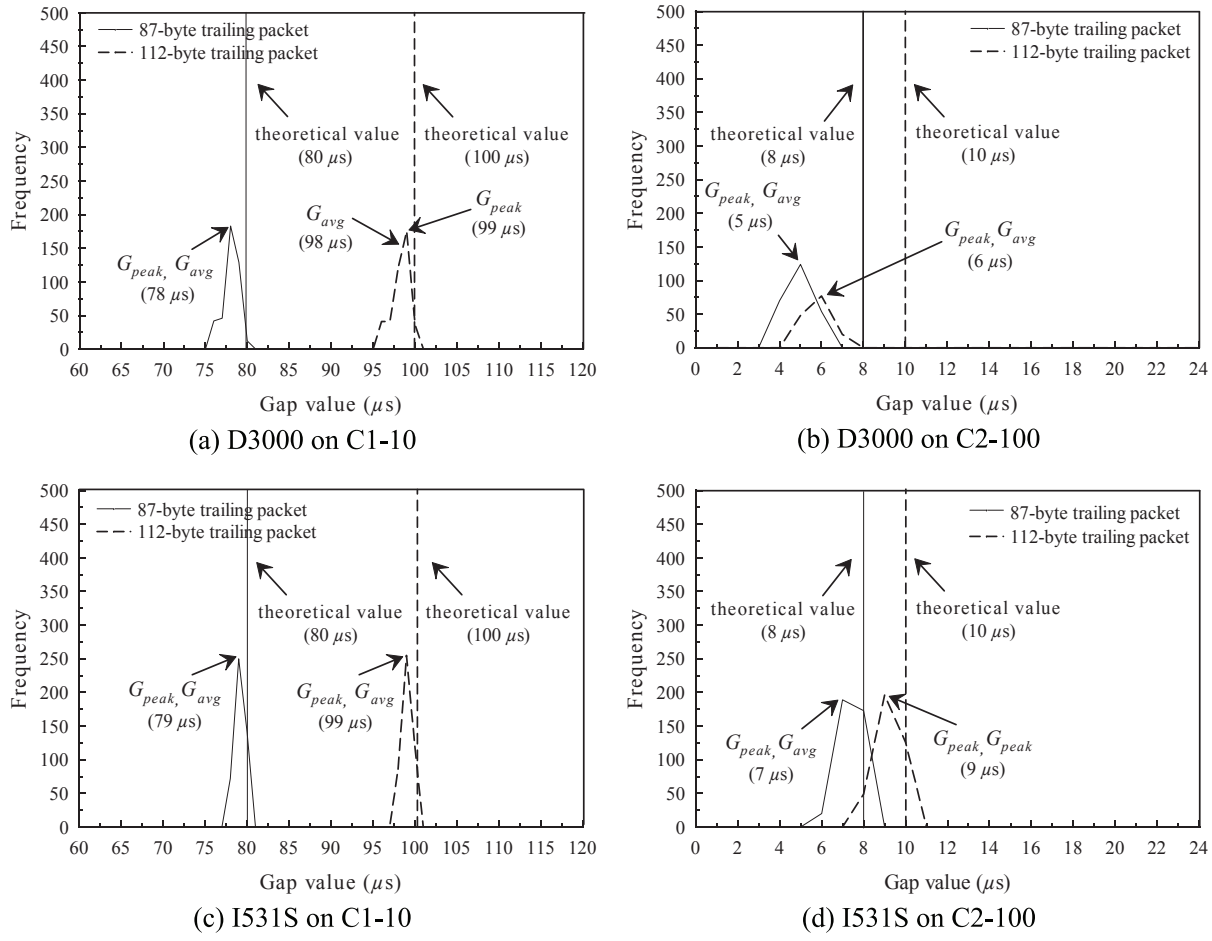


Figure 3.10 Distributions of filtered intra-probe gaps under 60% cross-traffic load measured by the D3000 workstation on: (a) C1-10 and (b) C2-100, and by the I531S workstation on: (c) C1-10 and (d) C2-100.

Table 3.4 Summary of Intra-probe Gaps of Testbed Experiments

dst	Path	Packet size s_t	Intra-probe gap $G(s_t)$ $[low,\ high]$	avg	Δ	Slope value m $[low,\ high]$	avg	Expected transmission time $\hat{t}_t$	Processing time ΔPPT avg	std
D3000	C1-10	87	[77, 80]	79	19	[0.92, 0.64]	0.78	77.46	-2	1.5
		112	[96, 100]	98				96.96		
D3000	C2-100	87	[4, 5]	5	1	[0.12, 0]	0.06	6.18	-2	0.7
		112	[5, 7]	6				7.68		
D3000	C3-100	87	[3, 5]	4	2	[0.16, 0]	0.08	7.92	-4	0.5
		112	[5, 7]	6				9.92		
D3000	C4-100	87	[4, 6]	5	1	[0.12, -0.04]	0.08	7.92	-3	1.7
		112	[5, 7]	6				9.92		
I531S	C1-10	87	[78, 80]	79	20	[0.88, 0.72]	0.8	79.2	-1	0.9
		112	[98, 100]	99				99.2		
I531S	C2-100	87	[6, 8]	7	2	[0.16, 0]	0.08	7.92	-1	1.0
		112	[8, 10]	9				9.92		
I531S	C3-100	87	[6, 8]	7	2	[0.16, 0]	0.08	7.92	-1	1.0
		112	[8, 10]	9				9.92		
I531S	C4-100	87	[6, 8]	7	2	[0.16, 0]	0.08	7.92	-1	1.0
		112	[8, 10]	9				9.92		

3.4.2 Internet Experiments

1) Quality of Intra-Probe Gap Measurements at dst. Figures 3.11 and 3.12 show samples of the intra-probe gaps, measured by the dst nodes (D3000 and I531S workstations) on the Internet paths with 10- and 100-Mb/s end links. The graphs in Figure 3.11 show the distributions of intra-probe gaps on the NYNJ path and the graphs in Figure 3.12 show the intra-probe gap distributions on the TWNJ path. The distributions of the intra-probe gaps

measured by both workstations on the Internet paths are similar to those measured on the testbed.

2) Summary of Intra-probe Gap Measurements at dst. The intermediate variables measured on the Internet paths are shown in Table 3.5. ΔPPT_{avg} of the D3000 workstation on the 10- and 100-Mb/s end links are -6 and -4 μs, respectively. ΔPPT_{avg} of the I531S workstation on both the 10- and 100-Mb/s end links are -1 μs. As for m_{avg}, the values measured by the D3000 workstation on the Internet paths are 0.84 and 0.08 when the end-link capacities are 10- and 100-Mb/s, respectively. These values are measured as 0.8 and 0.08 on the respective end-link capacities by the I531S workstation. These values show that the proposed scheme can also measure the end-link capacity in the Internet with a high accuracy.

3.5 Packet-Size Ratio for Keeping Zero-Dispersion Gap

In the presented PPT experiments in the Internet, a very large packet-size ratios was adopted as many of the link capacities along the path were unknown. However, the size of the packets in the compound probe can be determined, as described in this section, if the link capacities of the complete path are known.

The sizes of P_h and P_t in the compound probe are defined by the capacities of the links of an end-to-end path such that a zero-dispersion gap is obtained at the end link. Cross traffic may also affect the dispersion in the compound probe. A pair of packets is considered to be operating in joint queuing region (JQR) at at a network node (e.g., router or workstation) when both the packets are available at the output queue of the node simultaneously [54]. However, the JQR state for the compound probe is not sufficient to ensure a zero-dispersion gap since P_h and P_t must be available back-to-back at the output

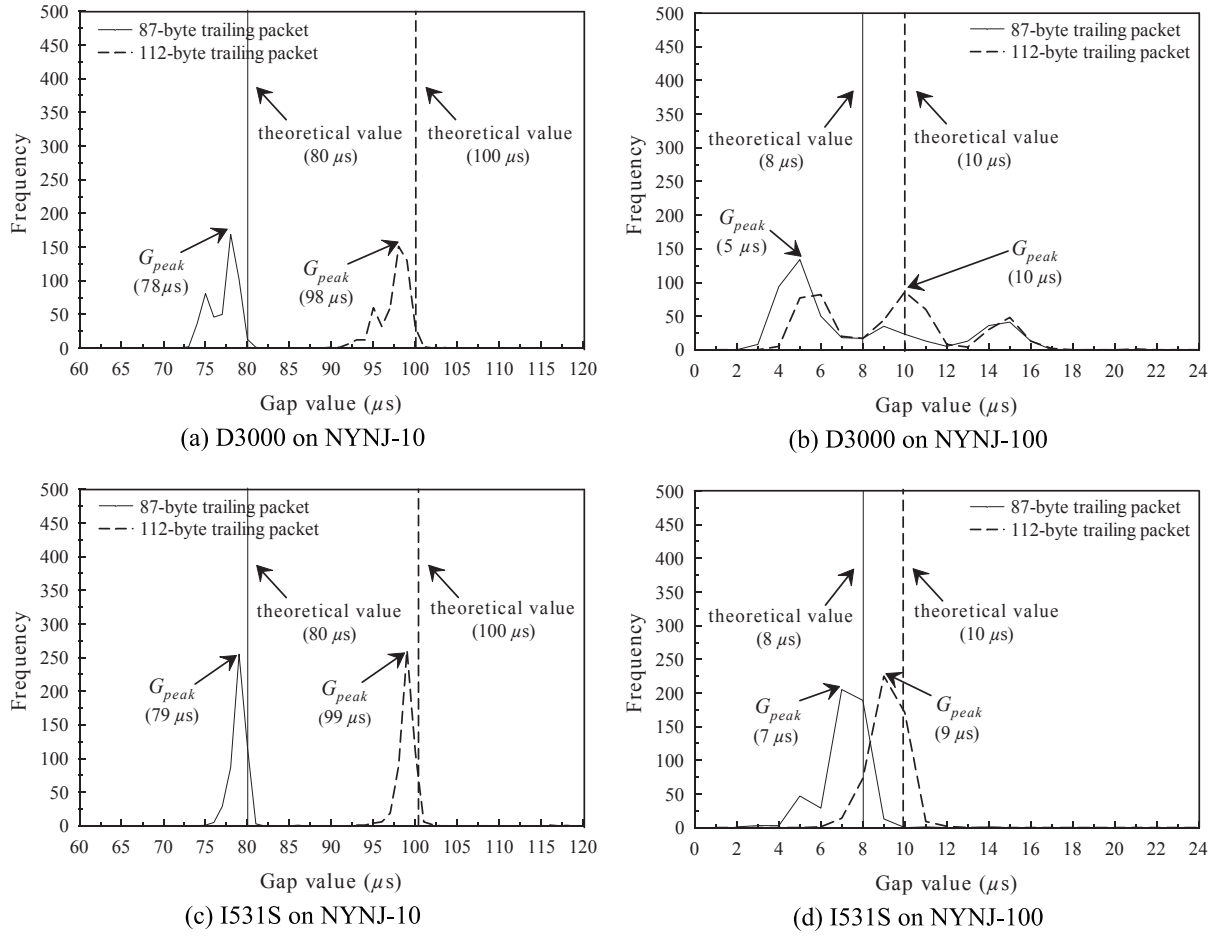


Figure 3.11 Distributions of intra-probe gaps measured at dst (NJIT) on NYNJ path: by the D3000 workstation with (a) 10-Mb/s end link and (b) 100-Mb/s end link, and by the I531S workstation with (c) 10-Mb/s end link and (d) 100-Mb/s end link.

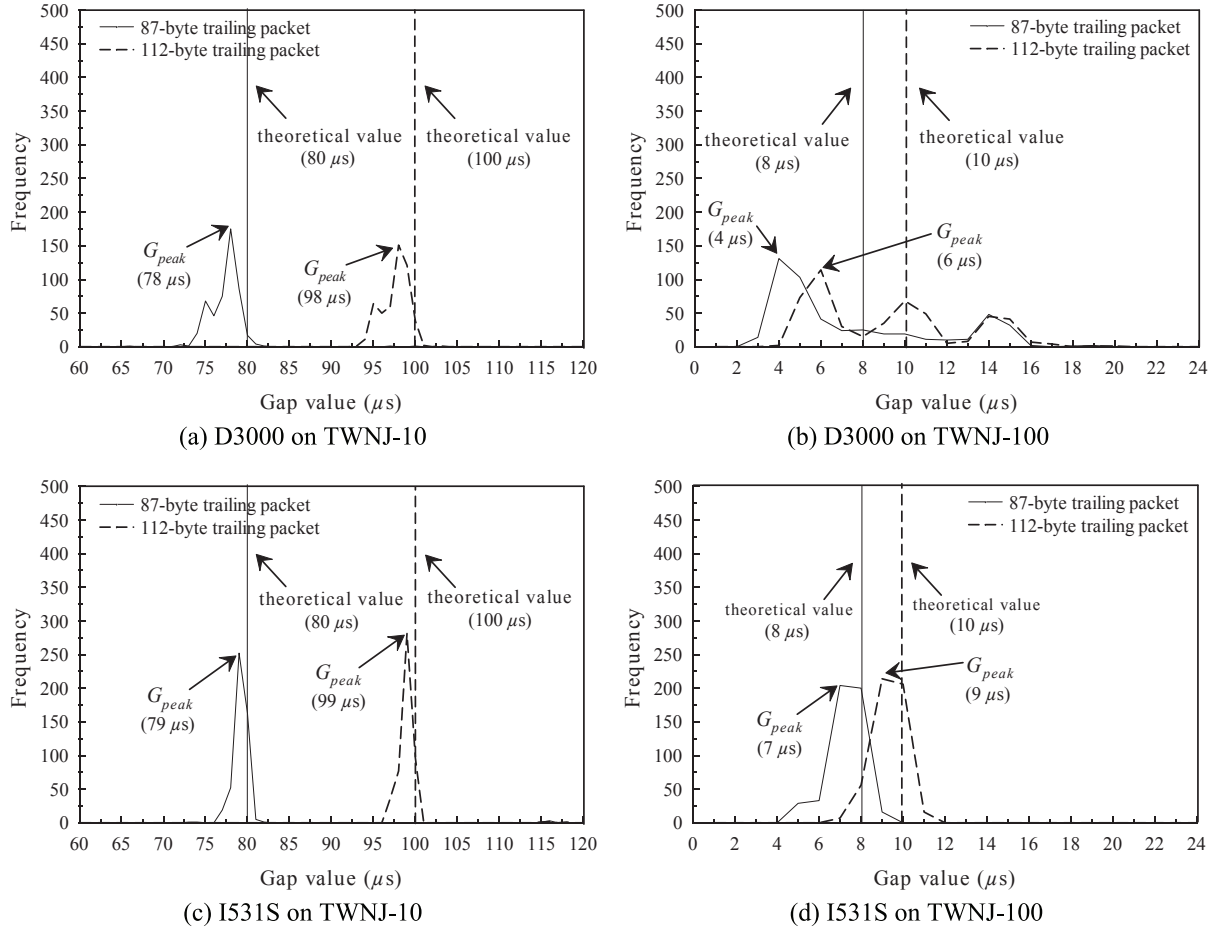


Figure 3.12 Distributions of intra-probe gaps measured at dst (NJIT) on TWNJ path: by the D3000 workstation with (a) 10-Mb/s end link and (b)100-Mb/s end link, and by the I531S workstation with (c) 10-Mb/s end link and (d) 100-Mb/s end link.

Table 3.5 Summary of Intra-probe Gaps of the Internet Experiments

dst	Path	Packet size s_t	Intra-probe gap $G(s_t)$ $[low,\ high]$	avg	Δ	Slope value m $[low,\ high]$	avg	Expected transmission time $\hat{t}_t$	Processing time ΔPPT avg	std
D3000	NYNJ-10	87	[75, 79]	77	21	[1, 0.68]	0.84	82.68	-6	1.6
		112	[96, 100]	98				103.68		
D3000	TWNJ-10	87	[75, 79]	77	21	[1, 0.68]	0.84	82.68	-6	1.6
		112	[96, 100]	98				103.68		
D3000	NYNJ-100	87	[4, 6]	5	1	[0.12, -0.04]	0.08	7.92	-4	1.5
		112	[5, 7]	6				9.92		
D3000	TWNJ-100	87	[3, 5]	4	2	[0.16, 0]	0.08	7.92	-4	0.7
		112	[5, 7]	6				9.92		
I531S	NYNJ-10	87	[78, 80]	79	20	[0.72, 0.88]	0.8	79.2	-1	1.3
		112	[98, 100]	99				99.2		
I531S	TWNJ-10	87	[78, 80]	79	20	[0.72, 0.88]	0.8	79.2	-1	0.8
		112	[98, 100]	99				99.2		
I531S	NYNJ-100	87	[6, 8]	7	2	[0, 0.16]	0.08	7.92	-1	0.9
		112	[8, 10]	9				9.92		
I531S	TWNJ-100	87	[6, 8]	7	2	[0, 0.16]	0.08	7.92	-1	0.8
		112	[8, 10]	9				9.92		

queue without having cross-traffic packets(s) in between them. A suitable packet-size ratio, α, can be used to make the compound probe immune to the heterogeneous link capacities and to a level of interference by cross traffic of an end-to-end path [33]. These issues are analyzed in this section.

3.5.1 Sizing of Probing Packets and Link Heterogeneity

Consider that the capacities of the links $L_1, L_2, \ ..., L_n$ between node 0 (src) and node n (dst) along an n-hop path are $c_1, \ ..., \ c_n$, as shown in Figure 3.4. To keep a zero-dispersion gap between P_h and P_t at dst when the compound probe is sent from src of the above stated end-to-end path, the following equation applies:

$$(\frac{s_h}{c_n} - \frac{s_h}{\alpha l_{n-1}}) + (\frac{s_h}{c_{n-1}} - \frac{s_h}{\alpha c_{n-2}}) + ... + (\frac{s_h}{c_{z+1}} - \frac{s_h}{\alpha c_z}) = 0 \tag{3.19}$$

where $\frac{s_h}{\alpha} = s_t$, and c_z is the capacity of a link connected to a node z, such that $cr_z = \frac{c_{z+1}}{c_z}$ is the largest link-capacity ratio along the path, located after the narrow link (in the direction from src to dst) and that c_z also is the link closest to dst (e.g., if two nodes following the narrow link closest to dst of the path have the largest link-capacity ratio, the node located the closest to dst is selected). Therefore, the index z is such that $1 \leq z \leq (n-1)$. Here, Equation 3.19 is valid as long as the narrow link is not the end link of the path.

The largest size of P_t is calculated from Equation 3.19 as:

$$s_t = s_h \frac{\sum_{j=z+1}^{n} \frac{1}{c_j}}{\sum_{j=z}^{n-1} \frac{1}{c_j}} \tag{3.20}$$

When the end-link capacity c_n is the path's narrow link, the required condition to achieve a zero-dispersion gap is:

$$\frac{s_h}{c_n} - \frac{s_h}{\alpha c_{n-1}} = 0 \tag{3.21}$$

and the largest size of P_t is calculated as:

$$s_t = \frac{s_h}{c_n} c_{n-1} \tag{3.22}$$

The rationale behind Equations 3.19 and 3.21 is that a compound probe may experience the largest dispersion at a network node over a path where the link-capacity ratio is the largest and the packet-size ratio is smaller than the largest link-capacity ratio. Therefore, s_t is determined with respect to s_h by finding a suitable packet-size ratio so that the dispersion gap reduces to zero upon arriving at node n.

3.5.2 Sizing of Probing Packet and Cross-Traffic Effect

Figure 3.13 shows an example of a compound probe forwarded from the input link to the output link by node i when two cross-traffic packets, denoted by D_h and D_t, intrude the compound probe. The capacity of the input and output links are c_i and c_{i+1}, respectively, where $c_i < c_{i+1}$ in this example. Here, P_h and P_t arrive at node i with a zero-dispersion gap. However, the compound probe experiences dispersion at the output link, as shown by the dispersion gap in the figure.

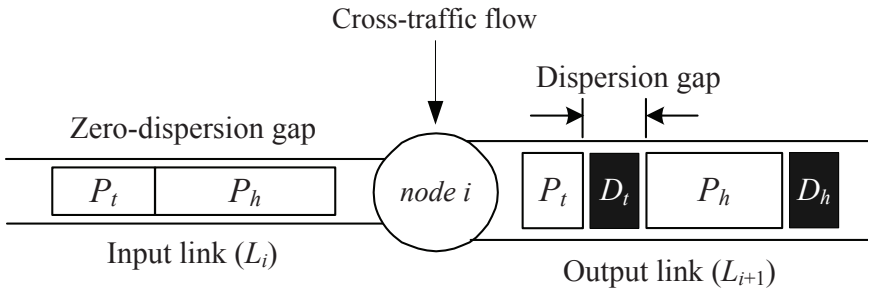


Figure 3.13 Forwarding of a compound probe by node i from its input link to its output link.

The intra-probe gap of the compound probe at the output link of node i is defined as:

$$G(s_t)_{i+1} = \frac{s_t}{c_{i+1}} + \delta_{i+1}, \tag{3.23}$$

where the dispersion gap at the output link, δ_{i+1}, is:

$$\delta_{i+1} = \begin{cases} \delta_i - \Delta tr(h,t)_i & \text{if } \delta_i - \Delta tr(h,t)_i > Qt_i \\ Qt_i & \text{else} \end{cases} \tag{3.24}$$

In the above equation, δ_i is the dispersion gap at the input link of node i, $\Delta tr(h,t)_i$ is the difference between the transmission time of P_h plus the queuing delay (Qh_i) caused by the cross-traffic packet(s) backlogged ahead of P_h at the output link and the transmission time of P_t at the input link of node i. Qt_i is the increment of the dispersion gap caused by the cross-traffic packet(s) inserted between P_h and P_t at node i. These terms are estimated as:

$$\Delta tr(h,t)_i = \frac{s_h}{c_{i+1}} + Qh_i - \frac{s_t}{c_i} \tag{3.25}$$

$$Qh_i = \sum_u \frac{\zeta_{u(i)}}{c_{i+1}};\ u \geq 0 \tag{3.26}$$

$$Qt_i = \sum_v \frac{\zeta_{v(i)}}{c_{i+1}};\ v \geq 0 \tag{3.27}$$

Here, $\zeta_{u(i)}$ and $\zeta_{v(i)}$ denote the sizes of the cross-traffic packets u and v ahead of P_h and P_t for Qh_i and Qt_i, respectively, at node i.

When node n of Figure 3.4 measures the intra-probe gap of the compound probe upon receiving it at the application layer, the measured intra-probe gap includes the IFG of the end link and it follows Equation 3.23:

$$G(s_t)_n = \frac{s_t}{c_n} + \delta_n + IFG \tag{3.28}$$

3.5.3 Numerical Evaluations of Packet-size Ratio

The upper bounds of α (i.e., the largest possible values of s_t against s_h) needed to keep the dispersion gap at zero using either Equation 3.20 or 3.22 over a four-hop end-to-end path, considering different configurations for the link capacities, is shown in Table 3.1. To verify the calculated values, the values of α was evaluated using the dispersion-gap model in Section 3.5.2, considering Qt_i and Qh_i equal to zero in Equations 3.24 and 3.25. To evaluate with different αs, the initial value of s_t is set to 50 bytes and it is gradually increased by 25 bytes until the length of s_h = 1500 bytes is reached. The last two columns of Table 3.1 show the lower bounds of both the calculated and evaluated values of α for each configuration, where both the calculated and evaluated values were obtained by using a single compound-probe over each path configuration. As Table 3.1 shows, the evaluated values are very close to the calculated ones; the calculated values of α over C1-10 to C4-100 are 1 (s_t = 1500 bytes), 10 (s_t = 150 bytes), 6.49 (s_t = 232 bytes), and 6.49 (s_t = 232 bytes), respectively. The evaluated values of α are 1 (s_t = 1500 bytes), 10 (s_t = 150 bytes), 6.67 (s_t = 225 bytes), and 6.67 (s_t = 225 bytes) over C1-10, C2-100, C3-100, and C4-100, respectively. The small over estimation in the evaluated link ratios for C3-100 and C4-100 are produced by the 25-byte step increase of s_t, as stated above.

Figures 3.14(a)-3.14(d) show detailed information about the estimated end-link capacities over each path configuration using different values of α, after considering $IFG = 0$ in Equation 3.28. The acceptable packet-size ratios are those representing the actual end-link capacity, as indicated by the arrows in the figures (where the zero-dispersion gap can be kept).

The impact of cross traffic in the measurement of end-link capacity was evaluated considering a symmetric cross-traffic load over the second and third links of each path configurations to emulated the testbed scenario of Figure 3.7. In this evaluation, the same values of α, as used in the above evaluation, were used along with three different values for u and v, i.e., 1, 5, and 10 packets, each packet is 64 bytes, in Equations 3.26 and 3.27. Figures 3.15(a)-3.15(d) show the estimated end-link capacities over C1-10 to C4-100, respectively. The estimated end-link capacities show that the cross-traffic interference increases the upper bounds of α for accurate link-capacity measurement except over C1-10, where the end link is the narrow link of the path. For example, when P_h and P_t of the compound probe is interfered by a single cross-traffic packet (i.e., when $u = 1$ and $v = 1$), the upper bound of α increases from 10 to 20 (i.e., s_t decreases to 75 bytes) over C2-100, and from 6.67 to 8.57 (i.e., s_t decreases to 175 bytes) over both C3-100 and C4-100, as shown in Figures 3.15(b)-3.15(d), respectively. These graphs show that if the narrow link is other than the end link, cross traffic produces dispersion. Another observation is that a large α simplifies the selection of the sizes of P_h and P_t. In this case, the largest value of α achievable on Ethernet is sufficient to achieve a zero-dispersion gap in the considered path configurations.

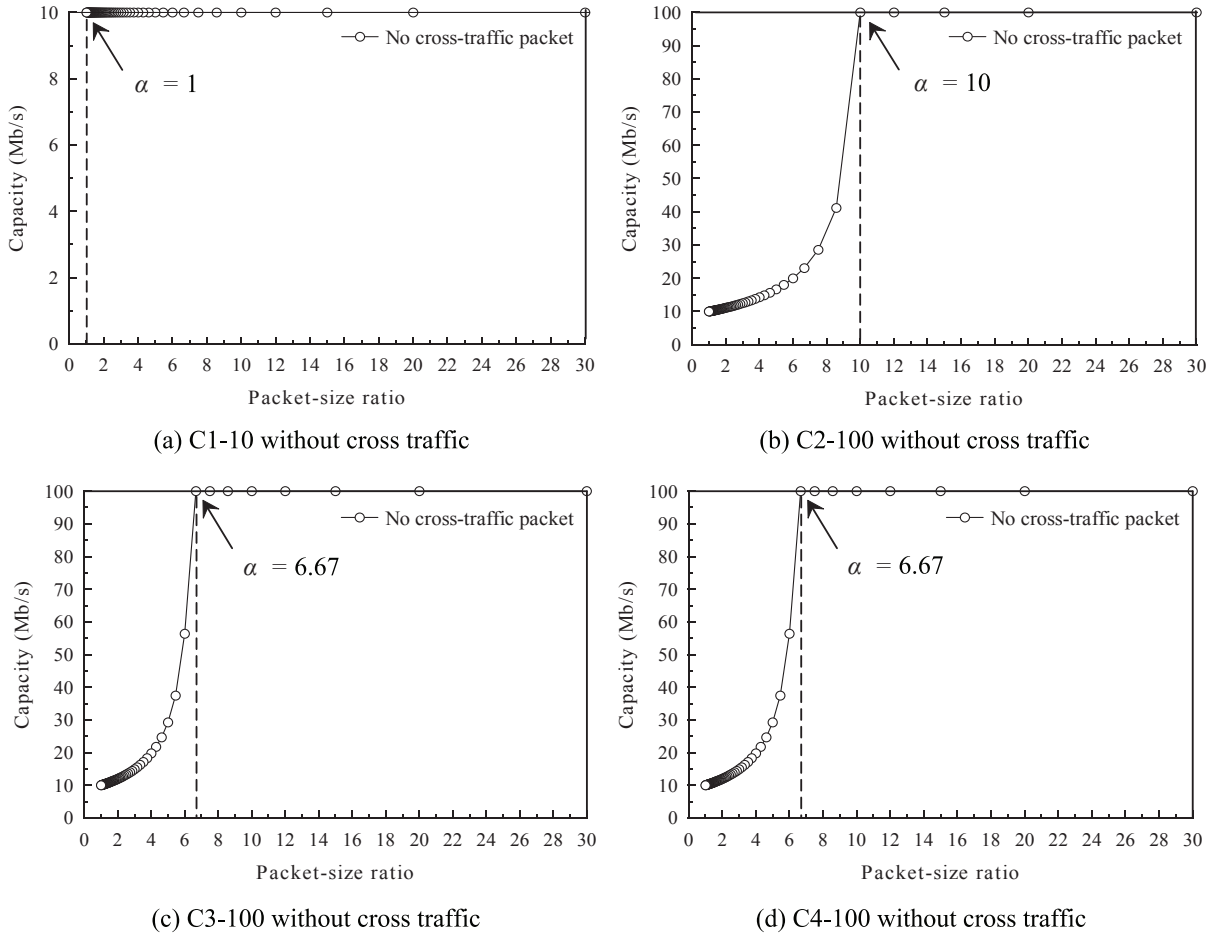
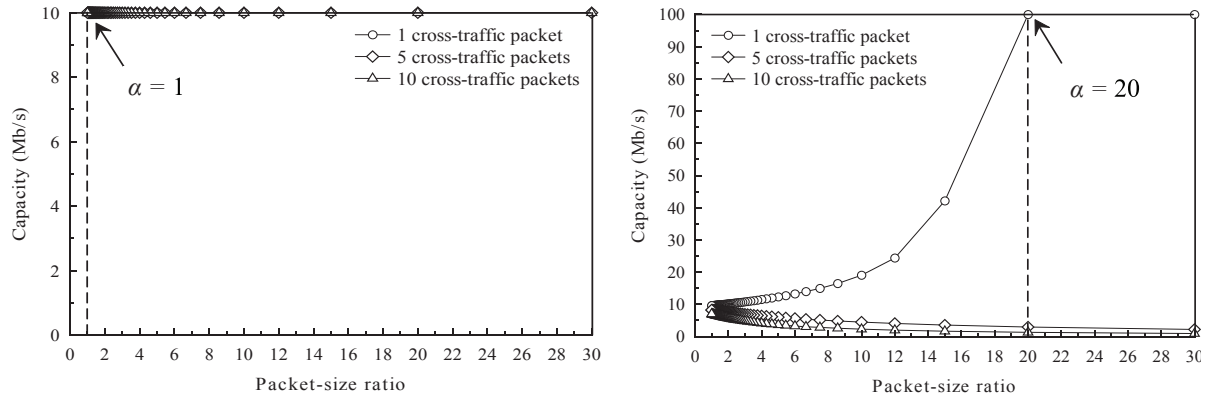
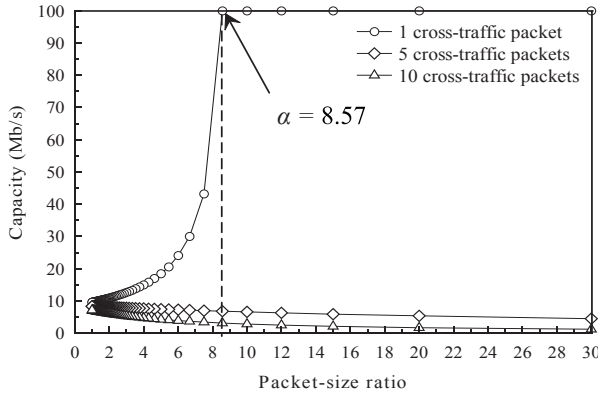


Figure 3.14 Evaluated packet-size ratios to measure end-link capacity (c_4) on four different path configurations: (a) C1-10, (b) C2-100, (c) C3-100, and (d) C4-100 without cross traffic.

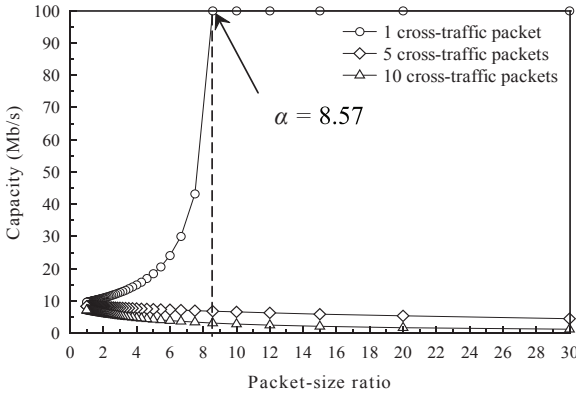


(a) C1-10 with cross traffic

(b) C2-100 with cross traffic



(c) C3-100 with cross traffic



(d) C4-100 with cross traffic

Figure 3.15 Evaluated packet-size ratios to measure end-link capacity (c_4) on four different path configurations: (a) C1-10, (b) C2-100, (c) C3-100, and (d) C4-100 with cross traffic.

3.6 Related Work on Measurement of Link Capacity

A plethora of active schemes to measure link capacity has been proposed [55, 29, 30, 56, 31, 57, 58, 59, 60, 61, 44, 48, 32]. In the measurement of PPT, the aim was at measuring the capacity of the end link, therefore, the following discussion limits the discussion to the schemes that may be applicable to this case.

The link-capacity measurement schemes can be coarsely classified based on performing hop-by-hop [58, 60, 59, 44, 32] and specific-link [48] measurement for this particular case. Pathchar, an early hop-by-hop scheme to measure link capacity, is based on measuring the round-trip time, RTT (the traveling times of a packet sent from a source to the destination node plus the traveling time on the opposite direction) [58]. In this scheme, the network node (i.e., router) directly connected to the source node is used as the first destination, and the capacity of the first link is obtained. The same procedure is applied to the nodes farther away from the source until the remote end host is reached. However, the single packet approach and the measurement of RTTs have been reported to produce large errors [12]. There are other schemes that follow similar approach to Pathchar, with, however, different statistical analysis, aimed at reducing probing load [60, 59].

A subsequent hop-by-hop scheme, called Nettimer, uses an approach based on the accumulated delay of an end-to-end path [44]. Nettimer uses a tailgating technique, where two probing packets are sent back-to-back, with a small packet tailgating a large packet, to identify the contribution to the accumulated delay from each single link in the path. The large packet is dropped right before reaching the link of interest and the small packet is left to continue towards the end node. The scheme is based on measuring the delays of the small packet. However, the measurement of the path delay using the tailgating technique is affected by large errors in the estimation of link capacity due to cross traffic [44].

Another hop-by-hop scheme, which is resilient to cross traffic and that produces a small probing load, was recently proposed [32]. The scheme is also based on the tailgating technique but each probing packet consists of two small ICMP packets behind a large data packet, and all sent back-to-back. The time stamps of the ICMP packets, provided by the destination node, are used to measure the gap in each probing packet and the measure gaps are used to estimate the link capacity. Even though the measurement accuracy of the scheme is high, the low resolution of ICMP time stamps (i.e., 1 ms [62]) bounds its applicability to slow link rates.

Different from hop-by-hop measurement, a scheme that measures the link capacity of a target link in a path was proposed [48]. This scheme sends a train of probing packets in pairs, where each pair consists of a large packet tailgated by a small packet, similar to a single pair in Nettimer. All packets in the probing train are dropped before reaching the link of interest while the small packets of the first and last packet pairs reach the end node. This scheme measures the gap between the two small packets at the end node of a one-way transmission. The reported accuracy is higher than that of Nettimer when the measurement path is lightly congested with cross traffic.

3.7 Discussions

3.7.1 Clock Resolution

The main limitation of the workstations for the PPT measurement is the low resolution of the clock as provided by the operating systems. Higher accuracy in the measurement of PPT can be achieved by using a nanosecond-resolution clock, but the implementation of this resolution into the operating systems may degrade the performance of the currently available workstations [12]. Although the two workstations with 100-Mb/s links were

tested successfully in the reported experiments, a higher (host) clock resolution than the one used in the experiments is needed to measure PPTs for this link capacity and higher rates. In the experiments with end-link capacities of 10 Mb/s, the microsecond-resolution clock is sufficient for measuring P_ts as the transmissions times are 10 times larger than the clock resolution.

3.7.2 Internet Link-Capacity Ratios

The Internet experiments were performed without the knowledge of capacities of the intermediate links; therefore, two large packet-size ratios were used rather than exploring the largest link-capacity ratio of the Internet paths. The used packet-size ratios (i.e., $\alpha_1 = 17.37$ and $\alpha_2 = 13.5$) for the compound probes and the consistent measurements show that the Internet may not have link-capacity ratio(s) larger than the used packet-size ratios. Moreover, the Internet backbone is reported to have link capacities between 1- and 10-Gb/s speeds [28], [63]. It is, therefore, very unlikely to have large link-capacity ratio(s) in an Internet path that could produce dispersion in the intra-probe gap. Even with very large link-capacity ratio(s) along a path (or when approaching to the network core on aggregation links), it is possible to keep a zero-dispersion gap as end links are expected to have small capacities.

3.8 Conclusions

A scheme to measure the PPT of an end host that is connected over a multiple-hop path was proposed. To do this, two sets of compound probes, each set with a different trailing packet size and a zero-dispersion gap, is sent from a source host to the remote end host. The intra-probe gaps of the probing packets were used to estimate the end-link capacity.

PPT at the remote end host was estimated from the observed deviation of expected end-link capacity measurement.

In the proposed scheme, the zero-dispersion gaps, provided at the generation of the compound probes from the source host, were also used to detect the probes affected by cross traffic along the traveling path. This feature permits the filtering of these affected gaps and provides immunity against cross traffic. To use this property, a model to calculate the dispersion gap under cross traffic and a filtering scheme to remove intra-probe gaps affected by cross traffic were introduced.

The proposed scheme is implemented as a Linux application to perform experimental tests on a controlled testbed and in the Internet. The Internet experiments included two paths, one between New York and New Jersey, and the other between Taiwan and the U.S. (New Jersey). Same workstations in both experimental setups using 10- and 100-Mb/s end links were used and obtained consistent PPTs under different network conditions. The experimental results suggest that the accuracy of PPT measurement also depends on the clock resolution of the host capturing packets, and provisioning general-purpose workstations with a high clock resolution remains an open problem.

The experimental results show that the proposed scheme achieves high accuracy on 10-Mb/s end links. The accuracy on 100-Mb/s end links decreases as higher clock resolution is desirable but the PPTs were measurable. Higher clock resolutions may be needed to apply this method on higher link rates (e.g., 1000 Mb/s) or when there is an interrupt coalescence in the end host under measurement. Further evaluation of the scheme using high clock resolution and with interrupt coalescence is left for future research.

The proposed scheme also proved to be practical for use over the Internet, as the link-capacity ratios of the Internet appear to be no larger than the packet-size ratio that

an Ethernet MTU can provide. The consistent accuracy of the proposed scheme on both testbed and Internet paths proves the applicability of the proposed measurement scheme.

CHAPTER 4

RELATIVE CLOCK SKEW

4.1 Introduction

Clock resolution (i.e., the inverse of the oscillating frequency) is the smallest instance of time that can be measured by a time-keeping device while recording the duration of an event. The accuracy of a clock depends on the variability of the clock resolution in a period of time, named clock skew. Clock skew can add errors in time measurement of an event between two network nodes if their clocks are not synchronized during the measurement period. For example, measurement of one way delay (OWD) [8, 38] can be affected by the relative clock skew between a pair of network nodes in the network [64]. Figure 4.1 shows an illustration of OWD measurement in reference to the clocks running on the source (src) and destination (dst) nodes of an end-to-end path. Here, P is the probing packet sent from src to dst for OWD measurement. C_{src} and C_{dst} denote the clocks of src and dst, respectively. In the example shown in the same figure, C_{dst} runs twice as fast as C_{src}. According to the references of C_{src} and C_{dst}, dst receives P at 80 μs after src transmits it. Therefore, the measured OWD is 80 μs (OWD = receiving time at dst - transmission time at src). However, the expected OWD is 40 μs, according to C_{src}. Due to the relative clock skew between the source and destination nodes, C_{src} advances two ticks (each tick at C_{src}, which is the clock resolution, is equivalent to 20 μs) while C_{dst} advances four during the same period of time. Therefore, it is important to measure clock skew between the clocks of two network nodes.

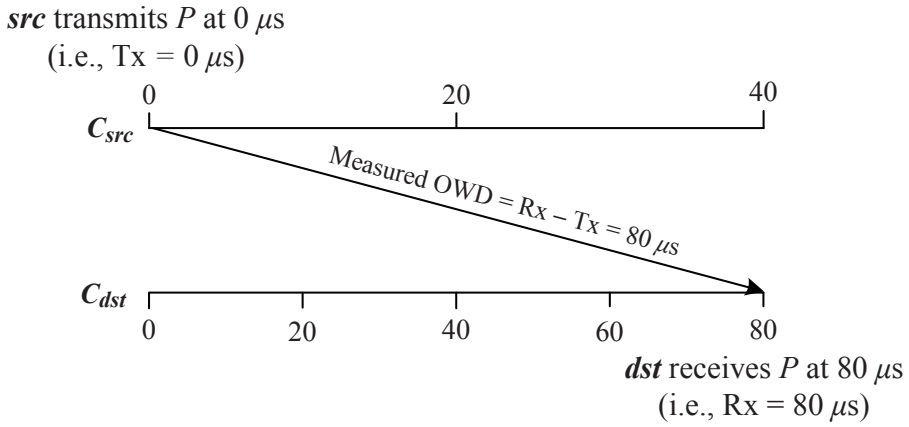


Figure 4.1 Effect of relative clock skew in OWD measurement between src and dst nodes, where the expected OWD is 40 μs.

In this chapter, a scheme for relative clock-skew measurement based on the intra-probe gap measurement of compound probes, as discussed in Chapter 3, at the end hosts (i.e., source and destination nodes) of an end-to-end path is proposed. The approach does not require complex statistical processing of the measured data unlike existing schemes [37, 64, 65, 66] because a) the compound probe is robust against cross traffic of the measurement path and b) it can detect the intra-probe gaps affected by the cross traffic so that the interference is statistically eliminated.

The remainder of the chapter is organized as follows: Section 4.4 discusses the existing clock-skew measurement schemes. Section 4.2 proposes a relative clock-skew measurement scheme and introduces a statistical method to filter the affected intra-probe gaps. Section 4.3 shows the performance evaluation of the proposed scheme in simulation environment. Section 4.5 concludes the discussion.

4.2 Proposed Scheme for Clock-skew Measurement

A scheme to measure relative clock skew between the end hosts of a path is proposed using intra-probe gap measurement of the compound probe at the hosts of a path. The proposed scheme is simple and measures clock skew accurately. The basic principle of the proposed scheme is that if the estimation of end link capacity, based on the intra-probe gap measurement of the compound probes, over a path is affected by a non-zero clock skew of the remote end host, as illustrated in Figure 4.2, the clock skew of the end host can be estimated by comparing the measured and actual capacity of the end link.

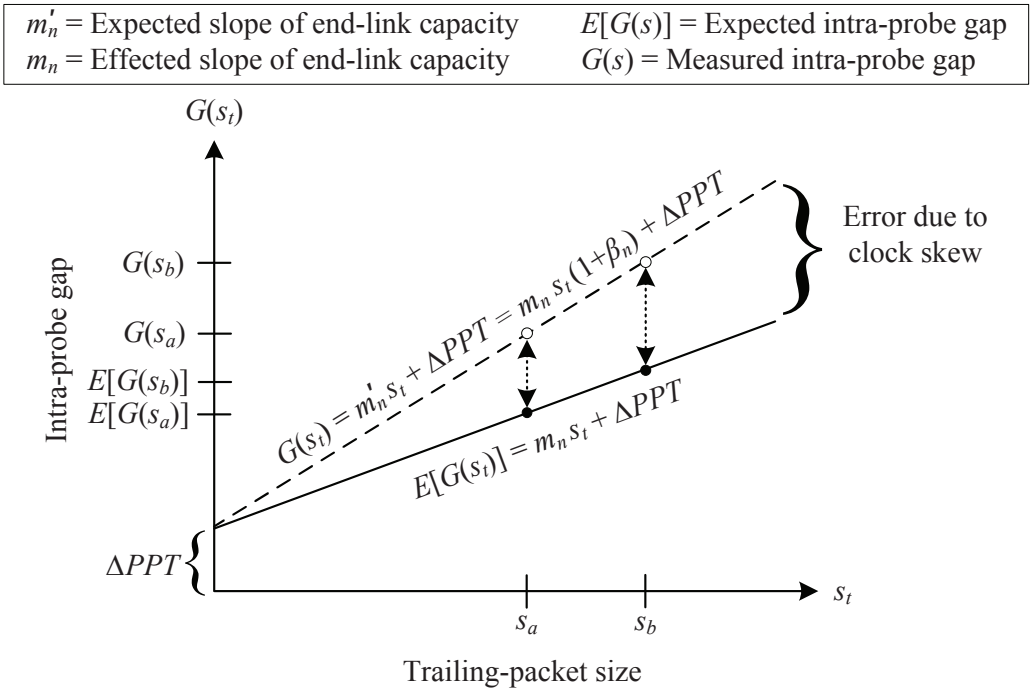


Figure 4.2 Effect of positive clock skew at $node_n$ in the estimation of the end-link capacity L_n using a pair of compound probes consisting of two different P_t sizes.

Figure 4.3 shows the steps to measure relative clock skew between src and dst of a n-hop end-to-end path based on end-link capacity measurements, as shown in Figure 4.4. src

Clock-skew measurement algorithm

1: Set s_h = Path MTU between *src* and *dst*

2 : Set $s_t = s_b$, where $s_b \leq s_t\,(max)$ over the forward path (*src* to *dst*)

3 : Send compound probes with s_h and s_t from *src* to *dst*

4 : Get the smallest intra-probe gap $G_{min}(s_b)$

5 : Set $s_t = s_a$, where $s_a < s_b$

6 : Send compound probes with s_h and s_t from *src* to *dst*

7 : Get the smallest intra-probe gap $G_{min}(s_a)$

8 : Estimate the capacity of the remote end link L_n, $\frac{1}{L_n'}$ or $m_n' = \frac{G_{min}(s_b) - G_{min}(s_a)}{(s_b - s_a)}$

9: Determine the expected intra-probe gap for s_b on L_n, $E[G(s_b)] = \frac{s_b}{L_n}$

10 : Estimate the clock skew of dst, $\beta_{dst} = \frac{(m_n'.s_b) - E[G(s_b)]}{E[G(s_b)]}$

11 : Repeat Steps 2 to 7 over the reverse path (*dst* to *src*), where $s_t = s_y$ and s_x, respectively, to send compound probes from *dst* to *src* for determining $G_{min}(s_y)$ and $G_{min}(s_x)$

12 : Estimate the capacity of the remote end link L_1, $\frac{1}{L_1'}$ or $m_1' = \frac{G_{min}(s_y) - G_{min}(s_x)}{(s_y - s_x)}$

13 : Determine the expected intra-probe gap for s_y on L_1, $E[G(s_y)] = \frac{s_y}{L_1}$

14 : Estimate the clock skew of src, $\beta_{src} = \frac{(m_1'.s_y) - E[G(s_y)]}{E[G(s_y)]}$

15 : Calculate the relative clock skew of dst with respect to *src*, $\beta = \beta_{dst} - \beta_{src}$

Figure 4.3 Proposed scheme for clock-skew measurement over an end-to-end path.

and dst send compound probes without any dispersion gap over the forward (src to dst) and reverse (dst to src) paths to measure the intra-probe gaps at dst and src, respectively. The

sizes of s_h and s_t in the compound probes are determined by the Maximum Transmission Unit (MTU) between the end hosts and the $s_t(max)$ value over each path direction, e.g., s_b and s_a for the forward path, and s_y and s_x for the reverse path, respectively. The sizes of $s_t(max)$ over forward and reverse path are determined based on the link capacities of the paths, as discussed in Chapter 3. Upon receiving the compound probes at each end host, the scheme determines the smallest intra-probe gaps $G_{min}(s_b)$ and $G_{min}(s_a)$ at dst, and $G_{min}(s_y)$ and $G_{min}(s_x)$ at src along with the slope values (i.e., the reciprocal of the link capacities) of the remote end links L_n and L_1, respectively. The expected intra-probe gaps of s_b and s_y on the remote end links are calculated using the actual link capacities of L_n and L_1, respectively. Clock skews of dst, β_{dst}, and of src, β_{src}, are estimated by comparing the measured and the expected intra-probe gaps of s_b and s_y on the respective end links. The relative clock skew, β, between src and dst is calculated from the difference between the estimated β_{src} and β_{dst} values.

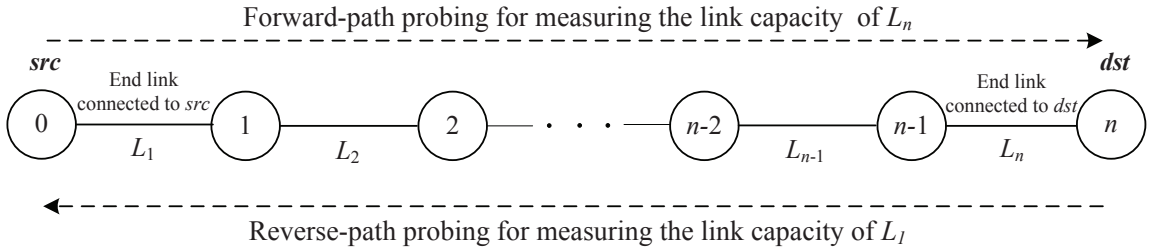


Figure 4.4 Bi-directional probing over an n-hop end-to-end path for measuring relative clock-skew between src and dst.

4.2.1 Filtering of Erroneous Intra-probe Gaps

In the proposed scheme, the smallest intra-probe gap at src and dst is determined by using a simple statistical analysis of the measured data sets. The intra-probe gap of a compound probe can have both decompression (caused by cross traffic over a path, if any) and compression (caused by the limited clock resolution in the operating system at the destination node), the following statistical analysis is performed to accurately determine the smallest gaps in the set (X) of intra-probe gaps by filtering out the affected gaps:

1. Calculate the mean $\bar{x}(j)$ and the standard deviation $\sigma(j)$ of X, where j is the iteration number such that $j \geq 1$.

2. If one of the following conditions is satisfied, stop. Else, go to Step 3.

   a. $\sigma(j) = 0$, for $j \geq 1$.

   b. $\sigma(j) => \sigma(j-1)$, for $j \geq 2$.

3. Discard all data elements in X greater than $\bar{x}(j)$ and go back to Step 1.

The mean value $\bar{x}(1)$ or $\bar{x}(j-1)$ is the smallest intra-probe gap in X if the algorithm terminates after one or j iterations, when $j > 1$, respectively.

4.3 Evaluation

The accuracy of the proposed scheme was evaluated in simulation environment using ns-2 [67]. The reasons for evaluating the scheme in simulation environment are: First, it is difficult to determine the ground-truth value of an end host's clock skew in real experiment [18]. Second, the smallest clock resolution of the available operating systems (e.g., 1 μs in Linux environment) running on end hosts is not sufficiently small [65], [4].

Figure 4.5 shows the simulation topology, an end-to-end path between end hosts src and dst, which was used to evaluate the proposed scheme. Relative clock skew between src and dst was measured under 75 and 80% bidirectional cross-traffic loads on the second ($L_2 = 10$ Mb/s) and third ($L_3 = 155$ Mb/s) links, respectively, of the end-to-end path using three different link capacities, e.g., 10, 100, and 1000 Mb/s, for the end links L_1 and L_4. Constant-Bit-Rate (CBR) traffic model with 100-byte packets (a dominant packet size in the Internet [19], [20]) was used as the cross-traffic packet size to create a demanding scenario for testing the effect of high traffic load on the proposed scheme. In the compound probes, $s_h = 1500$ bytes, and $s_t = 75$ and 100 bytes (on both forward and reverse paths) to adopt two large packet-size ratios, i.e., $\frac{1500\ bytes}{75\ bytes} = 20$ and $\frac{1500\ bytes}{100\ bytes} = 15$, respectively, to ensure a zero-dispersion gap in each path direction. 50 compound probes for both s_t sizes at each path direction were used to measure clock skews at src and dst to evaluate the robustness of the proposed scheme against the network congestion of the end-to-end path. $\Delta PPT = 1$ μs for every compound probe was considered both at src and dst.

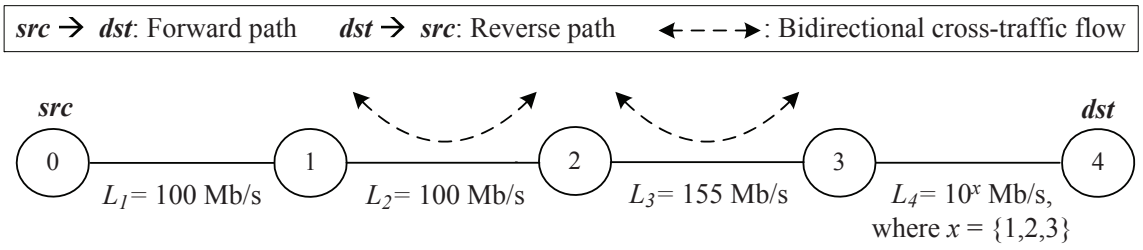


Figure 4.5 4-hop topology with bidirectional cross-traffic flows on the intermediate links.

The summary of the simulation results of the proposed scheme is presented in Table 4.1. According to the first column, $Clock\ skew$, of the table, the proposed scheme has been tested with four different combinations of clock skews at src and dst (i.e., β_{src} and β_{dst},

respectively) considering three different relative clock skew values, 50, 250, and 500 parts per million (ppm), between them. The $dst\ skew$ and $src\ skew$ columns show the measured clock skews of dst and src, respectively, and the last column, $Relative\ clock\ skew$, contains the relative clock skew calculated from the clock skews measured at the end hosts. According to the table, the clock skew values of src and dst measured by the proposed scheme are the same as those of the expected clock skews in the end hosts (see the first column); therefore, the measurement results validate both the accuracy and the robustness of the proposed scheme regardless of the heavy network congestion over the end-to-end path.

4.4 Related Work

The Network Time Protocol (NTP) [68, 37] is a widely used protocol for time synchronization between network nodes. It runs on a large distributed network of time servers consisting of different levels of clock accuracies. The time servers are connected in a hierarchical manner where the most accurate servers are located at the top tier of the NTP network. An end host connected to the NTP network synchronizes its clock to that of an upper-tier time server. NTP uses complex data-filtering and the so-called peer-selection and combining algorithms to synchronize the clocks throughout the network. NTP, however, cannot guarantee the accurate synchronization of clocks over the network [64], and the achieved synchronization resolution is rather coarse [37, 38].

A clock-skew measurement scheme based on OWD measurements of both forward (source to destination) and reverse (destination to source) paths over an end-to-end link was introduced [64]. The scheme shows that the OWD of a path increases or decreases over time in the presence of clock skew between a pair of end hosts, as shown in Figure 4.1. The

Table 4.1 Simulation Results of Clock-Skew Measurement

Clock skew (ppm)	End link (Mb/s)		Packet processing time (μs)	Forward path (src to dst) measurement							Reverse path (dst to src) measurement							Relative clock skew (ppm)
				s_t (bytes)		Smallest gap $G_{min}(s_t)$ (μs)		Slope	Expected gap (μs)	dst skew (ppm)	s_t (bytes)		Smallest gap $G_{min}(s_t)$ (μs)		Slope	Expected gap (μs)	src skew (ppm)	
β_{src}, β_{dst}	L_1	L_4	ΔPPT	s_a	s_b	$s_t = s_a$	$s_t = s_b$	m'_n	$E[G(s_b)]$	β_{dst}	s_x	s_y	$s_t = s_x$	$s_t = s_y$	m'_1	$E[G(s_y)]$	β_{src}	β
0, 50	100	10	1	75	100	61.003	81.004	0.80004	80	50	75	100	7	9	0.8	8	0	50
0, 50	100	100	1	75	100	7.0003	9.0004	0.080004	8	50	75	100	7	9	0.08	8	0	50
0, 50	100	1000	1	75	100	1.60003	1.80004	0.0080004	0.8	50	75	100	0.7	0.9	0.08	8	0	50
25, 75	100	10	1	75	100	61.0045	81.006	0.80006	80	75	75	100	7.00015	9.0002	0.080002	8	25	50
25, 75	100	100	1	75	100	7.00045	9.0006	0.080006	8	75	75	100	7.00015	9.0002	0.080002	8	25	50
25, 75	100	1000	1	75	100	1.600045	1.80006	0.0080006	0.8	75	75	100	7.00015	9.0002	0.080002	8	25	50
0, 500	100	10	1	75	100	61.03	81.04	0.8004	80	500	75	100	7	9	0.08	8	0	500
0, 500	100	100	1	75	100	7.003	9.004	0.08004	8	500	75	100	7	9	0.08	8	0	500
0, 500	100	1000	1	75	100	1.6003	1.8004	0.008004	0.8	500	75	100	7	9	0.08	8	0	500
250, 500	100	10	1	75	100	61.03	81.04	0.8004	80	500	75	100	7.0015	9.002	0.08002	8	250	250
250, 500	100	100	1	75	100	7.003	9.004	0.08004	8	500	75	100	7.0015	9.002	0.08002	8	250	250
250, 500	100	1000	1	75	100	1.6003	1.8004	0.008004	0.8	500	75	100	7.0015	9.002	0.08002	8	250	250

scheme estimates the relative clock skew between two end hosts of a path by determining the linear trends (i.e., the rate of increment or decrement) of the measured OWDs over the forward and reverse paths. Because the OWD measurement is affected by the cross traffic of a path, the detection of a linear trend in OWDs is difficult as it requires large data samples to eliminate measurement errors and to observe the time offset between the clocks of the remote hosts. The scheme performs rigorous statistical processing of the OWD data samples acquired over the measurement paths using complex data analysis.

Clock-skew measurement schemes based on unidirectional (i.e., forward path) OWD measurement were also proposed [65], [66]. In these schemes, errors in the OWD measurement are eliminated by using linear programming and convex hull techniques, respectively. These schemes also determine the relative clock skew between the workstations by detecting a linear trend of the measured OWDs. Therefore, these schemes also require large data samples for an accurate estimation. Although efforts have been made to reduce the size of the data sample, which is a major concern in active probing based measurement techniques [69], and to improve the accuracy of measurement [4, 18, 70], clock-skew estimation based on OWD measurements is still considered vulnerable to cross traffic over an end-to-end path [4, 65].

4.5 Conclusions

A scheme to measure the relative clock skew was proposed. The scheme sends compound probes over an end-to-end path in forward- and reverse-path directions and relies on receiving the compound probes at the end hosts with a zero-dispersion gap to measure the transmission speeds of the end links through intra-probe gap measurement. The comparison of the measured and expected transmission speeds is used to estimate the

clock skew of each host and the relative clock skew between them. The scheme is designed to measure the relative clock skew accurately because the compound probe and the data-filtering algorithm are designed to detect the intra-probe gaps affected by cross traffic on a path. The performance of the scheme was evaluated through simulation and the obtained value shows that it can estimate the relative clock skew accurately even under a heavy (e.g., 80%) cross-traffic load.

CHAPTER 5

WIRELESS THROUGHPUT IN HYBRID WIRED-WIRELESS NETWORK

5.1 Introduction

The shared-access mechanism of IEEE 802.11 networks along with collisions and channel fading make the measurement of the throughput of a wireless access link complex [71], [72]. Throughput is defined as the rate at which data bits can be successfully transmitted over a period of time for a given packet. If a packet of the maximum size (as defined by the MTU of an end-to-end path) is transmitted, the throughput is equivalent to the available bandwidth. A successful transmission of a packet pair, or a compound probe (see Chapter 3), over a wireless access link depends on the link capacity, unavailability of a wireless channel due to cross-traffic load, the number of retransmission attempts required to access a channel in case of collision, the time for receiving acknowledgment (ACK), and the delays contributed by the distributed coordination function interframe space (DIFS) and short interframe space (SIFS) [5]. Figure 5.1 shows some of these vulnerable intervals.

For a s_{tp}-byte P_t as Figure5.1 shows, the throughput (T) of the wireless access link is: $T = \frac{s_{tp}}{(t_2 - t_1)}$, where t_1 and t_2 are the arrival times of the last bits of P_h and P_t, respectively, of a compound probe at the wireless destination host and $t_2 - t_1$ is the intra-probe gap. The intra-probe gap might be affected by the bottleneck-link (i.e., the link with the smallest available bandwidth) location, cross traffic, and heterogeneous link capacities of the wired segment of a hybrid wired-wireless path, which is shown in Figure 5.2 [71, 73, 32]. A scheme to measure the download throughput of a wireless access link that is resilient to the above mentioned phenomena is needed.

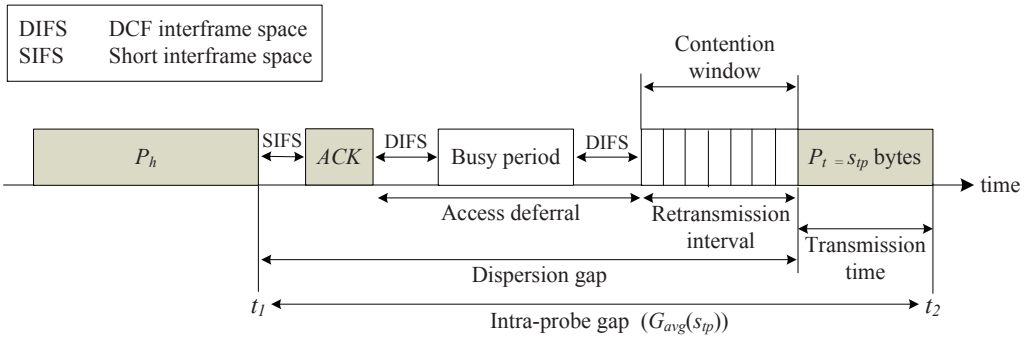


Figure 5.1 Intra-probe gap between the heading packet (P_h) and the trailing packet (P_t) of a compound probe over an IEEE 802.11 wireless access link.

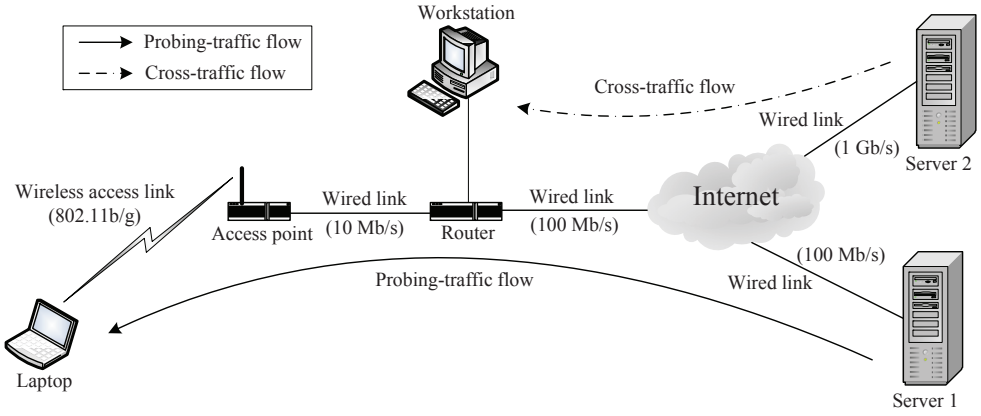


Figure 5.2 A hybrid wired-wireless path where a source host (Server 1) is connected to a wireless destination host (Laptop) through multiple wired links and a wireless access link.

In this chapter, a scheme to measure the download throughput of a wireless access link in a hybrid wired-wireless network is proposed. The scheme uses two compound probes with two different P_t sizes, $s_t = \{s_{ta}, s_{tb}\}$, to determine the smallest and average intra-probe gaps. The capacity of the wireless access link is then used to calculate the deviations on the expected intra-probe gaps. The deviation indicates the throughput of the

wireless access link. The scheme does not require the wireless link to be the bottleneck link of a path under measurement. Furthermore, the scheme is resilient against the presence of cross traffic on the wired links of the path.

The remainder of the chapter is organized as follows: Section 5.2 presents the proposed scheme to measure throughput of wireless access link. It also presents an analytical model for sizing the compound probes over a hybrid wired-wireless path and introduces a data-filtering algorithm to eliminate the cross-traffic affected intra-probe gaps in the compound probes for accurate throughput measurement. Section 5.3 presents the experimental results of the proposed scheme. Section 5.4 discusses existing schemes. Section 5.5 concludes the discussion.

5.2 Proposed Scheme for Throughput Measurement

In this section, the scheme to measure the throughput of wireless access link is proposed. An analysis of the conditions required for sizing of probing packets of the compound probe over a hybrid wired-wireless path for wireless throughput measurement also is presented. A filtering scheme is introduced to detect and remove the intra-probe gaps affected by cross traffic during measurement.

5.2.1 Measurement Scheme

Figure 5.3 shows the steps of the proposed measurement scheme. Two sets of compound probes are sent from the source host (src) to the wireless destination host (dst) of an end-to-end path using a large P_h size, s_h = Path MTU. Upon receiving the compound probes at dst, the scheme determines the smallest intra-probe gap $G_{min}(s_{tb})$ of the compound probes with $s_t = s_{tb}$ bytes, and the smallest and average intra-probe gaps $G_{min}(s_{ta})$ and $G_{avg}(s_{ta})$,

respectively, with $s_t = s_{ta}$ bytes, where $s_{ta} < s_{tb}$. The reciprocal of the wireless-link capacity $\frac{1}{c_n}$ is then determined from the smallest intra-probe gaps of the compound probes. The throughput is calculated as:

$$T = \frac{s_{tp}}{G_{avg}(s_{tp})} = \frac{s_{tp}}{G_{avg}(s_{ta}) - \frac{s_{ta}}{c_n} + \frac{s_{tp}}{c_n}} \tag{5.1}$$

where s_{tp} denotes the packet size for which the throughput is calculated. As stated in Equation 5.1, the throughput is the ratio between s_{tp} and the intra-probe gap $G_{avg}(s_{tp})$. The gap includes the normalized dispersion gap between P_h and P_t, defined as the gap between the last bit of P_h and the first bit of P_t, as shown in Figure 5.1. $G_{avg}(s_{ta}) - \frac{s_{ta}}{c_n}$ is the dispersion gap and $\frac{s_{tp}}{c_n}$ is the transmission time of a s_{tp}-byte packet on the wireless link. Figure 5.4 presents an illustration of a normalized dispersion gap on the wireless access link considering ΔPPT of dst in the intra-probe gap and the additional time, Δ, required to receive an ACK after the successful transmission of P_h on the wireless link, where [$E(Gs_t)$] refers to the case when there is no cross traffic on the link. Further details on Equation 5.1 can be found in [74].

Because the smallest and average intra-probe gaps of a compound probe might be different on a wireless link, the scheme sends multiple compound probes of each s_t size in a train for probing the wireless access link.

5.2.2 Filtering of Erroneous Intra-probe Gaps

In the proposed scheme, the smallest intra-probe gap of a compound probe is inversely proportional to the transmission rate of the wireless link when there is no contention for link access and, therefore, no dispersion in compound probes caused by cross traffic. In

Throughput measurement algorithm

1: Set s_h = Path MTU between *src* and *dst*

2: Set $s_t = s_{tb}$, where s_{tb} is determined by Equation 5.3 or 5.4

3: Send compound probes with s_h and s_t from *src* to *dst*

4: Get the smallest intra-probe gap $G_{min}(s_{tb})$

5: Set $s_t = s_{ta}$, where $s_{ta} < s_{tb}$

6: Send compound probes with s_h and s_t from *src* to *dst*

7: Get the smallest intra-probe gap $G_{min}(s_{ta})$

8: Get the average intra-probe gap $G_{avg}(s_{ta})$

9: Calculate the capacity c_n, $\frac{1}{c_n} = \frac{G_{min}(s_{tb}) - G_{min}(s_{ta})}{(s_{tb} - s_{ta})}$

10: Calculate the throughput T using Equation 5.1

Figure 5.3 Proposed scheme to measure the download throughput of a wireless access link in hybrid wired-wireless network.

wireless throughput measurement over an hybrid wired-wireless path, a compound probe can be affected by both decompression (caused by wireless channel contention by multiple wireless nodes, if any) and compression, as discussed in Chapter 4. To determine the smallest gap in the set (X) of measured intra-probe gaps, the same iterative method for filtering the affected gaps introduced for relative clock-skew measurement (see Section 4.2.1) is used.

On the other hand, the average intra-probe gap of X, which is proportional to the throughput of the wireless access link, is identified by determining the average of the most frequent intra-probe gap in the sample set where the data elements are distributed with a bin size of 9 μs. Here, the adopted 9-μs bin size is the smallest unit of retransmission interval following a collision on a wireless link as defined in the IEEE 802.11 standard [5], [75].

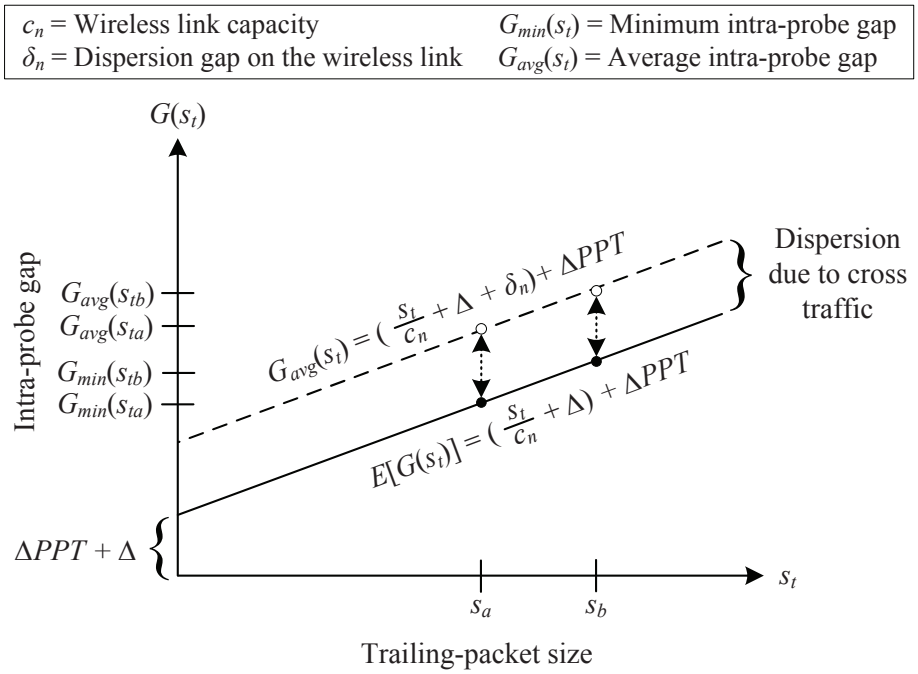


Figure 5.4 Normalized dispersion gap of a compound probe over wireless access link considering ΔPPT of dst and the additional time, Δ, required to receive an ACK in the intra-probe gap measurement.

5.2.3 Sizing Probing Packets to Ensure Zero-dispersion Gap

In a hybrid wired-wireless network with an IEEE 802.11 access link, a compound probe must arrive in the access point (AP) with a zero-dispersion gap so that any dispersion between P_h and P_t is the product of the access at the wireless link. If a compound probe experiences dispersion due to cross traffic and heterogeneous link capacities of the wired links [32], the intra-probe gap might not represent the throughput of the wireless link and this adds errors in the measurement.

Consider that the link capacities of the end-to-end path between the source host (src) and the wireless destination host (dst), as shown in Figure 5.5, consisting of multiple wired links, L_1, L_2, ..., L_{n-1}, and a wireless access link, L_n, are c_1, c_2, ..., c_n. To measure the

throughput of the wireless link of the end-to-end path, the required condition to obtain a zero-dispersion gap in a compound probe, with a heading-packet size s_h, at node $n-1$ (AP) is:

$$(\frac{s_h}{c_n} + \Delta - \frac{s_h}{\alpha c_{n-1}}) + (\frac{s_h}{c_{n-1}} - \frac{s_h}{\alpha c_{n-2}}) + ... + (\frac{s_h}{c_{z+1}} - \frac{s_h}{\alpha c_z}) = 0 \tag{5.2}$$

Equation 5.2 is motivated from the similar condition required at the destination host for measuring PPT over a wired end-to-end path, as derived by Equation 3.19.

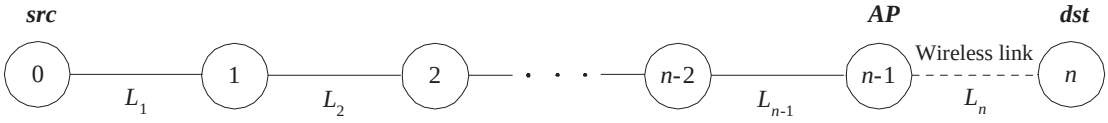


Figure 5.5 A multiple-hop path with wired (solid line) and wireless (dashed line) links.

From Equation 5.2, the maximum size of P_t, i.e., $s_t(max)$, over the wireless link is determined by:

$$s_t(max) = s_h \frac{\sum_{j=z+1}^{n} \frac{1}{c_j} + \Delta}{\sum_{j=z}^{n-1} \frac{1}{c_j}} \tag{5.3}$$

If the wireless access link is the narrow link of the path, $s_t(max)$ is calculated by:

$$s_t(max) = \left(\frac{s_h}{c_n} + \Delta\right) c_{n-1} \tag{5.4}$$

5.3 Experimental Results

Experimental evaluation of the proposed scheme has been performed in a testbed environment over two end-to-end path scenarios: a) single hop and b) multiple hops, as shown in Figure 5.6. The wireless links in these two scenarios were tested for IEEE 802.11b (11 Mb/s) and IEEE 802.11g (54 Mb/s) transmission rates. The single-hop path consists of a wired link and a wireless link without cross-traffic load along the path. The multiple-hop path has multiple wired links and a wireless access link with 50 and 75% cross-traffic loads on the second (L_2 = 155 Mb/s) and third (L_3 = 10 Mb/s) wired links, respectively. The above mentioned cross-traffic flows, each consisting of 128-byte User Datagram Protocol (UDP) packets, on L_2 and L_3 are generated by a Spirent Smartbits 6000C traffic generator. On the testbed, the wireless link constitutes the bottleneck link only in the single-hop scenario; in case of the multiple-hop scenario, the bottleneck link is located at the third wired link ($L_3 = 10$ Mb/s) of the path. A Belkin Wireless Cable/DSL Gateway Router (Model F5D7230-4) [76] was used as the AP in the testbed. The proposed scheme was implemented as a Linux application for the end hosts, src and dst, shown in Figure 5.6.

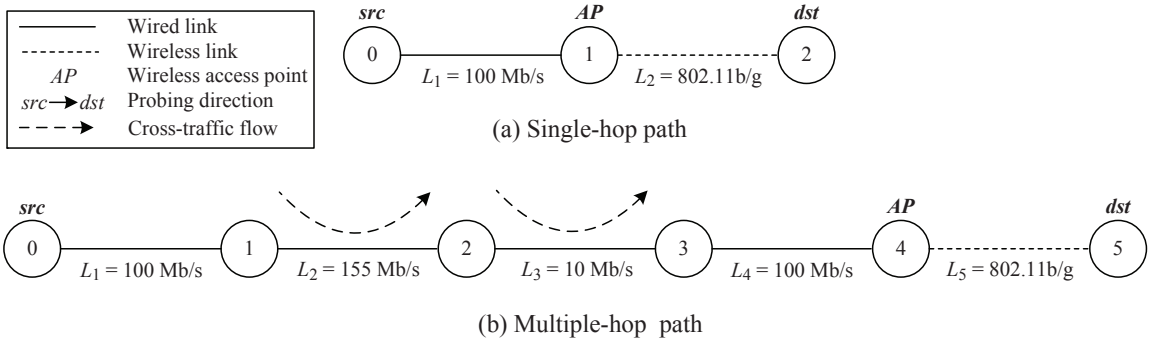


Figure 5.6 Hybrid wired-wireless testbed paths: a) single hop and b) multiple hops.

The measurement accuracy of the proposed scheme has been compared to WBest [73] and Iperf [77]. WBest is the only wireless scheme that provides source code in the public domain and Iperf is a widely used measurement tool [73]. Two sets of measurements for each scheme were performed using an IBM ThinkPad X40 (X40) and a Toshiba Satellite A105 (A105) laptops as dst nodes. Specifications of the laptops at dst, are shown in Table 5.1.

Table 5.1 Laptop Specifications at dst

	IBM ThinkPad X40	Toshiba Satellite A105
Name	X40	A105
Processor (speed)	Intel Pentium M (1 GHz)	Intel Pentium M (1.73 GHz)
RAM	512 MB	1024 MB
Linux Distribution	Fedora Core	Ubuntu
Linux kernel	2.6.18	2.6.35
NIC adapter	Intel Pro/Wireless 2200BG	Intel WM3B2200BG

The summary of the testbed experiments are presented in Table 5.2. In this table, the values refer to the average of 10 measurements performed by the proposed scheme, WBest, and Iperf. For throughput measurement, the proposed scheme adopted two different set values for s_{ta} and s_{tb}, including 8 bytes of UDP header, 20 bytes of IP header, and 14 bytes of MAC header, to be used as s_t in the compound probes. Considering s_h = 1500 bytes and an IEEE 802.11g link in the path configurations of Figure 5.6, $s_{ta} = 1392$ bytes and $s_{tb} = 1492$ bytes, respectively, were selected for the single-hop scenario, determined by Equation 5.4, and $s_{ta} = 288$ bytes and $s_{tb} = 388$ bytes, respectively, were selected for the multiple-hop scenario, determined by Equation 5.3[1]. Each probing train consisted of

[1]In s_{ta} and s_{ta} calculations, $\Delta = 0$ is considered in Equations 5.3 and 5.4.

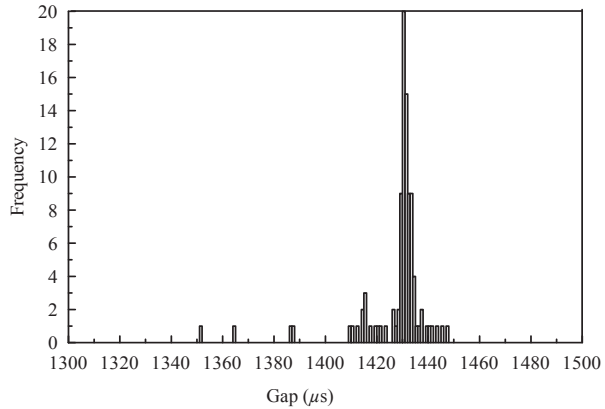
100 compound probes, which was found to be a suitable number through experimentation, inter-spaced with a constant interval of 100 ms. The same number of probing packets was used for WBest measurements. Because the probing-train size is not a tunable parameter in Iperf, each measurement iteration was run for 5 seconds. In WBest and Iperf, 1492-byte packets, including 42 bytes of protocol overhead, were used in the probing train.

In Table 5.2, the *Theoretical* column shows the theoretical throughputs of the IEEE 802.11b and 802.11g links for a traffic flow with 1450 bytes of UDP payload when there is no contention on the links. The theoretical throughputs have been determined by following the IEEE 802.11 standard [78]. These values are considered as ground truths in the testbed experiments. The throughput values measured by WBest and Iperf, using 1492-byte probing packets (IP payload size of the probing packets is also 1450 bytes) are shown in *WBest* and *Iperf* columns, respectively. *Proposed* column shows the throughput values of IEEE 802.11 links for a packet size s_{tp}, with a 1450-byte IP payload[2], which is obtained from the measured intra-probe gap values in the *Intra-probe gaps* column, wireless-link capacity values in the *Slope* column, and Equation 5.1. *Intra-probe gaps* column contains both the mean and the standard deviation of the measured intra-probe gaps, respectively. Figures 5.7 and 5.8 show samples of intra-probe gaps on the IEEE 802.11b and 802.11g links measured by the X40 and A105 laptops, respectively. The last three columns of Table 5.2 show the errors of WBest, Iperf, and the proposed scheme, respectively, in reference to the values of the $Theoretical$ column. The error is, therefore, defined as $|\frac{(Theoretical\ throughput\ -\ Measured\ throughput)}{Theoretical\ throughput}| \times 100\%$, where $Measured\ throughput$ is the throughput of the wireless link measured by WBest, Iperf, and the proposed scheme.

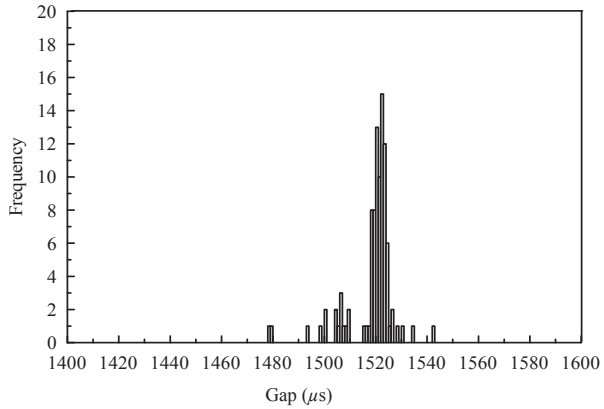
[2]Because throughput is calculated using IP payload, the header fields at the network and lower layers are not considered in Equation 5.1.

Table 5.2 Wireless Throughput Values of Testbed Experiments

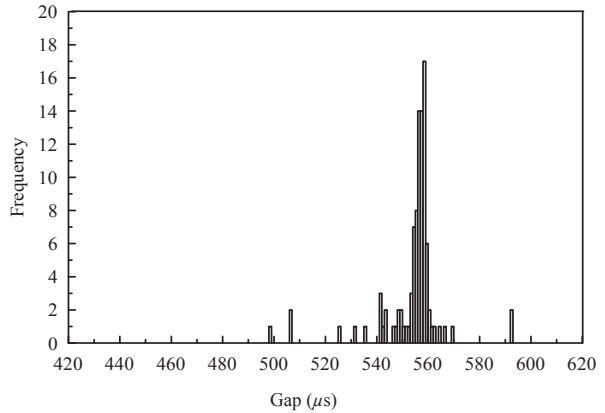
dst	Path	Wireless link	Intra-probe gaps (μs)			Slope	Throughput (Mb/s)				Error (%)		
	(hops)	(802.11x)	$G_{min}(s_{ta})$, std	$G_{min}(s_{tb})$, std	$G_{avg}(s_{ta})$, std	$\frac{1}{c_n}$	*Theoretical*	*WBest*	*Iperf*	*Proposed scheme*	*WBest*	*Iperf*	*Proposed scheme*
X40	Single	b	1427, 1.2	1516, 3.9	1430, 0.3	0.81	8.50	5.98	5.96	7.67	29.65	29.88	9.76
X40	Multiple	b	551, 6.7	628, 5.7	556, 0.3	0.77	8.50	4.96	5.65	7.82	41.65	33.53	8.00
X40	Single	g	485, 2.7	520, 1.2	495, 8.9	0.35	36.02	14.27	14.25	21.88	60.38	60.44	39.26
X40	Multiple	g	251, 9.2	274, 1.3	258, 0.4	0.23	36.02	5.24	8.19	22.28	85.45	77.26	38.15
A105	Single	b	1339, 56.9	1420, 52.7	1419, 14.3	0.81	8.50	5.63	5.95	7.73	33.76	30.00	9.06
A105	Multiple	b	519, 7.1	598, 6.8	547, 14.4	0.79	8.50	5.02	5.59	7.74	40.94	34.24	8.94
A105	Single	g	443, 30.4	484, 13.7	488, 45.0	0.41	36.02	14.84	12.8	21.92	58.80	64.46	39.14
A105	Multiple	g	222, 6.5	243, 4.8	242, 14.8	0.21	36.02	5.32	8.15	23.44	85.23	77.37	34.93



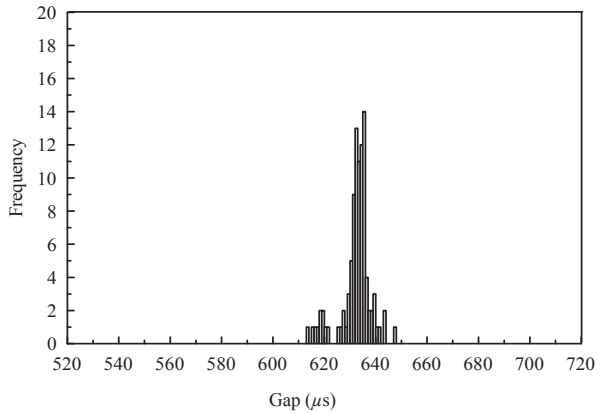
(a) X40 with IEEE 802.11b link on the single-hop path for s_t = 1392 byte



(b) X40 with IEEE 802.11b link on the single-hop path for s_t = 1492 byte

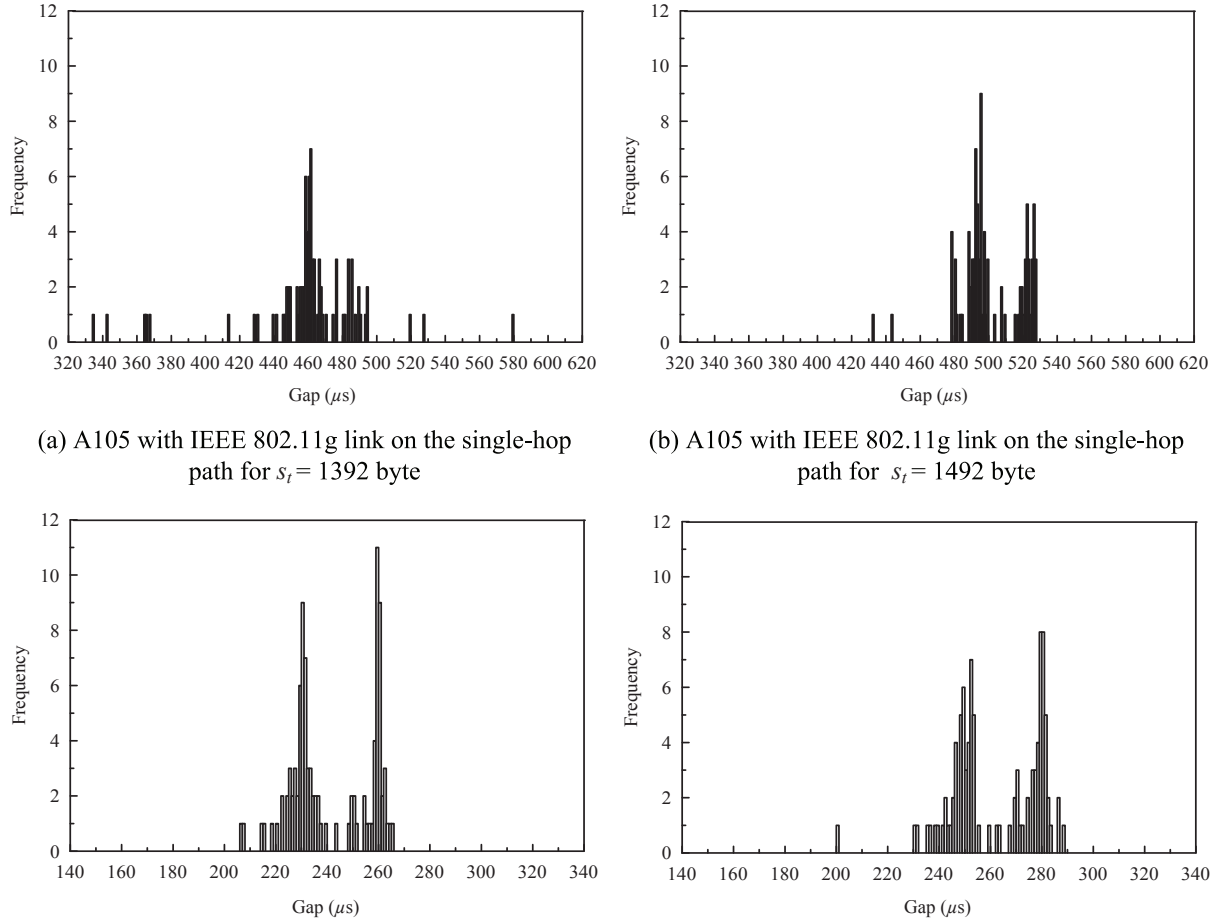


(c) X40 with IEEE 802.11b link on the multiple-hop path for s_t = 288 byte



(d) X40 with IEEE 802.11b link on the multiple-hop path for s_t = 388 byte

Figure 5.7 Samples of intra-probe gaps in the compound probes on IEEE 802.11b link measured by the IBM Thinkpad X40 laptop over single- and multiple-hop paths.



(a) A105 with IEEE 802.11g link on the single-hop path for s_t = 1392 byte

(b) A105 with IEEE 802.11g link on the single-hop path for s_t = 1492 byte

(c) A105 with IEEE 802.11g link on the multiple-hop path for s_t = 288 bytes

(d) A105 with IEEE 802.11g link on the multiple-hop path for s_t = 388 bytes

Figure 5.8 Samples of intra-probe gaps in the compound probes on IEEE 802.11g link measured by the Toshiba Satellite A105 laptop over single- and multiple-hop paths.

The error in the throughput values of the proposed scheme, measured by both laptops and on single- and multiple-hop paths are significantly smaller than those of WBest and Iperf. The errors of the proposed scheme's measurement are about 10 and 39% on IEEE 802.11b and 802.11g links, respectively, over the single-hop path. In the cases of WBest and Iperf measurements, the errors on IEEE 802.11b and 802.11g links are about 34 and 64%, respectively. The lower accuracy of WBest and Iperf measurements over the single-hop path may be the result of determining the throughput using the average intra-probe gap of the probing train, which can be affected by large intra-probe gaps.

While the high accuracy of the proposed scheme remains consistent in each path scenario, both WBest and Iperf are not designed to measure throughput on a multiple-hop path where the wireless link does not constitute the bottleneck link. The degradation of measurement accuracy of these schemes in multiple-hop scenario is more evident on the IEEE 802.11g link than on the IEEE 802.11b link. For example, the error in the WBest measurement increases from 59 to 85% when throughput is measured by the X40 laptop over the multiple-hop path, using IEEE 802.11g as the wireless access link. Overall, the testbed results show that the proposed scheme outperforms the existing schemes in both path scenarios, even when the wireless access link is the bottleneck link of the end-to-end path. The accuracy of the proposed scheme also remains constant under heavy cross-traffic conditions.

5.4 Related Work

Existing schemes for throughput measurement of wireless link (IEEE 802.11) can be broadly divided into two categories according to their working principles: a) one-way-delay

[71] and b) intra-probe gap [74]-[79] based schemes. Four schemes are discussed in detail below.

Probegap measures the wireless throughput of a hybrid wired-wireless path from the fraction of time a the wireless access link remains idle during a period of time [71]. The scheme determines the idle period of the wireless link by identifying the one-way delay (OWD) trend (e.g., continuing delay increments during a period of time) between the source and wireless destination nodes of the end-to-end path. The idle period is then multiplied with the predefined (i.e., known a priori) capacity of the wireless link to determine the wireless throughput. Experimental evaluation shows that the scheme overestimates throughput measurement under heavy cross load on the wireless access link even when the wireless access link constitutes the bottleneck link of an end-to-end path [71].

Normalized throughput is a packet-pair based scheme for measuring throughput in wireless ad-hoc networks [74]. The scheme shows that the intra-probe gap of a packet pair on a wireless link varies with the probing-packet sizes but the dispersion gap (i.e., the first and last bits of heading and trailing packets) remains constant. This scheme uses the predefined capacity of the wireless link to deduct the contribution of the transmission time in the measured intra-probe gap for estimating a normalized throughput of the wireless link.

WBest is another scheme that measures throughput of wireless access link using packet-pair structure [73]. The scheme first sends multiple packet pairs to determine the maximum transmission rate of the wireless access link from the measured intra-probe gaps. It then sends a train of probing packets at the maximum transmission rate to estimate the wireless throughput from the receiving rate of the probing train at the wireless destination

node. WBest also incorporates packet-loss rate in its throughput calculation and does not require predefined capacity of the wireless access link.

Netalyzr is a web-based utility tool that is dedicated for profiling the wireless access link of an end-to-end path [79]. It performs a number of measurements to deduce various network properties of the wireless connection, e.g., IP address, firewall, latency, buffer size, throughput, etc. For throughput measurement, Netalyzr periodically sends probing packets with different intra-probe gaps (i.e., different rates) using exponential load and provides a sustained transmission rate of the wireless access link. Detailed working principle of Netalyzr for throughput measurement is unknown.

A common drawback of the existing schemes is that they require the wireless access link constitute the bottleneck link of a hybrid wired-wireless path [71],[73],[74],[80]. If this condition is not satisfied, the accuracy of the schemes may decrease because the probing packets may experience delay or dispersion created by a bottleneck link located on the wired segment before reaching the AP of the path, as discussed in Section 5.1.

5.5 Conclusions

A scheme has been proposed to measure download throughput of wireless access links in a hybrid wired-wireless network consisting of IEEE 802.11 links. The scheme is based on sending compound probes with two different trailing-packet sizes. The scheme is simple and it does not require the wireless link to constitute the bottleneck of a hybrid wired-wireless path for throughput measurement. The scheme was experimentally tested on single-hop and multiple-hop paths, with different bottleneck-link locations and under different cross-traffic loads on the wired links. The experimental results show that the proposed scheme achieves an accuracy of 90 and 61% on IEEE 802.11b and 802.11g links,

respectively, and it is resilient to cross traffic on the wired links preceding the wireless access link.

CHAPTER 6

CONCLUSIONS

This dissertation was focused on the measurement of three different network parameters to better characterize the Internet. The contribution of this work spans both in wired and wireless environments.

In the wired network, a method to measure a new network parameter, called packet processing time (PPT) of end hosts, was introduced after presenting the significance of PPT for characterizing a network path. The proposed method uses ICMP packets and specialized packet-capture card to measured PPT in a LAN setup. The scheme was evaluated on three different workstations consisting of different specifications under 10-, 100-, and 1000-Mb/s transmission speeds on a controlled testbed. The experimental values show that the proposed scheme has a high measurement stability and does not require clock synchronization between the end host and the packet-capture card.

Another scheme to measure the PPT of an end host over a multiple-hop path in the Internet was proposed based on the capacity measurement of the link directly connected to the end host under interest using the intra-probe gap of a packet-pair structure, called compound probe. An analytical model for sizing the packets of a compound probe was also derived to avoid cross-traffic interference in the PPT measurement. The accuracy of the proposed scheme was verified through testbed and Internet measurements under two different (i.e., 10- and 100-Mb/s) transmission speeds considering different path configurations and cross-traffic loads. The consistent accuracy of proposed scheme proves the applicability of the scheme in the Internet.

A solution for measuring relative clock-skew between two end hosts connected over an end-to-end path in wired network was explored. The proposed scheme estimates the capacities of the end links of the hosts using compound probes to determine the relative clock skew. Unlike existing clock-skew measurement schemes, the proposed scheme is simple and does not require complex statistical processing of the sampled data sets since the probing structure can detect intra-probe gaps affected by the cross traffic of the path. The scheme was evaluated through simulation under high cross-traffic loads over an multiple-hop path. The simulation results validate the high measurement accuracy of the proposed scheme.

Finally, a scheme to measure the download throughput of IEEE 802.11 standard based wireless access links in a hybrid wired-wireless path using compound probes was proposed. This scheme showed that when a compound probe arrives in the access point of a hybrid wired-wireless path with a zero-dispersion gap, the intra-probe gap of the compound probe is determined by the throughput of the wireless access link regardless of the cross-traffic load on the wired links and bottleneck location of the path. Measurement accuracy of the proposed scheme was evaluated through experimentation in a testbed environment under different cross-traffic loads and bottleneck locations. The proposed scheme outperforms the state-of-art scheme in all testbed scenarios.

REFERENCES

[1] M. Crovella and B. Krishnamurthy, *Internet Measurement: Infrastructure, Traffic and Applications, ch. 5*. West Sussex, England: John Wiley and Sons., 2006.

[2] I. Csabai, A. Fekete, P. Haga, B. Hullar, G. Kurucz, and S. L. et al., "ETOMIC advanced network monitoring system for future internet experimentation," in *Proc. of International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities*, Berlin, Germany, pp. 243–254, 2010.

[3] S. Donnelly. Endace DAG time-stamping whitepaper. [Online]. Available: http://www.lcs.poli.usp.br/~jra/UPC/Traces/timestamping_whitepaper.pdf.

[4] J. Bi, Q. Wu, and Z. Li, "On estimating clock skew for one-way measurement," *Computer Communications*, vol. 29, no. 8, pp. 1213–1225, 2006.

[5] IEEE standard 802.11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications. [Online]. Available: http://standards.ieee.org/getieee802/download/802.11-2007.pdf.

[6] IEEE standard 802.3: Ethernet. [Online]. Available: http://standards.ieee.org/about/get/802/802.3.html

[7] K. Salehin and R. Rojas-Cessa, "Active scheme to measure throughput of wireless access link in hybrid wired-wireless network," *IEEE Wireless Communications Letters*, vol. 1, no. 6, pp. 645–648, 2012.

[8] 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.

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

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

[11] G. Jin and B. Tierney, "System capability effects on algorithms for network bandwidth measurements," in *Proc. of Internet Measurement Conference*, FL, USA, pp. 27–38, 2003.

[12] R. Prasad, M. Jain, and C. Dovrolis, "Effects of interrupt coalescence on network measurements," in *Proc. of Passive and Active Measruement Conference*, France, pp. 247–256, 2004.

[13] S. Savage. IP router design. [Online]. Available: http://cseweb.ucsd.edu/classes/wi05/cse123a/Lec8.pdf.

[14] A. Hernandez and E. Magana, "One-way delay measurement and characterization," in *Proc. of IEEE International Conference on Networking and Services*, Athens, Greece, pp. 114, 2007.

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

[16] B. Forouzan, *TCP/IP Protocol Suit, ch. 3*. New York, NY, USA: McGraw Hill, 2010.

[17] M. Garetto and D. Towsley, "Modeling, simulation and measurements of queuing delay under long-tail Internet traffic," in *Proc. of ACM Special Interest Group on Measurement and Evaluation Conference*, CA, USA, pp. 1–11, 2003.

[18] S. Zander and S. J. Murdoch, "An improved clock-skew measurement technique for revealing hidden services," in *Proc. of 17th USENIX Security Symposium*, CA, USA, pp. 211–225, 2008.

[19] Caida. Packet size distribution comparison between Internet links in 1998 and 2008. [Online]. Available: http://www.caida.org/research/traffic-analysis/pkt_size_distribution/graphs.xml.

[20] 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

[21] M. Lee, N. Duffield, and R. Kompella, "Not all microseconds are equal: Fine-grained per-flow measurements with reference latency interpolation," in *Proc. of ACM Special Interest Group on Data Communication Conference*, Dehli, India, pp. 27–38, 2010.

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

[23] V. Padmanabhan and L. Subramanian, "An investigation of geographic mapping techniques for Internet hosts," in *Proc. of ACM Special Interest Group on Data Communication Conference*, CA, USA, pp. 173–185, 2001.

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

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

[26] Z. Dong, R. Perera, R. Chandramouli, and K. Subbalaksmi, "Network measurement based modeling and optimization for IP geolocation," *Computer Networks*, vol. 56, no. 1, pp. 85–98, 2012.

[27] Internet2 network. [Online]. Available: http://www.internet2.edu/network/.

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

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

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

[31] C. Dovrolis, P. Ramanathan, and D. Moore, "Packet dispersion techniques and capacity estimation," *IEEE/ACM Transactions on Networking*, vol. 12, no. 6, pp. 963–977, 2004.

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

[33] ——, "Packet-pair sizing for controlling packet dispersion on wired heterogeneous networks," accepted to appear in *Proc. of IEEE International Conference on Computing, Networking and Communication*, CA, USA, pp. 1–5, 2013.

[34] J. Sommers and P. Barford, "An active measurement system for shared environments," in *Proc. of Internet Measurement Conference*, pp. 303–314, 2007.

[35] Z. Qin, R. Rojas-Cessa, and N. Ansari, "Task-execution scheduling schemes for network measurement and monitoring," *Computer Communications*, vol. 3, no. 2, pp. 124–135, 2009.

[36] Endace DAG 7.5G2 datasheet. [Online]. Available: http://www.endace.com/assets/files/resources/END_Datasheet_DAG7.5G2_3.0.pdf.

[37] 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

[38] L. D. Vito, S. Rapuano, and L. Tomaciello, "One-way delay measurement: State of art," *IEEE Transactions on Instrumentation and Measurements*, vol. 57, no. 12, pp. 2742–2750, 2008.

[39] P. Willman, H. Kim, S. Rixner, and V. Pai, "An efficient programmable 10 gigabit ethernet network interface card," in *Proc. of IEEE International Symposium on High-Performance Computer Architecture*, CA, USA, pp. 96–107, 2005.

[40] K. Ramakrishnan, "Performance considerations in designing network interfaces," *IEEE Journal on Selected Areas in Communications*, vol. 11, no. 2, pp. 203–219, 1993.

[41] R. Mandeville and J. Perser. RFC 2889 - Benchmarking methodology for LAN switching devices. [Online]. Available: http://www.ietf.org/rfc/rfc2889.txt.

[42] 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 Transations on Instrumentation and Measurement*, vol. 55, no. 4, pp. 1065–1076, 2006.

[43] R. Govindan and V. Paxson, "Estimating router ICMP generation time," in *Proc. of Passive and Active Measruement Conference*, CO, USA, pp. 1–8, 2002.

[44] K. Lai and M. Baker, "Measuring link bandwidths using a deterministic model of packet delay," in *Proc. of ACM Special Interest Group on Data Communication Conference*, Stockholm, Sweden, pp. 283–294, 2000.

[45] K. Salehin and R. Rojas-Cessa, "Scheme to measure relative clock skew of two Internet hosts based on end-link capacity," *IET Electronics Letters*, vol. 48, no. 20, pp. 1282–1284, 2012.

[46] Wireshark. [Online]. Available: http://www.wireshark.org/.

[47] K. Harfoush, A. Bestavros, and J. Byers, "Measuring bottleneck bandwidth of targeted path segments," in *Proc. of IEEE International Conference on Computer Communications*, CA, USA, pp. 2079–2089, 2003.

[48] ——, "Measuring bottleneck bandwidth of targeted path segments," *IEEE/ACM Transactions on Networking*, no. 1, pp. 80–92, 2009.

[49] Highest port density performance analysis system: SmartBits 6000C. [Online]. Available: http://www.spirent.com/~/media/Datasheets/Broadband/ObsoleteSMBTM/SmartBits%206000C.pdf.

[50] M. Crovella and B. Krishnamurthy, *Internet Measurement: Infrastructure, Traffic and Applications, ch. 4*. West Sussex, England: John Wiley and Sons., 2006.

[51] D. Bovet and M. Cesati, *Understanding the Linux Kernel, ch. 5*. Sebastopol, CA, USA: O'Reilly, 2001.

[52] Y. Ghiassi-Farrokhfal and J. Liebeherr, "Output characterization of constant bit rate traffic in FIFO networks," *IEEE Communications Letters*, vol. 13, no. 8, pp. 618–620, 2009.

[53] C. Choon. High speed networks and multimedia networking. [Online]. Available: http://www.comp.nus.edu.sg/~cs5224/lectures/traffic.pdf.

[54] N. Hu and P. Steenkiste, "Evaluation and characterization of available bandwidth probing techniques," *IEEE Journal of Selected Areas in Communications*, vol. 21, no. 6, pp. 879–894, 2003.

[55] J. Bolot, "End-to-end packet delay and loss behavior in the Internet," in *Proc. of ACM Special Interest Group on Data Communication Conference*, NY, USA, pp. 289–298, 1993.

[56] K. Lai, “Measuring bandwidth,” in *Proc. of IEEE International Conference on Computer Communications*, NY, USA, pp. 235–245, 1999.

[57] R. Kapoor, L. Chen, L. Lao, M. Gerla, and M. Sanadidi, “CapProbe: A simple and accurate capacity estimation technique,” in *Proc. of ACM Special Interest Group on Data Communication Conference*, OR, USA, pp. 67–78, 2004.

[58] V. Jacobson. Pathchar - A tool to infer characteristics of Internet paths. [Online]. Available: ftp://ftp.ee.lbl.gov/pathchar/msri-talk.pdf.

[59] A. Downey, “Using pathchar to estimate Internet link characteristics,” in *Proc. of ACM Special Interest Group on Data Communication Conference*, MA, USA, pp. 241–250, 1999.

[60] B. Mah. pchar: A tool for measuring Internet path characteristics. [Online]. Available: http://www.kitchenlab.org/www/bmah/Software/pchar/.

[61] Network characterization service (NCS). [Online]. Available: http://web.archive.org/web/20111229041049/http://www-didc.lbl.gov/NCS/

[62] J. Postel. RFC 792 - Internet Control Message Protocol. [Online]. Available: http://tools.ietf.org/html/rfc792.

[63] InterMapper Web Server. [Online]. Available: https://intermapper.engineering.cenic.org.

[64] V. Paxson, “On calibrating measurement of packet transit times,” in *Proc. of ACM Special Interest Group on Measurement and Evaluation Conference*, WI, USA, pp. 11–21, 1998.

[65] S. Moon, P. Skelly, and D. Towsley, “Estimation and removal of clock skew from network measurement delays,” in *Proc. of IEEE International Conference on Computer Communications*, CA, USA, pp. 227–234, 1998.

[66] L. Zhang, Z. Lui, and C. Xia, “Clock synchronization algorithm for network measurements,” in *Proc. of IEEE International Conference on Computer Communications*, NY, USA, pp. 160–169, 2002.

[67] The network simulator - ns-2. [Online]. Available: http://www.isi.edu/nsnam/ns/

[68] D. Mills, “Internet time synchronization: The network time protocol,” *IEEE Transactions on Communications*, vol. 39, no. 10, pp. 1482–1493, 1991.

[69] Z. Qin, R. Rojas-Cessa, and N. Ansari, “Task-execution scheduling schemes for network measurement and monitoring,” *Computer Communications*, vol. 33, no. 2, pp. 124–135, 2010.

[70] B. Ngamwongwattana and R. Thompson, “Sync & Sense: Voip measurement methodology for assessing one-way delay without clock synchronization,” *IEEE Transactions on Instrumentation and Mesurement*, vol. 59, no. 5, pp. 1318–1326, 2010.

[71] K. Lakshminarayanan, V. Padmanabhan, and J. Padhye, "Bandwidth estimation in broadband access networks," in *Proc. of Internet Measurement Conference*, Sicily, Italy, pp. 314–321, 2004.

[72] L. Angrisani, A. Napolitano, and M. Vadursi, "Modeling and measuring the capacity of communication networks," *IEEE Transactions on Instrumentation and Measurement*, vol. 59, no. 5, pp. 1065–1072, 2005.

[73] M. Li, M. Claypool, and R. Kinicki, "Wbest: A bandwidth estimation tool for ieee 802.11 wireless networks," in *Proc. of IEEE Conference on Local Computer Networks (LCN)*, Que., Canada, pp. 374–381, 2008.

[74] S. Shah, K. Chen, and K. Nahrstedt, "Available bandwidth estimation in ieee 802.11-based wireless networks," in *Proc. of First ISMA Workshop on Bandwidth Estimation*, CA, USA, p. 2003.

[75] D. Vassis, G. Kormentzas, A. Rouskas, and I. Maglogiannis, "The ieee 802.11g standard for high data rate wlans," *IEEE Network*, vol. 19, no. 3, pp. 21–26, 2005.

[76] Belkin wireless cable/dsl gateway router. [Online]. Available: http://www.belkin.com/support/dl/f5d6231-4v2-%20manual.pdf.

[77] Iperf. [Online]. Available: http://iperf.sourceforge.net/.

[78] M. Gast. When is 54 not equal to 54? a look at 802.11a, b, and g throughput. [Online]. Available: http://www.oreillynet.com/pub/a/wireless/2003/08/08/wireless_throughput.html\?page=2

[79] C. Kreibich, N. Weaver, B. Nechaev, and V. Paxson, "Netalyzr: Illuminating the edge network," in *Proc. of Internet Measurement Conference*, Melbourne, Australia, pp. 246–259, 2010.

[80] A. Johnsson, M. Bjrkman, and B. Melander, "An analysis of active end-to-end bandwidth measurements in wireless networks," in *Proc. of 4th IEEE/IFIP Workshop on End-to-End Monitoring Techinique and Services*, BC, Canada, pp. 74–81, 2006.