Method for Measuring the Packet Processing Time of Internet Workstations with the Detection of Interrupt Coalescence

Khondaker M. Salehin
Division of Science, Mathematics, and Computing
Bard College, Annandale-on-Hudson, NY 12504
Email: ksalehin@bard.edu

Roberto Rojas-Cessa
Department of Electrical and Computer Engineering
New Jersey Institute of Technology, Newark, NJ 07102
Email: rojas@njit.edu

***Abstract*—The packet processing time (PPT) of an end host (i.e., workstation) is the time elapsed between the arrival of a packet at the data-link layer and the time the packet is processed at the application layer (RFCs 2679). A recent work presented an active scheme to measure PPT over Internet paths using a packet-pair based probing structure considering its importance as a network parameter in the Internet. However, the existing scheme does not consider the effect of interrupt coalescence (IC) in network interface cards (NICs), which are configured with an IC under high transmission speeds. In this paper, we propose an enhancement to the existing scheme for measuring PPT when there is an IC available in the NIC of the workstation under test. The enhanced scheme first detects IC and then measures PPT using the measured gaps of the probing structure. We evaluated the enhanced scheme through testbed experiments using two different workstations and under different transmission speeds, i.e., 10, 100, and 1000 Mb/s. Our results show that the proposed scheme consistently measures PPT with high efficacy.**

***Index Terms*—Active measurement, packet pair, packet processing time, interrupt coalescence, compound probe, end-to-end delay, high-speed link, Internet path.**

I. Introduction

The packet processing time (PPT) of an end 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 Open Systems Interconnection Reference Model) and the time the packet is processed at the application layer [1]. As link rates increase faster than processing speeds [2], the role of PPT becomes more significant in the measurement of different network parameters.

For example, one-way delay (OWD) in a local area network (LAN) is a parameter that PPT may significantly impact [3]. Figure 1 illustrates the OWD of packet P over an end-to-end path, between two end workstations, the source (src) and the destination (dst) hosts. The figure shows different parameters associated with the OWD of P. According to RFC 2679 [1], OWD is the wire time that the packet experiences in the trip from src to dst. The wire time includes the transmission time (t_t), the queueing delay (t_q), and the propagation time (t_p), or $OWD = t_t + t_q + t_p$, as Figure 1 shows. However, as time stamping after packet creation at src (PPT_{src}) and packet receiving at dst (PPT_{dst}) takes place at the application layer, the coarsely measured OWD may include these PPTs plus additional buffering times of P at the data-link layer due to interrupt coalescence (IC) on the NICs [4], [5], as an apparent OWD (*OWD*') of packet P between src and dst, or:

$$OWD' = PPT_{src} + IC_{src} + t_t + t_q + t_p + PPT_{dst} + IC_{dst} \quad (1)$$

Moreover, because of the low transmission rates of legacy systems, PPT and IC have been considered so far negligible.

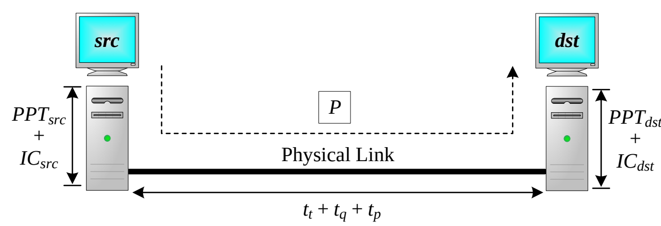


Fig. 1. Packet P travels over an end-to-end path between two directly connected workstations.

As transmission rates increase, the contribution of PPT on OWD increases, and the error in the measurement of OWD in high-speed LANs can be large if PPTs are neglected. For example, the measurement of OWD between end hosts connected over a 100-Mb/s link using 1500- and 40-byte packets would have errors of 2.5 and 9%, respectively, for $PPT_{src} + IC_{src} = PPT_{dst} + IC_{dst} = 2$ μs, where $IC_{src} = IC_{dst} \geq 0$, an average $t_q = 40$ μs [6], and $t_p = 0.5$ μs, considering a 100-m Fast-Ethernet cable. This error increases to 108% when the queuing delay is relieved (i.e., $t_q \simeq 0$ μs [6]) for the 40-byte packet. In a similar scenario, this error is 16% on a 1000-Mb/s link (as $t_p = 25$ μs for a 5-km optical cable in Gigabit Ethernet). Therefore, PPT and the presence of IC must be considered for an accurate measurement of OWD, which can be, for example, used to increase accuracy in IP geolocation. In IP geolocation, each microsecond of propagation delay varies the estimated geographic distance by 200 m between two end hosts connected over optical links [7].

In addition, the knowledge of PPT can also be utilized in electronic-trading data centers. For example, it may be possible to identify which electronic-trading servers comply with the required transaction speed if PPT of the servers are estimated. This information would eventually increase the confidence between the customers and providers of these services [8], [9].

The measurement of the PPT of a remote host over an end-to-end path is complex. For example, 1) 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), 2) instrumentation at the data-link layer is required since commodity NICs do not time stamp packets at the data-link layer, and 3) the clocks at the data-link and application layers should be synchronized [10], [11].

Besides the above stated complexities, the presence of IC in the NIC of a workstation may also affect the PPT measurement. IC is a time period over which a NIC buffers more than one packets at it's input queue before sending an interrupt to the central processing unit (CPU) for time stamping the packets at the application layer in order to reduce CPU load at the workstation [4], [5], [12]. IC is a common hardware artifact in commodity NICs under high transmission speeds, e.g., 100 Mb/s and above, where the packet buffering period is available in the order of 100s of microseconds [4], [12]. It is, therefore, essential to detect the presence of IC for accurate measurement of PPT under high transmission speeds.

In this paper, we propose an enhanced scheme that first detects the presence of IC before estimating PPT of a remote workstation over an Internet path using the measured gaps in pairs of packets at the workstation under test . The enhanced scheme is an improvement over the existing scheme [10] which considers that the IC is negligible at the workstation under test.

The rest of the paper is organized as follow: Section II discusses the existing schemes for PPT measurement. Section III proposes the enhanced scheme in detail. Section IV presents the experimental results and measurement accuracy of the enhanced scheme in a testbed environment. Section V discusses the utility of IC detection in NICs beyond PPT measurement. Section VI concludes our discussion.

II. Related Work

To the best of our knowledge, there exists only two schemes that measure PPT of workstations using 1) Internet Control Message Protocol (ICMP) and 2) User Datagram Protocol (UDP) packets, respectively. We describe these two schemes below.

The ICMP packet based scheme uses a specialized packet-capture line (PCL) card, e.g., Endace DAG card [13], at the data-link layer of the workstation under test for measuring PPT in a single-hop scenario [11]. In this scheme, a source node directly connected to the host under test exchanges ICMP echo reply and echo request packets between them to collect time stamps both at the data-link and application layers. These time stamps along with the knowledge of transmission speed and the ICMP-packet size are then used to estimate the PPT of the workstation under test. This scheme is not affected by the clock-synchronization between the workstation and PCL; however, the IC at the host is inclusive of the estimated value.

The UDP packet based scheme is designed to measure PPT of a remote workstation in a multiple-hop scenario [10]. This scheme does not require any instrumentation at the data-link layer and clock synchronization since it estimates the capacity of the link connected to the remote workstation (i.e., end link) in order to determine PPT. The end-link capacity is estimated using a packet-pair based probing structure, called compound probe [14]–[16], which shows a linear relationship in its intra-probe gap measured at the remote host. The intra-probe gap is determined from the time stamps generated at the application layer; it contains the transmission time of the trailing (i.e., second) packet of the compound probe plus an offset (i.e., differential PPT) when the heading (i.e., first) and trailing packets arrive at the workstation back to back. In this scheme, a source workstation sends compound probes with two different trailing-packet sizes to the remote workstation over an Internet path for determining PPT from the capacity measurement of the end link. However, the scheme does not consider the effect of IC, which can affect the measurement of PPT under high transmission speeds when the IC is non-negligible.

III. Proposed Scheme

The proposed scheme is an enhancement over the UDP based scheme for measuring PPT of remote workstations over Internet paths. The motivation of the enhanced scheme is to measure PPT when IC is present in the NIC of a workstation under test. This scheme first detects the presence of IC during the measurement process. The enhanced scheme does not estimate IC during PPT measurement because the detection of a non-negligible IC period is sufficient to measure PPT of a workstation under high transmission speed. Therefore, the measurement IC is beyond the scope of this paper.

The enhanced method also uses compound probe that consists of a large heading packet (P_h) followed by a small trailing packet (P_t). If s_t and l_n are the size of P_t and the capacity of the end link of an n-hop path, respectively, the intra-probe gap, $G(s_t)$ of a compound probe is defined as

$$G(s_t) = \frac{s_t}{l_n} + \delta_n + \Delta PPT, \tag{2}$$

where δ_n is the cumulative dispersion (separation) between P_h and P_t, and ΔPPT is the differential PPT for time stamping the compound probe [10]. $\delta_n = 0$ when P_h and P_t arrive back to back at the remote workstation. Figure 2 illustrates a non-zero-slope linear relationship in (2) when $\delta_n = 0$. However, the linear trend in Figure 2 does not hold in the presence of an IC at the workstation.

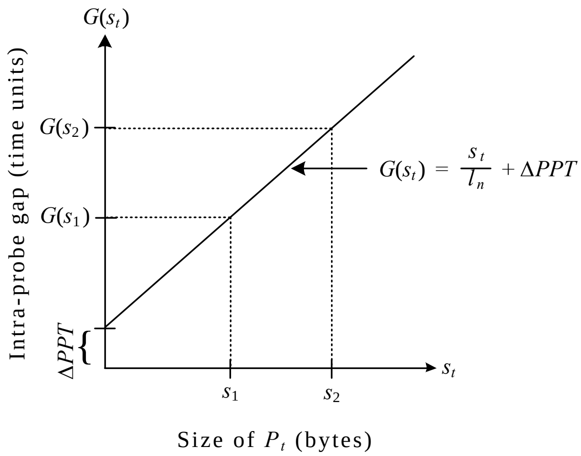


Fig. 2. Non-zero-slope linear relationship between $G(s_t)$ and s_t.

Figure 3 presents the effect of IC and the basic principle of the enhanced scheme for measuring PPT, respectively. Figure 3(a) shows the packet processing events considering a First-In-First-Out (FIFO) queueing model when a compound probe arrives at a remote workstation. Here, P_h and P_t arrive back-to-back during the interval of IC when the NIC is buffering packets in its input queue. After the expiration of the IC period, consecutive time stamps TS_h and TS_t, each with a constant delay, are generated for P_h and P_t, respectively, to acknowledge the packets. Because the time stamping delay corresponds to the traveling delay of a packet from the data-link layer to the application layer, the intra-probe gap is equivalent to the PPT of the workstation, or

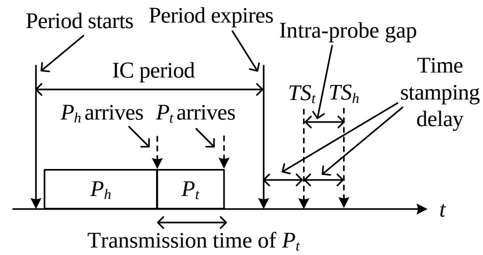
$$G(s_t) = PPT, \tag{3}$$

which shows a zero-slope relationship between the measured gap and s_t.

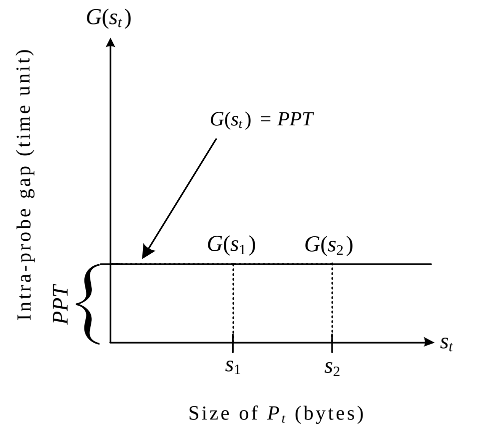
Figure 3(b) illustrates the zero-slop linear relationship between $G(s_t)$ and (s_t), based on (3), considering two different s_ts in the compound probe. This zero-slope linear relationship holds as long as the transmission times of both s_ts are smaller than the packet buffering period, as illustrated in Figure 3(a). Figure 3(b) shows that the PPT of a workstation with an IC period in the NIC can be measured by sending compound probes with more than one s_ts and then by detecting a zero-slope linear trend in the measured intra-probe gaps.

Let's consider a n-hop Internet path between two workstations src and dst. The detailed steps of the enhanced scheme for measuring PPT at dst are presented below:

1) Send a train of compound probes from src to dst using a P_h with a size $s_h = s_{max}$, where s_{max} is 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 probes is obtained. Sizing of s_t in the compound probe for ensuring $\delta_n = 0$ at dst over a multiple-hop scenario can be determined as



(a) Time stamping of compound probe under IC.



(b) Zero-slope linear relationship of intra-probe gaps.

Fig. 3. Effect of IC in PPT measurement: (a) Time stamping of compound probe and (b) Zero-slope linear relationship between $G(s_t)$ and s_t.

$$(\frac{s_h}{l_n} - \frac{s_h}{\alpha l_{n-1}}) + (\frac{s_h}{l_{n-1}} - \frac{s_h}{\alpha l_{n-2}}) + ... + (\frac{s_h}{l_{z+1}} - \frac{s_h}{\alpha l_z}) = 0, \tag{4}$$

where l_z and l_{z+1} are the capacities of the input and output links at a node z over the n-hop path, where P_h and P_t experience the largest dispersion of the path. A detailed discussion on the sizing in a compound probe considering different path parameters is available in [10], [15], [16].

2) Repeat Step 1 using P_ts with another $n-1$, where $n > 1$, different s_ts, such that $s_2 < s_3$ ... $< s_n$. The minimum difference between consecutive s_ts is determined by the resolution of the system clock at dst, e.g., 25 bytes over 100-Mb/s speed [10].

3) Determine the average intra-probe gap, $G_{avg}(s_t)$, of each compound-probe train from the intra-probe gaps measured at dst after discarding the gaps with $\delta_n > 0$ [10].

4) Check if $G_{avg}(s_t)$s for multiple and consecutive s_ts have a zero-slope linear relationship in order to detect IC, such that

$$G_{avg}(s_1) = G_{avg}(s_2) = \ ... \ = G_{avg}(s_n). \tag{5}$$

5) If IC is detected in Step 4, estimate PPT from the constant value of $G_{avg}(s_t)$, e.g.,

$$PPT_{dst} = G_{avg}(s_1). \quad (6)$$

Else, use the non-zero-slope linear relationship between $G_{avg}(s_1)$ and $G_{avg}(s_2)$, as shown in Figure 2, to estimate PPT [10].

IV. Experimental Evaluations

We evaluated the enhanced scheme in a testbed using a single-hop path consisting of a source (src) and a destination (dst) workstations, where dst is under test. We used a single-hop path in experimental evaluation because the performance of the scheme over a multiple-hop path is same if P_h and P_t arrive back-to-back, i.e., $\delta_n = 0$ in (2). Back-to-back arrival of P_h and P_t over multiple-hop paths in a testbed and the Internet under different network conditions has been successfully verified in [10], [16].

A. Testbed Setup and Probing Parameters

In the testbed setup, we use two workstations, Dell Optiplex 790 (DO790) and Dell Studio XPS 435 MT (DS435), as dst with different specifications. Table I presents the specifications of the workstations. The DO790 and DS435 workstations are equipped with integrated gigabit-Ethernet NICs: Intel 82579LM and Intel 82567LF-2, respectively. We measured PPTs of these workstations under three different transmission speeds, 10, 100, and 1000 Mb/s, in Linux environment, where each NIC has a default IC period of 50-250 μs [12].

TABLE I
WORKSTATION SPECIFICATIONS

	Dell Optiplex 790	Dell Studio XPS 435 MT
Name	DO790	DS435
Processor (speed)	Intel Core i3 (3.30 GHz)	Intel Core i7 (2.67 GHz)
RAM (speed)	8148 MB (1333 MHz)	8725 MB (1066 MHz)
NIC type (socket)	Integrated (PCI)	Integrated (PCI)
IC Period	50-250 μs	50-250 μs
Transmission speed	10/100/1000 Mb/s	10/100/1000 Mb/s
Linux kernel version	2.6.18	3.11.0

For PPT measurements, a large heading-packet size, $s_h = 1512$ bytes, and seven trailing-packet sizes, $s_t = \{87, 112, 212, 512, 812, 1012, 1212\}$ bytes, are used in the compound probes. Each of the packet sizes includes 12 bytes of Ethernet encapsulation [10], [16]. s_h is chosen considering the MTU between src and dst, whereas s_ts are chosen in order to have different packet-size ratios in the probing structure for detecting IC. We used 5000 compound probes in 10 iterations for each s_t. In each iteration, compound probes are sent with an interval of 100 ms. Table II shows the theoretical intra-probe gap of the compound probe for each s_t at dst when the transmission speeds of the host under test are 10, 100, and 1000 Mb/s.

TABLE II
THEORETICAL INTRA-PROBE GAPS

Trailing-packet size (bytes)	Theoretical intra-probe gaps = $\frac{s_t}{l_1}$ (μs)		
	10 Mb/s	100 Mb/s	1000 Mb/s
87	69.6	6.96	0.69
112	89.6	8.96	0.89
212	169.6	16.96	1.69
512	409.6	40.96	4.09
812	649.6	64.96	6.49
1012	809.6	80.96	8.09
1212	969.6	96.96	9.69

B. Experimental Results

Figure 4 presents the summary of average intra-probe gaps, $G_{avg}(s_t)$s, measured on DO790, in a logarithmic scale and microseconds, for different s_ts. In this figure, the hollow and the solid circles are the theoretical and estimated $G_{avg}(s_t)$s, respectively. Figure 4(a) shows the experimental results under 10 Mb/s, where the estimated $G_{avg}(s_t)$s have a non-zero linear trend from $s_t = 212$ bytes and above. The non-zero linear trend in the estimated gaps is also similar to that in the corresponding theoretical gaps. However, the estimated gaps for $s_t = \{87, 112\}$ bytes are 3 and 4 μs, respective. These gaps can be considered to have a zero-slope linear trend even though there is a 1 μs variation, which is contributed by the 1-μs clock resolution in the Linux environment [18]. The estimated gaps for the smaller s_ts are also significantly smaller than their theoretical gaps.

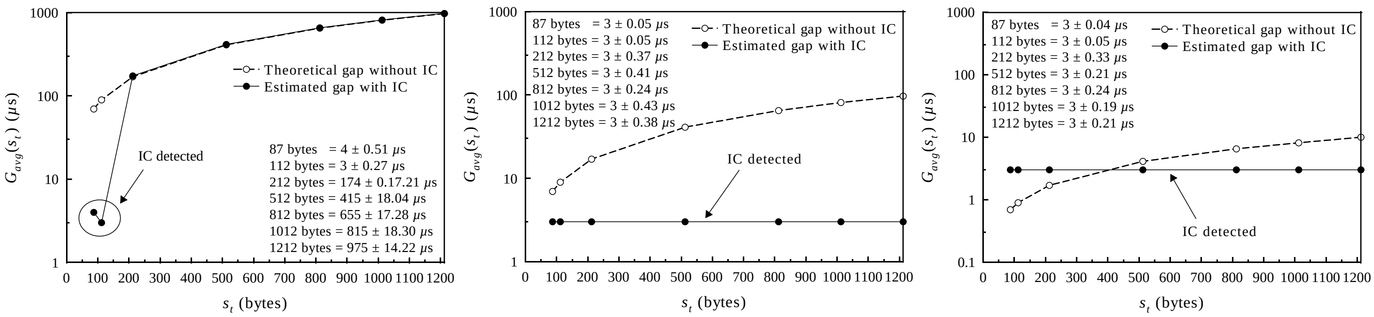
Figures 4(b) and 4(c) present the summary of $G_{avg}(s_t)$s for 100 and 1000 Mb/s, respectively. In these figures, the estimated $G_{avg}(s_t)$s have a zero-slope linear trend for all s_ts with a constant gap of 3 μs. Overall, the zero-slope linear trend in Figure 4 suggests that the enhanced scheme detects the available IC on DO790 successfully and estimates a PPT of 3 μs, according to (6).

The summary of average $G_{avg}(s_t)$s measured on DS435 for different s_ts is presented in Figure 5. Figure 5(a) shows that the estimated $G_{avg}(s_t)$s have a zero-slope linear trend for up to $s_t = 212$ bytes under 10 Mb/s with a constant gap of 11 μs. Figures 5(b) and 5(c) also show a zero-slope linear trend under 100 and 1000 Mb/s, respectively, where the estimated $G_{avg}(s_t)$s is around 11 μs, with a small variation for all s_ts. Based on the non-zero linear trend in Figure 5, the estimated PPT on DS435 for all transmission speeds is 11 μs.

C. PPT-Measurement Accuracy

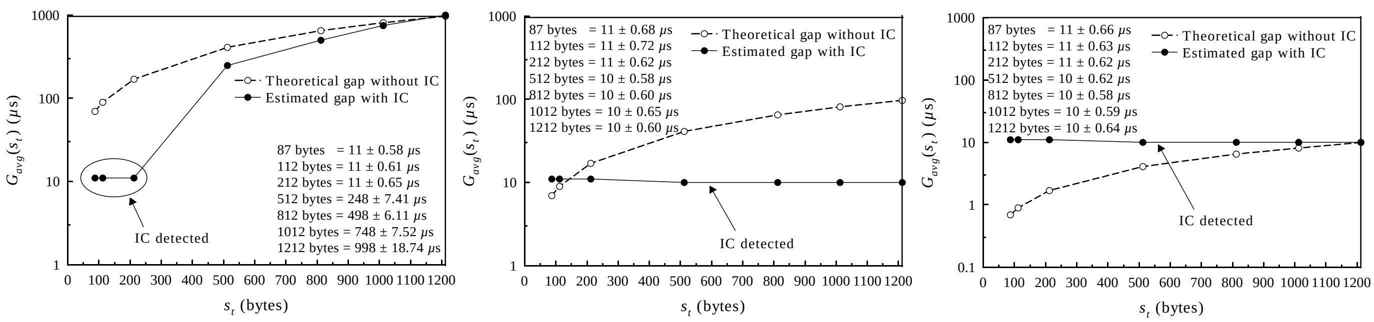
To verify the accuracy of the enhanced scheme, we set the IC period on DO790 and DS435 at a 125 μs and measured their actual PPTs using the ICMP-packet-based scheme, as the reference scheme, which estimates the sum of PPT and IC. Other than the reference scheme, there is no other scheme that measures PPT in the presence of IC according to the best of our knowledge. The actual PPTs are determined after removing the IC from the final estimates.

We verified the enhanced scheme only under 100- and 1000-Mb/s speeds because NICs with low transmission speeds, e.g., 10 Mb/s, does not use IC. The actual PPTs of DO790 under 100 and 1000 Mb/s are 6 μs $\pm$ 25 μs and 9 μs $\pm$ 25 μs, respectively. In case of DS435, these values are 37 μs $\pm$ 31 μs



(a) DO790 under 10 Mb/s (b) DO790 under 100 Mb/s (c) DO790 under 1000 Mb/s

Fig. 4. Summary and scatterplots of average intra-probe gaps (G_{avg}s) in logarithmic scale for different trailing-packet sizes (s_ts) measured on the DO790 workstation under different transmission speeds: (a) 10, (b) 100, and (c) 1000 Mb/s.



(a) DS435 under 10 Mb/s (b) DS435 under 100 Mb/s (c) DS435 under 1000 Mb/s

Fig. 5. Summary and scatterplots of average intra-probe gaps (G_{avg}s) in logarithmic scale for different trailing-packet sizes (s_ts) measured on the DS435 workstation under different transmission speeds: (a) 10, (b) 100, and (c) 1000 Mb/s.

and 20 μs $\pm$ 30 μs. Here, the large standard deviations in the actual PPTs are caused by the variations in the IC and/or ICMP packet generation time of the reference scheme. For example, the variation in the ICMP packet generation time can be in the order of milliseconds [19], [20]. Because the PPTs measured by the proposed method are 3 μs $\pm$ 1 μs and 11 μs $\pm$ 1 μs on DO790 and DS435, respectively, are right within the ranges of the actual PPTs, the experimental results verify that the proposed method consistently measures PPT in the presence of IC with high efficacy and achieves higher resolution than the existing scheme.

V. Utility of IC Detection

Even though the enhanced scheme detects IC in NICs for accurate measurement of PPT, the utility of IC detection is manifold. We discuss some cases below.

The prerequisite of IC measurement over an end-to-end path is that an IC period is detected at a workstation because IC is a configurable parameter of NICs [12]. Detection of IC using active probing, therefore, facilitates IC measurement without requiring any infrastructural support from the workstation under test. Based on the IC-detection method proposed in this work, an active scheme to estimate IC period has been recently proposed [17].

Like PPT, IC is an important parameter in end-to-end OWD measurement, which has myriad of useful applications on the Internet. In Section I, error in OWD measurement is calculated considering negligible ICs, i.e., $IC_{src} = IC_{dst} = 0$, at the end hosts, as illustrated in Figure 1. IC, however, is a non-negligible parameter under high transmission speeds and is a larger parameter than the PPT of the workstations. For example, the magnitude of IC period is 10 times higher that than of PPT at workstations [17]. Therefore, the aforementioned error calculation emphasizes the necessity of estimating IC period during OWD measurement because IC can affect the accuracy of delay measurement.

The congestion-control algorithm in transmission control protocol (TCP) is at the core of successful data communications on the Internet. TCP congestion control uses end-to-end delays to maintain an optimal data-transmission rate between pairs of end hosts [21]. However, the efficacy of the algorithm can be compromised by the presence of IC at the end hosts [4]. IC forces to generate acknowledgements (ACKs) at the receiving hosts in short bursts irrespective of the congestion condition in the network; this phenomenon can ramp up the data-transmission rate in a disproportionate manner [22]. This phenomenon can be avoided if the ICs at the end hosts are detected and estimated during TCP congestion control. For example, the congestion-control algorithm would then use the estimated IC period to filter out the bursty ACKs and produce a consistent performance.

The efficacy of different probe-gap based schemes that measure link bandwidth depends on the accurate time stamping of the probing packets at the end-hosts; however, not all existing schemes incorporate the knowledge of IC during bandwidth measurement to ensure high accuracy [5]. Anomaly in time stamping packets due to the presence of an IC has been already illustrated in Figure 3(a). If IC is detected and estimated a priori, probe-gap based schemes, e.g., bprobe and cprobe [23], nettimer [24], spruce [25], IGI and PTR [26], Pathrate [28], pathChirp [27], and capProbe [29], can design optimal probing-packet and probing-train sizes to produce a high measurement accuracy. Given that bandwidth plays a significant role in characterizing the Internet, detection of IC has a significant importance.

VI. Conclusions

Packet processing time (PPT) of workstations is an important network parameter. The existing scheme for measuring PPT does not consider the effect of interrupt coalescence (IC) in NICs of workstations even though IC is a common hardware artifact under high transmission speeds over Internet paths. In this paper, we proposed an enhancement to the existing scheme that first detects interrupt coalescence to estimate PPT at the workstation under test in the presence of an IC. The enhanced scheme uses the slope of the measured gaps of its packet-pair based probing structure for detecting IC then uses the measured gaps to estimate PPT. We evaluated the scheme through testbed experiments using two different hosts and under three different transmission speeds, e.g., 10, 100, and 1000 Mb/s. The experimental results show that it consistently detects IC and measures PPT with high efficacy without being affected by IC.

References

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

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

[3] A. Hernandez and E. Magana, "One-way delay measurement and characterization," in Proc. of *IEEE International Conference on Networking and Services*, Athens, Greece, 2007, p. 6.

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

[5] G. Jin and B. Tierney, "System capability effects on algorithms for network bandwidth measurements," in Proc. of *ACM Internet Measurement Conference*, FL, USA, 2003, pp. 27–83.

[6] 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 Area of Communications*, vol. 21, no. 6, pp. 908–921, 2003.

[7] 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.

[8] R. Kompella, K. Levchenko, A. Snoeren, and G. Varghese, "Every microsecond counts: Tracking fine-grain latencies with a lossy difference aggregator," in Proc. *ACM SIGCOMM*, Barcelona, Spain, 2009, pp. 255–266.

[9] InformationWeek, (2007). Wall street's quest to process data at the speed of light. [Online]. Available: http://www.informationweek.com/wall-streets-quest-to-process-data-at-the-speed-of-light/d/d-id/1054287?.

[10] 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," *IEEE Transactions on Computers*, vol. 64, no. 1, pp. 205–218, 2015.

[11] K. Salehin and R. Rojas-Cessa, "A method to measure packet processing time of hosts using high-speed transmission lines," *IEEE Systems Journal*, vol. PP, no. 99, p. 1, 2013.

[12] Intel. Interrupt moderation using Intel GbE controllers. [Online]. Available: http://www.intel.com/content/www/us/en/ethernetcontrollers/gbe-controllers-interrupt-moderation-appl-note.html.

[13] Endace DAG 7.5G2 Datasheet. (2014). [Online]. Available: http://www.emulex.com/artifacts/b4469f7d-ecee-4022-8232-295390c7c036/end_ds_all_dag_7.5g2.pdf.

[14] 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.

[15] 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.

[16] 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 Communications*, CA, USA, 2013, pp. 1031–1035.

[17] K. Salehin, V. Sahasrabudhe, and R. Rojas-Cessa, "Remote measurement of the interrupt-coalescence latency of Internet hosts," *under review*.

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

[19] R. Govindan and V. Paxson, "Estimating router ICMP generation delays," in Proc. of *Passive and Active Measurements*, CO, USA, 2002, pp. 1–8.

[20] K. Anagnostakis, M. Greenwald, and R. Ryger, "cing: Measuring network-internal delays using only existing infrastructure," in Proc. of *IEEE INFOCOM*, CA, USA, 2003, pp. 2112–2121.

[21] V. Jacobson, "Congestion avoidance and control," *SIGCOMM Comput. Commun. Rev.*, vol. 18, no. 4, pp. 314–329, Aug. 1988. [Online]. Available: http://doi.acm.org/10.1145/52325.52356.

[22] M. Zec, M. Mikuc, and M. Zagar, "M.: Estimating the impact of interrupt coalescing delays on steady state TCP throughput," in Proc. of *10-th SoftCOM*, 2002.

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

[24] K. Lai and M. Baker, "Nettimer: A tool for measuring bottleneck link bandwidth," in Proc. of *USENIX Symposium on Internet Technologies and Systems*, 2001, pp. 123–134.

[25] J. Strauss, D. Katabi, and F. Kaashoek, "A measurement study of available bandwidth estimation tools," in Proc. of *ACM SIGCOMM Conference on Internet Measurement*, 2003, pp. 39–44.

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

[27] V. Ribeiro, R. Riedi, R. Baraniuk, J. Navratil, and L. Cottrell, "pathChirp: Efficient available bandwidth estimation for network paths," in Proc. of Passive and Active Measurement, vol. 4, 2003, pp. 1–11.

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

[29] R. Kapoor, L. Chen, L. Lao, M. Gerla, and M. Sanadidi, "CapProbe: A simple and accurate capacity estimation technique," in Proc. of *ACM SIGCOMM*, OR, USA, 2004, pp. 67–78.