Usefulness of precise time-stamping for exposing network characteristics on high-speed links

Yoshinori Kitatsuji[a], Masato Tsuru[b], Satoshi Katsuno[c], and Yuji Oie[b]

[a]NICT Kitakyushu JGN2 Research Center, Fukuoka, Japan;
[b]Dept. of Computer Science and Electronics, Kyushu Institute of Technology, Fukuoka, Japan;
[c]KDDI R&D Laboratories, Inc., Saitama, Japan

ABSTRACT

To expose network characteristics by active/passive measurements, measuring some timing issues such as one-way delay, one-way queuing delay, and inter-packet time is essential, and is conducted by time-stamping for packets passing through an observation point. However, emerging high-speed networks require very high precision of time-stamping, far beyond the precision of conventional software-based time-stamping systems such as 'tcpdump'. For example, the inter-packet time of two consecutive 64-byte length packets on a giga-bit link can be less than 0.001 msec. In this paper, to demonstrate the usefulness and strong necessity of precise time-stamping on high-speed links, experiments of network measurements over a nation-wide IPv6 testbed in Japan have been performed, using a hardware-based time-stamping system that can synchronize to GPS with a high resolution of 0.0001 msec and within a small error of 0.0003 msec. In our experiments, several interesting results are seen, e.g., i) the distribution of one-way queuing delay exhibits a considerable difference depending on the size and the type (UDP/ICMP) of packets; ii) the minimal one-way delays for various sizes of UDP/ICMP packets give an accurate estimate of the transmission delay and the propagation delay; iii) the correlation between inter-packet times at the sender and the receiver sides in a sequence of TCP ACK packets clearly shows the degree of ACK compression; iv) the inter-packet time in a UDP stream generated by a DV streaming application shows three dominant sending rates and a very rare peak rate, which might provide crucial information to bandwidth dimensioning; all of which would indicate the usefulness of precise time-stamping.

Keywords: Network measurement, High-speed network, Timestamp, GPS, IPv6

1. INTRODUCTION

Since the Internet has already become an indispensable infrastructure for social and economic activities, it needs to be operated in a reliable and efficient way, and thus should be measurable in terms of its characteristics. What to be measured would be classified into two types of characteristics: one is the quality (such as performance) of individual end-to-end communications over a network, and the other is the condition of the network itself, which includes local states of network-internal portions and global behaviors of traffic flows over the network. Knowledge of both types of characteristics is of practical importance not only for reliable, efficient and QoS-aware network operations (in a static or a dynamic way) but also for research and development for new network technologies, which need understanding the hidden nature of networks.

In order to expose such network characteristics by active and/or passive measurements, measuring some timing issues such as RTT (round-trip-time), one-way delay, one-way queuing delay, and inter-packet time (the time interval between consecutive packets on a traffic flow) is essential, and is conducted by time-stamping, that is, recording the time of a packet passing through an observation point with the header and/or contents of the packet (or some information to identify

Further author information: (Send correspondence to Yoshinori Kitatsuji)
Yoshinori Kitatsuji: kitaji@kyushu.jgn2.jp (Address: AIM bldg. 7F, 3-8-1 Asano, Kokurakita-ku, Kitakyushu-shi, Fukuoka 802-0001, JAPAN), Masato Tsuru: tsuru@cse.kyutech.ac.jp, Satoshi Katsuno: katsuno@kddilabs.jp, Yuji Oie: oie@cse.kyutech.ac.jp.

the packet). However, emerging high-speed networks require very high precision of time-stamping. For example, the inter-packet time of two consecutive 64-byte length packets on a giga-bit link can be less than 0.001 msec (about 0.0005 msec), and the maximum queuing delay experienced at a 50-Kbyte (50 packets of 1000 bytes) length output buffer on a ten giga-bit link is about 0.04msec, both of which are far beyond the precision of conventional software-based time-stamping systems such as 'tcpdump'.

There are two kinds of errors in time-stamping. One is the error derived from the inaccuracy and imprecision of clocks, e.g., resolution, offset, skew, or drift. The other is the difference between the exact instance of packet arriving and the instance of referring and recording the time by the clock, that is, processing delay in a time-stamping system. The impact of those errors to the final results depends on what we would like to measure: RTT, one-way delay, one-way queuing delay, or inter-packet time. In fact, the one-way delay experienced on a path from point A to point B is measured as a difference between the time of a packet arriving at A by the clock of A and the time of the packet arriving at B by the clock of B, and thus, the absolute value of measured one-way delay is meaningless unless the offset between two clocks is negligible. To eliminate such an offset, two clock should accurately synchronize to some common precise time source such as GPS (Global Positioning System). On the other hand, since one-way queuing delay from A to B is estimated as a difference between the each value in measured one-way delays from A to B and the minimal value in them, accuracy of the estimation strongly depends on the skew (and drift) between two clocks at A and B, regardless of the clock offset.

In general, unfortunately, the conventional software-based time-stamping system such as 'tcpdump' on an off-the-shelf PC (whose clock is usually based on an unstable crystal oscillator) would be insufficient in accuracy of the time-stamping on high-speed links. For example, we previously evaluated the accuracy of one-way queuing delays measured by 'tcpdump' (after removing clock skew between two measurement PCs by some calibration algorithm) by comparing with the one-way queuing delays measured by a hardware-based passive monitoring device having the time-stamping function by a very stable (very little skewed) clock with 0.05 msec resolution.[1] In the experiment, we found the errors in one-way queuing delays measured by 'tcpdump' were likely to be less than 0.3 msec, which might come from the residual skew and the variation of processing delays in two PCs.

We should note that some study proposed an approach to precise time-stamping on two off-the-shelf PCs without GPS, by using a hardware register counting CPU cycles.[2] This approach, however, focuses on accurately calibrating clock skew, but not clock offset. In addition, errors from the variation of processing delays in PCs still remain. Some other study pointed out the limitation of off-the-shelf PC based measurement systems for network bandwidth estimation.[3]

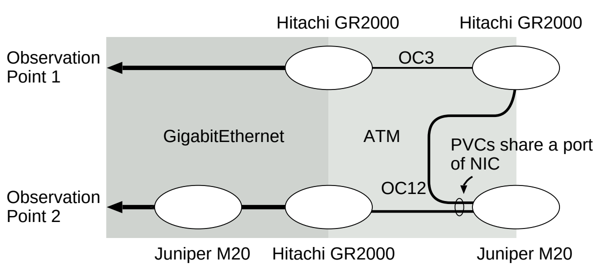
In this paper, therefore, to demonstrate the usefulness and strong necessity of the hardware-based precise time-stamping on high-speed links conveying high-speed application traffic, experiments of network measurements over a nation-wide IPv6 testbed are conducted. We employ the High-speed IP Meter (HIM), a hardware-based time-stamping system developed by KDDI R&D Laboratories, Inc. and Hitachi Ltd., which can synchronize to UTC (Universal Time) by using GPS receiver with a high resolution of 0.0001 msec (100 nano-sec) and within a small error of 0.0003 msec.[4] A HIM captures each packet (after being filtered by some configurable rules) on a giga-bit ethernet link, and sends a copy of the packet with a 64-bit accurate time-stamp embedded in a jumbo frame of ethernet to a storage PC connected via ethernet. Furthermore, if a HIM fails to synchronize to GPS, the de-synchronizing and re-synchronizing times will be recorded to ensure the integrity of the measured data. The storage PC stores the packet information in the tcpdump-compatible format, which allows us to employ tcpdump and a variety of tcpdump-compatible tools in off-line visualization and analysis. For packet capturing, a HIM can be operated either in pass-through mode (in which the original packets passing through the HIM are measured) or in mirroring mode (in which the copies of the original packets fed by a mirror port of a switch/router are measured). Note that, precisely speaking, the time-stamp by HIM refers to the arrival time of the last bit in a packet, although some standards (e.g., RFC2679[5]) require the ability of time-stamp referring to the arrival times of both the first bit and the last bit.

Several studies on network measurements using accurate and precise time-stamping have already been reported, especially by using a DAG measurement card, which is a special purpose network interface card (NIC) with accurate time-stamping that can synchronize to GPS.[6–8] For example, a highly accurate packet proving experiment was conducted, which was applied to bottleneck bandwidth estimation based on precise one-way delay measurement.[9] Along this line, we report our experiment on network characteristics measurements requiring the precise time-stamping on an IPv6 environment in Japan as described in Sect. 2. Unlike the studies previously reported, our focus here is on demonstrating the usefulness of precise time-stamping on high-speed links instead of on developing a specific measurement method for a specific network characteristic. Thus, our experiments are covering all of the three basic types of time-stamping related measurements, i.e., one-way delay, one-way queuing delay, and inter-packet time, where some interesting results are seen as described in Section 3. Finally Sect. 4 concludes this work.

2. MEASUREMENT ENVIRONMENT

To expose network characteristics by measuring some timing issues such as one-way delay, one-way queuing delay, and inter-packet time, we settle two High-speed IP Meter (HIM) on giga-bit ethernet links near to both ends of a path between observation points 1 (Kitakyushu) and 2 (Tokyo), which traverses a nation-wide IPv6 testbed on JGN (Japan Gigabit Network) * in Japan,

Figure 1 shows the whole path configuration. The path consists of five high-performance core routers (Hitachi and Juniper) and various high-speed links of 100 base-TX, 1000 base-SX, OC-3/ATM, and OC-12/ATM, and is, on the average, lightly loaded by traffic in some other experiments on the testbed. In the detailed configurations of two points illustrated in the right-hand side of Fig. 1, "Traffic Generator" is a PC for sending/receiving the targeted traffic, on which *ping6* command, *iperf* command † and *DVTS* ‡ (Digital Video Transport System, an IEEE 1394 digital-video format streaming application)[10] are running. Observation point 1 and 2 are the sender-side and the receiver-side, respectively. "IPmeter" represents a main-component (which measures the targeted traffic) of a HIM with GPS antenna and a storage PC as a sub-component for storing the measured data. "TCPDUMP" means a PC for performing "tcpdump" just to confirm the (in)accuracy of time-stamping performed by tcpdump. In the next section, however, we will omit showing the results of tcpdump measurements. The results definitely indicate that tcpdump on an off-the-shelf PC is not suitable for measuring timing issues on high-speed links, even though the PC is a high-performance Pentium machine operated by FreeBSD OS, due to the errors that might be a 0.1 msec order.



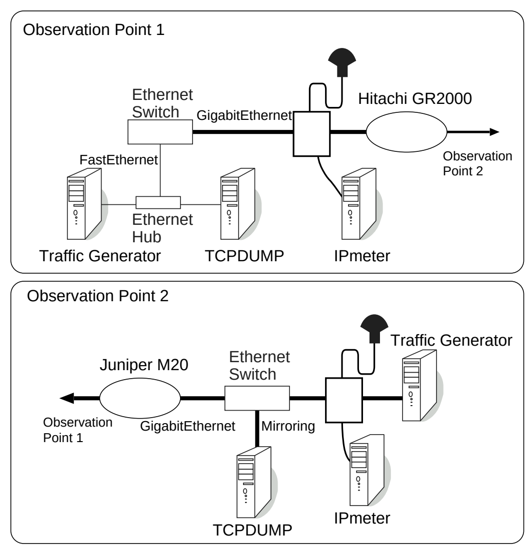


Figure 1. Path configuration between two observation points.

We conduct three types of measurements as follows.

- Measuring one-way delay for various sizes of UDP and ICMP packets by using *iperf* command for sending/receiving UDP packets and *ping6* command for sending/receiving ICMP echo packets. The arrival times of each UDP/ICMP packet observed by two HIMs at point 1 and point 2 are gathered, and the one-way delay of the packet is calculated. From the measured (absolute) one-way delays, we estimate one-way queuing delay as a difference between the each value in measured one-way delays and the minimal value in them.

*http://www.jgn.nict.go.jp/english/index_E.html
†http://dast.nlanr.net/Projects/Iperf/
‡http://www.sfc.wide.ad.jp/DVTS/index.html

Note that a UDP packet consists of 40-byte IPv6 header (without any option), 8-byte UDP header, and the payload. Similarly, an ICMP packet consists of 40-byte IPv6 header, 8-byte ICMP header, and the payload. The payload length varies from 100 to 1400 bytes. For UDP packets, while the stream with small 100 byte payload is sent at a high rate of 1 Mb/s from point 1 to point 2, the streams with other payload sizes are sent at a moderate rate of 100 Kb/s. For ICMP packets, one echo request packet is sent per 0.1 sec (i.e., 10 pps) for 1000 seconds and flies from point 1 to point 2, and then the corresponding echo reply packet flies from point 2 to point 1.

- Measuring inter-packet time for a bi-directional TCP packet stream as an elastic data transfer generated by *iperf* command. The IP packet size of the forward stream (the sequence of TCP data packets) is mainly equal to 1488 bytes, while that of the backward stream (the sequence of TCP ACK packets without data) is always equal to 60 bytes (40-byte IPv6 header $+$ 20-byte TCP header).
- Measuring inter-packet time for a uni-directional UDP packet stream generated by *DVTS*. The IP packet size of the stream is likely equal to 1354 bytes but sometimes 540 bytes.

3. EXPERIMENTAL RESULTS

3.1. One-way delay for UDP/ICMP packets

We show the results of one-way delay measurements for various sizes of UDP and ICMP packets.

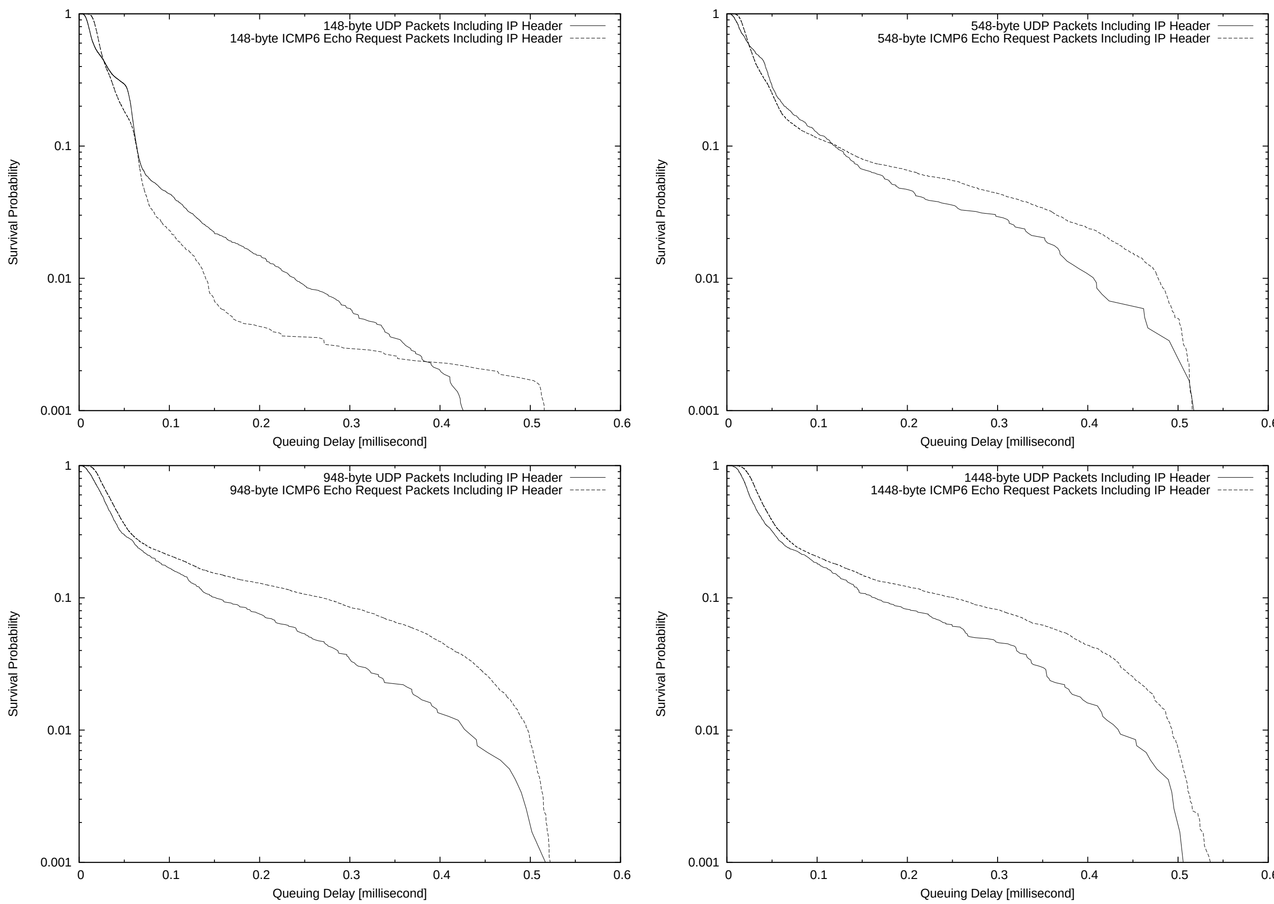


Figure 2. Survival distribution of one-way queuing delay of UDP and Echo request.

Table 1. Mean, standard deviation, and 99-tile of one-way queuing delay of UDP and Echo request with the bit and packet rates.

size / type	bit rate [b/s]	packet rate [pps]	mean [msec]	stdev [msec]	99-tile [msec]
148/udp	1050K	1300	0.0362	0.0432	0.239
148/icmp6	12K	10	0.0350	0.0386	0.140
548/udp	107K	27	0.0546	0.0743	0.410
548/icmp6	44K	10	0.0580	0.0882	0.478
948/udp	106K	15	0.0622	0.0845	0.433
948/icmp6	76K	10	0.0857	0.114	0.498
1448/udp	104K	9	0.0664	0.0892	0.436
1448/icmp6	116K	10	0.0860	0.111	0.494

First, the distributions of one-way queuing delays are examined. The top left, top right, bottom left, and bottom right of Fig. 2 show the survival distributions of one-way queuing delay of UDP and ICMP echo request packets with payload of 100, 500, 900, and 1400 bytes, respectively. Table 1 shows the mean, standard deviation, and 99-percentile of the one-way queuing delay with the bit and packet rates. Those results indicate that the distribution of one-way queuing delay exhibits a considerable difference depending on the size and the type (UDP/ICMP) of packets. For example, we can see the packet size dependence in queuing delay of UDP streams of 148 byte and 548 (or more) byte packets, where the delay clearly increases as the packet size increases, although the bit rate of 148 byte packets is ten times larger than that of 548 (or more) byte packets. In addition, we also see the packet type dependence in the queuing delay of UDP and ICMP streams of 148 byte, 548 byte, 948 byte, and 1448 byte packets. On one hand, the mean values of the queuing delay of UDP and ICMP streams of 148 byte packets are relatively close, although the bit rate of UDP stream is a hundred times larger than that of ICMP stream. On the other hand, the ICMP stream of 548 (or more) byte packets seems to experience a longer queuing delay compared with the UDP stream of the same size packets, although the bit and packet rates of the ICMP stream are lower than or nearly equal to those of the UDP stream. Although we suspect that there may exist a difference in the queuing process in some routers depending not only on the size but also the type of packets, further investigation should be required.

Next, the minimal one-way delays for various sizes of UDP/ICMP packets are examined. Figure 3 shows the minimal one-way delays for various sizes of packets form a clear straight line, from which the transmission delay (proportional to the size of a packet) and the propagation delay (independent of the size of a packet) can be precisely estimated. Since they reflect the path-internal structure (the former is related with link bandwidth of each link and the latter is mainly affected by the physical distance of each link), the precise time-stamping might be useful for detecting route changes or hidden store-

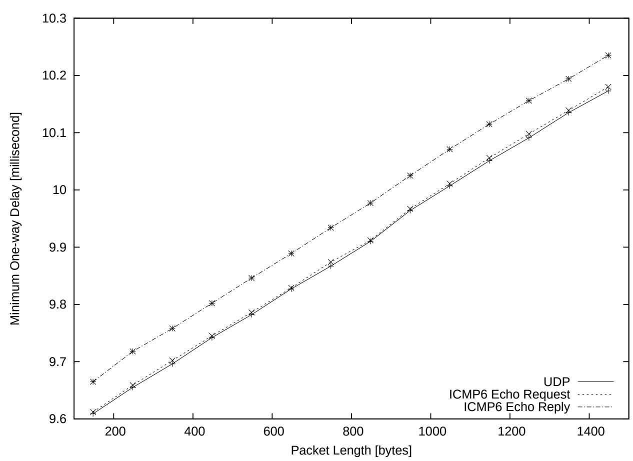


Figure 3. Minimal one-way delays for various sizes of UDP and Echo request packets on the forward path and those of Echo reply packets on the backward path.

and-forward devices. For example, in Fig. 3, while the lines produced by UDP and ICMP echo request on the forward path are almost consistent, the line by ICMP echo reply on the backward path has the same incline but is shifted in parallel with an offset of about 0.06 msec. This, at least, implies some asymmetric property related with the propagation delay of the paths.

3.2. Inter-packet time for a bi-directional TCP packet stream

We measure the time interval of each pair of consecutive packets in one direction at the sender side and that at the receiver side, and compare them.

The left-hand side of Fig. 4 shows the distributions of inter-packet time for a TCP data (forward) stream at the sender and the receiver sides. The shortest inter-packet time of TCP data packets is about 0.12 msec, which is most dominant (more than 60 %) at the sender-side. This might come from consecutive 1488-byte packets at the maximum rate of 98 Mb/s corresponding to the back-to-back packets sent from an interface of 100 base-TX. The next shortest and dominant inter-packet time at the sender-side is about 0.2 msec. These two dominant inter-packet times, at least, imply the bursty nature of TCP data sending mechanism.

The right-hand side of Fig. 4 shows the correlation between inter-packet times at the sender and the receiver sides. Apart from the case of back-to-back packets that have an initial inter-packet time less than 0.2 msec at the sender-side, it

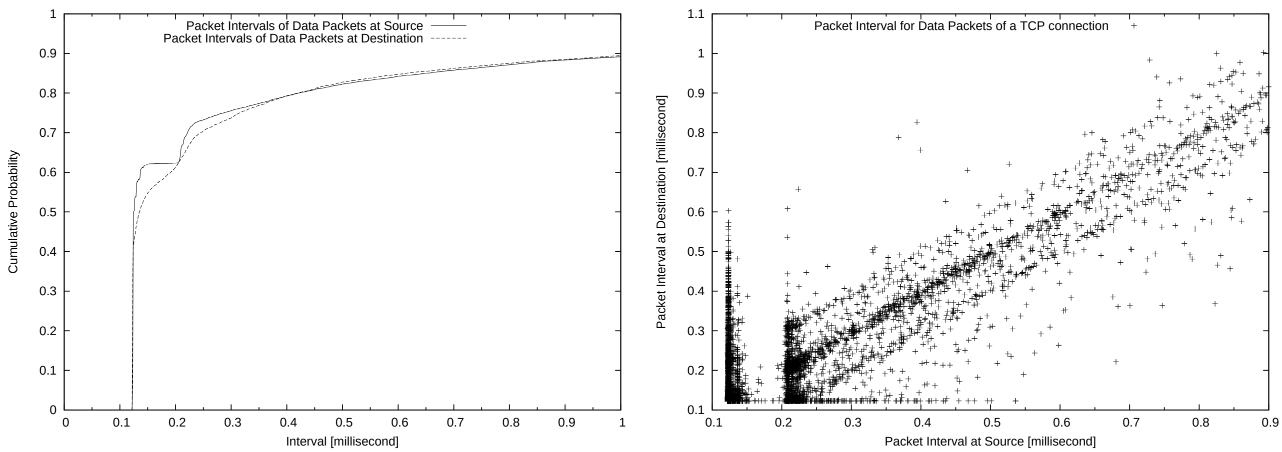


Figure 4. Inter-packet time for a TCP data stream at the sender-side and the receiver-side.

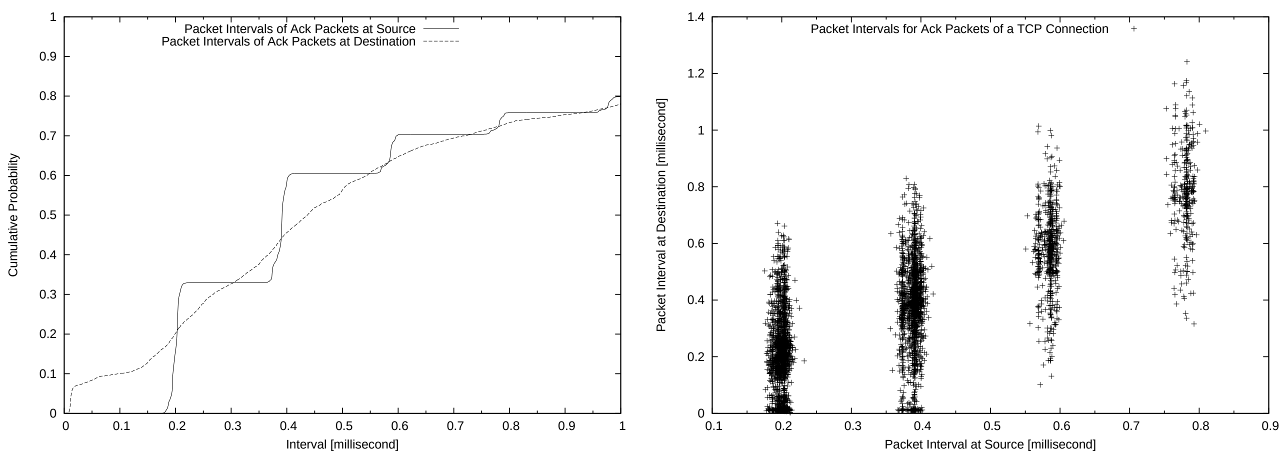


Figure 5. Inter-packet time for a TCP ACK stream at the sender-side and the receiver-side.

can be seen that the inter-packet time of two sending packets tends to increase or decrease in a degree of 0.1 msec at the receiver-side. The increment and the decrement might imply that some packets are likely to experience an additional delay of 0.1 msec.

The left-hand side of Fig. 5 shows the distributions of inter-packet time for a TCP ACK (backward) stream at the sender and the receiver sides, while the right-hand side shows the correlation between inter-packet times at the sender and the receiver sides. From these figures, the inter-packet time of TCP ACK stream at the data receiver (i.e., the ACK sender) is likely to be 0.2, 0.4, 0.6, or 0.8 msec. One possible scenario is that the inter-arrival time of TCP data at the receiver-side TCP stack would be 0.2 msec (although the most dominant inter-packet time of TCP data at the receiver-side network is 0.12 msec as shown in Fig. 4), and the receiver TCP would send back an ACK every one, two, three, or four data packets received. On the other hand, at the data sender (i.e., the ACK receiver), more than 5% of TCP ACK packets have very short inter-packet times (less than 0.01 msec), which clearly indicates the degree of ACK compression.

3.3. Inter-packet time for a uni-directional UDP packet stream by a DV application

Figure 6 shows the distributions of inter-packet time in a UDP stream at the sender and the receiver sides, generated by a DV streaming application. At the sender-side, the inter-packet time of 0.37, 0.25, and 0.5 msecs, that is, three levels of sending rates (29 Mb/s, 43 Mb/s, and 22 Mb/s), are significantly dominant. The sub-figure within Fig. 6 shows the distribution of very few packets with nearly peak rates, which indicates that the peak rate of 98 Mb/s (limited by the sender's network interface) rarely appears. To know about such properties of the sending rate of high-speed applications might provide crucial information to bandwidth dimensioning.

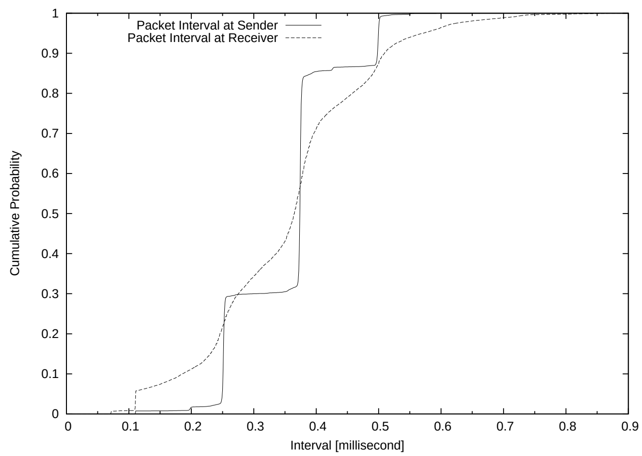


Figure 6. Inter-packet time for a DV streaming application at the sender-side and the receiver-side.

4. CONCLUSION

We conducted experiments on network characteristic measurements using a hardware-based precise time-stamping system, which could synchronize to GPS with a high resolution of 0.0001 msec and within a small error of 0.0003 msec.

From the measured data, several interesting results were obtained: i) the distribution of one-way queuing delay exhibits a considerable difference depending on the size and the type (UDP/ICMP) of packets; ii) the minimal one-way delays for various sizes of UDP/ICMP packets could give an accurate estimate of the transmission delay (proportional to the size of a packet) and the propagation delay (independent of the size of a packet), which might be useful for detecting route changes or hidden store-and-forward devices; iii) the correlation between inter-packet times at the sender and the receiver sides in a TCP data stream indicated the bursty nature of TCP data sending mechanism, and the correlation between inter-packet times at the sender and the receiver sides in a TCP ACK stream clearly showed the degree of ACK compression; iv) the distribution of inter-packet time in a UDP stream generated by a DV streaming application at the sender-side indicated that three levels of sending rates (29 Mb/s, 43 Mb/s, and 22 Mb/s) were completely dominant, and the peak rate of 98 Mb/s (limited by the sender's network interface) rarely appeared.

Although the further investigation on these results remains at this time, they would, at least, indicate the usefulness and strong necessity of precise time-stamping with high resolution on current and future high-speed links conveying high-speed application traffic.

ACKNOWLEDGMENTS

We would like to thank Katsuyuki Yamazaki at KDDI R&D Laboratories, Inc., Kiminori Sugauchi at Hitachi Ltd., and Jyunichi Kuramatsu at Kyushu Institute of Technology for their kindly cooperation in the experiment. This work was supported in part by the Japan Society for the Promotion of Science, Grant-in-Aid for Scientific Research (A), 15200005.

REFERENCES

1. M. Tsuru, N. Ryoki, and Y. Oie, "On the correlation of multiple path properties simultaneously measured on the internet," in *Proc. SPIE ITcom*, **5244**, pp. 206–213, (Orlando, USA), Sept. 2003.
2. A. Pasztor and D. Veitch, "PC based precision timing without GPS," in *Proc. ACM SIGMETRICS*, (Marina Del Rey, USA), June 2002.
3. G. Jin and B. Tierney, "System capability effect on algorithms for network bandwidth," in *Proc. ACM SIGCOMM Internet Measurement Conference*, (Miami, USA), Oct. 2003.
4. S. Katsuno, K. Sugauchi, O. Tsunehiro, K. Yamazaki, K. Yoshida, and H. Esaki, "Asymmetric characteristics of internet based on traffic measurement and analysis," *IEICE Trans. Inf. Syst.* **E86-D**, Nov. 2003.
5. G. Almes, S. Kalidindi, and M. Zekauskas, "A one-way delay metric for IPPM," RFC 2679, Network Working Group, Sept. 1999.
6. J. Micheel, I. Graham, and S. Donnelly, "Precision timestamping of network packets," in *Proc. ACM SIGCOMM Internet Measurement Workshop*, (San Francisco), Nov. 2001.
7. K. Mochalski, J. Micheel, and S. Donnelly, "Packet delay and loss at the auckland internet access path," in *Proc. PAM workshop*, (Fort Collins, Colorado), Mar. 2002.
8. D. Agarwal, J. M. Gonzalez, G. Jin, and B. Tierney, "An infrastructure for passive network monitoring of application data streams," in *Proc. PAM workshop*, (La Jolla, California), Apr. 2003.
9. A. Pasztor and D. Veitch, "High precision active probing for internet measurement," in *Proc. INET*, (Stockholm), June 2001.
10. K. Kobayashi, A. Ogawa, S. Casner, and C. Bormann, "Rtp payload format for dv (iec 61834) video," rfc 3189, Network Working Group, Jan. 2002.