PathRefiner: Accurate Bandwidth Estimation Using Large-Sized Virtual Packets for High-Speed Networks

Jingcheng Huang
Nagoya University
Aichi, Japan
huang.jingcheng.p4@s.mail.nagoya-u.ac.jp

Naoyuki Ito
Nagoya University
Aichi, Japan
ito.naoyuki.w5@s.mail.nagoya-u.ac.jp

Takashi Oshiba
NEC Corporation
Kanagawa, Japan
oshiba@nec.com

Kozo Satoda
NEC Corporation
Kanagawa, Japan
satoda@nec.com

Tutomu Murase
Nagoya University
Aichi, Japan
tom@itc.nagoya-u.ac.jp

***Abstract*—To estimate available bandwidth of high-speed lines with high accuracy, we propose PathRefiner, a bandwidth estimation method with two techniques: (1) packet concatenation technique (2) iterative resolution refinement technique. It is one of the so-called packet train methods that estimate available bandwidth by transmitting multiple probing packets. (1) The packet concatenation technique virtually generates large packets necessary for estimating available bandwidth on high-speed lines by concatenating multiple packets of MTU size or smaller. (2) The iterative resolution refinement technique improves the resolution of estimation and enables highly accurate estimation of available bandwidth while minimizing the network load for estimation by finding the optimal number of large-sized virtual packets to be concatenated. As a result, the average value of the relative error between the ground truth and the estimated available bandwidth is relatively small (14.4%), confirming that PathRefiner can accurately estimate the available bandwidth of 1 Gbps high-speed lines.**

Keywords—available bandwidth estimation, packet train, high-speed networks, high-accuracy

I. Introduction

For users and service providers, it is significant to monitor network conditions to obtain desirable service quality by sending appropriate traffic. With the wide variety of applications and services appearing in today's Internet, network loads and low bottleneck bandwidth are unpredictable. If it is possible to know how much bandwidth can be used (available bandwidth [1]) before it is consumed, it will be possible to adjust service and application parameters according to the available bandwidth, and more useful excellent streaming applications and cloud applications can be introduced into networks. Available bandwidth estimation methods have been proposed, for example [2][3][4][5][6][7][8] using packet trains [9] to easily investigate the available bandwidth without burden on the network.

However, these methods have difficulties in supporting the 1Gbps class high-speed lines which have become popular in recent years. The reason for this is that in high-speed lines, it is necessary to transmit large-sized packets corresponding to the line speed at short intervals. Since the transmission interval depends on the processing performance of the terminal, the packet size should be increased. But it is difficult to increase the packet size due to the MTU limitation. In addition, since the available bandwidth to be estimated becomes large in high-speed lines, the resolution is lower, i.e., the estimation is coarser and the error is larger. Although some methods have been proposed to improve accuracy by reducing the estimation range, they still cannot correspond to high-speed lines [10][11][12]. Here, the resolution corresponds to the scale unit of the measurement instrument and represents the limit of the estimation accuracy.

We propose the PathRefiner method, which has two functions: (1) packet concatenation and (2) iterative resolution refinement, as a method for estimating available bandwidth with high speed and high accuracy. PathRefiner follows the estimation method of PathQuick3 [2], which is a packet train method for estimating available bandwidth. (1) The packet concatenation technique has the ability to virtually generate large-sized packets required for high-speed line estimation by concatenating multiple packets of MTU or less. (2) The iterative resolution refinement technique has the function to improve the resolution while minimizing the network load for estimation by finding the optimal number C of large size virtual packets to be concatenated to improve the estimation accuracy.

Section 2 describes the conventional technology and its issues, Section 3 explains the proposed method, Section 4 evaluates the effectiveness of the proposed method through experiments on actual equipment, and Section 5 provides a summary.

II. Conventional Technology and Its Problems

This section first describes general techniques for measuring and estimating available bandwidth. Next, the principles and specific examples of estimating available bandwidth using conventional packet train techniques are described. Finally, the challenges of conventional techniques are described.

A. General techniques for measuring and estimating available bandwidth

Available bandwidth is the unused bandwidth remaining after subtracting the bandwidth actually used from the line capacity [1]. Since available bandwidth is theoretically unused bandwidth, it is different from bandwidth that can be used.

However, it is difficult to uniquely calculate bandwidth that can be used because it depends on the transport protocol and the data delivery pattern of the application. Here, the available bandwidth shall be used as one indicator of the bandwidth that can be used by the application.

Techniques for measuring or estimating available bandwidth include measuring available bandwidth from packets passing through routers and switches, such as MRTG [13], and a technique called active estimation, which sends inspection packets and estimates available bandwidth based on delay and other factors. Measurement methods such as MRTG are not useful if the bottleneck on the path is not known beforehand, because only the information of the link directly linked to the node such as the switch can be obtained, and the bottleneck point from an end to another is unknown. In addition, MRTG is often used only by network administrators, making it difficult for general users to use. Several active estimation methods have been proposed, including iperf [14], Kite[13] [15], and the techniques using packet trains such as PathQuick3 [2][3], pathChirp [4], Pathload [5][6][7][8], in which three or more UDP packets constitute a packet sequence. However, since iperf occupies a large amount of network resources, it adversely affects the communications of other users. Kate needs a lot of time to complete the estimation and cannot achieve accurate estimation in a short period of time. In the following, we will limit our discussion to packet-train-based available bandwidth estimation techniques that estimate available bandwidth by instantaneously congesting the network.

B. *Estimation of available bandwidth using packet trains*

The principle of estimating the available bandwidth of a packet train system is as follows.

If packets are sent at a certain interval and the reception interval is the same as the transmission interval, it means that all packets in the packet train arrived without queuing delay in the network. On the other hand, if the interval differs between transmission and reception, it means that packets suffered queuing delays due to instantaneous congestion. The aforementioned study is performed by increasing the per-packet transmission rate accordingly. The transmission rate obtained by dividing the packet size at the time of instantaneous congestion by the transmission interval is the estimated available bandwidth.

C. *Mechanism of available bandwidth estimation in PathQuick3*

PathQuick3 is a packet train method that can estimate the available bandwidth with high accuracy, short time, and low load. It has already been shown to provide highly accurate estimation when the available bandwidth is less than 100 Mbps [3]. PathQuick3 first performs delay detection as described in the previous section, then performs fitting and calculates the estimated value.

PathQuick3 linearly increases the per-packet transmission rate by linearly increasing the size of each packet in the packet train, as shown in Fig. 1 The transmission rate at which queuing delays begin to occur is used as an estimate of the available bandwidth. For example, as shown in Fig. 2, when the packet size is increased by g Bytes, the per-packet transmission rate exceeds the available bandwidth for the first time with the fourth packet; up to the third packet, the receive interval and transmission interval are the same because no queuing delay occurs. On the other hand, since the per-packet transmission rate for the fourth and subsequent packets exceeds the available bandwidth, congestion occurs instantaneously at the router or switch at the bottleneck point on the network path, and the packets suffer queuing delays. As a result, a longer receive interval is observed than the interval up to the third packet. The transmission rate of the third packet just before the reception interval begins to widen is used as the estimated available bandwidth.

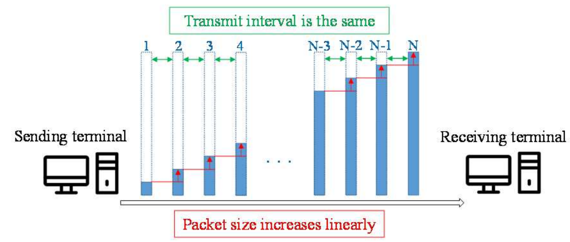

Fig. 1. Packet transmission with PathQuick3

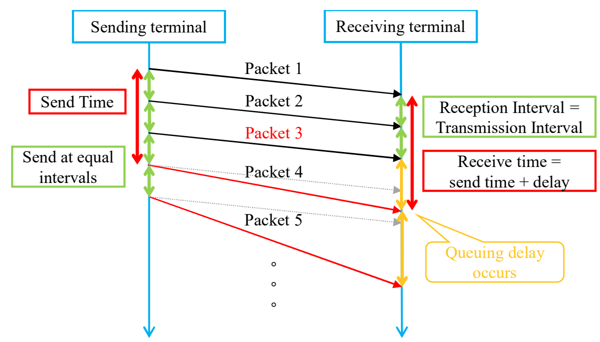

Fig. 2. Queuing delay caused by PathQuick3

Here, the queuing delay includes delays that are noise to the above estimation principle due to cross-traffic fluctuations and buffer control at switches, etc., so it is necessary to compensate for this. Therefore, a graph of the theoretically derived delay series [2] under ideal noise-free conditions is prepared in advance, and the effect of noise is mitigated by fitting the observed delays using the least squares method. In Fig. 3, the logical delay (delay point) is depicted for each i, assuming that no delay is observed up to the i-th packet and that queuing delay is observed starting with the i-th packet. The dotted network connecting the points calculated for each i (dotted line in Fig. 3) is called the delay point line. The logical delay is calculated by computation [3]. The dotted line with the smallest squared error between the delay point of the j-th packet and each point of the delay observed in the j-th packet (marked with X in Fig. 3) on the delay point line is selected. For example, Fig. 3 illustrates the fitting estimation when the available bandwidth is 46 Mbps: the squared error between the estimated packet and the dotted line with $i = 6$ is the smallest compared to the other packet dotted lines when $i = 6$, resulting in an estimate of 48 Mbps for a 46 Mbps available bandwidth. As in this example, an estimate of 46 Mbps cannot be obtained. In this case, the error is 2 Mbps (= 48 - 46 Mbps). Similarly, if the available bandwidth is 52 Mbps, the estimate would be 48 Mbps. In this example, the resolution determined by the incremental packet size used would be 4 Mbps (= 52 - 48 Mbps). When the resolution is 4 Mbps, the maximum error due to the resolution is also 4 Mbps.

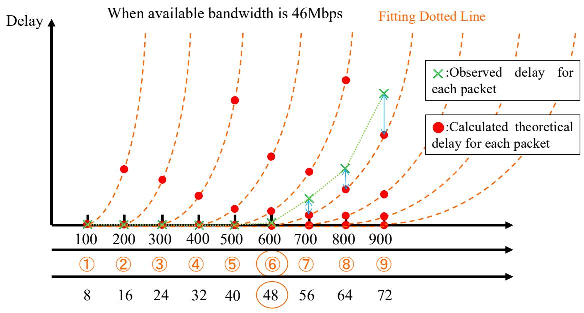

Fig. 3. Fitting with PathQuick3

D. *Issues with conventional methods*

Conventional packet train methods, including PathQuick3, have two problems in estimating the available bandwidth on 1 Gbps-class high-speed lines: first, they cannot support high-speed lines and cannot set the upper limit of estimation to a high value such as 1 Gbps; The second is the "problem of high-precision estimation", which is that the estimation resolution is lower for 1 Gbps-class high-speed lines than for 100 Mbps-class lines, resulting in a larger estimation error. This is explained below.

1) *Issues with high-speed line support*

To set the upper estimation limit to a higher value, the interval between packet transmissions must be shortened or the packet size must be increased. However, since the packet transmission interval depends on the processing performance of the terminal, there is a limit on how short it can be for each terminal. The use of Interrupt Coalesce on the network card can distort receiver timing of packets, which has a negative impact on active and passive network measurements [8][16][17][18]. On the other hand, it is difficult to increase the packet size as well because of the MTU limitation. For example, the MTU of current networks is often 1,500 Bytes. Therefore, conventional packet train methods have the problem that the estimation upper bound cannot be high enough. Note that although lines are currently getting faster and faster, the above MTU (1,500 Byte) is not likely to change in the future, and the transmission interval constraint is not likely to be relaxed much.

Therefore, there are two ways to send large packets, both of which may not be suitable for estimation: One method is to send packets larger than the MTU and fragment them when the MTU limit is hit. Since fragmentation is often processed by software by hosts and routers, there is a possibility that the exact reception time cannot be obtained. The other is to send packets smaller than the MTU and fragment them where they get caught in the MTU limit. Some devices also implement jumbo frames [19]. Jumbo frames allow communication of packets as large as MTU = 9,000 Bytes. However, it is not standardized in IEEE802 and may not be used in practice. Thus, it is difficult to use packets larger than 1,500 Bytes.

2) *Issues of highly accurate estimation*

Estimation error due to the resolution of the estimation must also be considered; in the case of PathQuick3, the resolution depends on the incremental packet size. In the example in Fig. 2, the packet size that saturates the network could be "the size of the third packet" + 1 Byte, or it could be "the size of the third packet" + g Bytes. In other words, the difference in transmission rate corresponding to g Bytes is the resolution. Specifically, in Fig. 3, the packet size increment g is 100 Bytes, so the resolution is 8 Mbps (see Reference [2] for calculation method). If g were 1,000 Bytes, the resolution would be 80 Mbps, which is 10 times lower resolution. From this, assuming that the same number of packets are used on a high-speed line as on a low-speed line, a higher transmission rate should be sent out. If the number of packets sent is constant, the packet size increment g should be increased to increase the estimation upper bound. On the other hand, the larger the increment g Byte, the lower the resolution. Considering the load on the network, it is difficult to increase the number of packets, so there is a trade-off between resolution and estimation upper bound. In addition, as discussed in Section 2.3, the maximum size of the resolution may enter into the estimation error.

III. Proposal of PathRefiner

In this paper, we propose PathRefiner, a bandwidth estimation method that simultaneously solves (1) the problem of high-speed bandwidth and (2) the problem of accurate estimation, which conventional methods have. (1) the packet concatenation technique, which solves the problem of supporting high-speed lines, and (2) the iterative resolution refinement technique, which solves the problem of accurate estimation. Each is described below.

A. *High-speed line support: (1) Packet concatenation Technique*

The packet concatenation technique creates a virtual large packet (virtual packet) by concatenating multiple packets to enable high-speed line estimation with packets larger than the MTU, Packets within a packet group are denoted as "packet group number" - "packet number within packet group". The transmission interval is the difference between the transmission time of the first packet in each packet group. The receive interval is the difference between the receive time of the last packet in each packet group. Specifically, as shown in Fig. 4, the transmission interval between virtual packets, or packet groups, is kept constant, and the packet size of each packet group is linearly increased while packets with the same concatenation number C are continuously transmitted within the packet group. The packet size of all multiple packets within a packet group shall be the same. As shown in Fig. 5, when the transmission rate per packet exceeds the available bandwidth for the first time in packet group 4-3, the receive interval is equal to the transmission interval because no queuing delay occurs up to packet group 3. On the other hand, since the transmission rate of packet group 4 exceeds the available bandwidth, queuing delays occur at the routers and switches at the bottleneck points on the network path. As a result, the receive interval becomes longer than the send interval. At the receiving terminal, the packet group for which the receive interval begins to increase is packet group 4, so the transmission rate of packet group 3 before it is used as the estimated value of the available bandwidth. By increasing the number of concatenations C, theoretically, any high-speed line can be supported. However, in reality, due to the accuracy problem described below, the number of concatenations C should be set at the upper limit of the line capacity, which is the bottleneck.

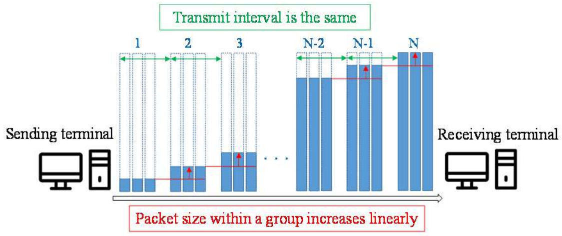

Fig. 4. Packet train structure for packet concatenation ($C = 3$)

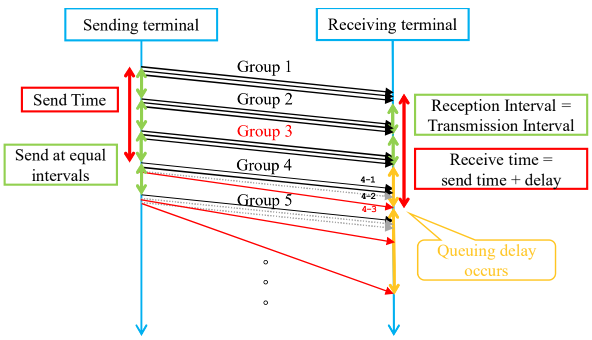

Fig. 5. Sequence diagram of packet transmission and reception (when $C = 3$)

B. *Accurate Estimation: (2) Iterative Resolution Refinement Technique*

In the iterative resolution refinement technique, the optimal number of linkages C is narrowed down through trial and error to reduce estimation errors caused by resolution while maintaining the function of high-speed line. In the following, we first discuss the relationship between the number of linkages C and resolution, and then explain the iterative resolution refinement technique to find the optimal C.

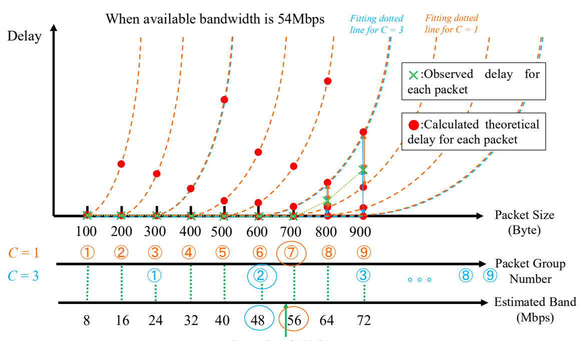

Fig. 6. Differences in estimation errors due to differences in the number of connections C

First, Fig. 6 is used to show the need to optimize the number of linkages C. This is because different number of linkages C have different resolutions and estimation errors. An example is given below. Assume that the line capacity is 300 Mbps and the available bandwidth is 54 Mbps. In this case, the error due to resolution is compared between the case where $C = 1$ and the case where $C = 3$. The transmission rate at which queuing delay begins to occur is 54 Mbps, but the delay is observed in packet group 2 for $C = 3$ and in packet group 7 for $C = 1$. Consequently, when fitting is performed, solutions with estimated results of 48 Mbps and 56 Mbps, respectively, and errors with the true value of 6 Mbps and 2 Mbps. From this, the concatenation number $C = 1$ can obtain better fitting results.

Thus, the smaller C is, the higher the resolution can be. However, for high-speed line support, C must be increased. Therefore, to find the optimal C, it is necessary to narrow down the selection range of C using the iterative resolution refinement technique to find the optimal value.

The iterative resolution refinement technique finds the optimal C by decreasing the value of C according to the estimated value and transmitting the packet train several times to iterate the estimation. Each time the estimation is repeated, a new C value is set such that the estimated available bandwidth is just barely included. The termination condition is when the same C value is obtained consecutively, and the estimation is terminated by determining the C to be the optimal number of divisions, *Copt*. However, to account for statistical variations in the estimation delay and to make this technique robust, a margin is taken and the value of C is set to one larger value. The example in Fig. 7 shows the procedure for refining the number of linkages C. First, the first time, C is set to 9 and estimated. The second time, C is estimated by setting 7, one larger than the previous $C = 6$. The third time, C is estimated by setting $C = 4$, which is one higher than the previous $C = 3$. The fourth time, $C = 5$, which is one greater than the previous $C = 4$, is set and estimated. Since the obtained $C = 4$ is the same as the previous one, the optimal number of divisions, *Copt,* is determined to be C and the estimation is terminated.Evaluation experiment using actual equipment on high-speed line

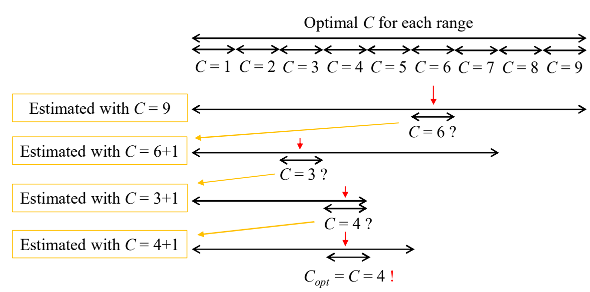

Fig. 7. The iterative resolution refinement technique

C. *Evaluation model and experimental configuration*

Experiments were conducted using actual equipment to demonstrate that the proposed available bandwidth estimation method is capable of supporting high-speed lines and is highly accurate. Fig. 8 shows the equipment and connection configuration for the evaluation experiment. We estimated the available bandwidth by sending packet trains from the PathRefiner's sending PC to the PathRefiner's receiving PC while other users were communicating. Cross-traffic is sent using iperf from the cross-traffic sending PC to the cross-traffic receiving PC, sending a UDP stream with a specified transmission rate. The test network was configured using Gig

Ethernet lines and Gig Ethernet switches to create various available bandwidths within a 1 Gbps line capacity. Due to the availability of experimental equipment, we set the range of available bandwidth to 200-1000 Mbps, which is a high communication speed, for our estimation. For this purpose, the communication speed of the cross-traffic m was varied from 0 Mbps to 800 Mbps in 1 Mbps increments. With these 801 patterns, PathRefiner packet trains were sent and estimates were recorded. By using ordinary PC which are not exclusive equipment, we try to estimate as accurately as possible. The true value of the available bandwidth is expressed by (1).

$$\textit{True Value} = \textit{Line Capacity} - \textit{Cross-traffic } m \tag{1}$$

The estimation error is defined by Equation (2), the relative error by Equation (3), and the standard deviation by Equation (4). n in equation (4) is the number of estimations.

$$Estimation\ Error = |Estimate\ Value - True\ Value| \tag{2}$$

$$Relative\ Error = \frac{Estimation\ Error}{True\ Value} \tag{3}$$

$$SD = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(RE_i - \overline{RE})} \tag{4}$$

In addition, the parameters used for this estimation are as follows. The packet transmission interval is 0.1 ms, the size of a single packet in the first packet group is 32 Bytes, the incremental packet size of adjacent packet groups is 12 Bytes, the number of packet group is 120, and the initial value of the concatenation number C is 9. The reasons for using these parameters are as follows. The time that can be accurately transmitted and estimated by the PC used is in units of 0.1 ms, which is the limit for using a normal PC; to transmit at 0.1 ms intervals, a virtual packet size of at least 12500 Bytes would be required to estimate 1 Gbps. To keep a single packet below an MTU, we set the concatenation number C to 9. Since we used 120 packet groups, we set the packet size for group 1 at the start to 32 Bytes, with an increment of 12 Bytes, so that the packet size within the final group was about an MTU. The estimated lower and upper limits calculated from these values are 2.6 Mbps and 1044 Mbps, respectively (see reference [2] for calculation method).

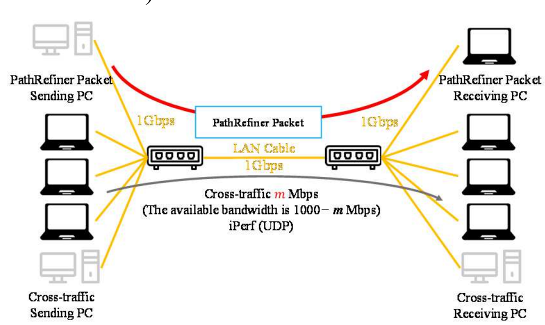

Fig. 8. Evaluation model and parameters for available bandwidth estimation

D. Evaluation results

1) High-speed line support

The experimental results show that the available bandwidth can be estimated even for high-speed lines with a line capacity of 1 Gbps, as described below. It is also confirmed that the conventional method PathQuick3 can only estimate the available bandwidth up to 120 Mbps for the same transmission interval. With the implemented PathRefiner, theoretically, the available bandwidth of any high-speed line can be estimated regardless of the MTU size.

2) Estimation accuracy

Fig. 9 shows the PathRefiner estimation results. Each point in the figure is the average estimated value for each pattern with various cross-traffic. The length of the line above and below each point represents the variation of the estimated value in each pattern with ±1 standard deviation. When the estimated value perfectly matches the ground truth, the points are plotted on a straight line at an angle of 45°. 100 estimates were made for each of the 801 patterns of cross-traffic communication speeds, i.e., 80,100 estimates in total. Overall for all 801 patterns, the average estimation error is 68.2 Mbps, the average relative error is 14.4%, and the average standard deviation is 89.3 Mbps.

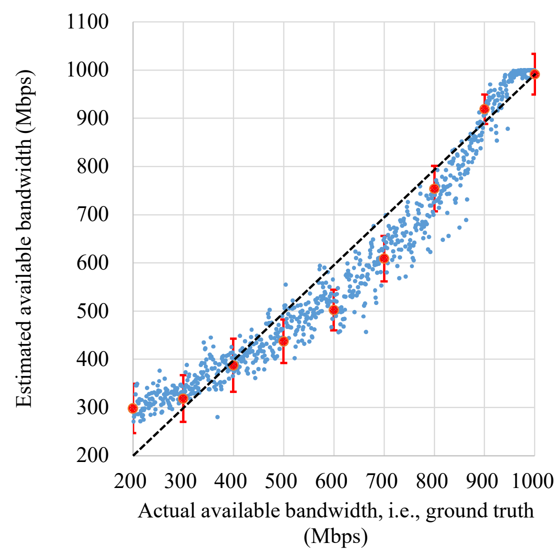

Fig. 9. Estimation Results with PathRefiner

IV. Conclusion

In this paper, we propose PathRefiner, a method for estimating available bandwidth for high-speed lines with high accuracy. It can also improve the resolution by using a technique to find the optimal number of concatenated large-sized virtual packets to be transmitted, which enables highly accurate estimation.

To demonstrate that PathRefiner can estimate with high accuracy on high-speed lines, we evaluate the estimation error on a 1 Gbps line using actual testbed. The results show that the average value of the relative error is relatively small (14.4%), confirming that PathRefiner can estimate with high accuracy on a high-speed line.

ACKNOWLEDGMENT

The authors would like to give heartfelt thanks to all the people who have ever helped in this paper, especially grateful to professors and students of Murase laboratry, Naogya Univeristy for their technical advices on the experiments.

REFERENCES

[1] R. Prasad, C. Dovrolis, M. Murray, and K.Claffy, “Bandwidth estimation: metrics, measurement techniques and tools,” IEEE network, 17(6): pp. 27-35, 2003.

[2] T. Oshiba, K. Nogami, K. Nihei, and K. Satoda, “Robust available bandwidth estimation against dynamic behavior of packet scheduler in operational LTE networks,” IEEE Symposium on Computers and Communication (ISCC), pp. 1276-1283, Italy, August 2016.

[3] N. Ito, T. Oshiba, K. Satoda, and T. murase, “PathRakeTCP: Available Bandwidth Estimation Using Multiple TCP Connections for Passing through Firewalls,” IEEE International Conference on Consumer Electronics (ICCE). IEEE: 1-6, 2023.

[4] V. J. Ribeiro, R. H. Riedi, R.G. Baraniuk, J. Navratil, and L. Cottrell, “pathChirp: Efficient Available Bandwidth Estimation for Network Paths,” Passive and active measurement workshop, USA, April 2003.

[5] C.L.T. Man, G. Hasegawa, and M. Murata, “An inline network measurement mechanism for highspeed networks,” IEEE/IFIP Workshop on End-to-End Monitoring Techniques and Services. IEEE, pp. 66-73, 2006.

[6] M. Jain, and C. Dovrolis, “End-to-end available bandwidth: measurement methodology, dynamics, and relation with TCP throughput,” ACM SIGCOMM, pp. 295-308, USA, October 2002.

[7] T. Tsugawa, C.L.T. Man, G. Hasegawa, and M. Murata, “Implementation issues on inline network measurement algorithms in gigabit networks,” IEICE Technical Report; IEICE Tech. Rep., 106(151): 73-78, 2006.

[8] C.L.T. Man, G. Hasegawa, and M. Murata, “Inline bandwidth measurement techniques for gigabit networks,” International Journal of Internet Protocol Technology, 3(2): 81-94, 2008.

[9] R. Jain, and S. Routhier, “Packet trains--measurements and a new model for computer network traffic. IEEE journal on selected areas in Communications, ” 4(6): 986-995, 1986.

[10] Q. Wang, L. Cheng, “FEAT: improving accuracy in end-to-end available bandwidth estimation,” Passive and Active Measurement, 2005.

[11] J. He, “Available bandwidth measurement, implementation and evaluation,” Proceedings. 2004 12th IEEE International Conference on Networks (ICON 2004)(IEEE Cat. No. 04EX955). IEEE, 1: 226-230, 2004.

[12] J. He, C. E. Chow, J. Yang, T. Chujo, “An algorithm for available bandwidth measurement,” International Conference on Networking. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 753-761, 2001.

[13] T. Oetiker, and D. Rand, “MRTG: The Multi Router Traffic Grapher,” LISA., 98: 141-148, 1998.

[14] A. Tirumala, “Iperf: The TCP/UDP bandwidth measurement tool,” http://dast.nlanr.net/Projects/Iperf/, 1999.

[15] D. Kachan, E. Siemens, V. Shuvalov, “Available bandwidth measurement for 10 Gbps networks,” 2015 International Siberian Conference on Control and Communications (SIBCON). IEEE, pp. 1-10, 2015.

[16] R. Prasad, M. Jain, C. Dovrolis, “Effects of interrupt coalescence on network measurements,” International Workshop on Passive and Active Network Measurement. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 247-256, 2004.

[17] Q. Yin, J. Kaur, F.D. Smith, “Scaling bandwidth estimation to high speed networks,” Passive and Active Measurement: 15th International Conference, PAM 2014, Los Angeles, CA, USA, March 10-11, 2014, Proceedings 15. Springer International Publishing, pp. 258-261, 2014.

[18] V. Tran, J. Tourrilhes, K. K. Ramakrishnan, P. Sharma, “Accurate available bandwidth measurement with packet batching mitigation for high speed networks,” 2021 IEEE International Symposium on Local and Metropolitan Area Networks (LANMAN). IEEE, pp. 1-6, 2021.

[19] R. Winter, et al., “Ethernet jumbo frames,” Ethernet Alliance, 2009.