Locating Causal Segments Along A Performance-Degraded End-to-End Path

Yuki Sakai †
Kyushu Institute of Technology
Fukuoka 820–8502, Japan

Masato Uchida
Kyushu Institute of Technology
Hibiya Kokusai Bldg. 1F–107
Tokyo, 100–0011, Japan

Masato Tsuru
Kyushu Institute of Technology
Fukuoka 820–8502, Japan

Yuji Oie
Kyushu Institute of Technology
Fukuoka 820–8502, Japan

1. INTRODUCTION

Transient network performance (quality) degradation resulting from packet delay variations and/or packet losses, often called as "congestion", is still likely to occur on the Internet due to the inherent feature of statistical and loosely controlled shares of network resources. Typically, the performance of an end-to-end path is not responsible to a single Internet Service Provider (ISP) because the path traverses multiple ISPs and user-managed networks at both sides. Thus, it is of practical importance for end-to-end service application users and/or providers to find the causes of transient network performance degradation by themselves.

In response to this problem, in this paper, we propose and implement a practical tool based on network tomography. This tool runs on user hosts at both sides of a performance-degraded end-to-end path and infers the packet loss rate and delay variance on each individual directional segment to finally locate the causal segments along the path. A segment here is defined as a network portion segmented by two routers and consisting of a sequence of any number of routers, switches, and/or physical links. Our tool might look like *pathping*, an Windows-based network command that combines *traceroute* and *ping* [1]. However, our tool can more quantitatively and precisely infer the performance status of each segment using a theoretical basis with combination of one-way and round-trip measurements.

2. A TOOL LOCATING CAUSAL SEGMENTS ALONG A PATH USING PACKET PAIRS

There is an well-known approach to inferring the characteristics of network internal segments based on the network tomography, using multicast or unicast packet pair probing (e.g. [2, 3]) over a tree topology. We apply this approach using packet pairs along a single end-to-end path.

Let us explain an example in Fig. 1. In the packet pair measurements illustrated in the upper half of the figure, the tool at end-node S (server) sends a number of probing packet pairs bound for the other end-node C (client). Since TTL value of the second packet in each pair is set to 1, this packet vanishes at router R_0 and the ICMP time-exceed packet returns to the origin S, while the first packet can reach the destination C unless the packet is dropped.

Based on the measurement of N trials of packet pair probing, packet loss rates r_{S0} and r_{0C} on segments $[S, R_0]$ and $[R_0, C]$, respectively, are inferred. Suppose that (i) packet losses on segments $[S, R_0]$, $[R_0, C]$, and $[R_0, S]$ occur independently; and (ii) on common segment $[S, R_0]$, if the second packet in a pair is not dropped, the first one is unlikely to be dropped. Then, we have the following simple estimators:

$$\hat{r}_{S0} = 1 - \frac{N_C N_S}{N_{C+S} N}, \; \hat{r}_{0C} = 1 - \frac{N_{C+S}}{N_S} \tag{1}$$

where N_C, N_S, and N_{C+S} are the number of packet pair trials in which the first packet reaches node C, the router reply packet for the second one reaches back node S, and the both packets are not dropped until reaching their final destinations (C and S), respectively.

At the same time, packet delay variances v_{S0} and v_{0C} are inferred from the measurement of the above N_{C+S} packet pairs. Suppose that (i) packet delays on $[S, R_0]$, $[R_0, C]$, and $[R_0, S]$ vary independently; (ii) on common segment $[S, R_0]$, the delay experienced by the first packet in a pair is similar to that by the second one. Then, we have the following:

$$\begin{aligned} \hat{v}_{S0} &= \frac{1}{N_{C+S} - 1} \sum_{k=1}^{N_{C+S}} (d_C(k) - \overline{d_C})(d^*_S(k) - \overline{d^*_S}), \\ \hat{v}_{0C} &= \frac{1}{N_{C+S} - 1} \sum_{k=1}^{N_{C+S}} (d_C(k) - \overline{d_C})^2 - \hat{v}_{S0} \end{aligned} \tag{2}$$

where $d_C(k)$ (resp. $d^*_S(k)$) is the one-way delay time of the first packet from nodes S to C (resp. the round-trip delay time of the second packet and the router reply to it) in the k-th packet pair, and $\overline{d_C}$ (resp. $\overline{d[*]_S}$) is the sample mean of them for $k = 1, 2, \ldots, N_{C+S}$.

Similarly, from different packet pair measurements illustrated in the lower half of Fig. 1, the tool infers packet loss rates r_{S1} and r_{1C} and packet delay variances v_{S1} and v_{1C} on segments $[S, R_1]$ and $[R_1, C]$, respectively.

Finally, thanks to an additive property, the tool can infer packet loss rate and packet delay variance on intermediate segment $[R_0, R_1]$ from the previously obtained results.

---

† yuki@infonet.cse.kyutech.ac.jp

PAM2009, April 1-3, 2009, Seoul, Korea.
.

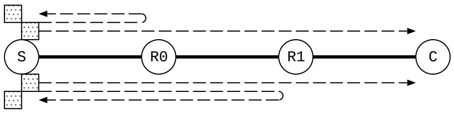

Figure 1: Packet pairs inferring the loss rate and delay variance on each segment along a path

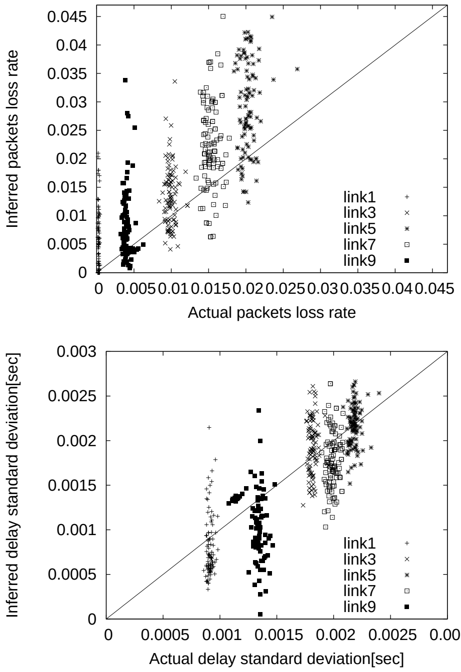

Figure 2: Actual and inferred loss rates (top) and delay standard deviations (bottom)

3. PRELIMINARY EXPERIMENTS

A NS-2 simulation was conducted to evaluate the inference accuracy of the proposed method [4]. A simple sequential topology with eleven segments was used. The background traffic was generated by a series of file transfers according to exponentially distributed time-intervals and a Pareto-distributed file sizes.

Figure 2 compares the actual and the inferred values on five segments. The inference error varies depending on the case but may be acceptable for the purpose of locating the causal segments except for an irregular under-estimation in delay standard deviation on link 9.

We implemented a proto-type tool running on Linux, and a preliminary laboratory experiment was performed to evaluate and improve the implemented tool in real network environments. Two end hosts with two intermediate routers were connected by 100 [Mbps] links according to Fig. 1, some Layer-2 switches were put into $[R_0, R_1]$, and various degree and type of background traffic was injected through the switches to make congestion. We run the tool for ten minutes to infer the packet loss rate and packet delay variance on the intermediate segment $[R_0, R_1]$ during the measurement period. The size, sending rate, and the number of the measurement packets were 64 [Bytes], 10 [Kbps], and 12000 (6000 pairs), respectively.

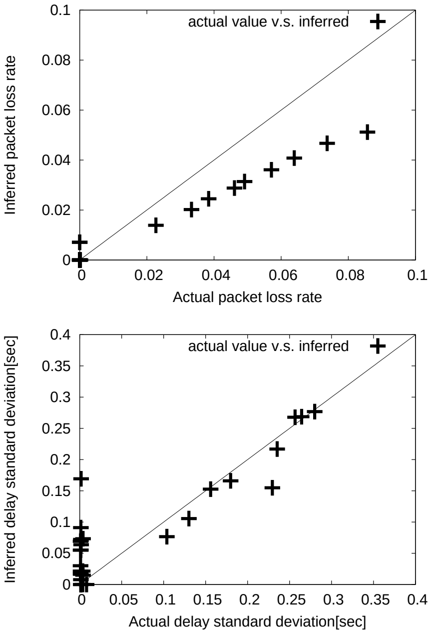

Figure 3: Actual and inferred loss rates (top) and delay standard deviations (bottom)

In Fig. 3, the potential of our tool is indicated but considerable biases are also seen: under-estimation in cases for large loss rates and over-estimation in cases for almost nearly zero delay variances, which should be investigated and improved.

4. DISCUSSION AND FUTURE WORK

The current proto-type tool can just quantitatively infer packet loss rate and delay variance on each segment. In this line, we are trying to improve the inference performance by introducing more accurate and robust estimators.

However, since the inference principle relies on the strong assumptions, an inference error likely remains to some extent. So, as the next step, we will focus on making the tool capable to automatically, promptly, and reliably locate the causal segments. For this, it is desired to handle losses and delay variations together rather than separately.

This work was partly supported by the National Institute of Information and Communications Technology, and the JSPS, Grant-in-Aid for Scientific Research (C) (No. 19500057).

5. REFERENCES

[1] pathping, `http://technet.microsoft.com/en-us/library/bb490964.aspx`

[2] M. Tsuru, T. Takine and Y. Oie, “Inferring link loss rates from unicast-based end-to-end measurement”, *IEICE Trans. Commun.*, E85-B(1):70–78, Jan. 2002.

[3] Y. Tsang, M. Yildiz, R. Nowak, and Paul Barford, “Network radar: Tomography from round trip time measurement”, in *Proc. ACM SIGCOMM Internet Measurement Conference*, Taormina, Sicily, Italy, pp. 175–180, Oct. 2004.

[4] NS-2, `http://www.isi.edu/nsnam/ns/`

[5] perfSONAR, `http://www.perfsonar.net/`