Measurement of Loss Pairs in Network Paths

Edmond W. W. Chan[‡], Xiapu Luo[§], Weichao Li[‡], Waiting W. T. Fok[‡], and Rocky K. C. Chang[‡]
Department of Computing[‡]
The Hong Kong Polytechnic University
{cswwchan|csweicli|cswtfok|csrchang}@comp.polyu.edu.hk
College of Computing[§]
Georgia Institute of Technology
csxpluo@cc.gatech.edu

ABSTRACT

Loss-pair measurement was proposed a decade ago for discovering network path properties, such as a router's buffer size. A packet pair is regarded as a loss pair if exactly one packet is lost. Therefore, the residual packet's delay can be used to infer the lost packet's delay. Despite this unique advantage shared by no other methods, no loss-pair measurement in actual networks has ever been reported. In this paper, we further develop the loss-pair measurement and make the following contributions. First, we characterize the residual packet's delay by including other important factors (such as the impact of the first packet in the pair) which were ignored before. Second, we employ a novel TCP-based probing method to measure from a single endpoint all four possible loss pairs for a round-trip network path. Third, we conducted loss-pair measurement for 88 round-trip paths continuously for almost three weeks. Being the first set of loss-pair measurement, we obtained a number of original results, such as prevalence of loss pairs, distribution of different types of loss pairs, and effect of route change on the paths' congestion state.

Categories and Subject Descriptors

C.2.3 [**Computer-Communication Networks**]: Network Operations; C.4 [**Performance of Systems**]: Measurement Techniques

General Terms

Experimentation, Measurement, Performance

Keywords

Loss pair, Packet pair, Non-cooperative, Delay

1. INTRODUCTION

Packet loss behavior in network paths has been extensively studied for the last twenty years. Most of the efforts focus on packet losses as a result of router congestion. The packet loss behavior has been characterized by loss rates, loss stationarity [53], loss episodes [53, 47], and loss correlation [52, 37]. Both active (e.g., ZING [4], Sting [44], Badabing [46, 47], OneProbe [33], and Queen [50]) and passive (e.g., [5, 39]) measurement methods have been proposed for measuring losses on end-to-end paths. For active methods, the probing process is an important consideration for minimizing measurement errors [6, 48, 7]. Moreover, various tomography techniques have been proposed for measuring packet losses on the link level [13, 17, 15].

Besides the packet loss measurement, it is also useful to study the correlation between loss and other important metrics. However, the correlation problem has so far received much less attention. A notable exception is using a packet pair to correlate a packet loss event and the delay that would have been experienced by the lost packet. A *packet pair* is referred to as a *loss pair* [31, 30] if exactly one packet (the first or second) in the pair is lost. If the two packets traverse the path close to each other, then the residual packet's delay can be used to infer the lost packet's delay. The loss-pair analysis was originally motivated by the problem of estimating buffer size of the congested node responsible for dropping the packet. Other possible applications of the lost-pair measurement include characterizing packet dropping behavior [31], classifying the type of packet loss [32], detecting dominant congestion links [51], and detecting common congestion points [21, 41].

Although loss pairs could be considered rare events in typical network paths, they can be detected by many existing measurement methods without extra cost. For example, the path capacity measurement methods, such as [42, 16, 26, 11], send a sequence of packet pairs to capture the packet dispersion from the bottleneck link. But they usually discard the loss pairs, which fail to provide the dispersion information. Other measurement methods for packet loss (e.g., [44, 35]), packet reordering (e.g., [34, 8]), Internet traffic characterization (e.g., [12]), and path fingerprinting (e.g., [45]) also send packet pairs for their measurement. Therefore, loss-pair measurement is considered a bonus feature for these tools, and the previously regarded useless probes can now be exploited for discovering additional path properties.

However, loss pairs have not been reported in actual network path measurement. In [31, 32, 30], only ns-2 simulation and emulated testbed experiments were performed to evaluate the effectiveness of using loss pairs to measure path properties. As a result, the behavior of loss pairs in Internet paths is largely unknown. Moreover, some important delay

IMC'10, November 1–3, 2010, Melbourne, Australia.

components, such as the impact of the first packet on the second, have not been taken into consideration. In this paper, we revisit the loss-pair measurement method and make three main contributions:

1. Delay characterization We conducted a more detailed analysis for the residual packets' delays by including the impacts of cross traffic and the first packet. The new analysis invalidates the previous claim that the first and second residual packets give the same result [31, 30]. We instead show that using the first packet's delay is generally more accurate than the second packet's delay on inferring the congested router's queueing delay upon packet loss. Moreover, we show that the delay variation of the first and second residual packets can be used to estimate the link capacity of a hop preceding the congested router.

2. Method for measuring loss pairs We exploited OneProbe's capability [33] of detecting path events from a single endpoint to measure all four possible loss pairs on a round-trip path: two for the forward path and the other two for the reverse path. To the best of our knowledge, OneProbe is the first non-cooperative method capable of performing comprehensive loss-pair measurement. Previous loss-pair measurement considered only two possible loss pairs on a round-trip path [31, 30]. We also utilized OneProbe's facility of packet size configuration to validate that a smaller packet size generally increases the accuracy of delay inference.

3. Loss-pair measurement in the Internet We conducted loss-pair measurement using HTTP/OneProbe (an OneProbe implementation based on HTTP/1.1 [19]) for 88 round-trip paths between eight universities in Hong Kong and 11 PlanetLab nodes located at eight countries. Our measurement shows that loss pairs were prevalent in the packet pairs that suffered packet loss, and a loss-pair analysis can help infer additional properties about the lossy paths. Besides, we show that loss pairs' delays provide path signatures for correlating multiple path measurements.

In §2, we first discuss previous works related to this paper. In §3, we review the loss-pair measurement method and describe how OneProbe detects the loss-pair events. In §4, we analyze the residual packets' delays and relate the results to the problem of estimating the queueing delay at the congested router upon packet drop. In §5, we report our findings of measuring 88 paths continuously for almost three weeks. In §6, we conclude this paper with a few potential directions to extend this work.

2. RELATED WORK

The notion of packet pair was first defined in [24] as a pair of back-to-back packets of the same size dispatched by a source to a destination. Each packet arrived at the destination is acknowledged by an acknowledgement packet, which travels along a returning path back to the source. Assuming that the pair traverses the path close to each other, they may observe similar states of the congested hop before a packet is discarded. Therefore, the residual packet in the loss pair has been used to estimate the packet dropping mechanism of the congested hop governed by various active queue management schemes (e.g., droptail, RED [20], and BLUE [18]) and a droptail queue's buffer size [31], and to classify the causes for packet loss [32]. In this paper, we analyze the accuracy of two possible residual packets' delays for inferring the network path properties.

Previous studies used packet pairs to measure various path performance metrics, including packet reordering [34, 8], packet loss [44, 35], available bandwidth [36, 23], and capacity [10, 43, 16, 26]. The pioneer work by Keshav [28] exploited the dispersion of a packet pair to measure the capacity of rate allocating servers. The packet-pair dispersion observed from the destination contains the latency of the packet pair after leaving the bottleneck link. Pásztor and Veitch [38] analyzed several types of components embedded in the packet-pair dispersion. On the other hand, Bolot [9] relied on the queueing of the first packet in a packet pair at the bottleneck hop to measure the bottleneck link's capacity. MultiQ [27] exploited the congestion experienced by packet pairs to infer the capacity of congested links. In this paper, we propose a method of estimating the capacity of a network link preceding the congested hop based on the two residual packets' delays.

A number of methods were proposed for monitoring congested network links, including Pathload [25] and Pong [14] that detect network congestions by observing increasing queueing delays of its probe packets. Besides, various methods [41, 21, 51] were proposed to detect the shared network congestion point in the paths. However, these methods were evaluated based on either simulation or cooperative measurement. In this paper, by using OneProbe's probing technique [33], our loss-pair measurement and analysis can infer the packet loss behavior for both forward and reverse paths in a non-cooperative manner and identify artifacts, such as packet reordering, that may affect the measurement results.

3. ACTIVE LOSS-PAIR MEASUREMENT

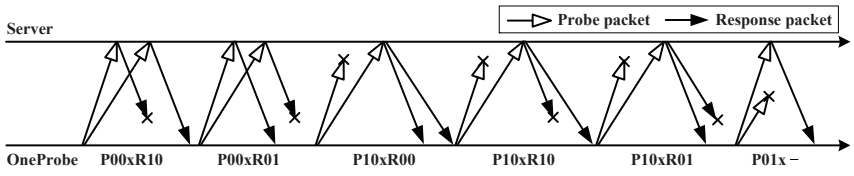
In loss-pair measurement, a source node sends a sequence of *probe pairs*, each pair consisting of two back-to-back probe packets, to a destination node. The possible delivery statuses of a probe pair are `00` (both received), `01` (only the first is received), `10` (only the second is received), or `11` (none is received). The cases of `01` and `10` are referred to as *loss pairs* in [31]. Moreover, the destination node may be induced to send a sequence of *response pairs*, each pair consisting of two back-to-back response packets, to the source node. There are four similar delivery statuses for each response pair. As a result, there are generally four possible loss pairs for a round-trip path: P`10` and P`01` for a probe pair, and R`10` and R`01` for a response pair.

Both passive and active methods could be used for measuring loss pairs. An active loss-pair measurement of a path can be performed on both endpoints of the path or from only a single endpoint. In this section, we use OneProbe [33] to illustrate how the four types of loss pairs can be measured from only one endpoint. We also deployed HTTP/OneProbe [33] to measure loss pairs on Internet paths, and the results will be presented in §5.

OneProbe sends a sequence of probe pairs, each consisting of two TCP data packets, to a remote server. If both packets are received in the same order, each packet elicits a response TCP packet, thus returning a response pair. Even if one or more probe packets is lost, at least one response TCP packet will be elicited immediately. Moreover, by predetermining the number, types, and order of the response packets elicited under each delivery status (`00`, `01`, `10`, or `11`) of the probe pair, OneProbe can distinguish the delivery statuses for both probe and response pairs just based on the elicited response packets. Figure 1(a) shows two cases. For those marked by '✓', OneProbe can simultaneously detect the probe pair's

	R00	R10	R01	R11
P00	✓	✓	✓	✓
P10	✓	✓	✓	✓
P01	–	–	–	–
P11	–	–	–	–

(a) The delivery statuses.



(b) The six loss-pair events.

Figure 1: The delivery statuses of probe and response pairs and six loss-pair events measured by OneProbe.

and response pair's delivery statuses. For those marked by '– ', OneProbe can only detect the probe pair's status, because at most one response packet can be elicited for those cases.

Six cases in Figure 1(a) involve at least one loss pair, and they are illustrated in Figure 1(b). For P00 and P10, two response packets can be elicited from the server. As a result, OneProbe can detect the forward-path and reverse-path loss pairs at the same time. However, in the absence of a response pair, OneProbe can detect only the forward-path loss pair for P01. Furthermore, packet reordering does not affect the loss-pair measurement, because OneProbe can also identify from the response packets end-to-end packet ordering events for the probe and response pairs.

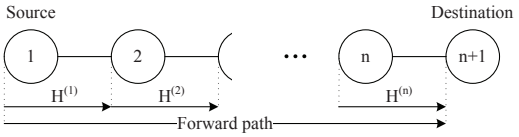
4. ANALYSIS OF LOSS PAIRS' DELAYS

Since a packet pair's delay is used for inferring path properties, in this section we analyze the first and second packets' delay, and their difference. In the following analysis, we consider the four loss-pair events (P10xR00, P01x–, P00xR10, and P00xR01) for which a loss pair exists in only one unidirectional path, and a similar analysis can be performed for the other events. To simplify the notations, we also use LP_{10} to denote a loss pair with the delivery status 10 (i.e., P10xR00 and P00xR10), and LP_{01} to denote that with the status 01 (i.e., P01x– and P00xR01).

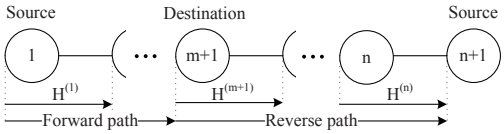
After presenting the network models in §4.1, we first derive in §4.2 the residual packets' delays in the LP_{10} and LP_{01}, taking into consideration the queueing delay at all hops. In §4.3, we then extend the analysis to the problem of using the delay to characterize the congested node's queueing delay upon packet drops. Finally in §4.4, we show that the LP_{10}'s and LP_{01}'s delays can be utilized to estimate the capacity of a link preceding the congested node.

4.1 Network models

Consider a sequence of probe pairs dispatched on a network path of n hops (where $n \geq 1$) which also admits other cross traffic. The network path is assumed unchanged throughout the measurement. Each hop in the path consists of a store-and-forward node and its outgoing link connecting to the next hop. We use $H^{(h)}$ to denote the h^{th} hop that transmits (i.e., serializes) packets to the outgoing link with capacity of $C^{(h)}$ bits/s. Each node is configured with a drop-tail queue which is modeled as a single-server queue with a buffer size of $B^{(h)}$ bits for $H^{(h)}$ and a First-Come-First-Serve (FCFS) queueing discipline. For convenience, we label the hops on the path sequentially, starting from 1 at the source node. The n-hop network path can be either a one-way path (forward path) depicted in Figure 2(a) or a round-trip path (forward path and reverse path) in Figure 2(b) in which the destination node is located at $H^{(m+1)}$, $1 \leq m < n$.



(a) One-way path.



(b) Round-trip path.

Figure 2: Two models for the loss-pair analysis.

We use $\{p_{j-1}, p_j\},\ j = 2i, i = 1, 2, \ldots$, to denote the i^{th} probe pair with p_{j-1} being the first packet in the pair. Each probe packet is of S bits long, including the IP header. Therefore, sending a probe packet on $H^{(h)}$ incurs at least a packet transmission delay of $X^{(h)}$ $(= S/C^{(h)})$ and a constant propagation delay denoted by $T^{(h)}$. Besides, adjacent probe pairs are assumed to be sufficiently spaced out, so that a packet is never queued behind the preceding packet pair, and the probe packets are not out-of-ordered due to the FCFS queueing discipline. In the case of round-trip path, we also make similar assumptions for the response packet pairs. Moreover, we use the same notations and packet size for the response pairs to simplify our ensuing discussion. However, the analysis can be easily adapted to different probe and response packet sizes.

We start the analysis by considering the total delay for p_j to traverse the first h hops of the path, denoted by $d_j^{(h)},\ h = 0, 1, \ldots, n$, and $d_j^{(0)} = 0$. We also let $t_j^{(h)},\ h = 1, \ldots, n+1$, be the time for p_j's to fully arrive (including the last bit of the packet) at $H^{(h)}$. Therefore,

$$\begin{aligned} d_j^{(h)} &= t_j^{(h+1)} - t_j^{(1)}, \\ &= d_j^{(h-1)} + \left(w_j^{(h)} + X^{(h)} + T^{(h)}\right), \end{aligned} \tag{1}$$

where $w_j^{(h)}$ is the queueing delay experienced at $H^{(h)}$. The recursive expression in Eqn. (1) also applies to p_{j-1} after updating the subscripts.

Moreover, we can relate $d_j^{(h)}$ and $d_{j-1}^{(h)}$ as

$$\begin{aligned} d_j^{(h)} &= \left(t_j^{(h+1)} - t_{j-1}^{(h+1)}\right) + \left(t_{j-1}^{(h+1)} - t_{j-1}^{(1)}\right), \\ &= \tau_{j-1,j}^{(h+1)} + d_{j-1}^{(h)}. \end{aligned} \tag{2}$$

where $\tau_{j-1,j}^{(h+1)}$ is the $\{p_{j-1}, p_j\}$'s inter-arrival time at $H^{(h+1)}$.

4.2 Analyzing the residual packets' delays

In the following, we consider a packet in $\{p_{j-1}, p_j\}$ being dropped at $H^{(h')}$ and the other packet delivered successfully. Thus, the loss pair is either an LP_{10} or LP_{01}. We also assume that the packet losses are due to node congestion. We obtain their residual packets' delays by including the queueing delay incurred from each hop. For the LP_{10}, it is also important to include p_{j-1}'s delay on the first $h'-1$ hops.

4.2.1 LP_{10}

To obtain p_j's delay for the LP_{10}, we first apply Eqn. (1) recursively until reaching the $(h'-1)^{th}$ node (since p_{j-1} is discarded at the h'^{th} node):

$$d_j^{(n)} = d_j^{(h'-1)} + \sum_{h=h'}^{n} \left(w_j^{(h)} + X^{(h)} + T^{(h)} \right). \quad (3)$$

By using Eqn. (2) for $d_j^{(h'-1)}$ and then applying Eqn. (1) recursively for $d_{j-1}^{(h'-1)}$, we obtain

$$\begin{aligned} d_j^{(n)} &= d_{j-1}^{(h'-1)} + \tau_{j-1,j}^{(h')} + \sum_{h=h'}^{n} \left(w_j^{(h)} + X^{(h)} + T^{(h)} \right), \\ &= \sum_{h=1}^{h'-1} w_{j-1}^{(h)} + \sum_{h=h'}^{n} w_j^{(h)} + \tau_{j-1,j}^{(h')} + \\ &\quad \sum_{h=1}^{n} \left(X^{(h)} + T^{(h)} \right). \quad (4) \end{aligned}$$

In addition to the queueing delay at all the nodes [31], Eqn. (4) also shows that the residual packet's delay contains $\tau_{j-1,j}^{(h')}$ which, as will be seen shortly, depends on a number of delay components in the preceding hops.

4.2.2 LP_{01}

To obtain p_{j-1}'s delay for the LP_{01}, we apply Eqn. (1) recursively for $d_{j-1}^{(h)}$ to obtain

$$d_{j-1}^{(n)} = \sum_{h=1}^{n} w_{j-1}^{(h)} + \sum_{h=1}^{n} \left(X^{(h)} + T^{(h)} \right). \quad (5)$$

Since the first packet is the residual packet, its delay is not affected by the second packet and does not contain $\tau_{j-1,j}^{(h')}$.

4.2.3 *Testbed experiments*

We conducted testbed experiments to evaluate the impact of $\tau_{j-1,j}^{(h')}$ on the residual packet's delay. The testbed, shown in Figure 3, was configured with a 12-hop round-trip path ($n = 12$), consisting of a probe sender, a web server running Apache v2.2.3 as the destination node, three cross-traffic clients $\mathbb{X}_1 - \mathbb{X}_3$, and five forwarding devices: two Linux 2.6.26 routers $\mathbb{R}_1 - \mathbb{R}_2$ and three 100 Mbits/s Ethernet switches $\mathbb{S}_1 - \mathbb{S}_3$. We designated $H^{(5)}$ ($\mathbb{R}_2$ and its link to $\mathbb{S}_3$) to be the only congested node on the path (i.e., $h' = 5$). We achieved this by running TC/Netem [22] in $\mathbb{R}_2$ to emulate $C^{(5)} = 50$ Mbits/s and a FCFS queue to accommodate approximately 100 ms of packets, and generating forward-path cross traffic (from $\mathbb{X}_2$ to $\mathbb{X}_3$) to congest $H^{(5)}$. Moreover, we designated $H^{(3)}$ ($\mathbb{R}_1$ and its link to $\mathbb{S}_2$) to be a bottleneck link by configuring Click v1.8 [29] in $\mathbb{R}_1$ to emulate $C^{(3)} = 1$ Mbit/s. The Click router was also configured to set the RTT between the probe sender and web server to 200 ms.

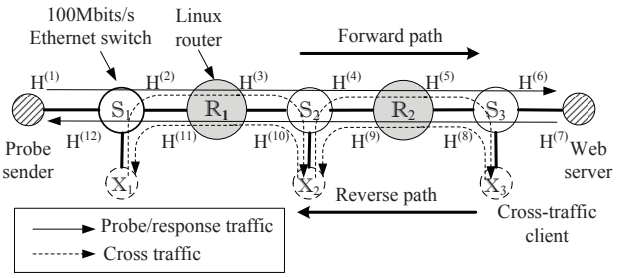


Figure 3: The testbed for the loss-pair experiments.

For this set of experiments, except for $H^{(5)}$, we did not generate cross traffic for other hops (i.e., $w_j^{(h)} = 0$, $\forall h \neq h'$, in Eqns. (4) and (5)). We ran HTTP/OneProbe from the probe sender to dispatch a sequence of 5000 Poisson-modulated probe pairs with a mean probing rate of 5 Hz. The probe sender was equipped with a DAG 4.5 passive network monitoring card [1] to obtain the RTT samples in microsecond resolution which is limited by the `pcap` header structure [2]. Similar to [26], the cross-traffic sources entered Pareto-distributed `ON` and `OFF` states with a shape $\alpha = 1.9$ and had a fixed packet size of 1500 bytes.

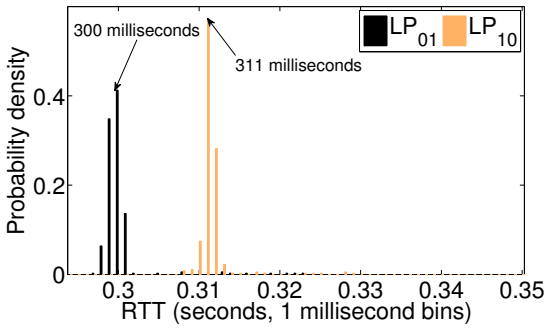


Figure 4: Residual packets' delays for LP_{10} and LP_{01} with $S = 1500$ bytes on the testbed for which $C^{(3)} = 1$ Mbit/s, $C^{(h')} = 50$ Mbits/s, and $h' = 5$.

Figure 4 shows the distributions of the residual packets' delays for the LP_{10} (i.e., `P10xR00`) and LP_{01} (i.e., `P01x–`) with $S = 1500$ bytes. Similar to [31], we applied a small bin size of 1 ms to mitigate the noise introduced by the non-congested hops. The figure shows that the residual packets' delays are dominated by the congested node's queueing delay of 100 ms, because most of them center around 300 ms and 311 ms for the LP_{01} and LP_{10}, respectively. We also note that many delay samples for the LP_{10} include an additional quantity of 11 ms which, according to Eqn. (4), came from $\tau_{j-1,j}^{(h')}$. Unlike other noises, this quantity cannot be filtered out by choosing a small bin size.

4.3 Characterizing the congested node's state

4.3.1 LP_{10}

A packet is dropped at $H^{(h')}$ when the node's buffer is full after the instantaneous input traffic rate exceeds $C^{(h')}$ for some time. We let $\{Q^{(h)}(t),\ t \geq 0\}$ be the continuous-

time process of its queue length in terms of bits, and $\mathbb{Q}_j^{(h)} = Q^{(h)}(t_j^{(h)-})$ (i.e., the queue length just prior to the arrival of p_j). When $\{p_{j-1}, p_j\}$ is an LP_{10}, we have

$$\mathbb{Q}_{j-1}^{(h')} + S > B^{(h')}, \text{ and} \quad (6)$$

$$\mathbb{Q}_j^{(h')} + S \leq B^{(h')}, \quad (7)$$

where

$$\mathbb{Q}_j^{(h')} = \left(\mathbb{Q}_{j-1}^{(h')} + A_{j-1,j}^{(h')} - D_{j-1,j}^{(h')}\right)^+, \quad (8)$$

and $(x)^+ = \max\{0, x\}$. $A_{j-1,j}^{(h')}$ is the amount of packets (in bits) arriving to and buffered at the queue during $(t_{j-1}^{(h')}, t_j^{(h')})$, and $D_{j-1,j}^{(h')}$ is the total amount of packets (in bits) departed from the node during $[t_{j-1}^{(h')}, t_j^{(h')})$. Therefore, the queueing delay of p_j at $H^{(h')}$ can be expressed as

$$w_j^{(h')} = \frac{\mathbb{Q}_j^{(h')}}{C^{(h')}} + R_j^{(h')}, \quad (9)$$

where $R_j^{(h')}$ is the residual service time upon p_j's arrival.

Moreover, $\tau_{j-1,j}^{(h')}$, $h' > 1$, can be expressed as [38]:

$$\tau_{j-1,j}^{(h')} = X^{(h^*)} + q_{j-1,j}^{(h^*)} + \sum_{h=h^*+1}^{h'-1} \left(w_j^{(h)} - w_{j-1}^{(h)}\right), \quad (10)$$

where $H^{(h^*)}$, $1 \leq h^* \leq h'-1$, is the last hop preceding $H^{(h')}$ for which p_j arrives before p_{j-1}'s full departure from the node. That is, both belong to the same busy period of the queue at $H^{(h^*)}$ [38]. Moreover, $q_{j-1,j}^{(h^*)}$ is p_j's queueing delay at $H^{(h^*)}$ due to intervening cross traffic arriving between p_{j-1} and p_j, and $X^{(h^*)}$ is the time for transmitting p_j at $H^{(h^*)}$.

For the purpose of estimating $\mathbb{Q}_j^{(h')}/C^{(h')}$, it is useful to consider p_j's *path queueing delay* defined by $\Theta_j = d_j^{(n)} - \min_{\forall i,\ j=2i}\{d_{j-1}^{(n)}\}$. Assuming that the minimum observable delay of p_{j-1}, $j = 2i$, $i = 1, 2, \ldots$, precludes the cross-traffic-induced queueing delay and using Eqns. (4), (9), and (10), we have

$$\begin{aligned}\Theta_j &= d_j^{(n)} - \sum_{h=1}^{n} \left(X^{(h)} + T^{(h)}\right), \\ &= \frac{\mathbb{Q}_j^{(h')}}{C^{(h')}} + R_j^{(h')} + X^{(h^*)} + \zeta_j, \end{aligned} \quad (11)$$

where $\zeta_j (= \sum_{h=1}^{h^*} w_{j-1}^{(h)} + q_{j-1,j}^{(h^*)} + \sum_{h=h^*+1}^{h'-1} w_j^{(h)} + \sum_{h=h'+1}^{n} w_j^{(h)})$ is the queueing delay contributed by the cross traffic present at $H^{(h')}$'s upstream and downstream hops.

From Eqn. (11), Θ_j can be used to estimate $\mathbb{Q}_j^{(h')}/C^{(h')}$, and the estimation is biased by the residual service time, $X^{(h^*)}$, and cross traffic. Furthermore, $\mathbb{Q}_j^{(h')}/C^{(h')}$ is a good approximation for $\mathbb{Q}_{j-1}^{(h')}/C^{(h')}$ under certain conditions. For instance, when $\tau_{j-1,j}^{(h')}$ is small enough and Eqn. (7) still holds, $\mathbb{Q}_j^{(h')}$ is expected to be very close to $\mathbb{Q}_{j-1}^{(h')}$, thus making $\mathbb{Q}_j^{(h')}$ a tight lower bound for $B^{(h')} - S$. As a result, if $\mathbb{Q}_j^{(h')}/C^{(h')} \gg R_j^{(h')} + X^{(h^*)} + \zeta_j$ and $B^{(h')} \gg S$, then $\Theta_j \approx B^{(h')}/C^{(h')}$ which was first given in [31]. On the other hand, according to Eqn. (8), $\mathbb{Q}_j^{(h')}$ could be dampened when $A_{j-1,j}^{(h')} \ll D_{j-1,j}^{(h')}$ and $\tau_{j-1,j}^{(h')}$ becomes large, because the congestion may be relieved by the time p_j arrives.

4.3.2 LP_{01}

The analysis for the LP_{01} is similar to the above. When $\{p_{j-1}, p_j\}$ is an LP_{01}, we have

$$\mathbb{Q}_{j-1}^{(h')} + S \leq B^{(h')}, \text{ and} \quad (12)$$

$$\mathbb{Q}_j^{(h')} + S > B^{(h')}, \quad (13)$$

where

$$\mathbb{Q}_j^{(h')} = \left(\mathbb{Q}_{j-1}^{(h')} + S + A_{j-1,j}^{(h')} - D_{j-1,j}^{(h')}\right)^+. \quad (14)$$

By replacing $w_{j-1}^{(h')}$ with a similar expression as Eqn. (9), we obtain p_{j-1}'s path queueing delay, defined by $\Theta_{j-1} = d_{j-1}^{(n)} - \min_{\forall i,\ j=2i}\{d_{j-1}^{(n)}\}$:

$$\Theta_{j-1} = \frac{\mathbb{Q}_{j-1}^{(h')}}{C^{(h')}} + R_{j-1}^{(h')} + \zeta_{j-1}, \quad (15)$$

where $\zeta_{j-1} = \sum_{h=1}^{h'-1} w_{j-1}^{(h)} + \sum_{h=h'+1}^{n} w_{j-1}^{(h)}$ and $R_{j-1}^{(h')}$ is the residual service time upon p_{j-1}'s arrival. Unlike the LP_{10}, the LP_{01}'s path queueing delay does not contain $X^{(h^*)}$, and ζ_{j-1} contains fewer components.

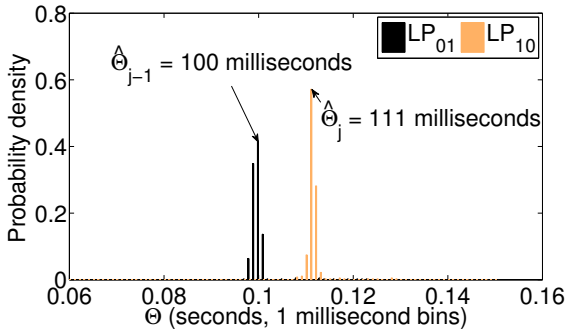
To estimate $B^{(h')}/C^{(h')}$ by the LP_{01}, note that $\mathbb{Q}_{j-1}^{(h')}$ serves as a tight lower bound for $\mathbb{Q}_j^{(h')}$ if $A_{j-1,j}^{(h')}$ is close to $D_{j-1,j}^{(h')}$ or $\tau_{j-1,j}^{(h')}$ is small enough. If p_{j-1} arrives at $H^{(h')}$ with its queue length not close to $B^{(h')}$ and $\tau_{j-1,j}^{(h')}$ is small, then Eqn. (13) will not hold with a high probability. However, if p_{j-1} arrives at an almost full queue and $\tau_{j-1,j}^{(h')}$ is small, it is more likely that Eqn. (13) will hold. As a result, if $\mathbb{Q}_{j-1}^{(h')}/C^{(h')} \gg R_{j-1}^{(h')} + \zeta_{j-1}$, $\tau_{j-1,j}^{(h')}$ is small enough, and $B^{(h')} \gg S$, then $\Theta_{j-1} \approx B^{(h')}/C^{(h')}$.

4.3.3 Testbed results

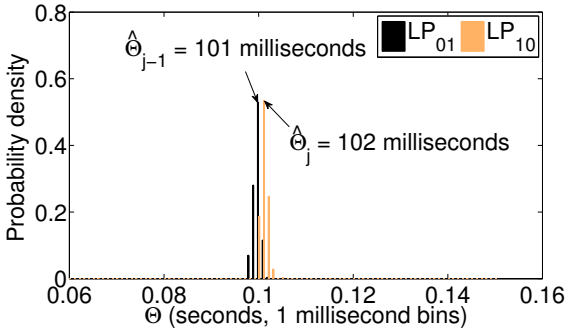
Figure 5(a) plots the path queueing delays for the LP_{10} and LP_{01} with $S = 1500$ bytes which are obtained from the previous set of testbed experiments. Notice that $B^{(h')}/C^{(h')}$ (= 100 ms) is much greater than $S/C^{(h')}$ (= 240 μs). We denote the bin with the highest count for the LP_{10} as $\hat{\Theta}_j$ and that for the LP_{01} as $\hat{\Theta}_{j-1}$. As shown, $\hat{\Theta}_{j-1}$ is the same as $B^{(h')}/C^{(h')}$. However, $\hat{\Theta}_j$ deviates from $B^{(h')}/C^{(h')}$ by about 11 ms, which is close to $X^{(h^*)}$ (= 1500 bytes/1 Mbit/s = 12 ms). Therefore, the results validate the contribution of $X^{(h^*)}$ to Θ_j, as modeled in Eqn. (11).

We also repeated the experiments by using a small probe packet size $S = 240$ bytes (with the same bottleneck link capacity $C^{(h^*)} = 1$ Mbit/s) and a larger bottleneck link capacity $C^{(h^*)} = 10$ Mbits/s (with the same probe packet size $S = 1500$ bytes), where $B^{(h')}/C^{(h')} \gg S/C^{(h')}$ for both cases. As shown in Figures 5(b) and 5(c), $\hat{\Theta}_{j-1}$ remains very close to $B^{(h')}/C^{(h')}$. Although $\hat{\Theta}_j$ may still deviate from $B^{(h')}/C^{(h')}$, the degree of the deviation becomes smaller, because of the decrease in $X^{(h^*)}$.

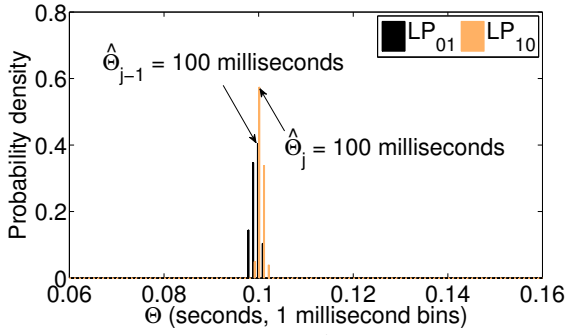
The estimates of $B^{(h')}/C^{(h')}$ made by the LP_{10} and LP_{01} are both prone to queueing delay at the non-congested nodes. However, the effect on the LP_{10}'s estimate is generally more



(a) $S = 1500$ bytes, $C^{(h^*)} = 1$ Mbit/s.



(b) $S = 240$ bytes, $C^{(h^*)} = 1$ Mbit/s.



(c) $S = 1500$ bytes, $C^{(h^*)} = 10$ Mbits/s.

Figure 5: Path queueing delays for the LP_{10} and LP_{01} on the testbed for which $C^{(h^*)} = \{1, 10\}$ Mbits/s, $C^{(h')} = 50$ Mbits/s, $h^* = 3$, and $h' = 5$.

significant than the LP_{01}'s, because the LP_{10}'s delay always contains $X^{(h^*)}$ which cannot be eliminated. Note that $X^{(h^*)}$ could be significant if the measurement is conducted using a low-bandwidth residential link. Though the impact of $X^{(h^*)}$ can be mitigated by choosing a smaller packet size for active loss-pair measurement, this is not feasible for passive loss-pair measurement. Whenever the packet size is not configurable, the LP_{01} should be used to avoid the bias.

4.4 Estimating $H^{(h^*)}$'s link capacity

In this section, we show that another benefit of the loss-pair analysis is estimating $H^{(h^*)}$'s link capacity from both LP_{10}'s delay and LP_{01}'s delay, assuming that both LP_{10} and LP_{01} observe the same congested hop $H^{(h')}$. Subtracting Eqn. (15) from Eqn. (11) gives

$$\Delta_{j-1,j} = \Theta_j - \Theta_{j-1} \quad = \quad X^{(h^*)} + \epsilon, \tag{16}$$

where $\epsilon = \frac{\mathbb{Q}_j^{(h')}}{C^{(h')}} - \frac{\mathbb{Q}_{j-1}^{(h')}}{C^{(h')}} + \zeta_j - \zeta_{j-1} + R_j^{(h')} - R_{j-1}^{(h')}$. Eqn. (16) shows that $\Delta_{j-1,j}$ includes a signature for $X^{(h^*)}$ and a noise term ϵ. Since the queueing delay and residual service times of p_{j-1} and p_j are contributed from different busy periods of the nodes, ϵ can be reasonably regarded as a random noise.

4.4.1 *Testbed results*

We conducted a new set of testbed experiments to evaluate this capability by configuring $\mathbb{R}_1$ to emulate $C^{(3)} = C^{(11)} = 10$ Mbits/s, and keeping $C^{(5)} = 50$ Mbits/s and $B^{(5)}/C^{(5)} = 100$ ms unchanged. Besides $H^{(5)}$, we also introduced the Pareto `On/Off` cross traffic between $\mathbb{X}_1$ and $\mathbb{X}_2$ in the forward and reverse paths. Other configuration settings were unchanged. As a result, $h^* = 3$ and $h' = 5$. We obtain the distribution of $\Delta_{j-1,j}$ by a mutual subtraction between Θ_{j-1} and Θ_j measured from `P01x–` and `P10xR00`, respectively, with $S = 1500$ bytes.

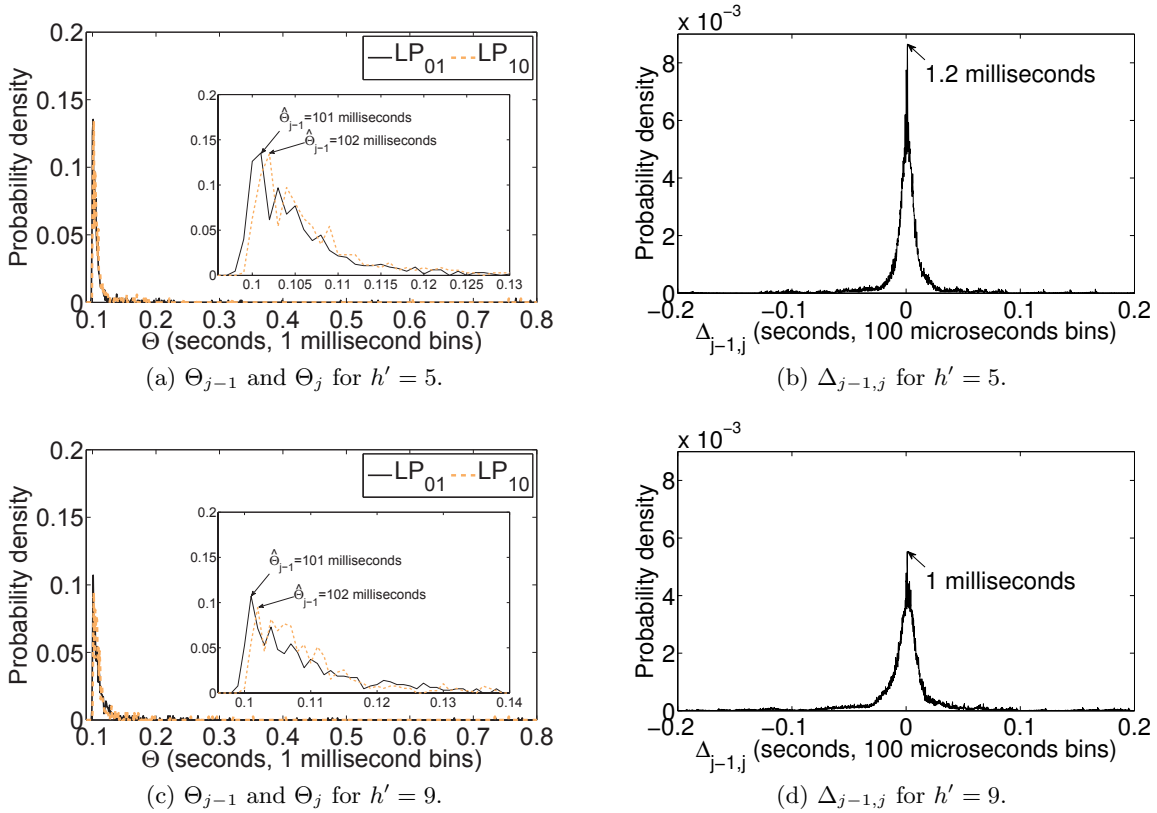
As shown in Figure 6(a), although $\hat{\Theta}_{j-1}$ and $\hat{\Theta}_j$ are relatively close to $B^{(h')}/C^{(h')} = 100$ ms, they also experience a higher variation due to the more significant cross traffic throughout the round-trip path. On the other hand, the probability density distribution of $\Delta_{j-1,j}$, shown in Figure 6(b), is symmetric about the peak at around 1.2 ms, which corresponds to the transmission delay of $H^{(h^*)}$ (i.e., $X^{(h^*)} = 1500$ bytes/10 Mbits/s). Thus, the peak of the distribution, together with the packet size, gives an accurate estimation of $H^{(h^*)}$'s link capacity. We also note from other testbed results (which are not shown in the paper) that $\Delta_{j-1,j}$ diminishes with the $H^{(h^*)}$'s link capacity and increases with the probe packet size. For example, for $S = 40$ bytes, we expect to use a microsecond bin size to make the transmission delay stand out in the distribution of $\Delta_{j-1,j}$.

We also include the results for the reverse-path loss pairs based on `P00xR10` and `P00xR01` in Figures 6(c)-6(d), and they are obtained by configuring $C^{(9)} = 50$ Mbits/s and $B^{(9)}/C^{(9)} = 100$ ms in the same testbed, and restoring the link capacity of $H^{(5)}$ to 100 Mbits/s with unlimited buffer. As a result, $h^* = 3$ remains unchanged, but $h' = 9$. Figure 6(c) shows that the corresponding $\hat{\Theta}_{j-1}$ and $\hat{\Theta}_j$ are still relatively close to $B^{(h')}/C^{(h')}$ and experience a similar variation due to the significant cross traffic introduced by $\mathbb{X}_1$ and $\mathbb{X}_2$. Moreover, as shown in Figure 6(d), $\Delta_{j-1,j}$ is quite similar to the forward-path results.

5. LOSS PAIRS IN THE INTERNET

We conducted end-to-end Internet path measurement between 26 February 2010 20:00 UTC and 17 March 2010 09:00 UTC, inclusively, using HTTP/OneProbe. The measurement covered a total of 112 (= 8×14) network paths between eight local universities in Hong Kong, denoted by UA–UH, as the sources of the paths and the 14 PlanetLab nodes listed in Table 1 as the destinations. Since HTTP/OneProbe performs measurement in a legitimate web session, we installed a `mini_httpd` (a web server) [40] at each PlanetLab node.

To monitor the path measurement from multiple sources to multiple destinations, we deployed a management system to dispatch the measurement tasks to the measurement



(a) Θ_{j-1} and Θ_j for $h' = 5$.

(b) $\Delta_{j-1,j}$ for $h' = 5$.

(c) Θ_{j-1} and Θ_j for $h' = 9$.

(d) $\Delta_{j-1,j}$ for $h' = 9$.

Figure 6: Path queueing delays and their differences for LP_{10} and LP_{01} with $S = 1500$ bytes on the testbed for which $C^{(h^*)} = 10$ Mbits/s, $C^{(h')} = 50$ Mbits/s, $h^* = 3$, and $h' = \{5, 9\}$.

Table 1: PlanetLab nodes used for the Internet path measurements.

Aliases	IP addresses	Locations	Average RTTs
PL001	212.235.18.114	Israel	308.24 ms
PL002	216.48.80.14	Canada	244.31 ms
PL003	202.112.28.98	China	83.67 ms
PL004	131.179.50.70	United States	–
PL005	128.143.6.134	United States	–
PL006	165.91.83.23	United States	229.55 ms
PL007	132.72.23.10	Israel	358.54 ms
PL008	210.123.39.168	Korea	53.34 ms
PL009	140.123.230.248	Taiwan	50.54 ms
PL010	134.151.255.181	UK	273.62 ms
PL011	142.104.21.241	Canada	248.99 ms
PL012	194.117.20.214	Portugal	–
PL013	198.82.160.239	United States	237.59 ms
PL014	137.132.80.110	Singapore	38.30 ms

nodes, monitor the resource usages in the nodes, and retrieve measurement data from the nodes. Each measurement node executed the measurement tasks to measure the network paths to the 14 destinations. To avoid self-induced network congestion, the destinations were evenly divided into two groups. The sources performed concurrent measurement for the paths in a group for one minute. Specifically, the sources launched HTTP/OneProbe to dispatch a sequence of Poisson-modulated probe pairs with a mean rate of 5 Hz and $S = 576$ bytes to each destination. To augment the path measurement with route information, tcptraceroute [49] was performed at both the sources and destinations. At the end of the minute, the nodes switched to the other group and repeated the same process. As a result, the average measurement traffic generated by each source was less than 48 KB/s (and less than 7 KB/s for each destination).

The measurement was conducted in the period during which HARNET [3]—a network through which the eight universities peered with one another—changed the service provider. In the switch-over process, the eight universities' networks were first switched to a temporary network one by one between 24 February 2010 14:00 UTC and 27 February 2010 23:00 UTC. They were then migrated back to the new service provider's network between 5 March 2010 11:00 UTC and 7 March 2010 2:00 UTC. As a result of these changes, we observed diverse network path characteristics even for the same source-destination pair.

In the stage of pre-processing the measurement data, we identified and removed a number of measurement artifacts. In particular, we identified artifacts associated with each source by correlating its measurement results. If there is a consistent pattern, such as persistent packet reordering, appearing in all the results, we conclude that the pattern is originated from the source or the path segment close to the source. This diverse-path-correlation method reveals the following measurement artifacts:

1. Forward-path and reverse-path reordering for UF between 27 February 2010 and 05 March 2010, and

2. Forward-path and reverse-path reordering for UH during the entire period.

Besides the artifacts, we observed system failures in three PlanetLab nodes PL004, PL005, and PL012 during the measurement period. After eliminating the paths to these destinations, we analyzed the remaining 88 paths for general packet loss statistics and loss-pair measurement.

5.1 Packet loss behavior: An overview

In this section, we present the overview results on the packet loss behavior, particularly the loss pairs, observed from the network paths. We consider the forward and reverse paths separately, because HTTP/OneProbe can distinguish the two paths for loss measurement. For the forward path, we define a *loss frequency* $f_{FL} = \sum_{i=1}^{M} \mathbf{1}_{\{L_i>0\}}/M$, where L_i represents the delivery status (i.e., `00`, `01`, `10`, or `11`) of the i^{th} probe pair, $\mathbf{1}$ is the indicator function, and M is the total number of packet pairs dispatched during a given time period. $L_i > 0$ if there is at least a packet loss in the pair, and $L_i = 0$, otherwise. For the reverse path, we apply a similar procedure to compute the loss frequency denoted by f_{RL}.

5.1.1 Prevalence of packet losses and loss pairs

We summarize in Table 2 the packet-loss and loss-pair statistics measured from all the paths for the entire measurement period. The table is organized based on the destinations. That is, the statistics for each destination is computed based on an aggregation of the path measurement from the eight sources to the destination. The statistics are also separated into forward and reverse paths. Besides the f_{FL} and f_{RL}, we also report $f_{\mathtt{P10}}$, $f_{\mathtt{P01}}$, and $f_{\mathtt{P11}}$ which give the respective percentages of `P10`, `P01`, and `P11` in the set of lossy probe pairs. The columns for $f_{\mathtt{R10}}$, $f_{\mathtt{R01}}$, and $f_{\mathtt{R11}}$ give similar statistics for the reverse path.

We observe the following results from Table 2:

1. Both f_{PL} and f_{RL} were less than 1% for most of the paths, and the highest loss frequency was 2.2% (i.e., PL003's f_{RL}).
2. Except for PL008 and PL011, the reverse paths suffered from more severe packet loss than the corresponding forward paths according to the loss frequencies. This result, however, is most likely location dependent.
3. Loss pairs were prevalent in the lossy packet pairs, because $f_{\mathtt{P11}}$ and $f_{\mathtt{R11}}$ were generally below 50% (except for PL007's $f_{\mathtt{P11}}$). The frequencies for some of the paths were even below 10%.
4. The LP_{01} dominated the LP_{10} in both forward paths and reverse paths, because $f_{\mathtt{P01}}$ ($f_{\mathtt{R01}}$) was consistently higher than $f_{\mathtt{P10}}$ ($f_{\mathtt{R10}}$).

5.1.2 Time series for packet loss events

To analyze the packet loss statistics as a function of time, we divide the entire measurement period into one-hour bins for each path. Each bin's value is set to 1 if there exists at least a one-minute session with loss frequency greater than 1%; otherwise, the bin value is set to 0. As a result, we obtain a time series of bin values for each path. We can combine the eight sources' time series for a given destination by adding their bin values. Alternatively, we can combine the 11 destinations' time series for a given source by also adding their bin values.

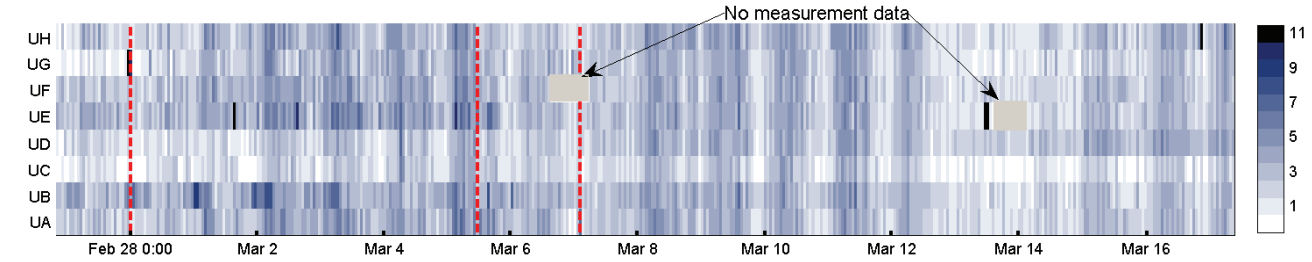
Table 2: Packet loss and loss pair statistics (in %) grouped by destinations.

	Forward Paths				Reverse Paths			
	f_{FL}	$f_{\mathtt{P10}}$	$f_{\mathtt{P01}}$	$f_{\mathtt{P11}}$	f_{RL}	$f_{\mathtt{R10}}$	$f_{\mathtt{R01}}$	$f_{\mathtt{R11}}$
PL001	0.04	29.01	33.02	37.97	0.13	24.19	33.60	42.21
PL002	0.16	13.09	57.52	29.39	0.35	19.41	42.53	38.06
PL003	0.23	47.72	51.03	1.25	2.22	41.73	41.76	16.51
PL006	0.01	20.56	43.89	35.55	0.10	30.84	38.10	31.06
PL007	0.07	23.08	25.73	51.19	0.15	25.86	33.86	40.28
PL008	0.67	15.25	34.80	49.95	0.33	25.56	31.26	43.18
PL009	0.29	44.41	44.97	10.62	0.69	44.50	45.80	9.71
PL010	0.01	21.72	36.55	41.73	0.16	29.03	37.38	33.59
PL011	0.17	44.75	49.62	5.62	0.09	36.05	41.61	22.34
PL013	0.04	33.15	40.08	26.77	0.11	29.06	36.78	34.16
PL014	0.93	44.46	47.06	8.49	1.93	47.38	48.02	4.59

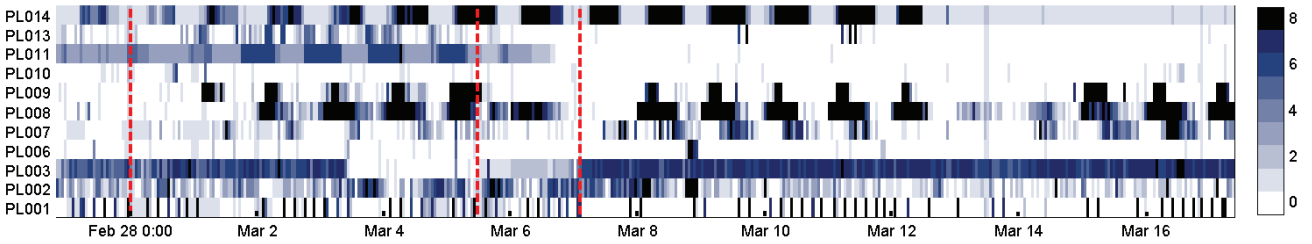
To effectively visualize the time series, we resort to heat-map diagrams. Figures 7(a) and 7(b) show the heat-map time series for the packet loss events in the forward paths grouped by the sources and destinations, respectively. Since there are 11 paths per source, the possible values in Figure 7(a) are $0, 1, \ldots 11$. A darker color corresponds to a higher value. We also grey out all the bins with no measurement data. Similarly, the possible values in Figure 7(b) are $0, 1, \ldots 8$. Moreover, there are three vertical dash lines: the first indicates the completion time for the transition to the temporary network, the second the beginning of the transition to the new service provider, and the third the completion time for the transition to the new service provider. We also show the diagrams for the reverse paths in Figures 7(c) and 7(d).

The heat-map diagrams enable us to effectively evaluate the loss behavior in the spatial and temporal domains:

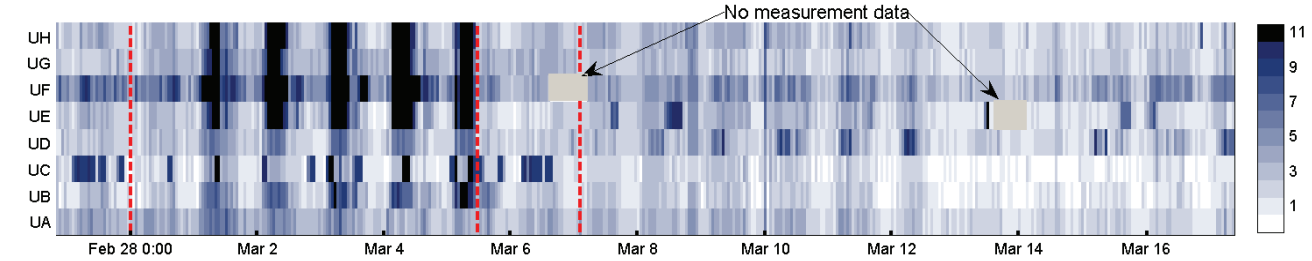
1. (Loss patterns) The heat maps can quickly identify loss patterns for a set of paths. In our case, the set of paths share either the same source or the same destination for forward/reverse paths. Figure 7(a) shows that there is no clear loss pattern for all the eight sets of source-identical forward paths. However, Figure 7(c) shows intense loss for some of the reverse paths during the network transition (between the first two dotted lines). On the other hand, Figures 7(b) and 7(d) depict that the loss is much more prevalent for some destination-identical paths.
2. (Loss correlations) The heat maps also reveal strong correlation among different sets of source/destination-identical paths. The most notable one is the periodic, intense losses for the UE, UF, UG, and UH paths in Figure 7(c). The similar pattern suggests that they probably shared the same loss origins. Moreover, Figure 7(d) shows that all 11 sets of destination-identical paths share similar reverse-path loss patterns during the network transition, but they are no longer similar after migrating to the new service provider's network.
3. (Loss diagnosis) We use heat maps to further diagnose the loss behavior by correlating the source-identical paths and destination-identical paths. Going back to the intense losses for the UE, UF, UG, and UH paths in Figure 7(c), we can obtain more insights by comparing Figure 7(c) and Figure 7(d) in the same five periods of heavy losses. Figure 7(d) shows that some destinations contributed losses to most of the reverse paths



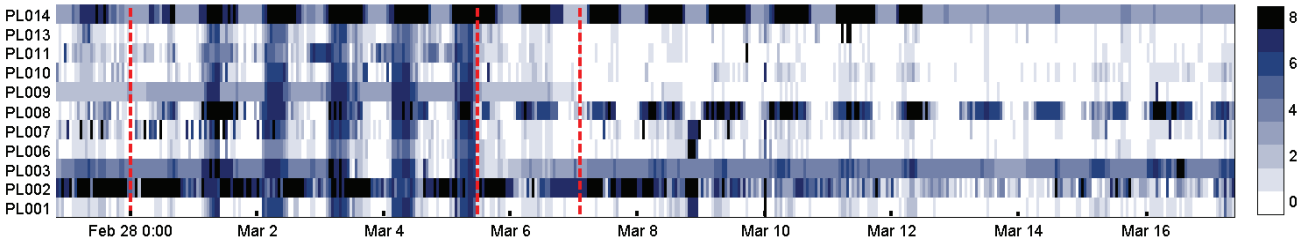
(a) Forward paths (grouped by sources).



(b) Forward paths (grouped by destinations).



(c) Reverse paths (grouped by sources).



(d) Reverse paths (grouped by destinations).

Figure 7: Heat-map time series for the packet loss events in the forward and reverse paths.

(notably PL014). Therefore, the losses for the four paths actually occurred on multiple locations: some on the destination side and others on the source side.

4. (Loss anomalies) The heat maps also help reveal loss anomalies. A time-correlation of the forward-path and reverse-path measurements based on the destinations shows that PL014 is a "congested" node. The paths to and from this node experienced high loss for all paths in a diurnal pattern until 13 March 2010. The loss could occur as a result of congestion at the node or the node's network. Since this path's loss measurement is heavily biased by the destination, a more useful path measurement can be obtained by replacing this with another node in the same vicinity. The forward paths to PL008 and PL009 also experienced periodic high losses. Unlike PL014, the loss patterns continued to the end of the measurement period.

5.2 Loss-pair analysis of the paths to PL009

In this section, we use the eight forward paths to PL009 as a case study of loss-pair analysis. Figure 8 shows the heat-map time series for the frequency of event P01x– obtained from the eight paths. We do not give the time series for the event P10xR00, because it exhibits a similar pattern shown in Figure 8. We compute the frequency for each path using one-hour bins and grey out all the bins with no measurement data. The loss-pair frequencies of the forward paths were 1-3%, and they distributed in several loss episodes, each of which lasted for several hours. In the ensuing discussion, we zoom into two loss episodes e_1 and e_2 in Figure 8 observed

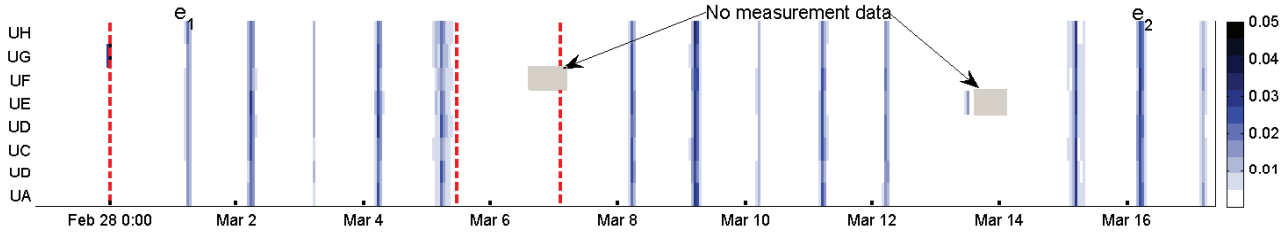


Figure 8: Heat-map time series for the frequency of event P01x– from UA–UH to PL009.

on 1 March 2010 (during the period of the temporary network operation) and 16 March 2010 (after the transition to the new service provider), respectively, with the same time period between 02:00 and 11:00 UTC on each day.

5.2.1 *The loss episode* e_1

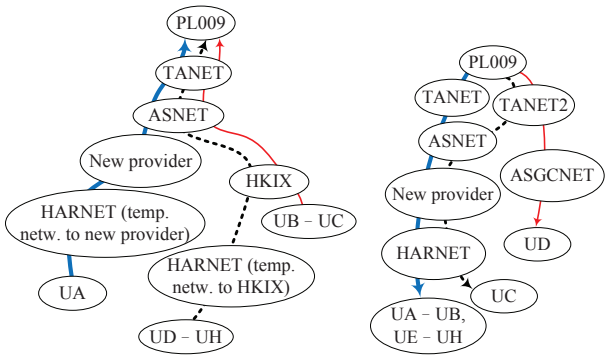
Figure 9 shows the RTT time series for the first packets (i.e., p_{j-1}) for the paths from UA, UB, UD, and UE during e_1. In each time series, we also superimpose the residual packets' RTTs for events P01x– and P10xR00 observed from the corresponding path. We do not show the time series for UC, UF, UG, and UH, because the time series for UC is similar to that for UB, and UF–UH to UE. The following highlights the main observations from Figure 9:

1. A minimum RTT (minRTT) of 30 ms was found for the path from UD and 32 ms for the others. Most of the RTTs were found below 100 ms for each path.
2. Except for the UA path, other paths experienced two RTT surges at around 02:15 and 06:15 UTC.
3. Forward-path loss pairs were observed between 03:00 and 07:45 UTC from all the paths.
4. Forward-path loss pairs were observed between 07:45 and 11:00 UTC only from the UD–UH paths.
5. The first packets' RTTs remained low and relatively stable between 02:35 and 06:15 UTC for all the paths. The loss pairs' RTTs clustered around the peaks and most of residual packets' RTTs for event P10xR00 were higher than that for event P01x– (which is consistent with our analysis in §4).
6. The first packets' RTTs became high and unstable after 06:15 UTC (especially between 06:15 and 09:45 UTC) except for the UD path. A significant variation was also observed from the loss pairs' RTTs, and many residual packets' RTTs for event P10xR00 were found lower than that for event P01x–.

Since diverse RTT characteristics were observed among the eight paths, we further analyze the tcptraceroute results for both the forward and reverse paths between the eight sources and the destination, and find that the sources actually used different IP routes to reach the destination. The tcptraceroute results also reveal that no IP route change occurred during e_1.

Figure 10(a) shows the eight forward paths to PL009. As shown, while the forward paths from UB–UH went through the peering of HKIX towards ASNET, the path from UA actually went through the new service provider to ASNET. As a result, the two RTT surges (point 2 above) were probably introduced by the HKIX network. Besides, we observe that only ASNET and TANET were involved in all the eight forward paths. Therefore, the loss pairs observed between 03:00 and 07:45 UTC (point 3) were probably introduced by a congestion point near the destination. However, since only UD–UH went through the temporary network to HKIX during e_1, the loss pairs observed between 07:45 and 11:00 UTC from their paths (point 4) were likely due to the congestion in this temporary network.

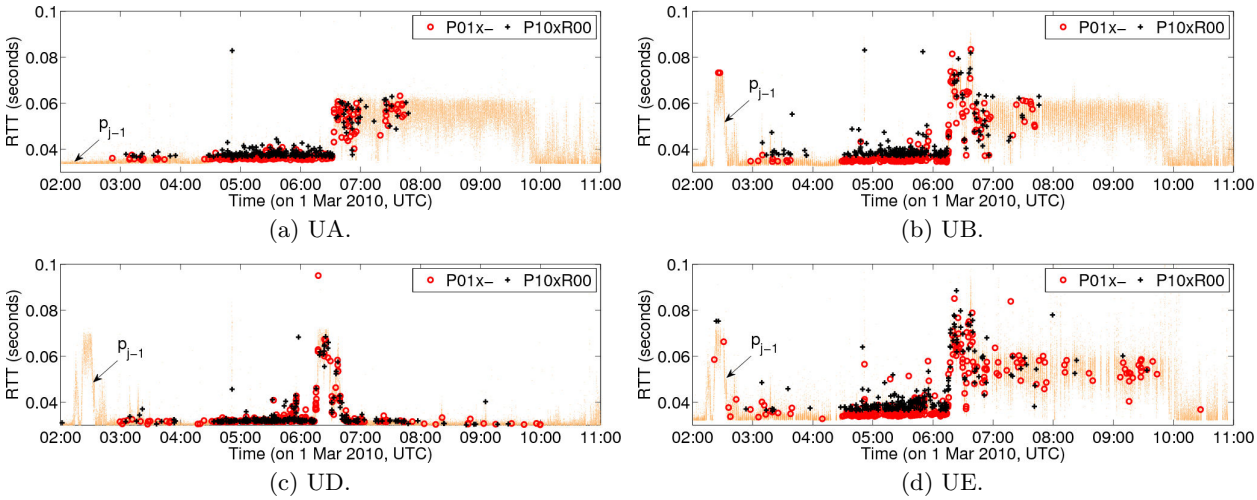
On the other hand, the tcptraceroute for the reverse paths provide additional information to reveal the effect of the reverse-path networks on the observed first packets' and loss pairs' RTTs. Figure 10(b) shows the reverse paths to the eight sources. As shown, only the reverse path to UD went through the ASGCNET network (with at least three router hops shorter). This observation suggests that the shorter minRTT observed from the UD path (point 1) was probably due to the shorter IP reverse route. While the other reverse paths from PL009 went through ASNET, the new service provider, and then HARNET to the sources, these paths actually shared only three common router hops in ASNET. Therefore, the RTT fluctuation after 06:15 UTC observed from most of the paths, except for the UD path (point 6), was introduced by another common congestion point in the ASNET network on the reverse paths.



(a) Forward paths. (b) Reverse paths.

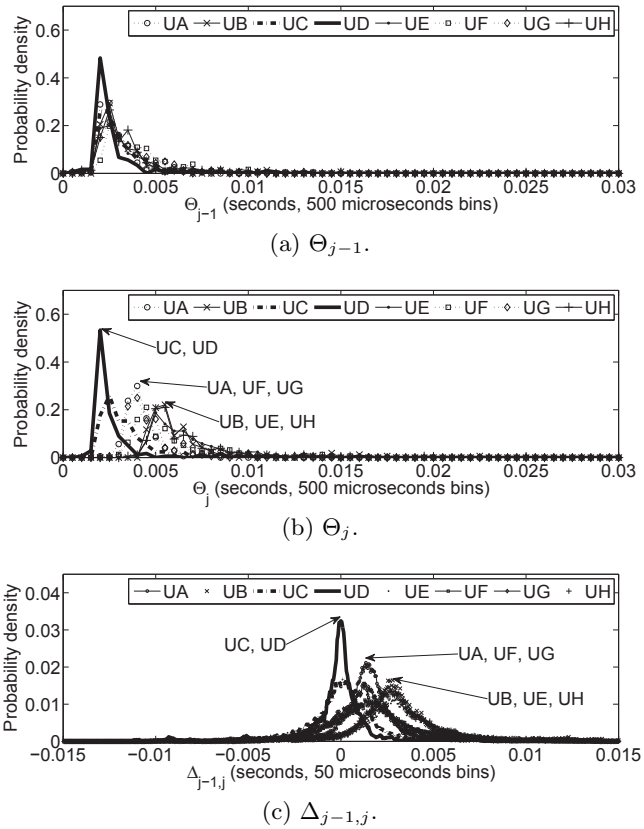
Figure 10: A comparison of forward and reverse paths between UA–UH and PL009 during e_1.

The observations above indicate that the eight paths exhibit relative stable RTTs and similar loss pairs' patterns between 02:35 and 06:15 UTC. To further characterize the properties for the eight forward paths, we compute the distributions of the residual packets' path queueing delays for events P01x– (i.e., Θ_{j-1}) and P10xR00 (i.e., Θ_j) in Figures 11(a)–11(b). Figure 11(a) shows that the modes of the path queueing delays for event P01x– were around 2 ms for the eight sources; therefore, the sources probably shared the same congestion point on their forward paths (which further supports our above findings). Moreover, by studying the



(a) UA. (b) UB. (c) UD. (d) UE.

Figure 9: RTT time series for the paths from UA, UB, UD, and UE to PL009 during e_1.



(a) Θ_{j-1}. (b) Θ_j. (c) $\Delta_{j-1,j}$.

Figure 11: Path queueing delays for loss-pair events P01x– and P10xR00 and their differences between 02:35 and 06:15 UTC during the loss episode e_1.

distributions of the path queueing delays for event P10xR00 shown in Figure 11(b), we obtain additional fingerprints for the eight paths and can further classify the sources into three groups: (i) UC and UD; (ii) UA, UF, and UG; and (iii) UB, UE, and UH.

Figure 11(c) shows the $\Delta_{j-1,j}$ distribution for each path based on the mutual differences between the corresponding residual packets' path queueing delays for events P01x– and P10xR00. As shown, the $\Delta_{j-1,j}$ distributions for the three groups are distinct from each other, meaning that they experienced different $H^{(h^*)}$'s configurations during the time period. For group (i), the figure shows that the corresponding link capacity was at least greater than 100 Mbits/s. However, we are unable to determine the exact value due to the coarse packet timestamp resolution. For groups (ii) and (iii), the estimated link capacities were at least 3 Mbits/s and 1.5 Mbits/s, respectively. Overall, the loss-pair analysis provides more comprehensive comparison of the eight paths and their characteristics which would not be easily discovered by considering only the loss frequencies (Figures 7(a) and 8) or the packet-pair RTTs (Figure 9).

5.2.2 *The loss episode e_2*

Figure 12 plots the time series of the first packets' RTTs observed from the UF's path to PL009 during e_2. Similarly, we superimpose the residual packets' RTTs for events P01x– and P10xR00 on the first packets' RTTs. Since this loss episode was located after the transition to the new service provider, it gives different path characteristics as compared with e_1. The figure shows that the minRTT for the UF path was around 65 ms, and most of the RTTs fell below 75 ms. We also observe similar RTT ranges for the other paths, except for the UC path whose RTTs ranged between 101 ms and 119 ms. Since all the eight paths exhibit very similar RTT time series patterns as in Figure 12, including two RTT surges at 04:50 and 05:30 UTC, we omit the time series for other paths.

Moreover, forward-path loss pairs were observed from the eight paths between 03:00 and 07:45 UTC. It is interesting to note that this time period is exactly the same as that in e_1 when forward-path loss pairs also existed in all the paths, although the loss pairs' RTTs found in e_2 mostly hit the highest values. This observation suggests that the transition event did *not* affect the congestion point in the forward path. Our tcptraceroute results for the forward paths obtained in e_2 also show that the forward paths still went through the same hops in ASNET and TANET observed during e_1.

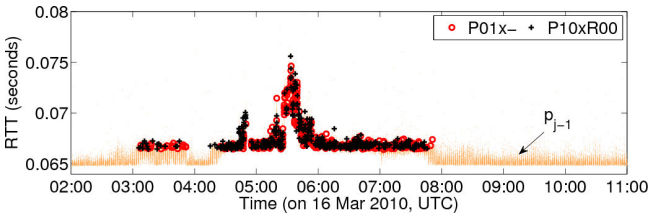


Figure 12: RTT time series for the path from UF to PL009 during the loss episode e_2.

We obtain the path queueing delays to further examine the impact of the transition to the new service provider. Figure 13 plots the distributions of the path queueing delays for events P01x– and P10xR00, and their differences obtained from the eight paths between 02:35 and 06:15 UTC during e_2. Figure 13(a) shows that the modes of the path queueing delays for event P01x– were still around 2 ms. Therefore, the transition probably had no impact on the congestion point encountered by the eight paths. However, Figures 13(b) and 13(c) show that the transition affected the configuration of $H^{(h^*)}$ for the eight paths (comparing with Figures 11(b) and 11(c)), where the distributions for both $\Delta_{j-1,j}$ and path queueing delays for event P10xR00 are very similar among the eight sources. As a result, the sources likely shared the same hop at $H^{(h^*)}$ after the transition.

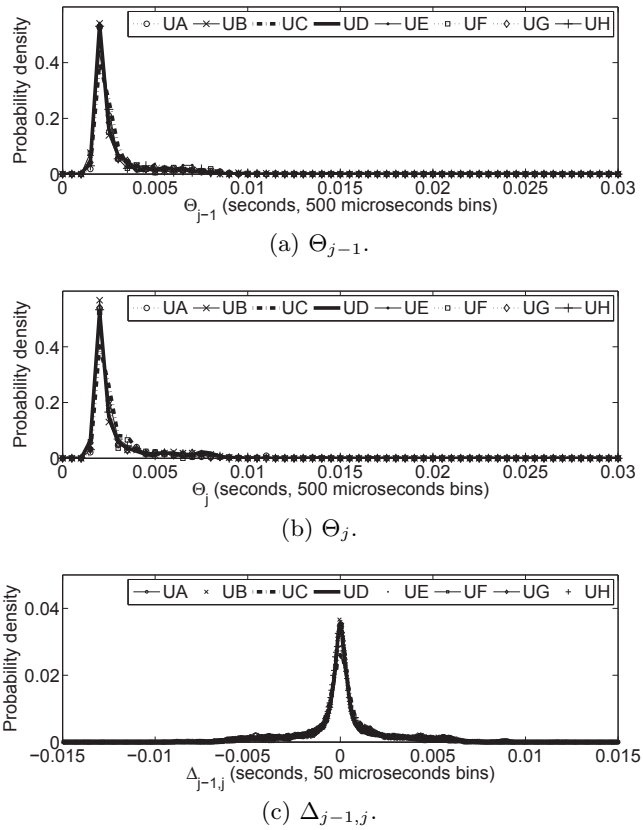
5.3 Loss-pair analysis of the paths to PL014

In this section, we apply the loss-pair analysis to the eight sources' reverse paths from PL014. Recall from Figure 7(d) that the reverse paths from PL014 exhibited significant packet loss during the measurement period. Figure 14 shows the heat-map time series for the frequency of event P00xR01. Similarly, the grey areas indicate the periods with no measurement data. We also note that the time series for the event P00xR10 (not shown in the figure) shows a similar pattern. Figure 14 shows that the frequency for each path was less than 4%. The reverse paths to UD, UF, and UG suffered a long-term loss episode for the entire measurement period, and the paths to others encountered several loss episodes before 13 March 2010.

5.3.1 *The loss episode e_3*

We analyze a reverse-path loss episode (labeled as e_3 in Figure 14) between 00:00 and 23:59 UTC on 8 March 2010. Figure 15 plots the RTT time series of the first packets and the residual packets' RTTs for events P00xR01 and P00xR10 obtained from the paths for UC and UF to PL014 during e_3. Figure 15(a) shows an RTT inflation period between 03:00 and 18:00 UTC. Note that most of the loss pairs were found within this RTT inflation period. Figure 15(b), on the other hand, also shows an RTT inflation period, but the loss pairs can be found throughout the measurement period. For other paths, the RTT time series for UA, UB, UE, and UH exhibit similar patterns as that for UC, whereas the time series for UD and UG are similar to that for UF. Therefore, we classify the eight sources into two groups: (i) UA, UB, UC, UE, and UH; and (ii) UD, UF, and UG. Moreover, we observe a minRTT of 50 ms for the UD path, and 35 ms for the other paths.

To characterize the congestion node's state encountered by the two groups of paths, we plot the path queueing delays



(a) Θ_{j-1}.

(b) Θ_j.

(c) $\Delta_{j-1,j}$.

Figure 13: Path queueing delays for loss-pair events P01x– and P10xR00 and their differences obtained from UA–UH to PL009 between 02:35 and 06:15 UTC during the loss episode e_2.

for events P00xR01 and P00xR10 and their differences during e_3 in Figures 16(a)-16(c). In particular, Figures 16(a)-16(b) show that group (i) exhibits a single mode at around 4 ms for the path queueing delays for both events P00xR01 and P00xR10. According to Figure 15(a), the path queueing delays represent the congestion experienced by the group of sources during the RTT inflation period, during which almost all the loss pairs were found. We also notice that group (ii) exhibits a similar (but weaker) mode at around 3.5 ms, but the mode vanishes if we only consider the loss pairs outside the RTT inflation period. Consequently, it seems that all eight sources suffered from the same congestion point during the RTT inflation. Based on the tcptraceroute results, the congestion point was very likely a router hop in the destination's network which was the only one present in all eight reverse routes.

Figures 16(a)-16(b) also show that group (ii) exhibits another stronger mode at around 500 μs. A further investigation finds that the mode was contributed by the loss pairs across the loss episode e_3 (instead of only outside the RTT inflation period). This observation suggests that multiple congestion points existed across e_3 for the paths in group (ii). Moreover, it is interesting to note from Figure 16(c) that the two groups experienced a similar $H^{(h^*)}$'s link capacity estimate of at least 100 Mbits/s.

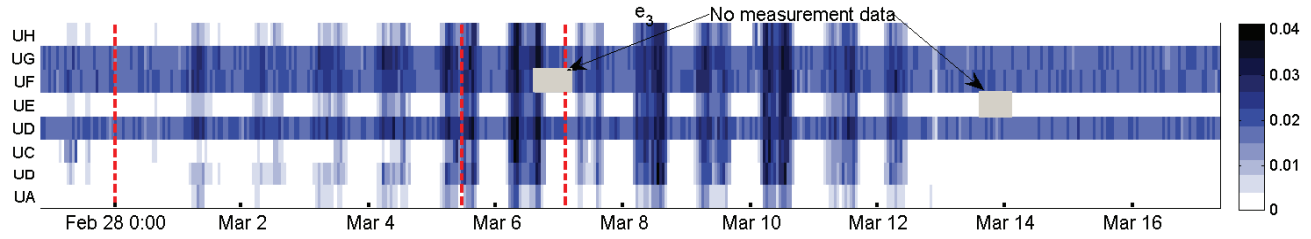
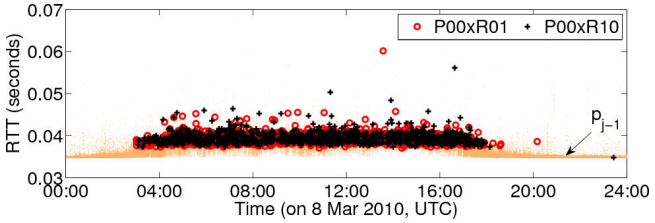
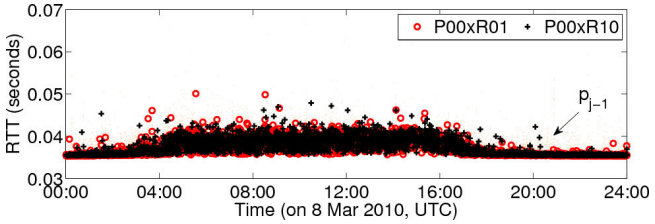


Figure 14: Heat-map time series for the frequency of event P00xR01 from UA–UH to PL014.

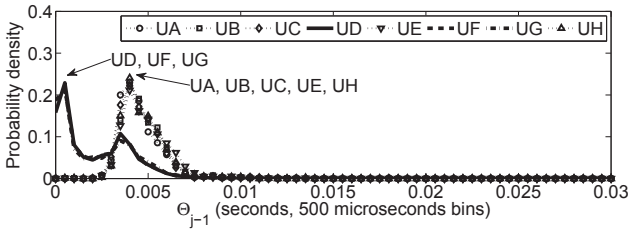


(a) UC.

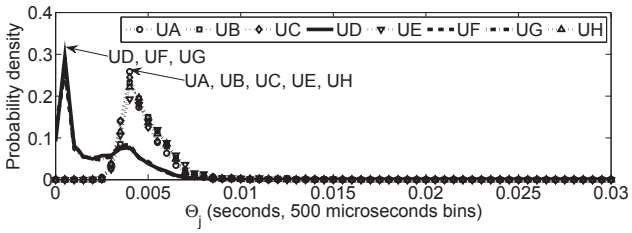


(b) UF.

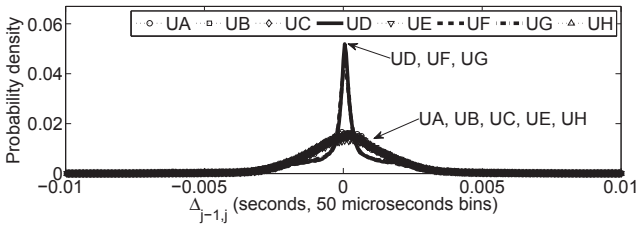
Figure 15: RTT time series for the paths from UC and UF to PL014 during the loss episode e_3.



(a) Θ_{j-1}.



(b) Θ_j.



(c) $\Delta_{j-1,j}$.

Figure 16: Path queueing delays for loss-pair events P00xR01 and P00xR10 and their differences obtained from UA–UH to PL014 during the loss episode e_3.

6. CONCLUSIONS

In this paper, we revisited the loss-pair measurement method proposed a decade ago. Based on our new analysis and Internet measurement results, we concluded that the loss-pair measurement is a very useful method for correlating a packet loss event and the delay that would be experienced by the lost packet. This correlation provides insight into, for example, the congested node's state upon packet drop, capacity of a link preceding the congested node, and loss asymmetry. Moreover, the loss-pair measurement could be incorporated in the existing measurement tools, such as capacity measurement based on packet-pair dispersion. A possible direction to extending this work is to integrate the loss-pair measurement with capacity measurement, and the other is to obtain useful path signatures for path fingerprinting and detecting common congestion points for multiple paths.

Acknowledgments

This work is partially supported by a grant (ref. no. ITS/355/09) from the Innovation Technology Fund in Hong Kong and a grant (ref. no. H-ZL17) from the Joint Universities Computer Centre of Hong Kong.

7. REFERENCES

[1] endace. http://www.endace.com/.
[2] TCPDUMP/LIBPCAP public repository. http://www.tcpdump.org/.
[3] The Joint Universities Computer Centre (JUCC). http://www.jucc.edu.hk/jucc/harnet.html, May 2010.
[4] A. Adams, J. Mahdavi, M. Mathis, and V. Paxson. Creating a scalable architecture for Internet measurement. In *Proc. INET*, 1998.
[5] M. Allman, W. Eddy, and S. Ostermann. Estimating loss rates with TCP. *ACM SIGMETRICS Perform. Eval. Rev.*, 31(3):12–24, 2003.
[6] F. Baccelli, S. Machiraju, D. Veitch, and J. Bolot. The role of PASTA in network measurement. In *Proc. ACM SIGCOMM*, 2006.
[7] F. Baccelli, S. Machiraju, D. Veitch, and J. Bolot. On optimal probing for delay and loss measurement. In *Proc. ACM/USENIX IMC*, 2007.
[8] J. Bellardo and S. Savage. Measuring packet reordering. In *Proc. ACM SIGCOMM IMW*, 2002.

[9] J. Bolot. End-to-end packet delay and loss behavior in the Internet. In *Proc. ACM SIGCOMM*, 1993.

[10] R. Carter and M. Crovella. Measuring bottleneck link speed in packet-switched networks. *Performance Evaluation*, 27-28:297–318, 1996.

[11] E. Chan, X. Luo, and R. Chang. A minimum-delay-difference method for mitigating cross-traffic impact on capacity measurement. In *Proc. ACM CoNEXT*, 2009.

[12] Y. Cheng, V. Ravindran, and A. Leon-Garcia. Internet traffic characterization using packet-pair probing. In *Proc. IEEE INFOCOM*, 2007.

[13] M. Coates and R. Nowak. Network loss inference using unicast end-to-end measurement. In *Proc. ITC Conf. IP Traffic, Modeling and Management*, 2000.

[14] L. Deng and A. Kuzmanovic. Monitoring persistently congested Internet links. In *Proc. IEEE ICNP*, 2008.

[15] G. Denisa, H. Nguyen, M. Kurant, K. Argyraki, and P. Thiran. Netscope: Practical network loss tomography. In *Proc. IEEE INFOCOM*, 2010.

[16] C. Dovrolis, P. Ramanathan, and D. Moore. Packet dispersion techniques and a capacity-estimation methodology. *IEEE/ACM Trans. Netw.*, 12(6):963–977, 2004.

[17] N. Duffield, L. Presti, V. Paxson, and D. Towsley. Network loss tomography using striped unicast probes. *IEEE/ACM Trans. Netw.*, 14(4):697–710, 2006.

[18] W. Feng, K. Shin, D. Kandlur, and D. Saha. The BLUE active queue management algorithms. *IEEE/ACM Trans. Netw.*, 10(4):513–528, 2002.

[19] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, and T. Berners-Lee. Hypertext transfer protocol – HTTP/1.1. RFC 2616, IETF, June 1999.

[20] S. Floyd and V. Jacobson. Random early detection gateways for congestion avoidance. *IEEE/ACM Trans. Netw.*, 1(4):397–413, 1993.

[21] K. Harfoush, A. Bestavros, and J. Byers. Robust identification of shared losses using end-to-end unicast probes. In *IEEE ICNP*, 2000.

[22] S. Hemminger. Network emulation with NetEm. In *Proc. Australia's National Linux Conference*, 2005.

[23] N. Hu and P. Steenkiste. Evaluation and characterization of available bandwidth probing techniques. *IEEE Journal on Selected Areas in Communications*, 21(6):879–894, 2003.

[24] V. Jacobson. Congestion avoidance and control. In *Proc. ACM SIGCOMM*, 1988.

[25] M. Jain and C. Dovrolis. End-to-end available bandwidth: Measurement methodology, dynamics, and relation with TCP throughput. *IEEE/ACM Trans. Netw.*, 11(4):537–549, 2003.

[26] R. Kapoor, L. Chen, L. Lao, M. Gerla, and M. Sanadidi. CapProbe: A simple and accurate capacity estimation technique. In *Proc. ACM SIGCOMM*, 2004.

[27] S. Katti, D. Katabi, C. Blake, E. Kohler, and J. Strauss. MultiQ: Automated detection of multiple bottleneck capacities along a path. In *Proc. ACM/USENIX IMC*, 2004.

[28] S. Keshav. A control-theoretic approach to flow control. In *Proc. ACM SIGCOMM*, 1991.

[29] E. Kohler. The Click Modular Router Project. http://read.cs.ucla.edu/click/.

[30] J. Liu. *Characterizing Network Elements and Paths Using Packet Loss Behavior*. PhD dissertation, Boston University, 2003.

[31] J. Liu and M. Crovella. Using loss pairs to discover network properties. In *Proc. ACM SIGCOMM IMW*, 2001.

[32] J. Liu, I. Matta, and M. Crovella. End-to-end inference of loss nature in a hybrid wired/wireless environment. In *Proc. WiOpt*, 2003.

[33] X. Luo, E. Chan, and R. Chang. Design and implementation of TCP data probes for reliable and metric-rich network path monitoring. In *Proc. USENIX Annual Tech. Conf.*, 2009.

[34] X. Luo and R. Chang. Novel approaches to end-to-end packet reordering measurement. In *Proc. ACM/USENIX IMC*, 2005.

[35] R. Mahajan, N. Spring, D. Wetherall, and T. Anderson. User-level Internet path diagnosis. In *Proc. ACM SOSP*, 2003.

[36] B. Melander, M. Bjorkman, and P. Gunningberg. A new end-to-end probing and analysis method for estimating bandwidth bottlenecks. In *Proc. IEEE GLOBECOM*, 2000.

[37] H. Nguyen and M. Roughan. On the correlation of Internet packet losses. In *Proc. ATNAC*, 2008.

[38] A. Pásztor and D. Veitch. The packet size dependence of packet-pair like methods. In *Proc. IWQoS*, 2002.

[39] K. Papagiannaki, R. Cruz, and C. Diot. Network performance monitoring at small time scales. In *Proc. ACM/USENIX IMC*, 2003.

[40] J. Poskanzer. mini_httpd: small HTTP server. http://www.acme.com/software/mini_httpd/.

[41] D. Rubenstein, J. Kurose, and D. Towsley. Detecting shared congestion of flows via end-to-end measurement. *IEEE/ACM Trans. Netw.*, 10(3):381–395, 2002.

[42] S. Saroiu, P. Gummadi, and S. Gribble. A measurement study of peer-to-peer file sharing systems. In *Proc. MMCN*, 2002.

[43] S. Saroiu, P. Gummadi, and S. Gribble. SProbe: A fast technique for measuring bottleneck bandwidth in uncooperative environments. In *Proc. IEEE INFOCOM*, 2002.

[44] S. Savage. Sting: A TCP-based network measurement tool. In *Proc. USENIX Symp. Internet Tech. and Sys.*, 1999.

[45] R. Sinha, C. Papadopoulos, and J. Heidemann. Fingerprinting Internet paths using packet pair dispersion. Technical Report 06-876, USC, Computer Science Dept., 2005.

[46] J. Sommers, P. Barford, N. Duffield, and A. Ron. Improving accuracy in end-to-end packet loss measurement. In *Proc. ACM SIGCOMM*, 2005.

[47] J. Sommers, P. Barford, N. Duffield, and A. Ron. A geometric approach to improving active packet loss measurement. *IEEE/ACM Trans. Netw.*, 16(2):307–320, 2008.

[48] M. Tariq, A. Dhamdhere, C. Dovrolis, and M. Ammar. Poisson versus periodic path probing (or, does PASTA matter?). In *Proc. ACM/USENIX IMC*, pages 10–10, 2005.

[49] M. Toren. tcptraceroute. http://michael.toren.net/code/tcptraceroute/.

[50] Y. Wang, C. Huang, J. Li, and K. Ross. Queen: Estimating packet loss rate between arbitrary Internet host. In *Proc. PAM*, 2009.

[51] W. Wei, B. Wang, D. Towsley, and J. Kurose. Model-based identification of dominant congested links. In *Proc. ACM/USENIX IMC*, 2003.

[52] M. Yajni, S. Moon, J. Kurose, and D. Towsley. Measurement and modelling of the temporal dependence in packet loss. In *Proc. IEEE INFOCOM*, 1999.

[53] Y. Zhang and N. Duffield. On the constancy of Internet path properties. In *Proc. ACM SIGCOMM IMW*, 2001.