

Unicast Probing to Estimate Shared Loss Rate

Dinh-Dung Luong¹, Attila Vidács¹, József Bíró¹, Daisuke Satoh², and Keisuke Ishibashi²

¹ Department of Telecommunications and Media Informatics
Budapest University of Technology and Economics
Magyar tudósok körútja 2, Budapest, H-1117 Hungary
{luong,vidacs,biro}@tmit.bme.hu

² NTT Corporation
3-9-11 Midori-cho, Musashino-shi, Tokyo, 180-8585 Japan
{satoh.daisuke,ishibashi.keisuke}@lab.ntt.co.jp

Abstract. The paper introduces a new receiver-based active end-to-end measurement technique, called the Single-Double Unicast Probing (SDUP), to estimate the rate of losses which occur on the shared network path of two flows. A comprehensive performance evaluation and a comparison between the SDUP and an existing technique having the same objective, the Striped Unicast Probing (SUP) [6], is carried out. We show that our proposed SDUP method has smaller bias (i.e., 20% smaller absolute error) and smaller variance compared to that of SUP. Besides the slightly higher accuracy, an important advantage of our method is that only one measurement equipment is needed at only one receiver, instead of deploying units to both receivers. Furthermore, with one sender and receiver pair we can measure not only one shared path, but any partial path that begins at the sender and locates on the path from the sender to the receiver. Our method is less intrusive and causes less bursty traffic compared to the SUP. The adaptation of these techniques into passive measurement is also considered.

1 Introduction

Packet losses inside communication networks are considered as the signals of traffic congestion or possible troubles, faults which need to be under control. Estimating different loss statistics has been a hot topic in network measurement. The well-known ping program, described in Chapter 7 of [9], sends ICMP *echo request* packets and receives *echo reply* packets to/from hosts in the networks. It can infer loss and delay statistics for the round-trip including forward and backward paths. The pathchar tool [5,3], by using TTL (Time-To-Live) field in IP header, is capable to estimate bandwidth, delay and loss properties of individual links in a network path. However, these estimated parameters are for two-way links. Besides, the pathchar has to face with several problems related to accuracy, measurement time and path asymmetry.

If one is interested in the loss rate of one-way path, he/she has to perform a receiver-based measurement. However, if the loss statistics of only one segment

in a network path are required, the simple receiver-based measurement is insufficient. The task of estimating the “one-way” loss rate in a segment of a network path is usually reduced to solving the following simpler problem: we have two flows of probe packets from one sender S to two receivers A and B (see Figure 1). The packets of both flows will firstly pass a common path then fork into two separate paths. We call this common path the shared path. The question is that how we can estimate the loss rates in the shared and separate paths.

To address the problem described above, a multicast-based measurement technique is proposed in [7,8]. The sender sends multicast probe packets to two receivers. The measurement technique uses the observation that if a probe packet is lost in the shared path, none of them will receive it (perfect correlation). Then the joint loss statistics at the receivers (e.g., the probability that both of the receivers receive a probe packet) depend not only on the loss rate at the shared path but on the two separate paths as well. From the derived relations, the three loss rates can be explicitly computed. The measurement technique can be applied for several multicast flows in a tree structure and the loss rate of each path segment can be inferred.

With unicast end-to-end measurement, estimating loss statistics in the shared and separate paths could not be done with the original technique in multipath probing. The problem is that when a multicast packet is replaced by unicast packets there is no perfect loss correlation between successive packets. Authors in [6] have used striped unicast probes (i.e., with no delay between transmission of successive packets within a stripe) to enhance the loss correlation. It has been reported that the typical absolute error of an estimate was about less than one percent when the size of stripe was increased to four packets.

This paper introduces an alternative unicast probing technique called the Single-Double Unicast Probing (SDUP) which sends single and pairs of packets. We give a comprehensive comparison between the SDUP and the SUP. The adaptation into passive measurements is also discussed.

The rest of the paper is organized as follows. Section 2 revisits the SUP. We describe the SDUP in Section 3. We evaluate and compare these two techniques in Section 4. Their adaptation into passive measurement is considered in Section 5. We conclude the paper in Section 6.

2 Previous Works

In [6] the authors adapt the multicast inference technique proposed in [7]. The idea here is the construction of composite probes of unicast packets, i.e., striping a group of unicast packets and dispatching them back-to-back, whose properties closely resemble those of a multicast packet. However, correlations within stripes may be less than perfect in practice (e.g., dispersion of stripes as they travel, loss events that are narrower than the stripe, interleaving of background traffic, or packet-dropping on the basis of Random Early Detection (RED)). We revisit the technique proposed in [6] next.

2.1 Binary Stripes

The main idea is illustrated using the two-leaf tree topology (see Figure 1) and a binary stripe. The server S sends packets back-to-back to clients A and B .

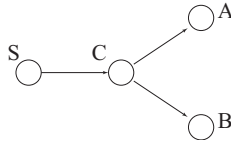


Fig. 1. Simple tree topology

Let $\alpha_k(D)$ denote the probability that packets in D ($D \subset \{1, 2\}$ with binary stripes) are successfully transmitted to node k ($k \in \{C, A, B\}$ in our case), conditioned upon having reached its previous node on the path. For example, $\alpha_A(1)$ is the probability that the first packet from the stripe successfully arrives to its destination node A , given that the packet was not lost on its way to node C . Thus, the probability that the first (second) packet reaches its destination, $g(1)$ ($g(2)$), can be calculated as

$$g(1) = \alpha_C(1)\alpha_A(1), \quad g(2) = \alpha_C(2)\alpha_B(2). \tag{1}$$

Note, that the probabilities $\alpha_k(i)$ are not necessarily the same for all i . Let $\beta_k(i|j)$ ($i \neq j$) denote the conditional probability that packet i is successfully transmitted to node k , given that packet j is successfully transmitted, all packets having reached the previous node. Thus we have

$$\beta_C(2|1) = \alpha_C(\{1, 2\})/\alpha_C(1), \quad \beta_C(1|2) = \alpha_C(\{1, 2\})/\alpha_C(2). \tag{2}$$

With perfect correlations the various β_k would be 1. The multicast loss model of [7] is statistically equivalent to this special case and hence $\alpha_k(i)$ all equal some α_k . The probability that both packets successfully arrive to the destination is the following:

$$g(12) = \alpha_C(12)\alpha_A(1)\alpha_B(2). \tag{3}$$

Here we used the notation $\alpha_k(12)$ instead of $\alpha_k(\{1, 2\})$ for simplicity. From Eqs.(2)-(3) we get the following relations

$$\alpha_C(1) = \frac{g(1)g(2)\beta_C(1|2)}{g(12)}, \quad \alpha_A(1) = \frac{g(12)}{g(2)\beta_C(1|2)}, \quad \alpha_B(2) = \frac{g(12)}{g(1)\beta_C(2|1)}. \tag{4}$$

Taking $\beta_C = 1$ yields the estimates

$$\hat{\alpha}_C = \frac{g(1)g(2)}{g(12)}, \quad \hat{\alpha}_A = \frac{g(12)}{g(2)}, \quad \hat{\alpha}_B = \frac{g(12)}{g(1)}. \tag{5}$$

However, with imperfect correlations β_C cannot be recovered from the $g(\cdot)$ success probabilities. Since $\beta_C \leq 1$, the estimations via Eq.(5) are biased, over-estimating α_C and underestimating α_A and α_B .

2.2 n -Packet Stripes

The authors in [6] propose a modified striping scheme for which the value of β is closer to one. The idea is to use longer stripes. For example, using three-packet stripes $\langle A, B, B \rangle$, it is more likely that packet 1 reaches C , upon reception of packets 2 and 3 at receiver B , rather than reception of packet 2 alone. Replacing the reception of packet 2 with reception of packets 2 and 3, the analogs of the estimates in Eq.(5) are

$$\hat{\alpha}_C = \frac{g(1)g(23)}{g(123)}, \quad \hat{\alpha}_A = \frac{g(123)}{g(23)}. \quad (6)$$

(Note that α_B can be estimated using the complementary stripe $\langle B, A, A \rangle$.) These estimates introduce less bias than those from two-packet stripes. In general, the bias can be further reduced by lengthening the stripe. A potential problem with this approach is that the statistical properties of stripes may not reflect those of actual traffic if their width is not negligible compared with buffer sizes. Using measurements and simulation the authors of [6] observed good accuracy even with packet pairs, with a typical error of about 1%, which significantly decreases as stripe length is increased to four packets.

3 The Proposed Single-Double Unicast Probing

3.1 The Model

The Single-Double Unicast Probing (SDUP) is found upon our observation that the loss probability for the second packet in a back-to-back packet pair at a buffer is approximately twice as large as the probability that the first packet is lost. We analyze the reasons and introduce the measurement technique using the following model and approximations. (We use the notations of [6].) Let us consider a set of links between a server and two clients in Figure 2. Together these links form a tree rooted at the server. We refer to the set of links en route to client A (B) as L_A (L_B) and the set of links that they share as L_S . We assume that all buffers use Drop-Tail buffer management. A 3-packet probe sequence $S_{i,j,i}(\Delta, \varepsilon)$ is a sequence of packets destined to clients i, j, i , respectively. The packet spacing between the two first packets is Δ time units, and is at most ε between the second and third packets (see Figure 3). The size of probe packets is small enough as compared to the average size of packets in the network. Let p_i^k be the steady-state probability that the queue at L_i can receive exactly k packets, and p_i^{k+} is the probability that the queue at L_i can accept k or more packets. Denote $g(1)$ the probability that the first packet in packet triples is successfully received by client B . Similarly, $g(3)$ is the probability that the third packet in packet triples arrive at the client B . Suppose that the loss processes at different links are independent. Then we have

$$g(1) = \prod_{i \in L_B} p_i^{1+}. \quad (7)$$

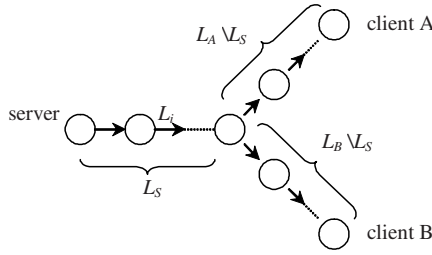


Fig. 2. Tree topology

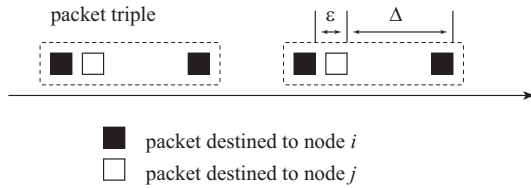


Fig. 3. 3-packet probe sequence

The third packet in each packet triple observes almost the same load condition at the shared path as the second packet does, since they have small size and are sent back-to-back. Then the third packet will pass the shared path if each queue in that path can accept at least two packets:

$$g(3) = \prod_{i \in L_S} p_i^{2+} \prod_{j \in L_B \setminus L_S} p_i^{1+}. \tag{8}$$

Combining Eq.(7) and Eq.(8) we get

$$Y \doteq \frac{g(1)}{g(3)} = \frac{\prod_{i \in L_S} p_i^{1+}}{\prod_{i \in L_S} p_i^{2+}} = \frac{\prod_{i \in L_S} (1 - p_i^0)}{\prod_{i \in L_S} (1 - p_i^0 - p_i^1)}. \tag{9}$$

The measure Y can be estimated from the estimates of $g(1)$ and $g(3)$. Eq. (9) deals with success probabilities. The probability that the first packet in a back-to-back packet pair is lost on link i is p_i^0 , while the probability that the following back-to-back packet is lost is $p_i^0 + p_i^1$. Our observation is that the latter loss probability is approximately twice as large as the probability that the first packet is lost. Thus we use the approximation that $p_i^0 \approx p_i^1$. (We will verify this approximation by simulation later.) In other words, it means that the loss probability at a queue is approximately equal to the probability that the queue can accept exactly one packet. Then we get

$$Y \approx \frac{\prod_{i \in L_S} (1 - p_i^0)}{\prod_{i \in L_S} (1 - 2p_i^0)}. \tag{10}$$

The loss probability at the shared path is $p_{sl} = 1 - \prod_{i \in L_S} (1 - p_i^0)$. We estimate this loss probability in two separate cases:

Case 1: Shared Losses Occur Only on One Link

In this case Eq.(10) reduces to

$$Y = \frac{1 - p_{sl}}{1 - 2p_{sl}}. \quad (11)$$

From that,

$$p_{sl} = \frac{Y - 1}{2Y - 1} = \frac{g(1) - g(3)}{2g(1) - g(3)}. \quad (12)$$

Case 2: Shared Losses Occur on Several Links

In this case, we use a further approximation:

$$1 - 2p_i^0 \approx (1 - p_i^0)^2. \quad (13)$$

Then

$$p_{sl} = 1 - \prod_{i \in L_S} (1 - p_i^0) \approx 1 - \frac{1}{Y} = \frac{g(1) - g(3)}{g(1)}. \quad (14)$$

3.2 Implementation

Assume that we would like to measure the loss rate on the shared path L_s (see Figure 2). Then the 3-packet sequence of UDP packets $S_{A,B,A}(\Delta, \varepsilon)$ should be generated at the server and sent to the receivers. (Note that the sequence $S_{B,A,B}(\Delta, \varepsilon)$ is an alternative probing where we have our receiver at node B .) The sending times of probing packets are scheduled as described above. For the case of $S_{A,B,A}$, each probe packet to client A carries a unique sequence number with that we can determine at node A whether a packet was lost or not. The first and third probe packets could be sent to two different port numbers in order to distinguish them easily.

Let us define $Z_1^{(k)}$ and $Z_3^{(k)}$ as binary variables indicating whether or not the first and third probe packets in packet triple k arrive at the destination, respectively. $Z_1^{(k)}$ and $Z_3^{(k)}$ will be one if the considered packet reaches its destination, and zero otherwise. Thus we get $EZ_1 = g(1)$ and $EZ_3 = g(3)$. The estimates of $g(1)$ and $g(3)$ will be

$$\hat{g}(1) = \frac{1}{n} \sum_{k=1}^n Z_1^{(k)}, \quad \hat{g}(3) = \frac{1}{n} \sum_{k=1}^n Z_3^{(k)}, \quad (15)$$

where n is the sample size. Then the estimate of shared loss rate can be observed using Eq.(12) or Eq.(14). (Note that Eq.(14) can also be used when the shared losses occur only on one link, but Eq.(12) is expected to be more accurate.) Furthermore, loss statistics of only the first and the third packet in packet triples are considered, while we ignore what happens with the second packet. We call the first and the third packets the *concerned packets* (packets in black in Figure 3) and the second packet will be named as the *unconcerned packet* (in white).

4 Evaluation and Comparison

4.1 Simulation Scenario

The performance of the SDUP have been evaluated by simulation and compared with that of the SUP from different perspectives: accuracy with Drop-Tail buffers, measurement complexity, impact on background traffic, and adaptation possibility into passive measurement. The evaluation and comparison are mainly based on simulation results, using the ns simulator [1].

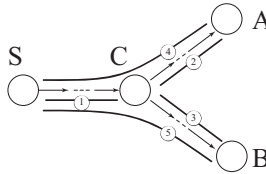


Fig. 4. Simulated tree topology with background flow groups

We set up a simple tree network configuration (see Figure 4) for simulation. The capacity of the shared links and the disjointed links were randomly chosen between 5 and 10 Mbps. The delay of each link was 50 ms. Five groups of background flows have been established (see Figure 4), the number of flows in each group was uniformly chosen from 10 to 20. A randomly chosen 75% of the flows use TCP, the others are CBR flows with on-off service times. The rate of each CBR flow was randomly chosen from 10 to 20 kbps, and the average on and off times were randomly chosen from 0.2 to 3.0 seconds. The background flows are started at time $t = -10s$, the probes start at time $t = 0$ and end at time $t = 120s$.

4.2 Queue Tail Approximation

Consider first the most important approximation in our loss estimation procedure. That is, to obtain Eq.(10), we assumed that $p_i^0 \approx p_i^1$, where p_i^0 is the probability that a queue at link i cannot accept any packet, and p_i^1 is the probability that a queue has space for exactly one more packet. We have verified this approximation by simulation using the scenario described above. Figure 5 shows the scatter plot representing p^1 versus p^0 , indicating that the probability p^0 approximates p^1 accurately with very small error.

4.3 Accuracy with Drop-Tail Buffers

The packet triples for the SDUP had parameter Δ of 8 ms and parameter ε of 1 μs . The inter-packet time between the first two packets is exponentially distributed with expected value of Δ time units. The time distance between two

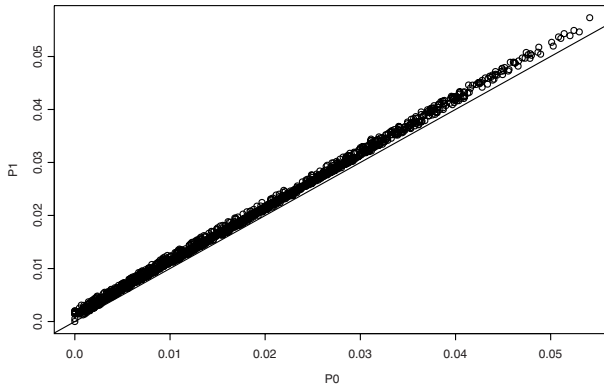


Fig. 5. Scatter plot of p^1 versus p^0 , where p^0 is the probability that a queue cannot accept any packet, p^1 is the probability that a queue has space for exactly one more packet. (The straight line is the line of $y = x$.)

successive packet triples is also exponentially distributed with expected value of Δ . The probe packets' size was 40 bytes. For the SUP experiments, the inter-stripe and inter-packet times were set as in [6]: 16 ms and 1 μ s, respectively. We considered 4-packet stripes for SUP since they are reported to give accurate estimations. We investigate the accuracy of the SDUP in two cases, we run 1000 experiments for each case:

Case 1: The shared path and the disjointed paths contain only one link. The estimated loss rate versus its actual value is shown in Figure 6.

Case 2: The number of links on the shared path as well as on the disjointed paths were randomly chosen between 1 and 5. The estimated loss rate versus its actual value is shown in Figure 7 for our SDUP method, and in Figure 8 for SUP. Comparing the results, there is no significant difference in accuracy between the two techniques, although the SDUP method seems to have smaller variance and less bias especially at low loss rates.

The average and the standard deviation of the absolute error (see Table 1) have changed slightly between the case of single link and the case of multiple links for the SDUP. Comparing with the results for the SUP, our proposed SDUP method has smaller bias (i.e., the absolute error is smaller by about 25%) and smaller variance compared to that of SUP.

Table 1. Statistics of the absolute error

Case	Average	Standard deviation
SDUP, DropTail, single shared link	0.0026	0.0032
SDUP, DropTail, multiple shared links	0.0033	0.0035
SUP, DropTail, multiple shared links	0.0044	0.0042

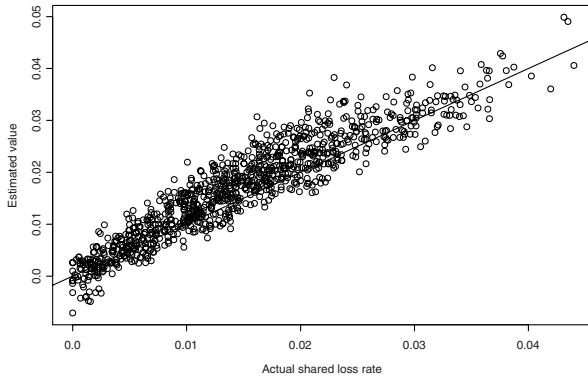


Fig. 6. The SDUP: Estimated loss rates versus their actual values for the case of single shared link. (The correlation coefficient of estimated and actual loss rate $R = 0.93$)

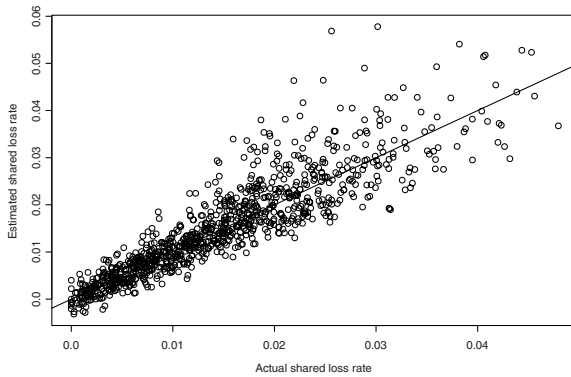


Fig. 7. The SDUP: Estimated loss rates versus their actual values for the case of multiple shared links. (Correlation coefficient $R = 0.89$)

4.4 Measurement Complexity and Flexibility

The measurement complexity intends to mean that how many measurement devices need to be set up and installed, and how complicated the measurement process is, including data collection and processing. The SDUP, as well as the SUP are receiver-based measurement techniques, then measurement equipments with installed measurement software are required at the receivers. The SUP requires measurement devices on both receivers. However, the SDUP needs it only on one receiver. Moreover, the processing of data is more simple since the SDUP takes into account only the packets received on one receiver.

An important advantage of the SDUP is that with one installed measurement device at the receiver, we can measure not only one shared path, but any partial path that begins at the sender and locates on the path from the sender to the receiver. We illustrate this in Figure 9: with a sender and receiver pair, we are

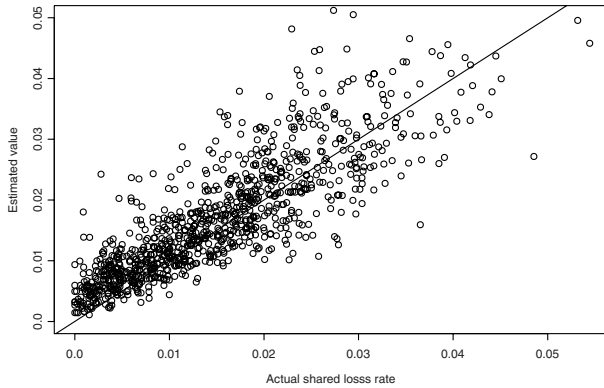


Fig. 8. The SUP: Estimated loss rates versus their actual values for the case of multiple shared links. (Correlation coefficient $R = 0.83$)

able to measure the loss rate of three shared segments: from S to node 1, from S to node 2 and from S to node 3. The implementation of SDUP in this case means that we need to deploy our measurement equipment only at the receiver, and generate probe packet triples at the sender. The only additional information needed is to choose the destination for the second (middle) probe packets such that they leave the shared path at node 1, 2 or 3, respectively, and register it in the probes that are to be received at the receiver. The advantage of the SDUP mentioned here is especially important if we cannot access other branches of the measurement tree, e.g., when there are branches belonging to another network.

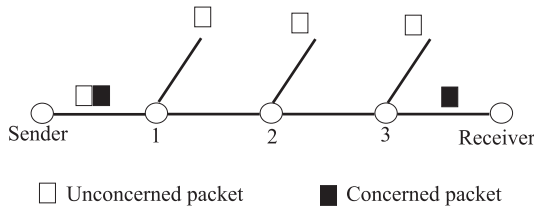


Fig. 9. With a sender and receiver set up once, SDUP can measure any shared path beginning at the sender and located on the path from the sender to the receiver

4.5 Impact on Background Traffic

In the SDUP, three packets are sent in each packet triple. In the SUP this number is at least the same, but four packets in a stripe would guarantee adequate accuracy. Suppose that the arrival rate of the triples is equal to that of the stripes, then the number of probe packets to be sent for the same measurement

Table 2. Some statistics of the extra loss rate caused by probe packets

Technique	Average extra loss rate	Standard deviation
SDUP	0.0041	0.0030
SUP	0.0049	0.0036

interval is larger in the case of SUP. Furthermore, the packets in stripes are back-to-back, then they would cause more bursty traffic than the triples do. We can guess that the SUP will cause at least an equal extra loss rate as the SDUP. In simulation experiments, we used 4-packet stripes and compared the extra loss rates generated by the two techniques in Table 2.

5 Adaptation Possibilities for Passive Measurement

Active measurements are intrusive since they inject probe packets into the network (e.g., they cause “extra loss”). With passive monitoring, we do not inject probe packets. However, we lose the ability to create packet patterns that are well controlled and fit our needs best. Instead, monitoring devices are placed inside or at the edges of the network. At the monitoring points packets can be captured, filtered and their collected statistics such as arrival time, packet size, header information (such as source and destination addresses) are stored and processed later.

There are mainly two kinds of packets in the Internet: UDP and TCP. The TCP header contains the sequence number and the source and destination information including address and port numbers that form the (nearly¹) unique identification of a TCP packet. We can determine that a TCP packet was lost or not using this identification. UDP packets do not have a sequence number or similar header fields to identify them uniquely. To measure the loss rate on a path, a monitoring device needs to be placed at the beginning of that path. The determination of the fact that a TCP packet is lost or not can be done by waiting for its retransmission packet. If a TCP packet is retransmitted, we can infer that it was lost somewhere on the path from the monitoring device to its destination. A TCP packet is considered successfully transmitted if it has no retransmission packet.

The SUP can be adapted to passive environment as follows. The monitoring device tries to capture stripes which contain only TCP packets, and where the first packet is targeted to a predefined destination while the remaining packets in the stripe are for another target. The difficulty of this solution is that such stripes are expected to be unlikely.

In the passive version of the SDUP it is not required to capture triples which are all TCP, that is, the unconcerned packet in the middle can be UDP because

¹ In theory two different TCP packets can have the same source, destination address, port number and sequence number. However, in a short measurement interval, it is very unlikely.

Table 3. Parameters observed from measurement

Parameter	LBL-4	LBL-5	DEC-1	DEC-2	DEC-3	DEC-4
$S(40)$	0.47	0.49	0.37	0.35	0.40	0.38
v_{tcp}	0.65	0.50	0.64	0.67	0.64	0.66
$q(2)$	0.17	0.19	0.24	0.23	0.19	0.22
$q(3)$	0.06	0.10	0.10	0.09	0.05	0.07
$Ra(40, 3)$	4.41	5.13	2.38	2.27	3.26	2.62
$Ra(40, 4)$	12.09	9.45	5.58	5.98	11.76	7.88

we do not have to concern its loss event. Moreover, the first (single) packet and the remaining back-to-back packets (the double packets) can be monitored separately. Due to the separate monitoring of single and double packets, the probability of successfully capturing them is considerably larger than that of striped packets, therefore the SDUP is easier to adapt into passive measurement.

To characterize the frequency of filtered stripes for passive SUP and double packets for passive SDUP, we use the following notations. Let $r(A)$ and $r(B)$ be the probabilities that a captured packet is destined to client A and client B , respectively. Let $S(s)$ denote the probability that a captured packet have smaller size than s (in bytes). Additionally, $q(n)$ denotes the probability that n successive captured packets have the same destination. For simplicity, we assume that these probabilities are mutually independent. Finally, v_{tcp} is the probability that a captured packet is TCP packet. Thus, the probabilities to capture appropriate SUP and SDUP probe packet sequences are:

$$Prob\{\text{double packets}\} = r(A)r(B)S(s)v_{tcp} \quad (16)$$

and

$$Prob\{n\text{-packet stripes}\} = r(A)r(B)q(n-1)v_{tcp}^2. \quad (17)$$

Then the ratio of the two probabilities above, denoted by $Ra(s, n)$, will be

$$Ra(s, n) = \frac{S(s)}{q(n-1)v_{tcp}}. \quad (18)$$

To evaluate Eq.(18) for real traffic, we used measured traffic traces. We analyzed the LBL-PKT and DEC-PKT traces provided by Lawrence Berkeley National Laboratory and are available from the Internet Traffic Archive [2]. The LBL-PKT traces were captured in 2 hours by monitoring all wide-area traffic into or out of the Lawrence Berkeley Laboratory, located in Berkeley, California, while the DEC-PKT traces are 5 hour-long wide-area traffic between DEC and the rest of the world, gathered at the primary Internet access point. Using these traces, the estimates of the related probabilities were calculated and are shown in Table 3. The results from the table indicate that if the size of the first packet within a SDUP packet pair must be smaller than 40 bytes, the probability of successfully capturing such packet pair is at least two times larger than that of capturing 3-packet stripes and at least five times larger than that of capturing 4-packet stripes.

6 Conclusion and Future Work

In the paper we introduced the Single-Double Unicast Probing (SDUP) to estimate the loss rate of shared path of two flows. We have evaluated the new technique and compared it with the Striped Unicast Probing (SUP)[6]. We have observed that our technique achieves slightly higher accuracy in the case of Drop-Tail buffers, but the main advantage of our proposed method is that SDUP provides higher level of measurement flexibility and low complexity, as well as it causes less measurement overhead than the SUP does. While the SUP requires measurement equipment devices on both receivers, the SDUP needs it only on one receiver. Moreover, with one installed measurement device we can measure not only one shared path, but any partial path that begins at the sender and is part of the sender-receiver path. The number of probe packets to be sent for the same measurement interval and accuracy is smaller with our method, and the generated probe traffic is less bursty, and thus less intrusive.

We also considered the adaptation of the two techniques into passive measurement. We have shown by analyzing measurement data that the probability of capturing appropriate packets for passive SDUP is considerably higher (e.g., ten times higher in certain cases) than that for passive SUP.

An important future work would be to examine and adapt the proposed method to the case when certain queue management methods (such as RED—Random Early Detection) are used in the network. In this case, packet losses not only occur when buffers overflow, and thus the correlation between back-to-back packets are weakened.

References

1. ns2: Network Simulator. <http://www-mash.cs.berkeley.edu/ns/ns.html>.
2. P. Danzig, J. Mogul, V. Paxson, and M. Schwartz. The internet traffic archive. <http://ita.ee.lbl.gov/>.
3. A. Downey. Using pathchar to estimate Internet link characteristics. In *Proceedings of the ACM SIGMETRICS International Conference on Measurement and Modeling of Computing Systems (SIGMETRICS-99)*, volume 27,1 of *SIGMETRICS Performance Evaluation Review*, pages 222–223, New York, May 1999. ACM Press.
4. S. Floyd and V. Jacobson. Random early detection gateways for congestion avoidance. *IEEE/ACM Transactions on Networking*, 1(4):397–413, August 1993.
5. V. Jacobson. Pathchar probe tool, 1997. <ftp://ftp.ee.lbl.gov/pathchar/>.
6. N.G. Duffield, F. Lo Presti, V. Paxson and D. Towsley. Inferring link loss using striped unicast probes. In *Proceedings of IEEE INFOCOM'01*, Anchorage, Alaska, April 2001.
7. R. Cáceres, N.G. Duffield, J. Horowitz and D. Towsley. Multicast-based inference of network-internal loss characteristics. *IEEE Transactions in Information Theory*, 45:2462–2480, 1999.
8. S. Ratnasamy and S. McCanne. Inference of multicast routing trees and bottleneck bandwidths using end-to-end measurements. In *Proceedings of IEEE INFOCOM'99*, New York, NY, March 1999.
9. W.R. Stevens. *TCP/IP Illustrated, Volume 1; The Protocols*. Addison Wesley, Reading, 1995.