Locating Congested Segments over the Internet Based on Multiple End-to-End Path Measurements

著者	Tachibana Atsuo, Ano Shigehiro, Hasegawa Toru, Tsuru Masato, Oie Yuji
journal or publication title	IEICE Transactions on Communications
volume	89
number	4
page range	1099-1109
year	2006-04-01
URL	http://hdl.handle.net/10228/00006304

PAPER *Special Section on Internet Technology VI*

Locating Congested Segments over the Internet Based on Multiple End-to-End Path Measurements

Atsuo TACHIBANA[†a)], Shigehiro ANO[†], Toru HASEGAWA[†], Masato TSURU[††], *Members*,
and Yuji OIE[††], *Fellow*

SUMMARY Since congestion is very likely to happen in the Internet, locating congested areas (path segments) along a congested path is vital to appropriate actions by Internet Service Providers to mitigate or prevent network performance degradation. We propose a practical method to locate congested segments by actively measuring one-way end-to-end packet losses on appropriate paths from multiple origins to multiple destinations, using a network tomographic approach. Then we conduct a long-term experiment measuring packet losses on multiple paths over the Japanese commercial Internet. The experimental results indicate that the proposed method is able to precisely locate congested segments. Some findings on congestion over the Japan Internet are also given based on the experiment.

key words: *packet loss, active measurement, locating congestion, network tomography*

1. Introduction

The Internet is currently serving as a communication infrastructure supporting various social and economic activities, and ISPs (Internet Service Providers) are thus required to provide Internet users with high quality communication. Nevertheless, in fact, congestion is still very likely to happen all over the Internet due to the inherent feature of providing best effort services. Network administrators of ISPs have to be aware of or predict network performance degradation, which can be due to congestion, and take some actions to mitigate or prevent it. However, a conventional way such as periodically monitoring the states of a number of individual network-internal points by SNMP (Simple Network Management Protocol) is not scalable for large networks in general, and impossible in some cases due to recent network service structures across multiple managed-domains. Therefore, our major concern here is to develop a way to detect whether or not congestion happens, and further infer which segments of the network are congested, by using multiple end-to-end performance measurements based on the network tomography, instead of directly monitoring the network-internal states. Each of segments can be a subnetwork and can include a number of links and routers. In response to finding congested network segments, the administrator managing the segment will mitigate the congestion in some fashion. Note that, for conciseness, we use a term "congested segment" as the segment in which too much packet losses happen, although the origins of packet losses are not only so-called "congestion" but routing problems (e.g., route changes) or hardware problems (e.g., router/switch interface failure). In fact, packet losses cause or indicate a considerable degradation of network performance in most cases, despite the origin of the losses.

In this paper, we propose a practical method to locate congested segments by actively measuring one-way end-to-end packet losses on appropriate paths from multiple origins to multiple destinations, which is intended to be employed in large and/or multiple managed-domain networks. Our interest is on inferring congested segments at each instant as well as likely (frequently) congested segments over a long term. In Sect. 2, we would address related work. In Sect. 3, we first describe the basic principle of our method on a simple topology where two end-to-end paths have a shared path segment. Then we explain our method generally using more than two paths so as to more precisely infer distinct congested segments. As described in Sects. 4 and 5, in order to assess the feasibility of our method, we conducted an experiment performing active measurements of 15-second averaged packet loss rates on 30 paths from multiple origins to multiple destinations, each of which traversed one or two ISPs, continually for a few months. In Sects. 5 and 6, we would also give some consideration and lessons tentatively obtained by the experiment. Finally, Sect. 7 concludes this work.

2. Related Work

A number of network tomographic approaches have been extensively studied to infer network-internal states from end-to-end measurement over multiple paths (e.g., [1]–[6]). Originally, such approaches exploited the correlation on behaviors of individual packets (e.g., a multicast packet or a pair of unicast packets) simultaneously traversing the multiple paths sharing a common part. However, as requiring precise (i.e., large) measurement data and/or complex computation, they might suffer from the scalability issues in large operational networks.

On the other hand, recent research has shown the possibility of inferring (locating) which segments are most likely to be congested solely by end-to-end path measurements without investigation of packet-level correlation. For ex-

Manuscript received July 8, 2005.
Manuscript revised October 3, 2005.

[†] The authors are with KDDI R&D Laboratories, Inc., Fujimino-shi, 356-8502 Japan.

[††] The authors are with the Dept. of Computer Science and Electronics, Kyushu Institute of Technology, Iizuka-shi, 820-8502 Japan.

a) E-mail: tachi@kddilabs.jp

DOI: 10.1093/ietcom/e89–b.4.1099

ample, Padmahbhan, et al. [1] identified, under some assumptions, the most lossy links on tree-structured downward paths from a web server to a number of clients based on packet loss rates of individual paths estimated by server-side observation on TCP packets. Note that the proposed inference method based on Bayesian estimation was accurate but computationally complex. Then, Duffield [2] clarified the reason (and general conditions) why such inference could work despite the problem not being statistically identifiable, and proposed a simple algorithm to infer the most relevant segment on a tree model.

For practically locating congested segments in a real-time manner, in the present paper, we propose the inference method along the line of the recent network tomography mentioned above. Our proposed method actively and continuously measures end-to-end packet loss rates in each measurement period on appropriate paths from multiple origins to multiple destinations. The contributions of this paper are as follows: (i) We employ a general model, instead of the tree model in [2], with tree, sink tree, and their combination, for simultaneous measurements of paths with multi-origins and multi-destinations. This enables us to efficiently and precisely locate the congested segments. (ii) We develop the concrete procedure and parameters in detail, in order to make the method usable in practice. And then we successfully examine and verify the proposed method through a real-world experiment. (iii) We give some findings on congestion over the Japan Internet based on analysis of the above experiment.

A recent study, based on network tomography similar to our method, extensively measures packet losses over the Internet [9]. However, since packet losses on just two end-to-end paths are measured, the path of interest is divided into just two segments, i.e. the access network and the backbone. The main objective is not to locate bad segments precisely, but to characterize the congestion behavior of access networks.

Some other related work from various view points should be addressed here. The constancy of packet losses on Internet end-to-end paths has been analyzed by many studies. For example, in [7], the packet loss rates over one-minute period are categorized into the following cases: $0-0.5\%$, $0.5-2\%$, $2-5\%$, $5-10\%$, $10-20\%$ and $20+\%$. Then, how long the packet loss rates remain in the same category is analyzed. The study showed that the packet loss rates over one-minute period were likely stable at least one hour. Some studies on locating congestion are not based on the network tomography. In [8], a bottleneck identification tool that issues many traceroute commands is developed. A bottleneck link is estimated as a link whose delays are becoming much larger than the minimum stable delay. The measurements over the Internet show that bottleneck links can be estimated; however, it is difficult to employ the tool in daily use due to extraordinary test packets.

3. Measurement Methodology

We explain our method of detecting and locating path segments that are likely to cause too much packet losses on a path solely from multiple end-to-end path measurements. Each path segment corresponds to a portion of an end-to-end path such as an ISP, an IX (Internet eXchange), or an access network. There are three steps to find such "congested segments."

1. The end-to-end packet losses on paths are measured to detect paths whose performance are considerably degraded. Test (probing) packets are sent at a small rate along each path continually and simultaneously, and packet losses of the test packets are measured. A path is regarded as being congested at a certain measurement period (e.g., 15 seconds), if the loss rate of the path in the period is more than a pre-defined threshold.
2. The candidate set of congested (bad) segments is inferred, which is most likely to cause performance degradation on bad paths in each measurement period. For example, in the left-hand side of Fig. 1, there are three segments S_1, S_2 and S_3, and two paths P_1 (from node o_1 to d_3) and P_2 (from node o_2 to d_3). Under appropriate settings and assumptions, we can employ the following rules:
   Rule (1):
   if P_i is good but P_j is bad ($i, j = 1, 2$) at the same period, then S_i and S_3 are good, but S_j is likely to be bad.
   Rule (2):
   if P_1 and P_2 are bad at the same period, then S_3 is more likely to be bad than S_1 and S_2 are.
3. After a series of measurement periods, the segments frequently congested are inferred and characterized based on the occurrences of the periods in which the segment is inferred as being bad in the previous step.

Hereafter, we explain the details of two rules in the second step mentioned above through a simple example. As shown in the right-hand side of Fig. 1, let N_0, N_1, and N_3 represent the number of the test packets leaving node $o1$, entering S_3, and arriving at $d3$ along path P_1, respectively. Similarly, N'_0, N'_2, and N'_3 are the number of the test packets leaving node $o2$, entering S_3, and arriving at $d3$ along path

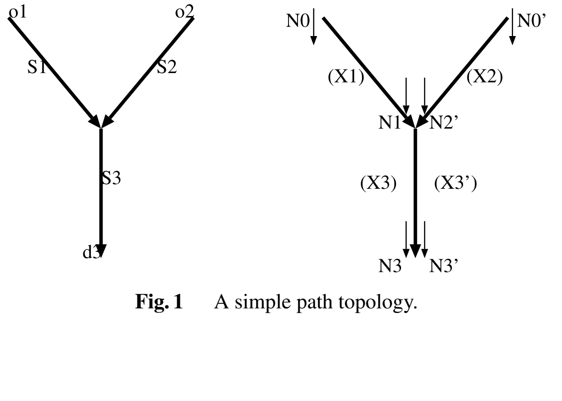


Fig. 1 A simple path topology.

P_2, respectively. Consider an empirical "non-loss rate" (or transmission rate) over a measurement period on each segment or each path as a random variable ranging from 0 to 1. Let $X_1 \equiv N_1/N_0$, $X_3 \equiv N_3/N_1$, $X_2 \equiv N_2'/N_0'$, $X_3' \equiv N_3'/N_2'$, which are (unobservable) non-loss rates for individual segments. On the other hand, $Y_1 \equiv N_3/N_0$ and $Y_2 \equiv N_3'/N_0'$ are (observable) non-loss rates for individual paths. Note that, "loss rate" is equal to one minus "non-loss rate," and the non-loss rate of a path is equal to the product of the non-loss rates of segments in the path (e.g., $Y_1 = X_1X_3$ and $Y_2 = X_2X_3'$).

Let us define l and h ($0 \ll l < h < 1$) as thresholds of non-loss rates representing "the bad condition" and "the good condition," respectively (e.g., $l = 0.99$ and $h = 0.995$). Segment i is called good if $X_i > h$, and bad otherwise, while path j is called good if $Y_j \geq h$, bad if $Y_j < l$, and medium otherwise. Basically we take the following assumptions (conditions) on non-loss rates of segments for sufficiently large N_0 and N_0':

- S_1, S_2, and S_3 are likely "good" so that "bad" is uncommon. In other words, X_1, X_2, X_3, and X_3' are likely near to 1.
- $X_3 = N_3/N_1$ is nearly equal to $X_3' = N_3'/N_2'$.
- X_1, X_2, and $X_3 (\approx X_3')$ are almost independent of each other.

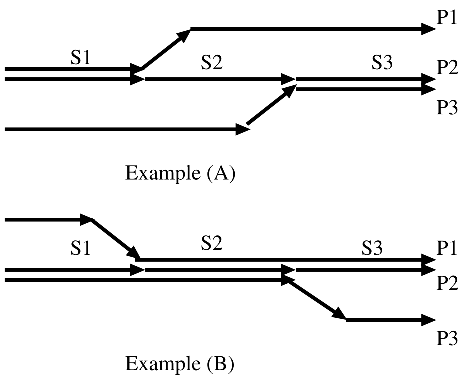
Rule (1)
For each measurement period, if we observe $Y_1 \geq h \wedge Y_2 \leq l$ (P_1 is good but P_2 is bad), then $X_1, X_3 \geq h$ because $X_1, X_3 \geq Y_1$. Furthermore, $X_2 < h$ is likely to hold (S_1 and S_3 are good, but S_2 is bad). This directly holds in case that $l < h^2$ (i.e., $l \leq l/h < \sqrt{l} < h \leq 1$) because $X_2 \leq l/X_3' \approx l/X_3 \leq lX_1/h < hX_1 \leq h$, which corresponds to "separability property" in [2]. For general topologies, this holds in case that $l < h^m$ where m is the maximum number of segments in a path.

On the other hand, in case that $h^2 \leq l$ (i.e., $l < h \leq \sqrt{l} \leq l/h < 1$), we can infer that S_2 is bad ($X_2 < h$) with some false positive probability, that is the conditional probability of $\{X_2 \geq h\}$ for given Y_1 and Y_2. Since $X_2 \geq h$ implies $hY_1/Y_2 \leq X_1 \leq 1$, $h \leq X_2 \leq Y_2/Y_1$, and $Y_1 \leq X_3 \leq Y_2/h$, if $Y_2/Y_1 < h$, then the above false positive probability is 0. Otherwise, the smaller Y_2/Y_1 is (close to h), the smaller the conditional probability of $\{X_2 \geq h\}$ is.

Rule (2)
For each measurement period, if we observe $Y_1 \leq l \wedge Y_2 \leq l$ (both P_1 and P_2 are bad), then $X_3 < h$ is more likely to hold (S_3 is likely to be bad) than X_1 and X_2 are. In this case, we choose S_3 as the candidate of bad segments, although we cannot say anything about the goodness/badness of S_1 and S_2. This is along the same line as the 'Smallest Consistent Failure Set Rule' in [2], which is justified when "bad" segments are uncommon and not correlated. Similarly to Rule (1), we can infer that S_3 is bad ($X_3 < h$) with some false positive probability, that is the conditional probability of $\{X_3 \geq h\}$ for given Y_1 and Y_2 (assuming $Y_1 \leq Y_2$). Since $X_3 \geq h$ implies $Y_1 \leq X_1 \leq Y_1/h$ and $Y_2 \leq X_2 \leq Y_2/h$, if Y_1 and Y_2 are small enough (e.g., $Y_2 < h^2$), both X_1 and X_2 are small (e.g., $X_1, X_2 \leq Y_2/h \leq h$), which unlikely happens under our assumption that S_1 and S_2 are independent and merely become bad. Hence the false positive probability is small. In general, as the number of measured paths sharing a target segment increases, the false positive probability of Rule (2) for the segment decreases.

In the following sections, we adopt this idea to a real-world case study where end-to-end measurements along several paths from multiple origins to multiple destinations are conducted. To do so, we extend the above rules to a general form of inferring a candidate set of bad segments from the observation of good/bad paths. That is, a candidate set of bad segments has the following properties: (i) each good path includes none of the segments in the candidate set; (ii) each bad path includes at least one segment in the candidate set; (iii) the number of segments in the candidate set is the minimum among all possible bad segment sets satisfying properties (i) and (ii). Note that there may be more than one candidate sets of bad segments.

In two examples shown in Fig. 2, suppose path P_2 is bad, and bad segments on the path should be inferred. Table 1 indicates the relations between observed goodness(G)/badness(B) of paths and inferred goodness(G)/badness(B) of each segment. For the segments inferred as being bad, "B" indicates a member of the candidate (inferred) set of bad segments that is the unique set satisfying properties (i) and (ii), while "(B)" indicates a member of the candidate set determined by using property (iii). "ND (Not be Determined)" indicates the segments whose state we cannot say anything about.



Example (A)

Example (B)

Fig. 2 Three paths P_1, P_2, P_3; and three segments S_1, S_2, S_3 on a bad path P_2.

Table 1 Mapping from observed badness of paths to inferred badness of segments.

case			example (A)			example (B)		
	P_1	P_3	S_1	S_2	S_3	S_1	S_2	S_3
i	G	G	G	B	G	(rarely occur)		
ii	G	B	G	ND	(B)	B	G	G
iii	B	G	(B)	ND	G	G	G	B
iv	B	B	(B)	ND	(B)	ND	(B)	ND

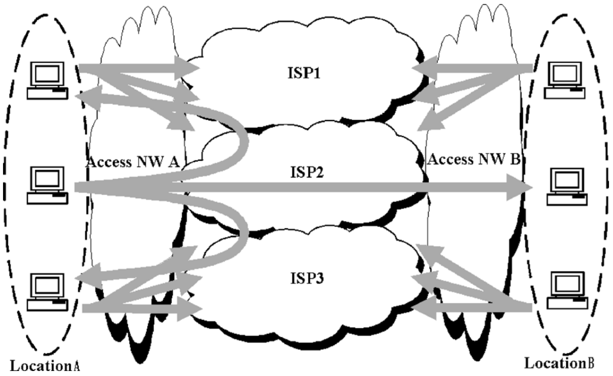


Fig. 3 Measurement infrastructure.

4. Measurement Experiments and Results

4.1 Measurement Experiments

(1) Measurement Infrastructure
We deploy a measurement infrastructure between two locations over the Japanese commercial Internet as shown in Fig. 3. The distance between them is about 1000 km. At the two locations, we subscribe access network services that provide accesses to ISPs (ISP1, ISP2 and ISP3) via optical fiber access links whose physical speed is 100 Mbps.
(2) Test Packets for Active Measurements
We simultaneously transmit test packets among six PCs (Personal Computers) on 30 paths in a full mesh manner. If a path is bad, we can use four paths of which origins are the same as the origin of the bad path, and four paths of which destinations are the same as the destination of the bad path. Hereafter, we call these eight paths "test paths" and the bad path "target path." We can infer bad segments based on the nine path measurements. The following UDP test packets are continually transmitted on all the above 30 paths.
• Test packets of measuring packet losses
UDP packets are sent not at a constant rate but with uniformly distributed intervals so that the UDP packets do not become synchronized with a particular queue behavior of the router.
- Packet interval: uniformly distributed between 10 ms and 90 ms. The average is thus 50 ms.
- Packet Size: 64 bytes
Twenty packets are transmitted on each path in one second. The bit rate of the test packets is only about 10 kbps. In our experiments, since the number of paths that traverse an identical link shared by these paths is at most five, the maximum bit rate of the test packets on such links is about 50 kbps, which is acceptable in broad-band network environments.
• Traceroute of measuring the route
Every one minute, a tester of origin issues a traceroute command to obtain the route information that is a list of routers' interface IP addresses. In the rest of paper, we call a series of them just a route.
(3) Analysis of Active Measurement Data

Table 2 10%-quantile of non-loss rate values of 30 paths for two ten-weeks.

10%tile of non-loss rate	number of paths	
	Period I	Period II
$P = 1$	17	20
$0.995 \leq P \leq 1$	6	0
$0.99 \leq P \leq 0.995$	2	2
$0.985 \leq P \leq 0.99$	4	6
$0.98 \leq P \leq 0.985$	1	2

We performed measurement experiments on all the 30 paths for two ten-weeks from April 4th to June 18th in 2004 and from February 14th to April 24th in 2005. In the rest of this paper, we call the first ten-weeks Period I and the second ones Period II.

4.2 Packet Loss Rates and Threshold

We choose non-loss rate thresholds of bad and good paths, i.e., l and h, based on 10%-quantile non-loss rate values for the two ten-week measurements. Table 2 classifies the 30 paths according to the values. Each row shows the number of paths of which the 10%-quantile non-loss rate value satisfies the condition of the left-most column. Since the proposed method cannot use medium paths whose non loss-rate is between l and h, there ought to be as few as possible. Since there are few paths whose 10%-quantile non-loss rate value is between 0.99 and 0.995 in both periods as shown in Table 2, we adopt 0.99 and 0.995 as l and h, respectively so that 0.99 (l) is equal to 0.995^2 (h^2). In addition, 1% packet loss rate, i.e., 0.99 non-loss rate, is feasible as a threshold for a bad path because many application performances such as TCP and VoIP (Voice over IP) may be degraded if the packet loss rate is more than 1%.

4.3 Method of Inferring Bad Segments

In this section, as an example, we show how bad segments are inferred on the path from ISP1 at Location A to ISP2 at Location B in Period I of which 10%-quantile of non-loss rate is 0.993. We adopt 15 seconds as a measurement period. The method performs the following procedure when the target path is bad.
(1) The method picks up the paths that are either good or bad. In other words, the paths such that $l <$ non-loss rate $< h$ are not used. If at least one of the test paths is medium, the method fails to infer bad segments at that period.
(2) The route might change on the Internet. Thus, the method checks whether the route of the current minute is the same as that of the previous one minute for every path. If these are not the same, it is probable that the route may change during the two-minute periods. If there is at least one path whose two routes are not the same, the method fails at the 15-second periods which are contained in that period.
(3) The method extracts distinct segments through which the target path is divided by the eight test paths. Each segment consists of some consecutive routers. Figure 4 shows the extracted segments for the target path from origin O, i.e.,

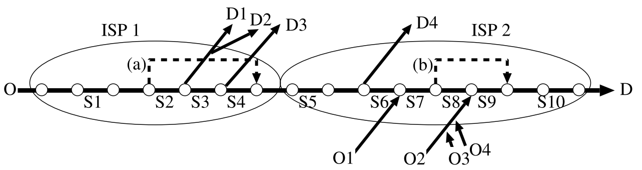


Fig. 4 Example of bad path's segments.

Table 3 Example of mapping table.

	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}
$O \rightarrow D$	bad	bad	bad	bad	bad	bad	bad	bad	bad	bad
$O \rightarrow D_1$	bad	bad								
$O \rightarrow D_2$	bad	bad								
$O \rightarrow D_3$	bad	bad	bad							
$O \rightarrow D_4$	good	-	-	-	good					
$O_1 \rightarrow D$							good	-	-	good
$O_2 \rightarrow D$									good	good
$O_3 \rightarrow D$									good	good
$O_4 \rightarrow D$									good	good
state	good	(bad)	ND	ND	good	ND	good	ND	good	good

ISP1 at Location A to destination D, i.e., ISP2 at Location B. It is divided into ten segments, i.e., segments S_1 to S_{10}, by the eight paths. The four test paths whose destinations are $D1$ to $D4$ have the same origin O as the target path. The other four test paths whose origins are $O1$ to $O4$ have the same destination as the target path. In Fig. 4, circles represent routers on the target path. As shown in Fig. 4, the path from O to $D4$ leaves the target path once from O to D, and then joins the path again as shown by the dashed arrow (a). On the contrary, the path from $O1$ to D also leaves and joins the path as shown by the dashed arrow (b). Then the method identifies an ISP which each segment belongs to using the whois server to all routers' interface IP addresses.

(4) The method creates a table whose rows and columns correspond to paths and segments, respectively. We call this table the mapping table. An entry records the state of the path if the path includes the segment. The state is good or bad according to the two non-loss rate thresholds as described in Sect. 4.2. On the contrary, if the path does not include the segment, the entry does not record anything, and it is not used to locate bad segments. Table 3 shows a mapping table for the actual bad path of Fig. 4. In this example, the path from O to $D4$ leaves the bad path and then joins the path again. In this case, the entry of the segment that the path does not include is marked as '-.'

(5) The method infers the state of each segment based on the mapping table according to the following steps.

- If at least one good path traverses a segment, the method infers the segment as being good by using Rule (1).
- Then the method deletes the above good segments from the table. If there remains only one segment in the table, the method infers the segment as being bad by using Rule (1).
- Otherwise (if bad paths traverse more than one segment), the method finds the minimum set of segments such that all paths are bad if the segments are bad. The method infers all the segments of the set are inferred as (bad) by using Rule (2). The other segments cannot be determined "ND."

Table 4 Periods of bad segments.

segment	**Rule(1)**	**Rule(2)**
S_1	0	922
S_2	0	493
S_3	0	41
S_5	0	209
S_7	0	0
S_9	0	0
S_{10}	0	11
S_4orS_6orS_8	381	0

4.4 Inference over 15-Second Period

During the two ten-weeks, there are totally 128661 periods in which at least one of the 30 target paths is bad. Bad segments are not inferred in 39090 periods due to the following reasons: First, in 25319 periods, the method cannot be used because a route change is observed. In particular, the target path from ISP1 at Location B to ISP2 at Location A and its test paths regularly change routes. However, the method can infer bad segments as long as no route change is observed during a period. Second, in 13771 periods, the method cannot infer bad segments because at least one of the test paths is medium.

We show the detailed result for the target path from ISP1 at Location A to ISP2 at Location B in Period I as an example. There are 2057 periods of no route change and no medium path out of 3214 periods of a target path's badness. Table 4 shows segments inferred as being bad. The first column shows segments illustrated in Fig. 4. The second and third column show the number of periods during which the segment of the first column is inferred as being bad by Rule (1) and Rule (2), respectively. As shown in Table 4, segments S_1, S_2, S_3 and S_4 are inferred as being bad by Rule (2). Segment S_4, S_6 or S_8 are inferred as being bad by Rule (1).

4.5 Inference over a Series of 15-Second Periods

Choosing a target path from ISP1 at Location A to ISP2 at Location B during Period I, we analyze correlation on the inference results and states of paths over a series of 15-second periods.

(1) Correlation on Inference Results

Table 5 shows the inference results over a series of 15-second periods. The first column shows a run length of 15-second periods in which the target path is bad. The second column whose top is marked as total shows the number of the series. The third column whose top is marked as inferred shows the number of series in which there is no medium test path so that the bad segments are inferred in all 15-second periods of the series. The fourth column whose

Table 5 Inference results over a series of 15-second.

run length	**frequency(times)**		
(periods)	**total**	**inferred**	**same segments**
1	1966	1405	-
2	542	217	195
3	43	19	14
4	5	2	2
5	3	0	0

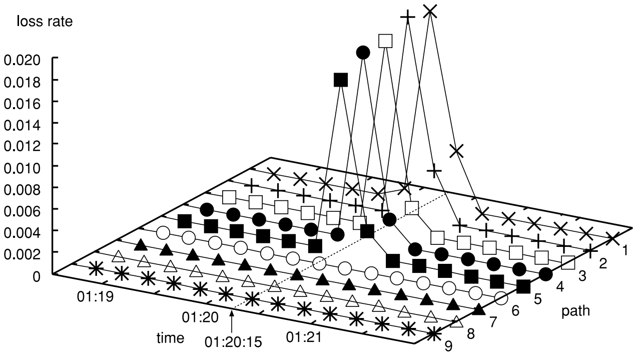


Fig. 5 Example of path synchronization.

top is marked as same segments shows the number of the series such that same segments are inferred as being bad in all the 15-second periods of the series.

Analysis of Table 5 is summarized as follows:

- A bad state does not usually continue for more than one period. A run length of the 1966 series is one 15-second period. On the contrary, only three series' run lengths are more than five periods.
- In the case where a run length is more than two 15-second periods, the same segments are likely to be continuously inferred as being bad. For example, as for the two 15-second series, the same segments are inferred as being bad in 195 series out of 217 series of "inferred." This continuity of inference may be indirect evidence that bad segments are correctly inferred.

(2) Correlation on Path States

To determine correlation on path states, we analyze the packet loss rate change of paths that become bad due to the same segment's badness. As an example, we choose segment S_1 of Fig. 4 and show the packet loss rate changes of nine paths of 01:19 to 01:22 in Fig. 5. The path states are synchronized. The packet loss rates of all five paths traversing segment S_1 suddenly increase at the same 15-second (i.e., 01:20:15) and decrease at 01:20:30. Besides which, we see a similar synchronization in most cases where the five paths become bad. In this case, our method infers that segment S_1 is bad. On the contrary, if segment S_1 were not bad, two segments S_2 and S_5 would become bad at the same period. However, this is very improbable because we assume that bad segments are uncommon and unlikely to be correlated.

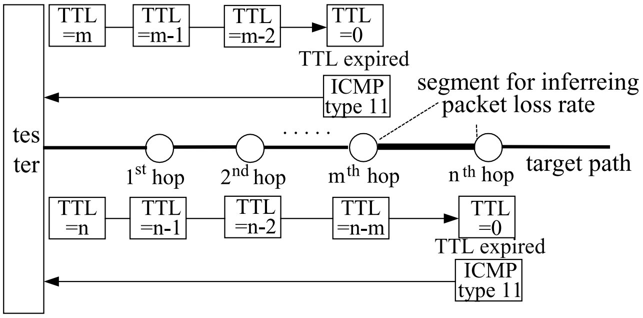


Fig. 6 Method of inferring packet loss rate.

4.6 Verification Based on Pathping-Like Method

Several tools such as pathping [10] and Bfind [8] were developed to infer states of links. They send many test packets to both the end routers of a link, and receive ICMP messages generated by the routers. The link's state is inferred based on the difference between the round-trip performance of test packets returned from the routers just before and just after the link. Although this approach is inefficient if we want to check the states of every link due to its brute-force manner and not suitable for daily use, it is feasible to use this approach for inferring the state of a specific segment. Therefore, we compare the inference results of our method and those of a pathping-like method.

(1) Pathping-like method

We develop a pathping-like method. First, a tester is set at the same place as that of an origin, e.g., O of Fig. 4.

- The tester sends a series of test packets to both routers at the edges of a specific segment as shown in Fig. 6. Test packets are small size UDP packets, (i.e., 40 byte packet in Period I and 64 byte in Period II). A TTL (Time To Live) value of the test packet is set at a hop number from the tester to the router. For example, the TTL value of a packet sent to a router of the left end, i.e., the router of the m-th hop in Fig. 6, is set at m.
- When a test packet arrives at the router, the TTL value becomes 0, i.e., it is expired, and the router sends back an ICMP type 11 message to the tester.
- The interval between test packets is 50 ms. We choose 50 ms based on the preliminary experiments. We sent test packets changing intervals from 10 ms to 50 ms on several paths that are usually good. We observed at what intervals routers skip sending back ICMP type 11 messages. As a result, we know that routers do not skip if an interval is equal to or more than 50 ms.
- The sending time of the two series of test packets is staggered in 10 ms intervals to avoid simultaneous packets from causing congestion. The test packets to the further hop are sent earlier than those to the nearer one from the tester.

Second, the tester counts the numbers of received ICMP messages. Since received ICMP messages mean that the test packet correctly arrives at the router, a difference between the ICMP message numbers is regarded as the number of packets lost on the link.

(2) Verification Experiments

We chose five and three segments that were frequently inferred as being bad during Period I and Period II respectively. These eight segments are on different paths. Then we simultaneously performed both measurements of our method and the pathping-like method for ten days during each period, which were from June 8th to 18th in 2004 and from April 1st to 10th in 2005 respectively. Since the pathping-like method inflicts a heavy load on routers, it is applied to only one segment at a time. We then compared the both results. In the case of 0.99 (l) and 0.995 (h) thresholds, badness means that the non-loss rate of a bad segment is likely to be less than 0.995. However, the absolute values of non-loss rates inferred by the pathping-like method are not accurate due to error factors: difference in the sending time of both series of packets, packet losses (ICMP message losses) on return direction and so forth. Therefore, we use the change of segment states in time. We consider that when a segment inferred as being bad by our method satisfies the following conditions, the segments is actually bad.

- Non-loss rates of the segment in both the previous and the next 15-second periods inferred by pathping-like method are higher than in 0.005 that in the current 15-second period.

Table 6 shows the inference results of both methods. Segments S'_1 to S'_5 are measured in Period I and segments S'_6 to S'_8 are measured in Period II. The second column shows the number of 15-second periods in which the targeted segment of the first column is inferred as being bad by our method and, at the same time, the pathping-like method measures the non-loss rate of the segment. Note that, since only one segment is measured by the pathping-like method at one period, the segment inferred as being bad by our method is not always measured by the pathping-like method. The third column shows the number of 15-second periods in which the targeted segment is inferred as being bad by our method and is also regarded as bad based on measurement by the pathping-like method as mentioned above. As a result, we observe that the bad segment inference by our method is verified by the pathping-like method in 116 periods out of 143 15-second periods. The high probability, i.e., 81%, that the both results are the same is also indirect evidence of the correctness of our method.

Table 6 Verification results for the proposed methods.

	inferred peirods		
segment	our method	both method	pathping-like method
S'_1	9	9	45
S'_2	40	32	141
S'_3	4	4	21
S'_4	31	20	40
S'_5	10	9	22
S'_6	29	21	46
S'_7	8	6	19
S'_8	16	15	34
total	143	116	369

The fourth column shows the number of 15-second periods in which the targeted segments' non-loss rates estimated by the pathping-like method are less than or equal to 0.995. By comparing the second and the fourth column, the number of periods inferred by our method is about 39% (143/369) of those inferred by pathping-like method. This is due to the following reasons: First, the non-loss rate measured by pathping-like method contains packet loss on return directions, then pathping-like method tends to estimate non-loss rate lower. Second, the packets series of each method are sent at deferent time independently, Third, our method stops the inference when at least one of the test paths is medium.

5. Congestion Trend of the Internet

In this section, we discuss our findings based on analysis of 30 paths of the two ten-week measurement data.

5.1 Congested Segments in Broadband Internet

Figure 7 shows which segments are likely to be bad on the 30 paths of the two ten-week measurements in a direction from Location A to Location B. Since some paths change routes often, we show the result for the most stable routes. In other words, the duration of the combination of the routes shown in Fig. 7 is longer than that of any other combination. Three ISPs are drawn by circles based on routers' domains that are obtained from the whois server. Segments that are likely to be bad are made wide and bold. On every path, we choose bad segments that satisfy the following condition: The number of 15-second periods in which the segment is inferred as being bad is more than 50 15-second periods a day on average.

(1) Location

We first analyze which segments often become bad among three types: ISP access segments, ISP backbone segments and inter ISP segments. Since we cannot identify precisely to which type each segment corresponds, we classify segments into three types as follows. Segments including links that are less than 5 hops away from both ends are regarded

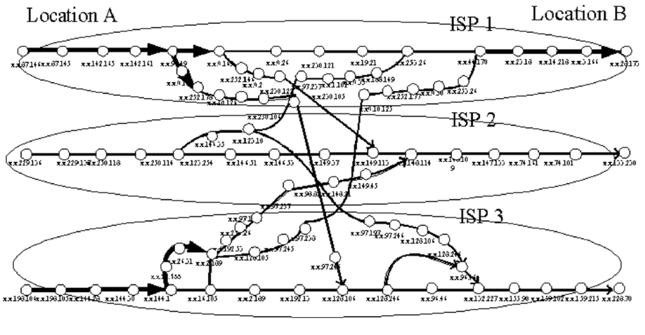


Fig. 7 Congested segments over experiment Internet.

Table 7 Classification of segments.

type of segment	the number of bad periods	the number of segments
ISP access	51574	62
inter ISP	3528	14
ISP backbone	3842	21
total	58944	97

as ISP access segments. Those including links that are one hop inside from an edge of ISP network are regarded as inter ISP segments. The other segments are regarded as ISP backbone segments. As a result, ISP access segments are more likely to be congested than inter ISP and ISP backbone segments. Table 7 shows the number of 15-second periods in which each segment type is inferred as being bad when the routes are most stable. The number of segments classified into each type is also shown in Table 7. There are 97 segments on the most stable network topology.

(2) Direction

Then we analyze the directions of bad links in ISP networks. We call links from an origin to an edge of an ISP network and those from an edge of an ISP network to a destination as up-links and down-links, respectively. As shown in Fig. 7, most bad links in ISP networks are up-links. In other words, down-links are seldom bad. We can extract 40284 periods and 13490 periods in which up-links and down-links are inferred as being bad respectively. In the past, down-links were likely to be congested due to the download WWW (World Wide Web) traffic from WWW servers that are usually located at ISP backbones. On the contrary, it has been currently reported that up-links of ISP networks may be congested due to P2P (Peer 2 Peer) traffic since P2P applications that are located at end points upload many files [11]. We consider that the up-links' congestion may foresee the explosion of P2P application spread.

5.2 Path State Constancy

To determine the constancy of path states, we analyze how the 10%-quantile non-loss rate per day changes day by day. We classify a path state of a day into three states based on a 10%-quantile non-loss rate value. A path is good if the 10%-quantile is more than 0.995. It is medium if the 10%-quantile value is between 0.995 and 0.99. It is bad if the 10%-quantile value is less than 0.99. Most paths' states are stable in days as follows.

- The states of most paths do not change in days for the ten weeks. During Period I, the states of 23 paths are always good or medium and those of 5 paths are always medium or bad. During Period II, the states of 20 paths are always good or medium and 10 paths are always medium or bad. In other words, the states of most paths are stable for the ten weeks.
- We analyze how the worst segment in a day changes in days for the above stable paths. We observe that the worst segment in days rarely changes.

Then we analyze the other two paths during Period I. These two paths change from a bad state to a good one when three days have passed from the beginning of the measurement. During the first three days, we observe more than 250 15-second periods in which the paths are bad per day on average. Then, during the last 66 days, the number of bad periods per day decreases to one to twenty. We guess that the ISP network operators of the bad segments take appropriate action to mitigate the congestion. Since our observation on the constancy of path states and bad segment location indicates that specific segments are likely bad chronically, we believe that the bad segment inference in 15-second periods is useful for ISPs to identify bad segments for which they should take action such as traffic engineering and network management. On the other hand, locating bad segments in real-time manner may also be able to help end users to select better routes. For example, on the overlay network applications, by sharing the information of bad segments with many users, each overlay paths may be able to be set so as not to traverse the congested segments.

6. Discussion

6.1 Measurement Period Length

In order to analyze the validity of measurement period length, in this section, we use one-minute measurement periods instead of 15-second periods and apply our method to the measurement data of the path from ISP1 at Location A to ISP2 at Location B during Period I. We infer bad segments every one-minute by adopting the same thresholds (l = 0.99, h = 0.995). The comparison between the results of 15-second and one-minute periods is as follows:

- There are 720 one-minute periods and 3214 15-second periods of the target path's badness. Bad segments are inferred in 461 one-minute periods and 2057 15-second periods, respectively. The 2057 15-second periods are distributed in 1772 minutes.
- We see the following observations by analyzing the correspondences between the inferred bad segments for one-minute and 15-second periods. We compare bad segments of each (bad) one-minute period and those of the 15-second periods which are included in the one-minute and in which bad segments are inferred. First, in 302 one-minute periods (66% of 461 one-minute periods), the same segments are inferred being bad as in all the included 15-second periods. Second, in 53 one-minute periods (11% of 461 one-minute periods), the same segments are inferred being bad as in one of the included 15-second periods. Third, different segments are inferred being bad in 48 one-minute periods (10% of 461 minute periods).

Then in order to know the reason of the difference, we analyze the packet loss rate changes of the last 48 one-minute periods. We observe that different segments independently become bad in different 15-second periods of

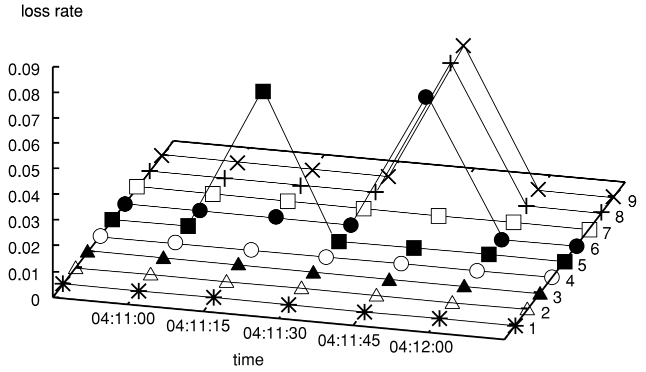


Fig. 8 Example of path non-synchronization.

the same one-minute period. As an example, we show the packet loss rate changes of nine paths of which 4 paths are bad in both one-minute and 15-second measurement period.

In Fig. 8, we observe that four paths become bad at the two 15-second periods at 04:11:15 and 04:11:45 due to the different segments' badness. Although one path's (P_5's) is bad at 04:11:15 and then three paths are bad at 04:11:45, all four paths seem to be bad in the one-minute period. Due to this, wrong segments are inferred being bad in the one-minute period.

As a result, we consider that the shorter the measurement period becomes, the more precise the inference of bad segment will be. However, a shorter measurement period may require a higher test packet rate to accurately measure the packet losses. Thus, in this paper, we experimentally use 15-second periods. As a conclusion, we believe that 15-second periods are adequate for the purpose of locating the bad segments which non-loss rate is lower than 0.995.

6.2 Segments and Test Paths

In our method, how a target path is divided into segments depends on the test paths' properties such as the number and routes. In the measurement of Sect. 4, we use eight test paths half of which have the same origin as the target path, and the other half of which have the same destination. We observe the following results.

- A target path is divided into 6.9 segments on average. A segment consists of 2.0 links on average. As a result, we can identify a bad segment whose average size is 2.0 links.
- Most links in a segment inferred as being bad are located in the same ISP. In this case, we can identify the ISP of a bad segment. On the contrary, most segments that span two ISPs consist of less than 4 links. In this case, the segments may correspond to inter ISP links.

As a result, our method is successful in identifying which ISP or inter ISP segment is bad (congested). Although the result depends on the test paths' routes, the experiment indicates the usefulness of multiple path measurement from multiple origins to multiple destinations. In addition, though our method sometimes infers some candidates of bad segments at one measurement period due to the combination of paths' states, as shown in the bottom row of Table 4 for example, we believe that the information about them is useful for ISP operators to take appropriate action to mitigate the congestion as soon as possible.

6.3 Method Improvement

Based on the analysis results, the proposed method can be improved in two main aspects.

(1) Congestion in Layer 2 Networks

We performed a preliminarily experiment under the same measurement settings using different access network services for a week in November 2003. We subscribed to access services where more users were accommodated on the access links than the current services. In the experiment, we observed synchronized performance degradation (high packet loss rates) on several paths that do not traverse a common segment. Due to this, the proposed method was not able to locate bad segments. After questioning the access service provider, we knew that there were hidden layer 2 switches that are shared (i.e., traversed) by all those paths in access networks. Since traceroute commands did not detect these layer 2 switches, the proposed method did not work well. Therefore, the detection of layer 2 switches that are not detected by traceroute commands is also required so that the proposed method properly obtains the physical topology and precisely locates the bad segments in such layer 2 networks. A similar problem might occur in MPLS or more complicated networks whose physical topologies could not be obtained by traceroute commands. It remains as future work to combine our method with additional information on topology obtained in any other way.

(2) Inference Error based on Route Change

Currently, our method locates bad segments every 15-second period independently, and abandons the inference at a 15-second period if a route change is detected by comparing the result of two traceroutes which interval (i.e., one minute) contains the period Hence, if the related paths are stable in a one-minute period, the bad segments can be inferred based on the topology at that one-minute period. Since periodical route changes in a short interval (e.g., less than one minute) can be regarded as FAILURE and merely occur regardless of the dynamic routing in the Internet, we believe that in many cases our periodical inference (i.e., 15-second periods) works well. One possible problem is that those two traceroutes cannot detect a quick route change where the route changes from an original one to another one, and immediately (within a one-minute period) changes back to the original one. To improve our method for detecting such very quick changes, we should check TTL values of the individual test packets. On the other hand, as explained above, since traceroute cannot detect Layer 2 switches, route changes in Layer 2 network cannot be detected. We may be able to improve our route change detection by measuring the variation of the minimum packet delay.

7. Concluding Remarks

In this paper, we have proposed a practical method to locate congested segments by actively and continually measuring end-to-end packet losses on appropriate paths from multiple origins to multiple destinations, using a network tomographic approach. The proposed method has been verified to be able to precisely locate congested segments through two of ten weeks experiments over the Japanese commercial Internet, whose preliminary results were partially presented at SAINT 2005 in [12].

To improve our method, we are investigating effective use of time series of per-period inference results in order to accurately and reliably characterize the bad segments. We are also developing a scheme to locate and investigate bad segments based not only on packet losses but on packet delay variations.

References

[1] V.N. Padmanabhan, L. Qiu, and H. Wang, "Server-based inference of Internet link lossiness," Proc. IEEE Infocom 2003, San Francisco, USA, 2003.

[2] N. Duffield, "Simple network performance tomography," Proc. ACM SIGCOMM Internet Measurement Conference, Miami, USA, 2003.

[3] M. Coates and R. Nowak, "Network loss inference using unicast end-to-end measurement," Proc. ITC Conference on IP Traffic, Modeling and Management, Monterey, CA, Sept. 2000.

[4] N.G. Duffield, F. Lo Presti, V. Paxson, and D. Towsley, "Inferring link loss using striped unicast probes," Proc. IEEE Infocom 2001, Anchorage, Alaska, April 2001.

[5] T. Bu, N. Duffield, F. Lo Presti, and D. Towsley, "Network tomography on general topology," Proc. ACM SIGMETRICS 2002.

[6] M. Tsuru, T. Takine, and Y. Oie, "Inferring link characteristics from end-to-end path measurements," Proc. IEEE ICC, pp.1534–1538, Helsinki, 2001.

[7] Y. Zhang, V. Paxson, and S. Shenker, "The stationarity of Internet path properties: Routing, loss and throughput," ACIRI Technical Report, May 2000. http://www.aciri.org/vern/papers/stationarity-May00.ps.gz

[8] A. Akella, S. Seshan, and A. Shaikh, "An empirical evaluation of wide-area Internet bottlenecks," Proc. ACM Internet Measurement Conference 2003, 2003.

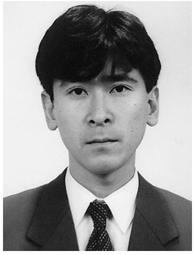
[9] Z. Cataltepe and P. Moghe, "Characterizing nature and location of congestion on the public Internet," Proc. IEEE ISCC 2003, pp.741–746, 2003.

[10] http://www.dabcc.com/docs/pathping.htm

[11] S. Sen and J. Wang, "Analyzing peer-to-peer traffic across large networks," Proc. ACM SIGCOMM Internet Measurement Workshop, 2002.

[12] A. Tachbana, S. Ano, T. Hasegawa, M. Tsuru, and Y. Oie, "Empirical study on locating congested segments over the Internet based on multiple end-to-end path measurements," Proc. IEEE/IPSJ SAINT, 2005.

Atsuo Tachibana received the B.E. and the M.E. degrees in electronic, information systems and energy engineering from Osaka University, Japan in 2000 and 2002, respectively. Since April 2002, he has worked at IP Communication Quality Labs in KDDI R&D Laboratories Inc. His current research interests are IP communication quality measurement and IP network management method.



Shigehiro Ano received the B.E. and the M.E. degrees in electronics and communication engineering from Waseda University, Japan in 1987 and 1989, respectively. Since joining KDD in 1989, he has been engaged in the field of ATM switching system and ATM networking. His current research interests are network management over the next generation Internet, QoS routing architecture and multicast protocol for IP broadcasting. He is currently the Senior Manager of IP Communication Quality Lab. in KDDI R&D Laboratories Inc. He received IPSJ Convention Award in 1995.

Toru Hasegawa received the B.E., the M.E. and Dr. Informatics degrees in information engineering from Kyoto University, Japan, in 1982, 1984 and 2000, respectively. Since joining KDD (now KDDI) in 1984, he has been working in the field of formal description technique (FDT) of communication protocols. From 1990 to 1991, he was a visiting researcher at Columbia University. His current interests are Internet measurement and routing protocols. He is currently the executive director of IP Network Division in KDDI R&D Laboratories Inc. He is also a guest professor at National Institute of Informatics. He received IPSJ Convention Award in 1987 and The Meritorious Award on Radio of ARIB in 2003.

Masato Tsuru received B.E. and M.E. degrees from Kyoto University, Japan in 1983 and 1985, respectively, and then received his D.E. degree from Kyushu Institute of Technology, Japan in 2002. He worked at Oki Electric Industry Co., Ltd., Information Science Center, Nagasaki University, Japan Telecom Information Service Co., Ltd., and Telecommunications Advancement Organization of Japan. Since April 2003, he has been an Associate Professor in the Department of Computer Science and Electronics, Kyushu Institute of Technology. His research interests include performance measurement, modeling and analysis of computer communication networks. He is a member of the IPSJ, JSSST, and ACM.

Yuji Oie received B.E., M.E. and D.E. degrees from Kyoto University, Kyoto, Japan in 1978, 1980 and 1987, respectively. From 1980 to 1983, he worked at Nippon Denso Company Ltd., Kariya. From 1983 to 1990, he was with the Department of Electrical Engineering, Sasebo College of Technology, Sasebo. From 1990 to 1995, he was an Associate Professor in the Department of Computer Science and Electronics, Faculty of Computer Science and Systems Engineering, Kyushu Institute of Technology, Iizuka. From 1995 to 1997, he was a Professor in the Information Technology Center, Nara Institute of Science and Technology. Since April 1997, he has been a Professor in the Department of Computer Science and Electronics, Faculty of Computer Science and Systems Engineering, Kyushu Institute of Technology. His research interests include performance evaluation of computer communication networks, high speed networks, and queueing systems. He is a fellow of the IPSJ and a member of the IEEE.