PAPER *Special Section on Networking Technologies for Overlay Networks*

# QoS-Aware Overlay Routing with Limited Number of Alternative Route Candidates and Its Evaluation*

**Masato UCHIDA**[†], ***Member***, **Satoshi KAMEI**[††,†††], ***Student Member***, **Ryoichi KAWAHARA**[††a)], ***and*** **Takeo ABE**[††], ***Members***

**SUMMARY** A recent trend in routing research is the use of overlay routing to improve end-to-end QoS without changing the network-level architecture. The key of this technology is to find an alternative route that can avoid congested routes, using an overlay network. Developing cost-efficient overlay routing in terms of calculation cost and information distribution cost needed to find an alternative route is important for deploying QoS-aware overlay routing. Thus, this paper evaluates how effective overlay routing can be when the number of alternative route candidates is limited to reduce costs. Evaluation results using actual measurement data indicate that overlay routing is still effective even if alternative route candidates are limited to 1/4 of all possible alternative routes. We also discuss an overlay routing algorithm to enable us to find an appropriate route under the constraint that the number of alternative route candidates is limited.

***key words:*** *overlay routing, QoS, overlay network*

## 1. Introduction

The Internet has developed to accommodate various applications. Recently, the accommodation of real-time applications, such as VoIP (voice over IP) and streaming services, has progressed. These applications are sensitive to QoS (quality of service), so establishing a method of avoiding congestion in the Internet to improve end-to-end QoS has become increasingly important. However, there are several problems in achieving such techniques, as follows.

- The Internet has already become a social infrastructure, so it is difficult to implement new functions that significantly change the existing architecture of the physical network (IP network).
- The Internet is composed of multiple Autonomous Systems (ASs) with different management organizations, so it is difficult to implement new functions on ASs all at once.

Against this background, controlling traffic using an overlay network, which is a logical network constructed over the physical network, is attracting much attention because it enables us to improve end-to-end QoS without changing the physical network. Examples of previous studies on overlay networks to improve end-to-end QoS are SON (Service Overlay Network) [1], OverQoS [2], and QRON (QoS-aware Routing in Overlay Networks) [3]. (We review these studies in more detail in Sect. 2.)

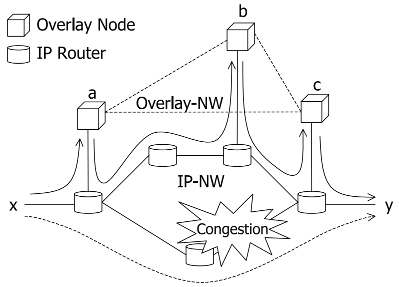


**Fig. 1** Concept of routing control using overlay network.

This paper focuses on routing control using an overlay network to improve end-to-end QoS because the key of QoS control using an overlay network is to find an alternative route that can avoid congested routes. This fundamental concept is illustrated in Fig. 1. In this figure, we assume that the traffic flows from x to y on the route illustrated by the dashed arrow. In addition, we assume that there is a congested router on the route; therefore, the QoS between x and y is degraded. Now, we can say that the congestion can be avoided using the alternative route by transiting overlay node b, i.e., the route illustrated by the solid arrow: x→a→b→c→y), where the alternative route is established using the overlay network composed of overlay nodes a, b, and c.

In [4]–[6], there were reports that there are many alternative routes improving end-to-end QoS based on actual traffic data in the Internet. This means that the distribution of traffic in the Internet is heterogeneous and existing routing controls in the IP networks are not optimal. These reports are important because they demonstrate the potential effectiveness of the routing control using an overlay network. However, in these previous studies, cost-efficient overlay routing controls in terms of calculation cost and information distribution cost needed to find an alternative route were not sufficiently investigated or evaluated, though such an investigation is crucial for developing and deploying QoS-aware

Manuscript received January 11, 2006.
Manuscript revised April 5, 2006.

[†] The author is with Network Design Research Center, Kyushu Institute of Technology, Kitakyushu-shi, 802-0001 Japan.

[††] The authors are with NTT Service Integration Laboratories, NTT Corporation, Musashino-shi, 180-8585 Japan.

[†††] The author is with the Department of Systems Science, Graduate School of Informatics, Kyoto University, Kyoto-shi, 606-8501 Japan.

[*] This paper is an extended version of our paper that was presented at ICOIN 2006.

a) E-mail: kawahara.ryoichi@lab.ntt.co.jp

DOI: 10.1093/ietcom/e89–b.9.2361

overlay routing. (We review these studies in more detail in Sect. 2.)

In this paper, we thus evaluate how effective overlay routing can be when the number of alternative route candidates is limited to reduce costs. First, using actual measurement data, we demonstrate that transit nodes offering optimal alternative routes are distributed non-uniformly. That is, a small number of transit nodes can offer optimal alternative routes to a large number of end-to-end pairs. Then, we evaluate how much end-to-end QoS can be improved when we appropriately limit the number of transit node candidates utilizing the above-mentioned non-uniformity. We show that, even if we limit the number of alternative route candidates, it is possible to improve end-to-end QoS to the same degree as when we can use all alternative route candidates. We also describe an overlay routing algorithm that enables us to find an appropriate route under the constraint that the number of alternative route candidates is limited.

## 2. Related Work

There have been some studies on overlay networks to improve end-to-end QoS. SON [1] assumes that a certain amount of bandwidth or other QoS guarantee is provided by the underlying IP network domain. That is, SON does not treat the case where there is an AS that does not support any QoS guarantee in the end-to-end path. QRON [3] gives a solution to path selection in an overlay network and evaluates its performance through simulation. However, it implicitly assumes that the traffic and the routing in the underlying IP layer are stable, which is unlikely in the actual Internet. In contrast to these studies, we focus on an overlay network that does not assume any QoS guarantee in the IP layer and investigate the performance of overlay routing taking account of actual Internet traffic.

OverQoS [2] provides a method of achieving a smaller packet-loss ratio on a logical link between overlay nodes using FEC (forward error correction) and ARQ (automatic repeat request). Methods of achieving high TCP throughput by dividing a TCP connection into parts at overlay nodes were proposed in [7], [8]. These studies investigated how to improve the performance of a logical link between overlay nodes, rather than the overlay routing on which we are focusing.

In [4]–[6], there were reports that there are many alternative routes that improve end-to-end QoS based on actual traffic data in the Internet. However, Banerjee et al. [5] evaluated the degree of improvement of the end-to-end QoS only when the optimal alternative route can be selected among all possible alternative route candidates. Unlike [5], [6], which only treated QoS such as delay and packet loss, RON [4] showed that the overlay network can detect underlying link or node failure and reroute traffic more quickly than BGP does. However, similarly to [5], RON also discussed optimized performance. Therefore, the evaluation of the improvement of end-to-end QoS when the number of alternative route candidates is limited is insufficient. Such an evaluation is important for implementing and deploying the overlay routing mechanism because the cost of finding the optimal alternative route among candidates increases as the number of candidates increases. On the other hand, Rewaskar et al. [6] evaluated the degree of improvement of end-to-end QoS when the number of alternative route candidates was limited. However, in this evaluation, alternative route candidates were simply randomly selected, and the influence of the way of selecting the candidates on the improvement of QoS was not discussed.

Nakao et al. [9] have proposed that the overlay interacts with the underlying network to reduce the cost in terms of QoS measurements. This kind of approach is effective when the underlying information such as BGP information is available. This method assumes that the delay time between overlay nodes is correlated with the AS hop count and overlay nodes within the same AS have similar QoS. In contrast, the overlay network on which we are focusing does not require any information or assumption about the underlying IP network.

## 3. Analysis and Evaluation of Traffic Data among ISPs

### 3.1 Traffic Data among ISPs

The data used in this paper was measured between 18 host-nodes (overlay nodes) in our related study [10][†].

Each node was connected directly to one of 18 geographically separated Japanese ISPs. We measured the delay time between each possible pair of hosts by executing one ping command per second for three minutes, i.e., 180 packets, every hour. This measurement was performed for two weeks. Six nodes were set up in Tokyo, four nodes were set up in Osaka, four nodes were set up in Sapporo, and four nodes were set up in Kumamoto (see Fig. 2). We analyzed the maximum delay time within each three-minute measurement. The reason we treat maximum delay rather

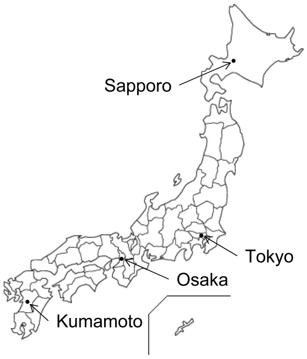


**Fig. 2** Map of Japan.

[†] In [10], we proposed an overlay architecture and discussed its effectiveness when the optimal alternative route can be selected among all possible alternative route candidates. In contrast, this paper focuses on the effect of the limitation of alternative route candidates on the performance.

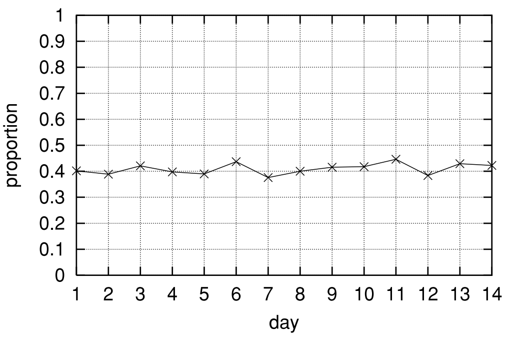


**Fig. 3** Proportion of node-pairs having better routes.

than average or minimum delay is that maximum delay has more impact on the performance of QoS-sensitive applications. (In Sect. 3.6, we also evaluate other QoS metrics such as packet loss ratio and throughput.)

We denote the set of the above 18 nodes as $V = \{v_1, v_2, \dots, v_{18}\}$. In addition, we denote the maximum delay time from source node $v_i$ to destination node $v_j$ $(i \neq j)$ at the $n$-th hour $(n = 1, 2, \dots, N_h)$ as ping_max$(n, v_i, v_j)$ and the route from $v_i$ to $v_j$ as $v_i \rightarrow v_j$.

## 3.2 Non-optimality of Default Route

In this section, we briefly review the results of our related study [10] showing that overlay routing could be effective in the Internet using actual measurement data.

First, we show the proportion of source-destination node pairs each for which there is at least one alternative route that has better end-to-end QoS than the default route at the IP layer. That is, we evaluate the proportion of pairs of source node $v_i$ and destination node $v_j$ $(i \neq j)$ each for which there is at least one $v_k$ $(k \neq i, j)$ satisfying

$$\begin{aligned}&\text{ping\_max}(n, v_i, v_j)\\&\quad > \text{ping\_max}(n, v_i, v_k) + \text{ping\_max}(n, v_k, v_j).\end{aligned} \tag{1}$$

(This value corresponds to $f_0$ defined in Sect. 3.3.) The value of this proportion obtained on each measurement day is shown in Fig. 3. As shown in this figure, the proportion is about 0.4 and this result does not depend on the measurement day.

Then, we also show how much end-to-end QoS improves using the optimal route. Here, let us define the optimal route from $v_i$ to $v_j$ as follows. If there is at least one $v_k$ $(k \neq i, j)$ satisfying Eq. (1), then denote $v_k$ minimizing the right hand side of this equation as $v_{n,i,j}$ and define the optimal route as $v_i \rightarrow v_{n,i,j} \rightarrow v_j$. Otherwise, define the optimal route as $v_i \rightarrow v_j$. The cumulative distributions of maximum delay times of default routes and optimal routes for all pairs of nodes $v_i$ and $v_j$ $(i \neq j)$ in one day are illustrated in Fig. 4. In addition, the correspondence of maximum delay times between those of default and those of optimal routes for all pairs of nodes $v_i$ and $v_j$ $(i \neq j)$ is shown in Fig. 5. As shown in these figures, the maximum delay time of the optimal route is much shorter than that of the default route. Note that this result did not depend on the measurement day.

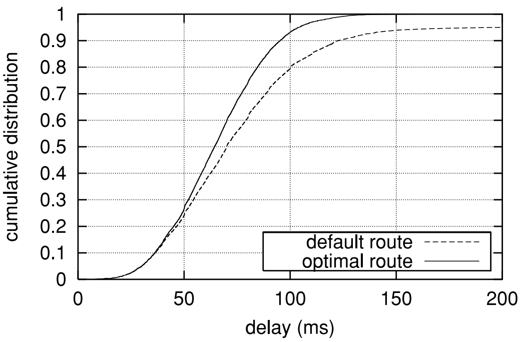


**Fig. 4** Cumulative distribution of maximum delay time.

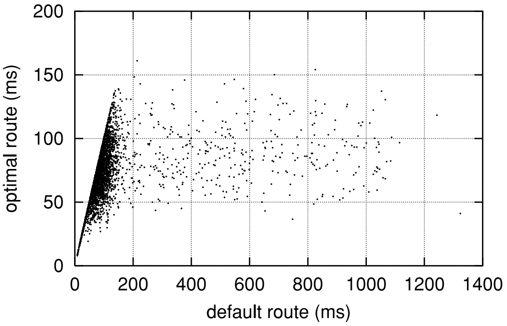


**Fig. 5** Correspondence of maximum delay times between default and optimal routes.

The result in this section indicates that the routing control at the IP layer is not necessarily optimal from the viewpoint of maximum delay time. We also evaluated other QoS metrics such as packet loss ratio and mean delay time in [10].

## 3.3 Non-uniformity of Optimal Transit Node

The results of our related study [10] shown in Sect. 3.2 indicate the potential effectiveness of routing control using an overlay network. In the evaluation of Sect. 3.2, we regarded all nodes, excluding the source node and the destination node, as transit node candidates. This means that we did not take the cost of selecting the alternative route into consideration in Sect. 3.2. This point becomes an important problem when the number of nodes that constitute the overlay network increases.

Now, we consider limiting the number of transit node candidates to solve the problem. That is, we reduce the cost of selecting the alternative route by using a limited number of nodes as transit node candidates. However, there is a possibility that a desirable alternative route, which has a better end-to-end QoS than the default route, cannot be found if we select transit node candidates at random. Therefore, appropriately deciding which nodes should be transit node candidates is necessary.

In this section, we evaluate which node has been selected as the optimal transit node in the evaluation of

Sect. 3.2. Based on the evaluation, we show that the frequency of a node being selected as the optimal transit node (i.e., $v_{n,i,j}$ in Sect. 3.2) is strongly biased.

When $v_k$ ($k \neq i, j$) does not satisfy ping_max$(n, v_i, v_j)$ > ping_max$(n, v_i, v_k)$ + ping_max$(n, v_k, v_j)$, we set $v_{n,i,j} = v_0$. Let us define the frequency $f_k$ of $v_k$ ($k = 0, 1, 2, \cdots, 18$) being selected as the optimal transit node by

$$f_k = \sum_{\substack{n=1,2,\cdots,N \\ i,j=1,2,\cdots,18,\ i\neq j}} \frac{I(v_{n,i,j} = v_k)}{N \times 18 \times 17}, \quad (k \neq i, j), \tag{2}$$

and

$$I(v_{n,i,j} = v_k) = \begin{cases} 1 & \text{if } v_{n,i,j} = v_k \\ 0 & \text{if } v_{n,i,j} \neq v_k \end{cases}, \quad (k \neq i, j). \tag{3}$$

In addition, let us define the arrangement of $f_k$ in descending order of value by $f_{[l]}$ ($l = 0, 1, 2, \cdots, 18$). Then, the cumulative value of $f_{[l]}$ is defined by

$$F_{[l]} = \sum_{x=0}^{l} f_{[x]}, \tag{4}$$

where $F_{[18]} = 1$ holds.

The value of $F_{[l]}$ calculated from the same data as that of Fig. 4 (i.e., $N_h = 24$) is shown in Fig. 6. We find that $F_{[0]}$ is about 0.6. In this case, we find that $F_{[0]} = f_{[0]} = f_0$ holds (see Fig. 3). In addition, we find that $F_{[4]}$ is about 0.9. This indicates that about 90% of the optimal routes (this percentage includes the case when the optimal route is the default route) can be covered even when transit node candidates are limited to the top four nodes with respect to the value of $f_k$. In other words, even when transit node candidates are limited to the top 1/4 (= 4/16) of all possible transit nodes (note that the number of the possible transit nodes excluding the source node and the destination node is 16), 3/4 (= (0.9 − 0.6)/(1.0 − 0.6)) of the optimal routes (excluding the case where the default route is the optimal route) can be covered.

Now, we discuss the above result from another viewpoint. The value of $1 - F_{[l]}$ calculated from the same data as that of Fig. 6 is shown in Fig. 7, where the vertical axis is a logarithm scale. Here, we also added the results for the other days to this figure. We can say that $F_{[l]}$ follows an exponential distribution because the shape of this figure is a straight line. This means that the value of $f_{[l]}$ decreases exponentially as the value of $l$ increases. In other words, a small number of nodes are selected with high frequency as optimal transit nodes, while other nodes are selected with low frequency.

On the other hand, there is an interesting relationship between $f_{[l]}$ and its corresponding geographic position. The relationship is shown in Table 1. The value of $f_{[l]}$ decreases in order of Tokyo, Osaka, Sapporo, and Kumamoto. Although this result seems to reflect the structure of the Internet in Japan, it also seems to depend on the ISPs to which the nodes are connected, so detailed observation about the above relationship is for further study.

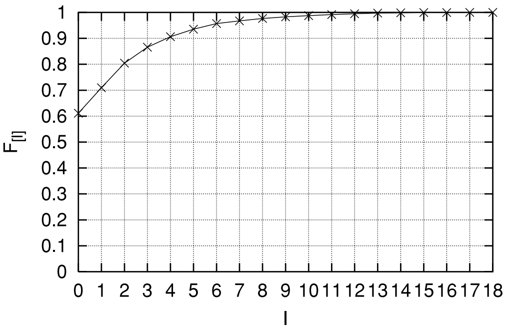


**Fig. 6** $F_{[l]}$ vs. $l$.

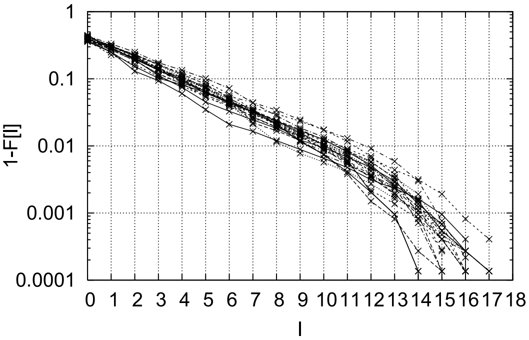


**Fig. 7** $1 - F_{[l]}$ vs. $l$.

**Table 1** Relationship between $f_{[l]}$ and the geographic position of corresponding node.

| Position | Tokyo | Osaka | Sapporo | Kumamoto |
|---|---|---|---|---|
| $l$ | 1,2,3,5,9,10 | 4,6,7,14 | 8,11,13,17 | 12,15,16,18 |

### 3.4 Impact of Transit Node Limitation

The discussion of Sects. 3.2 and 3.3 demonstrates (i) the non-optimality of default routes and (ii) the non-uniformity of optimal transit nodes. Forty percent of optimal routes are not default routes but alternative routes, and 3/4 of such alternative routes are composed of the top four transit nodes with respect to the value of $f_k$. These results indicate that end-to-end QoS can be improved sufficiently even when transit node candidates are limited to the above-mentioned four nodes. This section investigates that in detail. For convenience, let us denote the above-mentioned four nodes as $v_1$, $v_2$, $v_3$, and $v_4$. In the following, we use the same data as that in Sects. 3.2 and 3.3.

The procedure to select routes when transit node candidates are limited to $v_1$, $v_2$, $v_3$, and $v_4$ is given as follows. First, if there is at least one $v_k$ ($k \neq i, j,\ k = 1, 2, 3, 4$) satisfying ping_max$(n, v_i, v_j)$ > ping_max$(n, v_i, v_k)$ + ping_max$(n, v_k, v_j)$, let $v'_{n,i,j}$ denote $v_k$ minimizing the right hand side of the equation and $v_i \rightarrow v'_{n,i,j} \rightarrow v_j$ be the route of this case. Then, if there is no such $v_k$, let the route of this case be $v_i \rightarrow v_j$.

When the selected route is $v_i \rightarrow v_j$, we can categorize its meaning into the following two cases. That is, (i) there is no such $v_k$ satisfying $\mathrm{ping\_max}(n, v_i, v_j) > \mathrm{ping\_max}(n, v_i, v_k) + \mathrm{ping\_max}(n, v_k, v_j)$ for $k = 1, 2, \ldots, 18$ or (ii) there is no such $v_k$ for $k = 1, 2, 3, 4$, but there is for $k = 5, 6, \ldots, 18$. The selected route is $v_i \rightarrow v_j$ for both cases. We set $v'_{n,i,j} = v_0$ for case (i) and $v'_{n,i,j} = v_{19}$ for case (ii). Here, $v'_{n,i,j} = v_0$ means that there are no alternative routes that have better end-to-end QoS than the default route, regardless of the presence of the limitation in the number of transit node candidates. In contrast, $v'_{n,i,j} = v_{19}$ means that there are alternative routes that have better end-to-end QoS than the default route, but the alternative routes cannot be used due to the limitation in the number of transit node candidates, and as a result, the default route is selected. Note that the route selected using the above-mentioned route selection procedure is not necessarily the optimal route. This point is different from the procedure used in Sects. 3.2 and 3.3 because if $v_{n,i,j} = v_k$ $(k \neq 0, 1, 2, 3, 4)$ then $v_{n,i,j} \neq v'_{n,i,j}$, while if $v_{n,i,j} = v_k$ $(k = 0, 1, 2, 3, 4)$ then $v_{n,i,j} = v'_{n,i,j}$. Therefore, we can say that the selected route when the number of transit node candidates is limited is a suboptimal route.

As shown above, we cannot establish the optimal route using $v_5, v_6, \ldots, v_{18}$ as transit nodes when the transit node candidates are limited to $v_1$, $v_2$, $v_3$, and $v_4$. To evaluate the impact of the limitation, let us define the frequency of selecting $v_k$ as the suboptimal transit node by

$$f'_k = \sum_{\substack{n=1,2,\cdots,N\\ i,j=1,2,\cdots,18,\ i\neq j}} \frac{I(v'_{n,i,j} = v_k)}{N \times 18 \times 17},$$
$$(k \neq i, j,\ k = 0, 1, 2, 3, 4, 19). \tag{5}$$

The value of $f'_{19}$ is important for measuring the impact of the limitation in the number of transit nodes. This is because the value of $f'_{19}$ indicates the frequency that there exist alternative routes providing better end-to-end QoS than the default route, but the alternative route cannot be used when the number of transit node candidates is limited, and, as a result, the default route is selected. Using the same data used in Sects. 3.2 and 3.3, the value of $f'_{19}$ was about 0.05. This value seems to be sufficiently small, so we expect that the effect of limiting the number of transit node candidates will be negligible. In the following, we confirm this conjecture.

The cumulative distributions of the maximum delay times of the default routes, optimal routes, (these two lines are the same as those in Fig. 4) and suboptimal routes that were selected using the above-mentioned procedure are illustrated in Fig. 8. For reference, we also show the results when transit node candidates were selected at random ("random" in this figure), where there were four transit node candidates, similarly to "suboptimal." As shown in this figure, suboptimal routes enable us to obtain almost the same performance as that of the optimal routes[†], while the performance of the random routes is much worse than that of optimal routes and suboptimal routes.

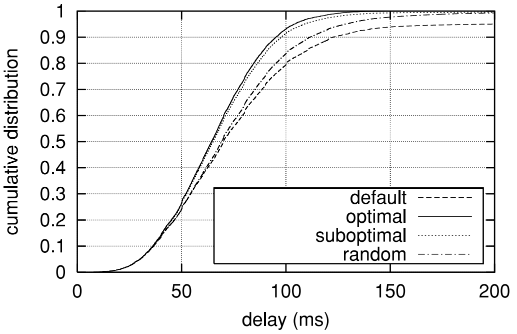


**Fig. 8** Cumulative distribution of maximum delay time (superimposed on Fig. 4).

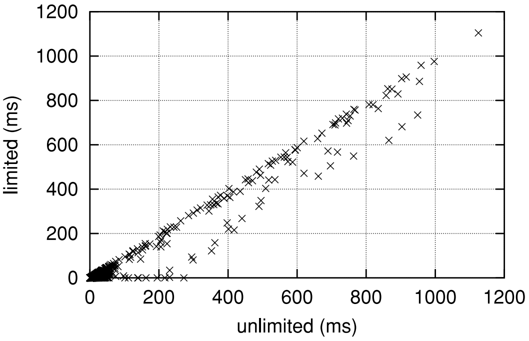


**Fig. 9** Comparison of reduction in maximum delay time (horizontal axis: without limitation, vertical axis: with limitation).

Here, we look at the above result in detail. When the number of transit node candidates is limited, the maximum delay time may not be improved sufficiently if the optimal route from $v_i$ to $v_j$ is the route using $v_5, v_6, \ldots, v_{18}$ as a transit node. To investigate this, we compared the degree of improvement in terms of maximum delay time when the number of transit node candidates is limited with that when unlimited. Figure 9 shows the results. The x-coordinate and y-coordinate of each point in this figure are $\{\mathrm{ping\_max}(n, v_i, v_j) - (\mathrm{ping\_max}(n, v_i, v_{n,i,j}) + \mathrm{ping\_max}(n, v_{n,i,j}, v_j))\}$ and $\{\mathrm{ping\_max}(n, v_i, v_j) - (\mathrm{ping\_max}(n, v_i, v'_{n,i,j}) + \mathrm{ping\_max}(n, v'_{n,i,j}, v_j))\}$ for each pair of nodes $v_i$ and $v_j$ satisfying $v_{n,i,j} = v_k$ $(k = 5, 6, \cdots, 18,\ k \neq i, j)$, respectively. That is, the x-coordinate and y-coordinate of each point indicate the reduction in maximum delay time without and with limiting the number of transit node candidates, respectively. Here, we set the y-coordinate to 0 when $v'_{n,i,j} = v_{19}$. As shown in this figure, many points are on a straight line with a slope of 1 and a $y$-intercept of 0. This indicates that a similar performance to that of the optimal routes can be achieved by using the suboptimal routes. This result also indicates that the number of transit node candidates used in this evaluation is appropriately limited.

Finally, to investigate the sensitivity of the number of transit node candidates $M$ to the performance, we evaluated the 95-percentile of the maximum delay times of all node-pairs when we calculated suboptimal routes using the top-$M$ transit nodes with respect to $f_k$ in Sect. 3.3. The results when we changed $M$ are shown in Fig. 10. This graph shows

[†] In Appendix, a similar evaluation is executed for cases where overlay nodes are locally allocated.

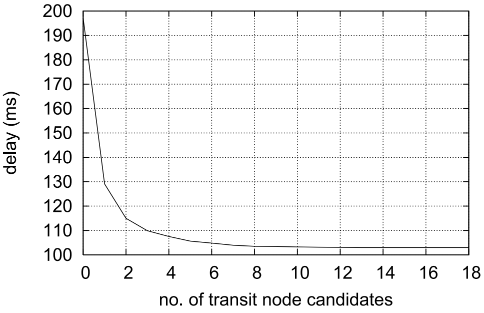


**Fig. 10** The 95-percentile of maximum delay times vs. no. of transit node candidates.

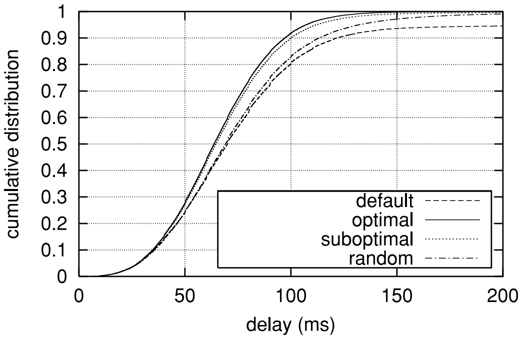


**Fig. 11** Cumulative distribution of maximum delay time (case of taking impact of daily variation into consideration).

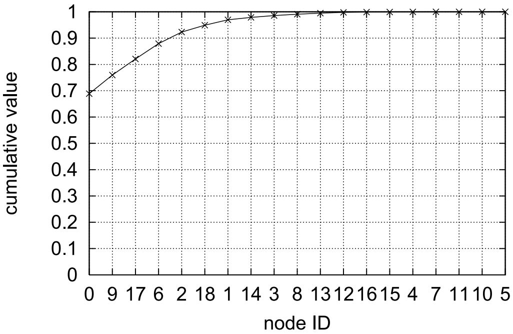


**Fig. 12** Frequency of being selected as optimal transit node in terms of packet loss ratio.

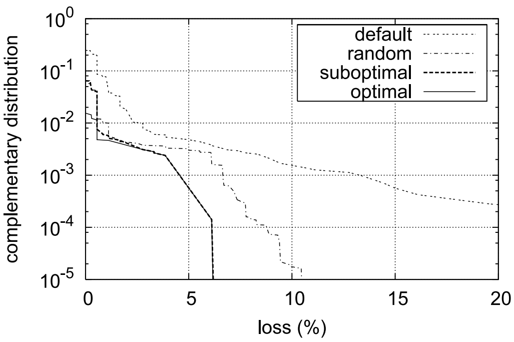


**Fig. 13** Complementary distribution of packet loss ratio.

that the degree of improvement in terms of delay times increases markedly as $M$ increases until $M$ reaches 4, while it becomes insensitive to $M$ when $M$ exceeds 4. Here, $M = 4$ corresponds to the minimum value satisfying $F_{[M]} > 0.9$ where $F_{[l]}$ is defined by Eq. (4), as explained in Sect. 3.3. This result also indicates that it is effective to limit the number of transit node candidates $M$ so that $M$ is equal to the minimum value satisfying $F_{[M]} > 0.9$.

### 3.5 Impact of Daily Variation

In Sect. 3.4, the data used to select optimal/suboptimal transit nodes was the same as that used to evaluate the performance of the selected transit nodes. Thus, we investigated whether the selected transit nodes are still effective for data on other days. The result of a similar evaluation using data taken for 2 weeks, including the day whose data was used in Sect. 3.4, is illustrated in Fig. 11. Here, transit node candidates were selected using the data of the first measurement day. Comparing Figs. 8 and 11, we find that the characteristics of both figures are almost the same. This means that the appropriate transit node candidates did not vary in time, at least during this two-week period.

### 3.6 Evaluation of Other QoS Metric

We also executed the same evaluation in terms of packet loss ratio as that in terms of delay time in Sects. 3.3 and 3.4. The results are shown in Figs. 12 and 13, which correspond to Figs. 6 and 8, respectively. To execute this evaluation, we used the packet loss ratio from node $v_i$ to $v_j$ at the $n$-th hour, which is denoted as $\text{ping\_loss}(n, v_i, v_j)$. Instead of Eq. (1), we used the following equation to find the optimal transit node $v_k$:

$$\text{ping\_loss}(n, v_i, v_j) > 1 - (1 - \text{ping\_loss}(n, v_i, v_k)) \times (1 - \text{ping\_loss}(n, v_k, v_j)). \quad (6)$$

Node ID, represented by the x-axis of Fig. 12, indicates the rank of nodes with respect to $f_k$ in terms of delay time, i.e., $l$ in Table 1. It is interesting to see that the top-$M$ nodes in terms of delay time are not always the top-$M$ nodes in terms of packet loss ratio. As shown in Figs. 12 and 13, similarly to the case of delay time, there is also a non-uniformity of optimal nodes in terms of packet loss ratio. That is, a small number of nodes are selected with high frequency as optimal transit nodes, while other nodes are selected with low frequency. In addition, the top-4 nodes also achieve almost the same performance as that of optimal routes in the case of packet loss ratio, which is similar to the case of delay time.

We further evaluated the case of throughput. Here, we consider TCP throughput ($Th$), which is estimated as follows [11]:

$$Th = \min\left(\frac{W}{\text{ping\_avg}}, \frac{MSS}{\text{ping\_avg} \times \sqrt{2/3\ \text{ping\_loss}}}\right), \quad (7)$$

where $W$ is the TCP window size and set to 64 KB, $MSS$ is the maximum segment size and set to 1460 bytes, ping_avg

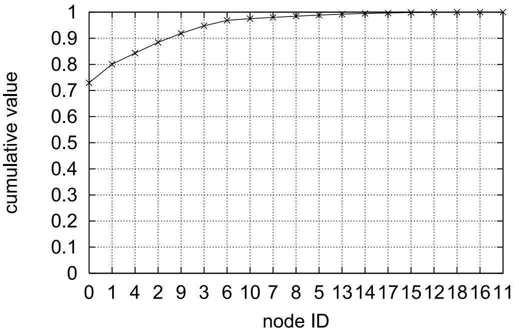


**Fig. 14** Frequency of being selected as optimal transit node in terms of throughput.

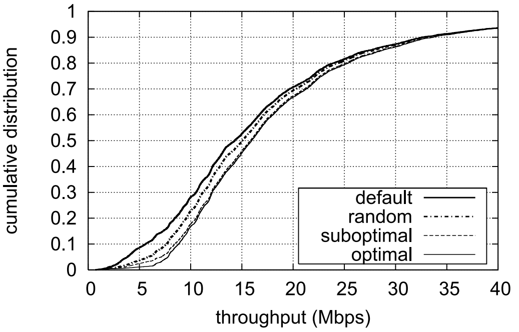


**Fig. 15** Cumulative distribution in terms of throughput.

is the average round-trip delay, and ping_loss is the packet loss ratio. The results are shown in Figs. 14 and 15, from which we found that there is also a non-uniformity of optimal nodes in terms of throughput and the top-$M$ nodes achieve almost the same performance as that of optimal routes.

## 4. Routing Algorithm Using Non-uniformity of Optimal Transit Nodes

In this section, we develop a cost-efficient routing algorithm based on the results shown in Sect. 3. The goal of the algorithm is to find optimal routes under the constraint that the number of transit node candidates is limited. In Sect. 3, we determined the top-$M$ optimal transit nodes by examining all the possible transit node candidates. In contrast, the algorithm described in this section learns the optimal top-$M$ nodes through the process of finding better routes under the constraint that the number of candidates is limited.

### 4.1 Procedure of Determining Route

A routing algorithm depends on how an overlay network is constructed and which mechanisms are allocated to the network. Thus, as an example, we assume the following mechanisms in the overlay network: (i) There are $N$ overlay nodes, e.g., $N = 18$ in Sect. 3, each of which behaves as both source and destination nodes. In addition, each node can also behave as a transit node if required, that is, each node has the capability of rerouting traffic from a source to a destination node. (ii) Each source node updates the routes to all other destination nodes periodically. (iii) In the network, there is a transit-node manager (TM) that maintains the transit-node-score list (TL), which indicates the possibility of overlay node $v_k$ being optimal. Here, we denote the score of $v_k$ at time $n$ as $C(n, v_k)$. First, we initialize $C(0, v_k) = 1$ for all $k$. Under such an overlay network, the proposed procedure of establishing the route is as follows.

Step 1) Each overlay node measures QoS, e.g., delay time, from itself to all other nodes at every predetermined period $\tau$, e.g., $\tau = 1$ hour in Sect. 3. Hereafter, we denote the measured QoS from source node $v_i$ to destination node $v_j$ at the $n$-th period as $d(n, v_i, v_j)$, which corresponds to "ping_max$(n, v_i, v_j)$" in Sect. 3.

Step 2) After that, overlay source node $v_i$ receives TL from TM to choose $M$ transit node candidates, e.g., $M = 4$ in Sect. 3. At this time, source node $v_i$ chooses node $v_k$ with probability $p_k = C(n, v_k) / \sum_{k \neq i} C(n, v_k)$. This procedure is repeated until $M - M'$ nodes have been chosen. In addition, node $v_i$ chooses $M'$ nodes among the remaining nodes with equal probability (regardless of the scores).

Step 3) Then, source node $v_i$ requires each transit node candidate $v_k$ to send $d(n, v_k, v_j)$.

Step 4) Source node $v_i$ determines the route to destination node $v_j$ using $d(n, v_i, v_k)$ and $d(n, v_k, v_j)$. If there is at least one $v_k$ satisfying $d(n, v_i, v_j) > d(n, v_i, v_k) + d(n, v_k, v_j)$, then node $v_i$ selects node $v_k$ minimizing the right-hand side of this equation as transit node $v_{k^*}$ from $v_i$ to $v_j$. Then, source node $v_i$ requires $v_{k^*}$ to set up the route from $v_i$ to $v_j$ via $v_{k^*}$. Otherwise, $v_i$ uses the default route to $v_j$. This procedure is executed for all $v_j$ $(j \neq i)$.

Step 5) After setting the route from $v_i$ to $v_j$ via $v_{k^*}$, transit node $v_{k^*}$ requires TM to update the score of $v_{k^*}$ in TL.

Step 6) TM counts up the point of $v_{k^*}$, $\tilde{C}(n, v_{k^*})$, as $\tilde{C}(n, v_{k^*}) \leftarrow \tilde{C}(n, v_{k^*}) + 1$ every time when it receives the request of updating the score from $v_{k^*}$. Here, $\tilde{C}(n, v_k)$ denotes the number of times that $v_k$ was selected as the transit node at $n$-th period, which is used to update score $C(n, v_k)$. When the next period comes (i.e., at $(n + 1)$-th period), TM updates score $C(n + 1, v_k)$ for all $k$ as follows:

$$C(n+1, v_k) = \max\left\{C_l, \left(1 - \frac{1}{t'}\right) C(n, v_k) + \frac{1}{t'} \tilde{C}(n, v_k)\right\}, \quad (8)$$

$$t' = \min(n + \beta, c), \quad (9)$$

where $c$ and $\beta$ are the predetermined parameters (e.g., $c = 4$ and $\beta = 0.5$), and $C_l$ is the predetermined lower bound for the score (e.g., $C_l = 0.1$) needed to avoid the score of a node being excessively decreased. After that, we reset $\tilde{C}(n, v_k) = 0$ for all $k$.

In the above procedure, we need to choose appropriate transit node candidates in Step 2. To achieve this, the process of updating the score in Step 6 is very important. Now, we explain about Eq. (8). The second term in Eq. (8) is used to increase the score according to the number of times of being selected as the transit node. By doing this, we expect that optimal transit nodes can get a large score. On the other hand, the first term in Eq. (8) is needed to decrease the score of a node that has been deselected as the transit node since the last time the node was selected. Now, we explain why we need to choose $M'$ nodes independently of the current score of each node in Step 2. This is because we need to give a chance of a low-score-node being chosen to cope with changes in traffic/QoS in the underlying IP layer. For example, there is some possiblity that the QoS in an IP network where a low-score-node is allocated will be improved by allocating more link-bandwidth capacity in the network. We expect that choosing nodes independently of the scores will enable us to cope with such situations. In Sect. 4.3, we evaluate the performance of our method when traffic conditions change.

As an alternative to Eq. (9), $t' = c$ may be straightforward. The reason we take the minimum of $n$ and $c$ is that we would like to use the current point $\tilde{C}(n, v_k)$, aggressively in the initial phase of updating in Eq. (8), i.e., when $n$ is small, to achieve quick convergence[†].

### 4.2 Performance Evaluation

We evaluate the performance of the proposed routing algorithm in Sect. 4.1, using the same data as that in Sect. 3. Here, we set the number of transit node candidates, $M$, to four and $M'$ to one. The evaluation result of the algorithm is shown by "learned" in Fig. 16, which is the same graph as that in Fig. 8 except "learned." As shown in this graph, the proposed algorithm achieves almost the same good performance as that of "suboptimal," which is the result when we used the top 4 nodes obtained in Sect. 3. For reference, we also show the score of each node at $n$ = 24-th hour and the transient behavior of score $C(n, v_k)$ in Figs. 17 and 18, respectively. In these graphs, node id $i$, i.e., the x-axis in Fig. 17 and graph-legend in Fig. 18, correspond to the ranking evaluated in Sect. 3.3. From these graphs, we see that the algorithm adequately learns the appropriate transit nodes. We also show the transient behavior of the 95-percentile of the maximum delay times of all node-pairs at the $n$-th hour (see Fig. 19). We found that the learning algorithm can adequately make the delay converge to that of "optimal" or "suboptimal."

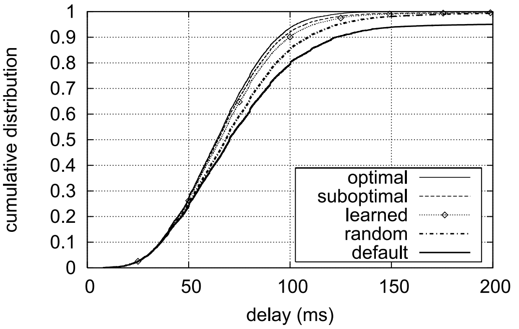


**Fig. 16** Results of proposed algorithm.

We also evaluated the performance of our algorithm in terms of packet loss and throughput; see Figs. 20 and 21. We confirmed that our algorithm is effective for both QoS metrics.

We also confirmed the effectiveness of our method using other network data. Here, we used packet trace data provided by PlanetLab, which serves as a large-scale testbed for

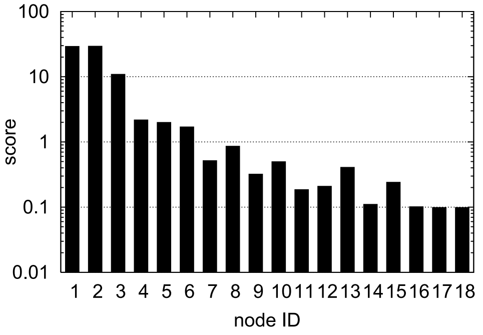


**Fig. 17** Score of each node.

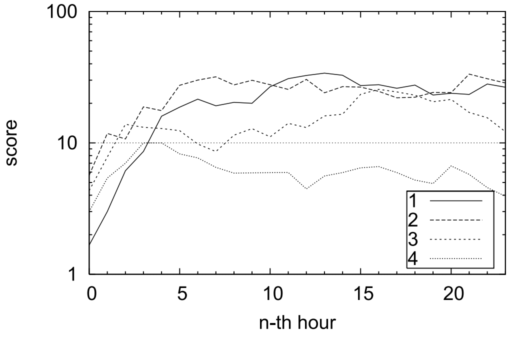


**Fig. 18** Transient behavior of scores.

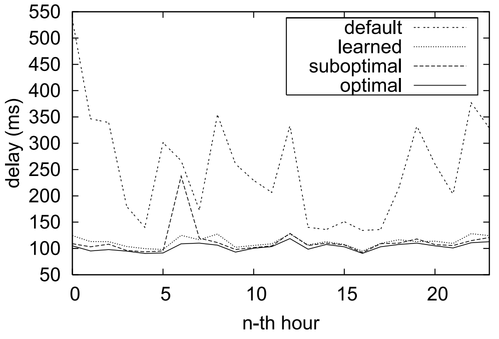


**Fig. 19** Transient behavior of delay time.

[†] The simplest way of aggressively using the current point is to set $t' = 1$. However, setting $t' = 1$ may cause inappropriate transit nodes to continue to be chosen. This is because if an inappropriate transit node is chosen and a point is assigned to that node, the score of the node increases (and never decreases), so the probability of the node being chosen in the next period also increases. To avoid such a negative spiral, we need to set $t' > 1$.

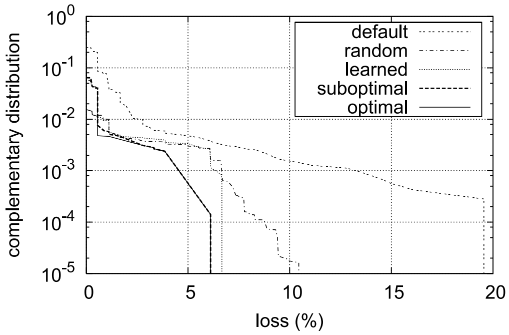


**Fig. 20** Results in terms of packet loss ratio.

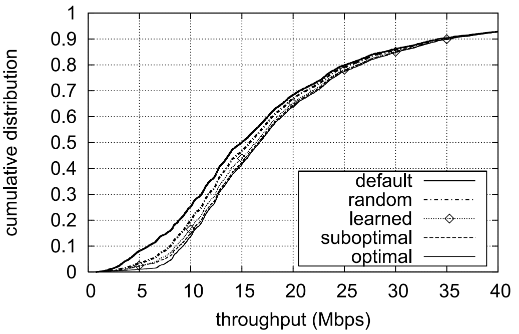


**Fig. 21** Results in terms of throughput.

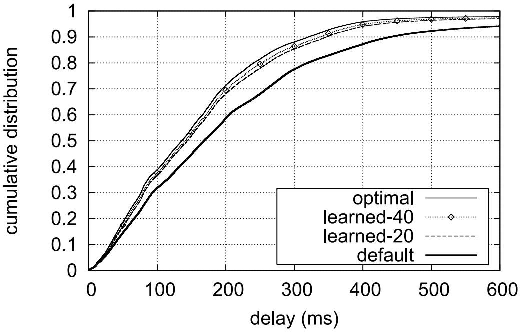


**Fig. 22** Evaluation using PlanetLab data.

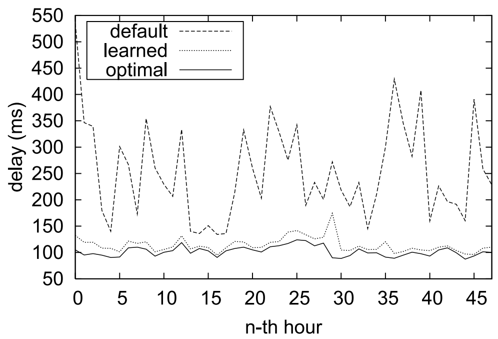


**Fig. 23** Behavior of delay time when traffc changes.

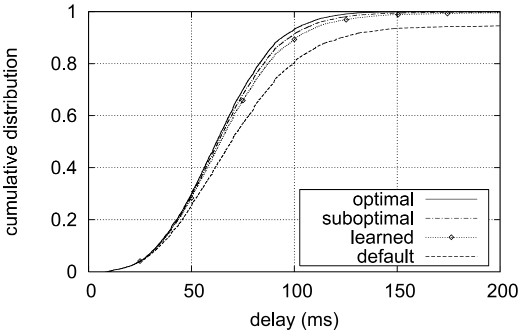


**Fig. 24** Distribution of delay times when traffc changes.

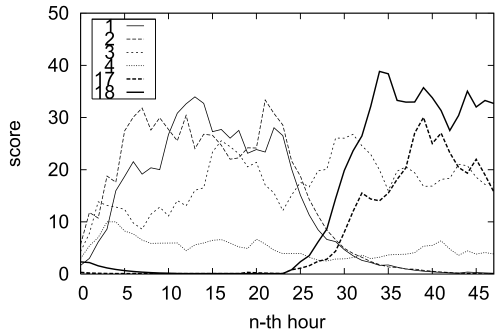


**Fig. 25** Behavior of scores when traffc changes.

world-wide overlay networks [12]. The data we used was maximum delay times between individual possible pairs of nodes in every 15 minutes for 7 hours on Novemer 2005 [13], where the number of overlay nodes was 496. The evaluation results are shown in Fig. 22. Here, we set the number of transit node candidates $(M, M')$ to $(20, 2)$ ("learned-20" in this figure) or $(40, 4)$ ("learned-40"), which correspond to 5% or 10% of all nodes, respectively. This graph shows that our method also works well for other network data.

### 4.3 Ability to Handle Changes in Traffic Patterns

In this section, we evaluate whether our method can cope with various patterns of traffic changes. Here, we consider the following case: At time $n = 24$-th hour, the delay times of nodes $v_1$ and $v_2$ become degraded while those of nodes $v_{17}$ and $v_{18}$ become improved, where $v_i$ denotes a node whose rank is $i$ with respect to $f_k$ in Sect. 3.3. To simulate this situation, we exchanged $d(n, v_1, v_j)$ with $d(n, v_{18}, v_j)$ when $n \geq 24$, where $d(n, v_i, v_j)$ corresponds to "ping_max$(n, v_i, v_j)$" in Sect. 3. (We also exchanged $d(n, v_i, v_1)$ with $d(n, v_i, v_{18})$.) We also exchanged $d(n, v_2, v_j)$ with $d(n, v_{17}, v_j)$ (and also $d(n, v_i, v_2)$ with $d(n, v_i, v_{17})$). Figures 23 and 24 show the transient behavior of the 95-percentile of the maximum delay times of all node-pairs at the $n$-th hour and the cumulative distribution of delay times between $n = 24$ and 48-th hours, respectively. We also show the behaviors of scores of $v_1$, $v_2$, $v_3$, $v_4$, $v_{17}$, and $v_{18}$ in Fig. 25. These graphs show that our method can achieve small delay times even after the traffic changes by adequately increasing the scores of new high-QoS nodes (i.e., $v_{17}$, and $v_{18}$).

We also evaluated other conditions shown in Table 2.

**Table 2** Patterns of traffic changes.

| | | | |
|---|---|---|---|
| (1) | global change | | top-2 ⇔ lowest-2 nodes |
| (2) | local | (2-1) | top-1 degraded |
| | change | (2-2) | lowest-1 improved |
| (3) | node | (3-1) | top-1 joins |
| | join | (3-2) | lowest-1 joins |

Here, pattern (1) corresponds to the above evaluation. In patten (2-1), we simulated the increased delay times of $v_1$ by multiplying the actual delay time by five when $n \geq 24$. That is, the simulated delay from $v_1$ to $v_j$ was set to "$5 \times d(n, v_1, v_j)$" (and we did the same thing to the delay from $v_i$ to $v_1$). In pattern (2-2), we simulated the decreased delay time of $v_{18}$ by dividing the actual delay time of $v_{18}$ by four. In pattern (3-1), we joined node $v_1$ at time $n = 24$-th hour while we joined node $v_{18}$ in pattern (3-2). The results for all patterns are shown in Fig. 26. These graphs show that our method can cope with any pattern of traffic changes in Table 2.

### 4.4 Case of Access Link Being Bottlenecked

In this section, we consider the case where the overlay routing could cause congestion. If a large amount of traffic rerouted by the overlay rouing concentrates on a particular link, the link may become bottlenecked, which causes excessive queueing delay at the link. Specifically, the access link of a transit overlay node could be bottlenecked if the link speed is not high. Therefore, the overlay routing algorithm needs to work well so that it can avoid such traffic concentration on a particular link. We thus evaluated whether our method can work well even in the above situation. To do this, we simulated the following case: we assume that there are two types of nodes: one is a node whose access link is optical fiber with 100 Mbps, and the other is a node whose access link is ADSL with 1.5 Mbps in the downstream direction and 512 kbps in the upstream direction. We modeled the queueing delay $Q_d$ at the access link of a node as

$$Q_d(N_q, C) = \text{p\_size}/C \times f(N_q, C), \qquad (10)$$

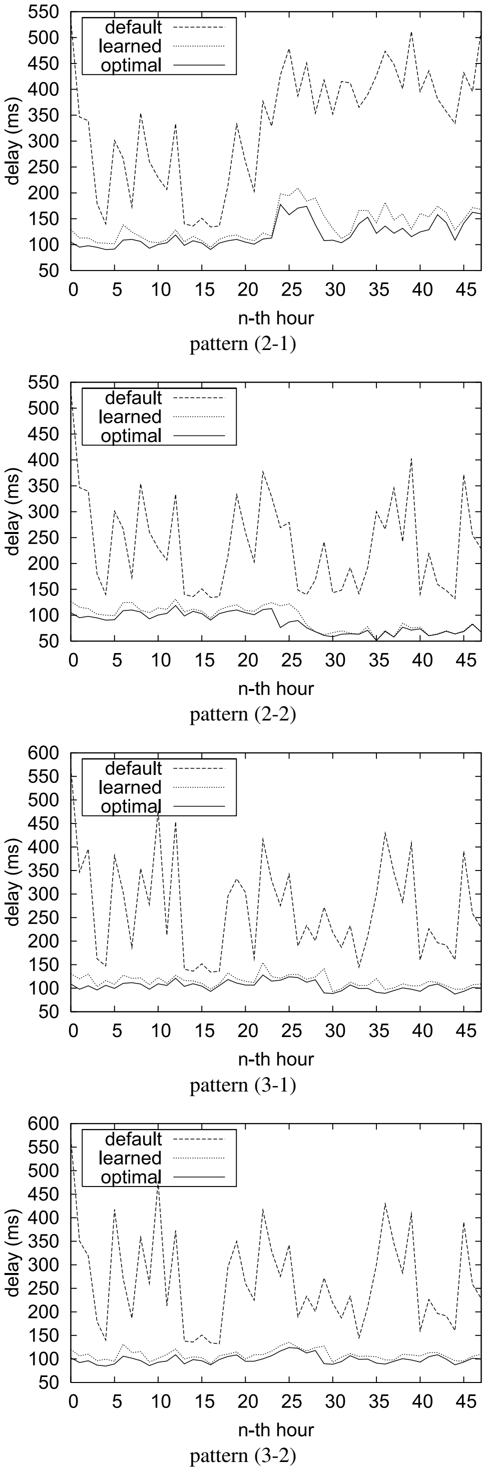
where $N_q$ is the number of node-pairs whose traffic is carried by the node, $C$ [bps] is the access link speed of the node, p_size is the average size of packets transmitted by the node, and $f(N_q, C)$ is an increasing function with $N_q$, which is given by

$$f(N_q, C) = \begin{cases} \dfrac{1}{1 - a \times N_q/C} & \text{if } a \times N_q < C \\ \infty & \text{otherwise.} \end{cases} \qquad (11)$$

(This is based on the M/M/1 model.) Here, $a$ [bps] indicates the traffic caused by one pair of nodes. By adding the queueing delays determined by Eq. (10) with the network delay $d(n, v_i, v_j)$ (which corresponds to the measured data in Sect. 3), we simulated the end-to-end delay between $v_i$ and $v_j$ including access link queueing delays. For example, if we simulate the end-to-end delay from $v_i$ to $v_j$ via $v_k$, we add the following queueing delays with network delays (i.e., $d(n, v_i, v_k)$ and $d(n, v_k, v_j)$):

$$Q_d(N_q(v_i), C_u(v_i)) + Q_d(N_q(v_k), C_d(v_k)) + Q_d(N_q(v_k), C_u(v_k)) + Q_d(N_q(v_j), C_d(v_j)). \qquad (12)$$

Here, $C_u(v_x)$ and $C_d(v_x)$ are the access link speeds of node $v_x$ in the upstream and downstream directions, respectively,



pattern (2-1)

pattern (2-2)

pattern (3-1)

pattern (3-2)

**Fig. 26** Behavior of delay time when traffc changes.

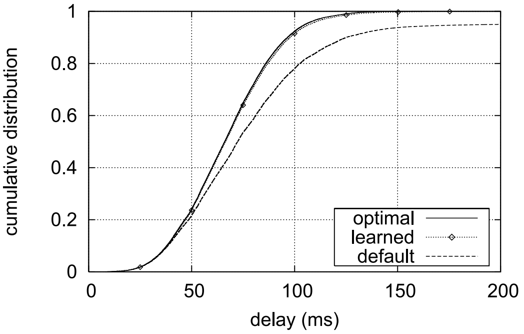


**Fig. 27** Distribution of delay times when access link is bottlenecked.

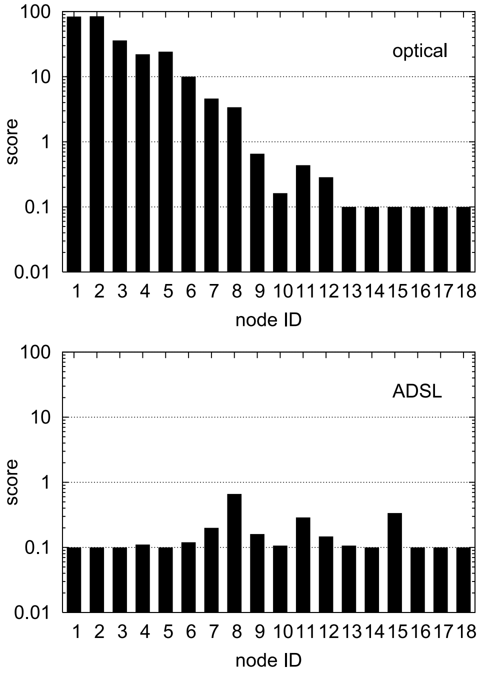


**Fig. 28** Node scores when access link is bottlenecked.

and $N_q(v_x)$ is the number of pairs whose traffic is carried by node $v_x$ at the current time. We set $C_u = C_d = 100$ Mbps for a node with an optical-link while we set $C_u = 512$ kbps and $C_d = 1.5$ Mbps for a node with an ADSL link. And we set p_size= 200 byte and $a = 80$ kbps (assuming VoIP communication between each pair of nodes). We assume that there are two nodes in each of 18 ISPs: one is a node whose access link is optical fiber, and the other is a node whose access link is ADSL. That is, the total number of nodes is $18 \times 2 = 36$ nodes. As for parameters in our method, we set $(M, M') = (8, 1)$ and the other parameters in our method to the same as those in Sect. 4.2. Figures 27 and 28 show the cumulative distribution of delay times for 24 hours and score of each node at $n = 24$-th hour, respectively. From these graphs, we found that our method can achieve almost the same performance as "optimal" by adequately selecting top nodes with optical fiber and not selecting nodes with ADSL, which makes it possible to avoid concentration of traffic on the low-speed link of ADSL.

**Table 3** Cost comparison.

| cost | | unlimited | limited |
|---|---|---|---|
| (a) | measurement | $N-1$ | $N-1$ |
| (b) | distribution | $(N-1)(N-2)$ | $M(N-2)$ |
| (c) | computation | $(N-1)(N-2)$ | $M(N-2)$ |

## 4.5 Discussion of Cost

In this section, we investigate the costs of the routing algorithm in Sect. 4.1. Here, we investigate three types of costs: measurement cost, information distribution cost, and route calculation cost. The summary of the cost evaluation is shown in Table 3. For comparison, we also show costs when we do not limit the number of node candidates ("unlimited" in this table).

Cost (a) in this table is the measurement cost per node, which indicates the number of destination nodes to be measured by a source node at every period. This value of "limited" is the same as that of "unlimited" because we need to grasp the QoS of the default route to each destination node, regardless of the limitation.

Cost (b) is the information distribution cost per node, which indicates the number of measured QoS results between transit and destination nodes that need to be sent to a source node. This value is the same as the number of possible alternative routes to individual destination nodes per source node. First, we consider the case of "unlimited." For each source node, the number of possible routes is $(N-1)$, i.e., the number of destination nodes, multiplied by $(N-2)$, i.e., the number of transit node candidates. On the other hand, in the case of "limited," the number of possible routes to the destination nodes each of which is a transit node candidate is $M(M-1)$. On the other hand, the number of possible routes to the destination nodes none of which are transit node candidates is $M$, i.e., the number of transit node candidates, multiplied by $(N-1-M)$, i.e., the number of destination nodes that do not belong to the group of transit node candidates. Therefore, for a source node, the total number of possible routes, which corresponds to the number of measured QoS results to be sent to the source node, is $M(M-1) + M(N-1-M) = M(N-2)$.

Cost (c) is the route calculation cost, which indicates the number of possible alternative routes to individual destination nodes per source node. That is, it is the same as the number in Cost (b).

Summarizing the above discussion, we conclude that we can reduce information distribution and route calculation costs by only limiting the number of transit node candidates. This cost reduction becomes significant when the number of overlay nodes $N$ increases. In this cost evaluation, we assumed that each source or destination overlay node needs to be a transit node if required. However, we can consider other network models such as a hierarchical model where transit nodes dedicated to routing are allocated. In addition, a technique of grouping some nodes may be useful to reduce costs further. For example, Krishnamurthy and Wang [14] utilized the underlying IP layer information, i.e., BGP information, to cluster the nodes. Similar to [9], this method assumes that the delay time between overlay nodes is correlated with the AS hop count and that overlay nodes within the same AS have similar QoS. In contrast, the overlay network on which we are focusing does not require any information or assumption about the underlying IP network. Under the same assumption as in our model, Zhang et al. [15] investigated how to group nodes so that the distances between individual nodes in the same group are close, where the distance means the propagation delay. Eugene Ng et al. [16] also treated the problem of predicting network distance with the coordinate-based approach. In contrast, we treat maximum delay and packet loss rather than propagation delay because they have more impact on the performance of QoS-sensitive applications. In our ongoing study [17], we are investigating how to reduce costs further by clustering nodes taking account of such QoS metrics, which is for further study.

### 4.6 Future Work

The algorithm in Sect. 4.1 uses the non-uniformity of optimal transit nodes, which makes it possible to achieve good performance even when we limit the number of transit node candidates. However, this may also cause processing loads required for transit nodes to be non-uniform. Therefore, to avoid the concentration of excessive load on a particular node, some additional mechanisms are required. For example, if CPU utilization of a transit node that is chosen in Step 2 in Sect. 4.1 exceeds a threshold, we need to choose another transit node. From Fig. 12, we found that optimal nodes in terms of delay time are different from those in terms of packet loss ratio. This implies that we can avoid load concentration on a particular node if we distinguish the traffic according to the required QoS metric. In addition to this kind of control mechanism, which makes use of current resources, we also need to consider how to design and extend the overlay network taking account of such non-uniformity of optimal nodes. For example, it may be a good idea to allocate new transit nodes close to existing optimal nodes.

Another issue to be considered is that we need to modify the algorithm so that it can cope with significant changes in traffic and/or routing at the underlying IP layer. If optimal nodes also change due to such changes, a lot of time may be taken to find a new optimal node. To solve this problem, for example, when the number of times that we can find the transit nodes offering better routes than the default routes is less than a threshold at the $n$-th period, we reinitialize the score, $C(n, v_i) = 1$ and $n = 0$. By doing this, we expect that we can shorten the time taken to find a new optimal node.

We also need to establish a method of determining an appropriate number of transit node candidates $M$ according to the traffic condition. One possible approach is to adjust $M$ by observing the degree of QoS improvement when we change $M$.

The above issues described in this section are for future study.

## 5. Conclusion

This paper gave the following evaluations about the effectiveness of routing control using an overlay network.

- Non-optimality of default route:
  We evaluated the end-to-end QoS when the optimal route can be selected. As a result, we found that the end-to-end QoS of the optimal route is much better than that of the default route at IP layer.
- Non-uniformity of optimal transit node:
  We discussed the limitation in the number of transit node candidates. This discussion is important for reducing the cost to select alternative routes that have better end-to-end QoS than the default route. As a result, we found that the selection of the optimal transit node is strongly biased.
- Impact of transit node limitation:
  We evaluated the improvement in end-to-end QoS when the number of transit node candidates is limited. As a result, we found that the performance of the sub-optimal route is almost the same as that of the optimal route if we select transit node candidates appropriately.

Based on the above results, we also described a cost-efficient overlay routing algorithm that learns optimal nodes through the process of finding better routes under the constraint that the number of transit node candidates is limited.

Our future work will focus on further reducing costs by clustering nodes. We will investigate how to improve the overlay routing algorithm so that it can avoid load concentration on a particular node and can cope with sudden changes in traffic and routing at the underlying layer.

## Acknowledgement

The evaluation section of this work was supported in part by the Ministry of Internal Affairs and Communications, Japan[†].

### References

[1] Y.T. Hou, Z. Duan, and Z. Zhang, "Service overlay networks: SLA, QoS and bandwidth provisioning," Proc. IEEE ICNP'02, Nov. 2002.

[2] L. Subramanian, I. Stoica, H. Balakrishnan, and R. Katz, "OverQoS: Offering QoS using overlays," Proc. HotNets-I, Oct. 2002.

[3] L. Zhi and P. Mohapatra, "QRON: QoS-aware routing in overlay networks," IEEE J. Sel. Areas Commun., vol.22, pp.29–40, Jan. 2004.

[4] D. Andersen, H. Balakrishnan, F. Kaashoek, and R. Morris, "Resilient overlay networks," 18th ACM SOSP, Oct. 2001.

[†] The method proposal and the evaluation of the proposed method in this paper were done when the first author was with NTT Service Integration Laboratories, NTT Corporation. On the other hand, a part of the evaluation of the proposed method in Sects. 3.6, 4.3 and 4.4 was done with a partial support by the Ministry of Internal Affairs and Communications, Japan, after he moved to Network Design Research Center, Kyushu Institute of Technology.

[5] S. Banerjee, T.G. Griffin, and M. Pias, "The interdomain connectivity of planetlab nodes," Proc. PAM 2004, April 2004.
[6] S. Rewaskar and J. Kaur, "Testing the scalability of overlay routing infrastructures," Proc. PAM 2004, April 2004.
[7] T. Murase, H. Shimonishi, and Y. Hasegawa, "TCP overlay network architecture," the 2002 IEICE Society Conference, B-7-49, 2002.
[8] Y. Amir, B. Awerbuch, C. Danilov, and J. Stanton, "Global flow control for wide area overlay networks: A cost benefit approach," IEEE OpenArch 2002, June 2002.
[9] A. Nakao, L. Peterson, and A. Bavier, "A routing underlay for overlay networks," ACM SIGCOMM'03, pp.11–18, Aug. 2003.
[10] S. Kamei, M. Uchida, R. Kawahara, and T. Abe, "Application of peer-to-peer technologies to overlay routing infrastructure," IEICE Technical Report, IN2005-38, vol.105, no.178, July 2005.
[11] J. Padhye, et al., "Modeling TCP Reno performance: a simple model and its empirical validation," IEEE/ACM Trans. Netw., vol.8, no.2, pp.133–145, 2000.
[12] http://www.planet-lab.org/php/overview.php
[13] https://wiki.planet-lab.org/twiki/bin/view/Planetlab/TraceLogs
[14] B. Krishnamurthy and J. Wang, "On network-aware clustering of web clients," ACM SIGCOMM 2000, pp.97–110, Aug. 2000.
[15] X.Y. Zhang, Q. Zhang, Z. Zhang, G. Song, and W. Zhu, "A construction of locality-aware overlay network: mOverlay and its performance," IEEE J. Sel. Areas Commun., vol.22, pp.18–28, Jan. 2004.
[16] T.S. Eugene Ng and H. Zhang, "Predicting Internet network distance with coordinates-based approaches," IEEE INFOCOM 2002, pp.170–179, 2002.
[17] S. Kamei and R. Kawahara, "Clustering method for IP QoS measurement on distributed environment," IEICE Technical Report, CQ2005-09, Nov. 2005.

## Appendix: Evaluation for Cases Where Ovelay Nodes Are Locally Allocated

Here, for cases where overlay nodes are locally allocated, we executed the same evaluation as in Sect. 3. We consider the following five cases: (I) overlay nodes are allocated only in Tokyo (6 nodes), (II) Osaka (4 nodes), (III) Tokyo and Osaka (10 nodes), (IV) Tokyo and Sapporo (10 nodes), and (V) Osaka and Kumamoto (8 nodes). The evaluation results are shown in Fig. A· 1, which corresponds to Fig. 8 in Sect. 3. Here, we calculated "suboptimal" results by using $M$ transit node candidates so that the nodes can cover 90% of the optimal routes, which were obtained when all nodes (e.g., 6 nodes in case (I)) were transit node candidates. That is, by calculating the same graph as in Fig. 6 in Sect. 3, we determined the transit node candidates. Figure A· 2 shows the results in case (I), for example. Here, the x-axis of this figure indicates the rank of nodes in Fig. 6. We chose two nodes (nodes 1 and 2) as transit node candidates in this case. The number of candidates $M$ in cases (II), (III), (IV), and (V) were 2, 4, 3, 2, respectively. Figure A· 1 shows that suboptimal routes can achieve much better performance than the default routes in any case and almost the same performance as the optimal routes. Therefore, we consider that overlay routing with a limited number of transit node candidates is also effective even in overlay networks where nodes are locally allocated.

We also evaluated case (VI) where Sapporo and Kumamoto (8 nodes) construct the overlay network. In contrast to cases (I)–(V), this case correponds to the case where overlay

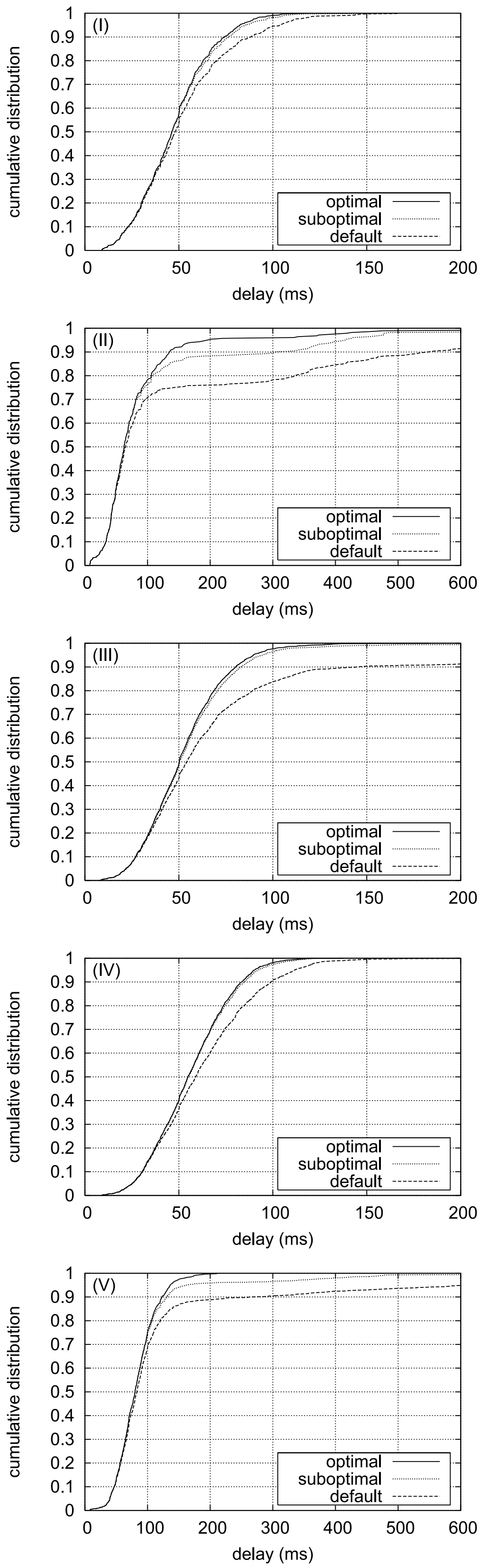


**Fig. A· 1** Results for local networks.

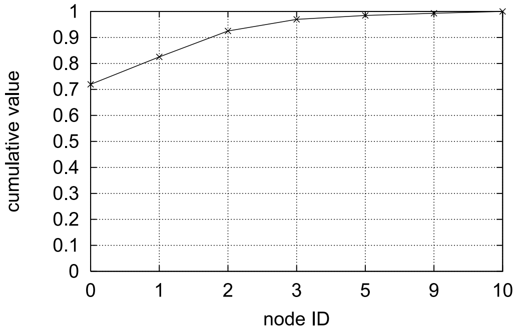


**Fig. A· 2** Frequecy of being optimal nodes in case (I).

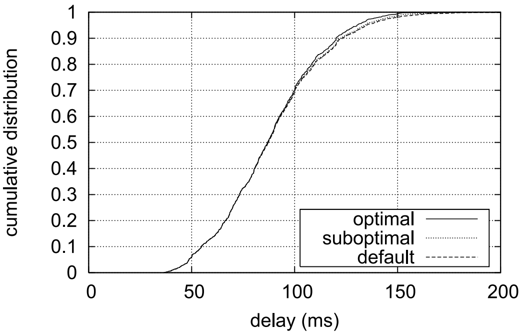


**Fig. A· 3** Results for nodes being remotely allocated.

nodes are remotely allocated. The evaluation result is shown in Fig. A· 3. In this case, we found that the overlay routing has little effect. The effectiveness of the overlay routing depends on the Internet structure such as topology. If there is no alternative route between a pair of overlay nodes, the overlay routing has no possibility of providing better routes than the default route. We consider that one of the reasons the overlay routing is not effective in case (VI) is that there may be few (or no) alternative routes because of the Internet structure.

Summarizing the above discussion, we conclude that, even when the number of nodes is small, the overlay routing is effective if the nodes are locally allocated, while the overlay routing has little effect if the nodes are remotely allocated like case (VI).

**Masato Uchida** received the B.E., M.E. and D.E. degrees from Hokkaido University, Sapporo, Hokkaido, in 1999, 2001, and 2005, respectively. In 2001, he joined NTT Service Integration Laboratories, Tokyo, Japan. Since August 2005, he has been an Associate Professor in Network Design Research Center, Kyushu Institute of Technology. He received the Research Award (IEICE Communication Quality Technical Group) in 2003, and the Young Investigators' Award (IEICE) in 2004. His research area includes teletraffic engineering and statistical learning theory.

**Satoshi Kamei** received the B.E. and M.E. degrees from Kyoto University, Kyoto, Japan in 1997 and 1999, respectively. In 1999 he joined NTT Service Integration Laboratories, Japan. Since 2004 he has been a doctoral course student in Kyoto University, Kyoto, Japan. He is engaged in research and development on QoS controllable IP networks, and end-host-based overlay-network such as peer-to-peer networks. He received the Young Investigators' Award (IEICE) in 2005. He is a member of Information Processing Society of Japan.

**Ryoichi Kawahara** received a B.E. in automatic control, an M.E. in automatic control, and a Ph.D. in telecommunication engineering from Waseda University, Tokyo, Japan, in 1990, 1992, and 2001, respectively. Since joining NTT in 1992, he has been engaged in research on traffic control for telecommunication networks. He is currently working on teletraffic issues in IP networks in NTT Service Integration Laboratories. Dr. Kawahara received IEICE's Young Investigators' Award in 1999 and Best Paper Award in 2003. He is a member of the Operations Research Society of Japan.

**Takeo Abe** received the B.E., M.E., and Dr. Eng. degrees in Applied Mathematics and Physics from Kyoto University, Kyoto, Japan, in 1978, 1980, and 1998, respectively. In 1980, he joined the Musashino Electrical Communication Laboratory of NTT Public Corporation (now NTT), where he has been engaged in research on network reliability design, traffic management, and congestion control. Currently, he is a professor of healthcare informatics at Tokyo Healthcare University. Dr. Abe received the IEICE Young Engineer Award in 1988 and has been chair of IEICE's Technical Committee on Communication Quality since 2005.