PAPER

Proposal for Adaptive Bandwidth Allocation Using One-Way Feedback Control for MPLS Networks*

Teruaki YOKOYAMA[†a)], Katsuyoshi IIDA[††], Hiroyuki KOGA[†††], *and* Suguru YAMAGUCHI[†], *Members*

SUMMARY In this research, we focused on fair bandwidth allocation on the Internet. The Internet provides communication services based on exchanged packets. The bandwidth available for each customer is often fluctuated. Fair bandwidth allocation is an important issue for ISPs to gain customer satisfaction. Static bandwidth allocation allows an exclusive bandwidth for specific traffic. Although it gives communications a QoS guarantee, it requires muany bandwidth resources as known as over-provisioning. In contrast with static control, dynamic control allocates bandwidth resources dynamically. It therefore utilizes bandwidth use more effectively. However, it needs control overhead in monitoring traffic and estimating the optimum allocation. The Transmission Control Protocol, or TCP is the dominant protocol on the Internet. It is also equipped with a traffic-rate-control mechanism. An adaptive bandwidth-allocation mechanism must control traffic that is under TCP control. Rapid feedback makes it possible to gain an advantage over TCP control. In this paper, we propose an Adaptive Bandwidth Allocation (ABA) mechanism as a feedback system for MPLS. Our proposal allows traffic to be regulated adaptively as its own weight value which can be assigned by administrators. The feedback bandwidth allocation in the previous work needs round-trip control delay in collecting network status along the communication path. We call this "round-trip feedback control." Our proposal, called "one-way feedback control," collects network status in half the time of roundtrip delay. We compare the performance of our one-way feedback-based mechanism and traditional round-trip feedback control under a simulation environment. We demonstrate the advantages of our rapid feedback control has using experimental results.

key words: *long-distance networks, adaptive bandwidth allocation, one-way feedback control, multi-protocol label switching (MPLS)*

1. Introduction

Many of today's companies, which have multiple branch offices that are geographically separated, construct and maintain Virtual Private Networks (VPNs) to exchange internal information. Moreover, most companies that run mission critical applications, such as financial and distribution industries, require Quality-of-Service (QoS) guarantees in VPN services. To attain this, one promising technology is Multi-Protocol Label Switching (MPLS) that enables QoS control including bandwidth allocation within service provider's networks [1].

To achieve QoS control in current MPLS, the Resource Reservation Protocol Traffic Engineering (RSVP-TE) protocol allows a path, which is established between the ingress and egress nodes for transmitting traffic, exclusively for bandwidth resources inside the MPLS network. This architecture enables fixed bandwidth allocation at the time the path is created and deleted. Fixed bandwidth allocation can ensure extremely accurate control in allocating bandwidths. However, it requires over-provisioning bandwidth resources for exclusive use. The current Internet provides a communication function that is based on resource-sharing networks. Therefore, it is difficult to apply static control directly to the Internet. Dynamic bandwidth allocation has the possibility of enabling both cost effectiveness and ensuring QoS. The Internet is already equipped with a bandwidth control mechanism, i.e., the Transmission Control Protocol (TCP). Dynamic bandwidth allocation has to control the bandwidth under TCP.

This paper proposes an Adaptive Bandwidth Allocation (ABA) mechanism for MPLS. There are some expected requirements to design this mechanism, i.e., a rapid convergence time for bandwidth allocation, a fair bandwidth allocation among customers, and higher bandwidth utilization. Therefore, we selected feedback congestion control, like the Available Bit Rate (ABR) in the Asynchronous Transfer Mode (ATM), which allows us to allocate bandwidth in a very short time scale to adapt to customer requirements and resource consumption [2]. As a result, it can fairly guarantee the minimum amount of bandwidth for each customer with a very small number of lost packets as well as high utilization of bandwidth.

Moreover, the existing feedback congestion control mechanisms use round-trip control instead of one-way control due to the difficulty of implementation, i.e., the route for the forward path is generally different from that for the return path on the Internet. However, MPLS has the capability of sending backward packets through the same routers in the forward path. It is necessary to construct one-way control to directly follow the communication path in the reverse direction. The benefit of one-way feedback control is expected to shorten propagation delay and to reduce the

Manuscript received November 10, 2006.
Manuscript revised February 22, 2007.

[†]The authors are with the Graduate School of Information Science, Nara Institute of Science and Technology, Ikoma-shi, 630-0192 Japan.
[††]The author is with Global Scientific Information and Computing Center, Tokyo Institute of Technology, Tokyo, 152-8550 Japan.
[†††]The author is with the Department of Information and Media Sciences, University of Kitakyushu, Kitakyushu-shi, 802-8577 Japan.

*A preliminary version of this paper was presented at the 1st Intn'l Forum on Information and Computer Technology (IFICT2003), Jan. 2003. This work was supported in part by the Ministry of Education, Science, Sports and Culture, Japan, Grant-in-Aid for Young Scientists, 18700054, 2006 and Grant-in-Aid for Scientific Research on Priority Areas, 18049063, 2006.

a) E-mail: terua-yo@is.naist.jp

DOI: 10.1093/ietcom/e90–b.12.3530

signaling overhead to about half. We evaluated the fundamental effect of one-way control in bandwidth allocation regardless of specific conditions such as different network interfaces and buffer sizes. In our proposal, each intermediate router reports congestion information to the upward neighbor router. After the ingress router obtains this information, the router limits the amount of traffic flow through the network according to the information.

Several past research projects have been done on adaptive traffic engineering for short-term-scale traffic allocation in MPLS. Elwalid et al. proposed [3] MPLS Adaptive Traffic Engineering (MATE), which dynamically balances the traffic load to multiple routes with only modifications to edge routers. Wang et al. proposed a feedback-based multi-path routing mechanism based on the Additive Increase and Multiplicative Decrease (AIMD) algorithm in MPLS networks [4]. These algorithms aim to use multiple routes simultaneously and efficiently. We, on the other hand, created a fundamental traffic-control mechanism, which is on a single path between the ingress and egress routers for TCP traffic. Another advantage is our one-way feedback control of congestion, which is expected to result in fast and fair allocations of bandwidth with higher network utilization.

Section 2 first introduces the Weighted Proportional Fair Rate Allocation (WPFRA) mechanism, the base mechanism in our proposal, which is feedback-based congestion control for DiffServ networks. We then propose ABA in Sect. 3, and then discuss our evaluation of it through simulations in Sect. 4. We specifically compare the performance of one-way and round-trip feedback control mechanisms, and consider parameter tuning under simple but extensive simulation scenarios. We also present simulation results from more complicated scenarios such as those with differing round-trip times (RTTs) and different weight settings. We conclude the paper in Sect. 5.

2. Weighted Proportional Fair Rate Allocation Mechanism

For service providers to allocate bandwidth on a short-term scale, the Weighted Proportional Fair Rate Allocation (WPFRA) mechanism has been proposed to allow us to dynamically allocate traffic with reasonably high utilization in DiffServ networks [5]. Basically speaking, WPFRA is a DiffServ version of ATM ABR traffic control that employs a resource management (RM) cell to distribute the congestion information throughout the network. We briefly describe WPFRA in this section.

2.1 Measurement and Traffic-Control Components

WPFRA is a path-based traffic-allocation mechanism. Specifically, a path, which is a unit of traffic allocation, is composed of a pair of ingress and egress routers. In WPFRA, a weight is assigned for each path to allow a different bandwidth allocation for different paths. Basically, WPFRA consists of four functions: measurement, calculation, notification, and enforcement. Measurement means traffic measurement on a core router, which is used to calculate the optimum bandwidth on that router. The optimum value is then reported to other routers. This notification is done repeatedly. Finally, the ingress router enforces the traffic allocation.

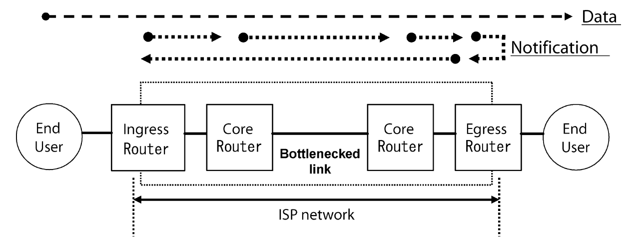


Fig. 1 Notification in roundtrip feedback control.

Figure 1 illustrates how these four functions are performed. First, ingress routers periodically send a control packet along each path. The control packet includes an explicit rate (ER) value that is the available bandwidth for the path. A core router periodically calculates the optimum bandwidth according to the measured amount of traffic. When the packet arrives at a core router, it rewrites the ER on the packet according to the previously calculated optimum bandwidth. This notification process is done repeatedly. When the packet, which includes the ER adjusted by a bottleneck router, arrives at the egress router, it sends back the packet to the ingress router. Note that the packet is not rewritten in the backward direction. Finally, the ingress router sets the policing parameter, which is the ER value multiplied by a pre-specified weight, on the path. In this way, the notification process is performed through a round trip, starting and ending at the ingress router.

2.2 Calculation, Notification, and Enforcement Algorithms

As explained above, core routers periodically calculate the optimum bandwidth for all paths. Here, the optimum bandwidth refers to the amount of link bandwidth divided by the sum of the weight values of active paths accommodated by the link. The optimum bandwidth thus represents the fair-share bandwidth for each weight. Here, we explain how the optimum bandwidth at core routers is calculated.

The optimum bandwidth is calculated just after periodic measurements of the traffic volume for each path. Let C be the bandwidth of a link, w_i be a weight value, and r_i be the optimum bandwidth given by

$$r_i = w_i \cdot \frac{C}{\sum_{j=1}^{n} w_j}, \tag{1}$$

where i indicates the path index. This function is based on the view that a path with n weights indicates that there are n virtual flows. However, as we cannot calculate r_i directly, we have to estimate this value using the estimated number of virtual flows n_{vf}. We then have

$$n_{vf} = \max\left(\frac{r_{arr}}{r_f}, 1\right), \tag{2}$$

where r_{arr} is the measured amount of traffic. We then calculate r'_f according to

$$r'_f = \frac{C}{n_{vf}}. \tag{3}$$

Since traffic in data networks is bursty in nature, we smooth r_f through an exponential weighted moving average that outputs $\hat{r_f}$ given by

$$\hat{r_f} \Leftarrow (1-\alpha)\cdot\hat{r_f} + \alpha\cdot r'_f, \tag{4}$$

$$r_f \Leftarrow \hat{r_f}, \tag{5}$$

where α is a smoothing factor within the range $0 \leq \alpha < 1$. α is a coefficient for the moving average in the repeated process of calculating ER. It takes a fixed value in the process. As shown by Eq. (4) in Sect. 2, it only influences the speed of convergence of ER calculations. When a core router receives a control packet, it chooses the minimum from the received ER value and r_f:

$$ER \Leftarrow \min(ER, r_f). \tag{6}$$

The next process is notification using the ER values of control packets. The initial ER value of the control packet is replaced by a new ER value and the packet is transmitted to the downward router along the path. A core router receives multiple ER values if multiple communication routes that share a common bottleneck path cross the router. In this situation, the router chooses the lowest value. As a result, the entire ingress node knows the ER, i.e., the least value in multiple routes, which represents a unit of bandwidth for a unit of weight value at the common bottleneck link. In other words, the ER value would be defined with each bottleneck link.

The final process is enforcement at the ingress routers. When an ingress router receives a control packet, it sets the ER value of the packet as the policing parameter. Since the ingress router has a leaky-bucket shaper, we adjust the mean traffic rate for the shaper according to the ER value.

The ER value will converge to an approximate value for the amount of optimum bandwidth for a unit of weight value through the iteration to calculate ER values. The ER takes a value of 1 if there is no congestion in the target link. We used 1 for the initial value of ER. The ER changes when other traffic arrives.

3. Proposal of ABA in MPLS

We found a way of constructing one-way control in dynamic bandwidth allocation on the Internet. We propose Adaptive Bandwidth Allocation (ABA) that is based on WPFRA in this section to guarantee the minimum bandwidth in MPLS. We first introduce our modifications to the MPLS mechanism and then describe the ABA algorithm as well as its protocol.

3.1 Modification to MPLS

The route of communication in typical IP networks is defined as an asymmetric route from the source to the destination node. That is, the communication route on the back path is sometimes different from that on the forwarding path. In MPLS, we found a way of constructing one-way feedback control for adaptive bandwidth allocation. In this section, we first explain the modifications to MPLS.

MPLS is a label-based extension of the IP network inside one administrative domain [6]. A pair of ingress and egress routers in an MPLS network constitutes a communication path inside the network as a label switched path (LSP) that includes a set of intermediate core routers. Incoming traffic can be handled by each LSP control. MPLS provides QoS functions through LSP-oriented control such as bandwidth allocation and traffic engineering for LSPs. It allows a way of finding a communication route. The egress and core routers can follow the LSP along the reverse direction of the communication path from the egress router to the ingress router hop-by-hop. Thus, the egress router can send a control signal to the ingress router directly as one-way feedback control.

MPLS uses the Label Distribution Protocol (LDP) [7] when it constructs LSP. LDP notifies of the mapping information between the label and LSP. This is used for LSP control. Once notification is done, all core routers have an appropriate incoming label map (ILM) table. This table is used as a forwarding information base to map an incoming label to the next hop and the next label. Since an ILM in point-to-point LSPs is a table of one-to-one relationships, all the reverse directions can be uniquely resolved. LDP also includes a mechanism that have availability for MPLS features extension. Kinds of LDP messages, i.e., “notification messages” are defined in the Request For Comments (RFC) of LDP specifications for transferring notification messages among neighboring routers. These two features, ILM and LDP extension can be used to construct a way to transfer the control signal that contains the available bandwidth along the reverse direction of the communication path.

3.2 Algorithm and Protocol in ABA

This section describes the ABA mechanism, which is one-way feedback control as well as its protocol. The proposed method is expected to solve unfair bandwidth use among LSPs on bottleneck links. In proposing ABA, we need to consider fairness (subject to constraints), the effective use of link bandwidths, and rapid convergence time.

The elements of the ABA algorithm are illustrated in Fig. 2. We propose ABA as a one-way feedback-based system in an MPLS network to solve the competition for bandwidth. Our proposal does not need to transmit a control signal on the round trip of the communication path. This reduces the control convergence time and contributes to improving the accuracy of bandwidth allocation. Our challenge was to reduce the convergence time by means of an autonomous and distributed algorithm. The approach we took was to create one-way feedback control that was exactly like the algorithm described in Sect. 2.2, except for the notification with MPLS architecture. We need to consider a mechanism that periodically monitors current bandwidth use for all communication paths in the network to allocate an appropriate amount of bandwidth adaptively for traffic competing for this. The monitoring mechanism has to follow the communication path from the ingress router to the egress router to obtain bandwidth for the target path. Using the communication path that follows to monitor network status contributes to removing synchronization management from network-based communication control. ER notifications are designed as event-driven mechanisms. ER calculations are processed independently of other node behaviors and are always repeated at Measurement Intervals (MIs). The Notification Interval (NI) affects the intervals ER notifications are transmitted at the egress node, from which ER notifications originate. All nodes transmit their calculated ER notifications when they receive an ER notification from other nodes.

The current routing system on the Internet cannot support the construction of one-way feedback control because it only stores a destination-based routing table and only knows the next-hop address in packet forwarding. MPLS, on the other hand, features a path-based traffic-forwarding mechanism. MPLS enables the communication path to be followed in the reverse direction of the path. This can be constructed when transmitting the control signal with the LDP extension and following the communication path from ILM information.

We designed a protocol for ER notifications in MPLS. Figure 3 shows how our protocol works with traffic across the network. When traffic flows in an MPLS network, paths are constructed from the ingress node to the egress node. The ingress node has to be aware of the competition for bandwidth on the paths to limit the bandwidth for traffic. With the ABA mechanism, all nodes are able to notify of their ER values in one-way control transmitting from the egress node to the ingress node. This explains how the ER value is transferred in the opposite direction to the traffic. Each node independently defines the ER value, which is calculated by the amount of traffic incoming to them. Nodes choose the lowest in receiving multiple ER values where several paths are crossing at the same node.

We used two characteristic features in MPLS, which are ILM information and notification messages, for our mechanism. MPLS keeps mapping information between incoming and outgoing communication paths as ILM. For example in Fig. 3, node Y can know about the relationship between Path A and B from ILM information. MPLS uses the protocol of the notification message to provide advisory information and to signal error information among MPLS-enabled routers. LDP has the capability to carry ER messages. We extended the messages to carry information for WPFRA and transfer the ER value to the upstreams of

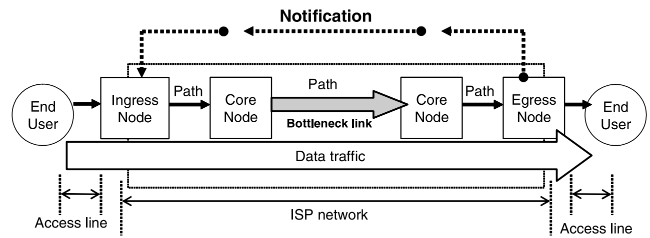


Fig. 2 Notification with one-way feedback control.

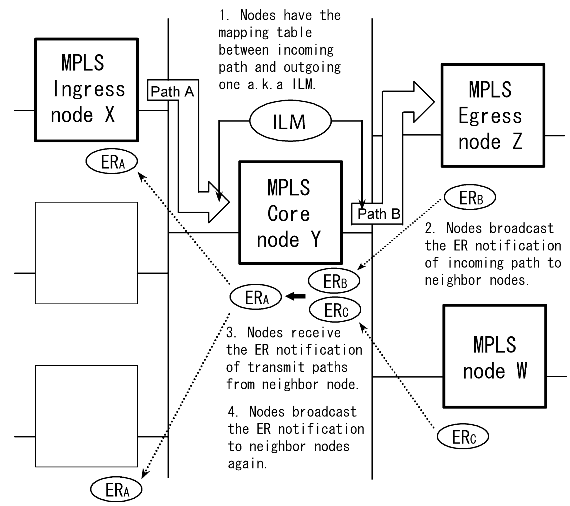


Fig. 3 Protocol of ER notification.

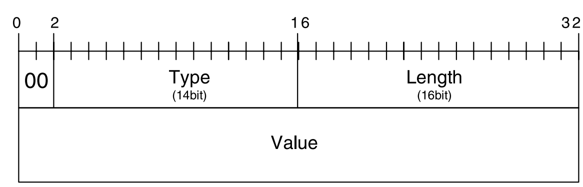


Fig. 4 An example of LDP message.

the communication path by using ILM information. Figure 4 has an example of the LDP message format we improved. The MPLS node periodically exchanges LDP messages among neighboring nodes. The messages are used for exchanging several kinds of information in MPLS. LDP can be extended to carry ER values in its value field.

The procedures in our proposal are:

1. All intermediate router keep ILM for path identification. We use this for identifying a pair of communication paths and ER values.
2. Unlike the original WPFRA, the egress router periodically creates a control packet that is destined for the ingress router. This period will be discussed as the Notification Interval (NI) in Sect. 4.
3. Corresponding to the calculation algorithm, each core router repeatedly calculates the ER value from the link bandwidth status at neighbor links at given Measurement Intervals (MIs). In other words, the ER value on each node is updated at MIs.

4. When the core node receives an ER notification, the node choose the minimum from received ER value and calculated values. It then resends the updated ER notification to an upward router on the target path.

As a result, the system allocates an appropriate bandwidth across all LSPs.

4. Simulation Results

In this section, we compare the performance of both forms of control. We first describe our simulation scenarios and the performance metrics we evaluated. We then explain the difference between one-way and round-trip feedback in a basic comparison, performance comparisons in several situations, the impact of network distance on the improvements, and the results of more complex simulations of realistic scenarios. With our proposal, we aim to improve bandwidth allocation within a shorter convergence time, with more stable and higher utilization, and better bandwidth distribution.

4.1 Simulation Environment

We implemented our proposal on the ns-2.27 simulator [8]. Our mechanism, called the Adaptive Bandwidth Allocation (ABA) module, is based on the algorithm described in Sect. 3. As the ABA module contains round-trip and one-way feedback controls, we could quantitatively evaluate the differences between these two forms of control.

Figure 5 outlines the simulation topology. We evaluated the effect of bandwidth allocation in one-way control against the volume of TCP traffic. Because such traffic causes unfair bandwidth allocation in bottleneck links, we designed a traffic source that included multiple TCP hosts in the topology. We confirmed the fundamental characteristics of our proposal in the topology.

The bandwidth of all links was 100 Mb/s, except that for the bottleneck link, which was only 10 Mb/s. To represent a site including multiple TCP source hosts, we introduced TCP source groups. Each group included five TCP source hosts, each of which was connected to the group gateway and generated six FTP flows; i.e., one TCP source group generated 30 FTP flows. To prevent TCP synchronization, the propagation delay between each TCP source host and the group gateway was varied within a range from 2.0 to 4.0 ms with 0.5-ms intervals. The group gateway was connected to the corresponding ingress router with a 2.0-ms propagation delay. Here, we had n TCP source groups. All ingress routers were connected to the core router with a propagation delay of a [ms]. The core router was connected to the egress router through the bottleneck link with a propagation delay of b [ms]. The egress router was connected with a 5.0-ms propagation delay to the destination nodes that terminated all traffic from the sources.

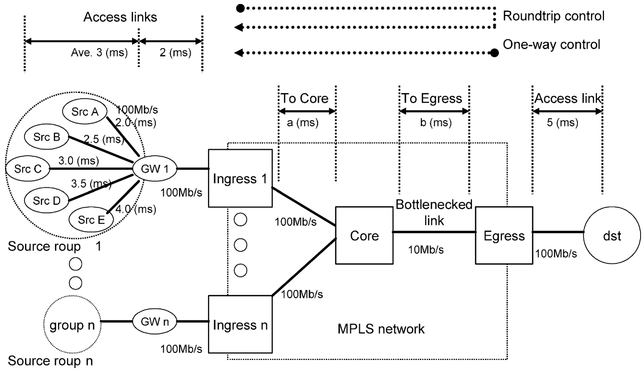


Fig. 5 Simulation topology.

The buffer sizes in the forwarding direction were 100 packets for the group gateways, 20 packets for the ingress routers, 30 for the core router, and 100 for the egress router. We assumed no packets were lost on the notification paths.

We further defined the communication path in the simulation. A path was composed of ingress, core, and egress routers to represent aggregated traffic. Path n was the traffic generated by TCP source group n. ABA was performed for all paths.

4.2 Evaluation Metrics

We employed two kinds of graphs in our experiments to visualize the experimental results. The graphs illustrate the performance metrics and behavior descriptions of bandwidth-allocation control in our experiments. We chose each according to the purpose of the experiments.

1. Performance metrics
   We evaluated three performance metrics to compare round-trip control and one-way control. We represented trends in the metrics with changing control parameters in both graphs. From the metrics, we aimed to find performance trends where the control parameters were variable. In other words, the graphs reflected what influence the control parameters had on performance. One point in the graphs represents the results of one experiment. The graphs were prepared with the results obtained from multiple experiments. We conducted our experiments with the metrics scenario in Sects. 4.4 and 4.5.
   The experimental scenario was to evaluate performance over longer time scales. We used two TCP sources groups, each of which was allocated a bandwidth by the network. The first TCP source group started flowing at 0 s, and the second group started at 2 s. The measurement period for calculating the three metrics ranged from 20 to 40 s.
   - Fairness: The first metric was the ratio of the allocated bandwidth for Path 1 to the required amount. This metric was to test the accuracy of bandwidth allocation. The ratio is given division allocated bandwidth by optimum allocation. The required metrics were close to 1.
   - Dropped packets: The second metric was the

number of dropped packets at the bottleneck link during the period of measurement. This metric indicates the amount of wasted-bandwidth resources. The last packets will be dropped at a bottleneck link after consuming bandwidth resources. Transmitting such packets causes bandwidth resources to be wasted because the wasted bandwidth could be used for transmitting other traffic. Ideally, the number of dropped packets at a bottleneck should be zero when network resources are used effectively.

- Utilization: The last metric was the average utilization of the bottleneck link during the measurement period. The utilization is given a ratio of used bandwidth at the bottleneck link. This metric is 1 when traffic across the bottleneck link occupies all the bandwidth at the bottleneck link.

2. Behavior description
To determine the behavior of bandwidth allocation, we focused on three metrics We prepared graphs with the metrics in Sects. 4.3 and 4.6. These graphs illustrate the state of the bandwidth-allocation system we found through an experiment. The inner workings of the system can be seen from the graph.

- Used bandwidth: This graph reflects how much traffic flow was in the target link in the experiment. It shows the bandwidth at the bottleneck link. Each path is represented in a different style in the cumulative graph.
- Dropped packets: This graph shows the dropped packets that occurred in the experiment. Packets are dropped when traffic that exceeds the bottleneck link's limits arrives. We can establish the timing for control to fail from the graph.
- ER value: This graph plots the changing ER values throughout the experiment. The ER value was repeatedly refreshed based on calculations at all nodes. The ER value indicates the optimal bandwidth allocation. The ER should be stable after the network situation has changed.

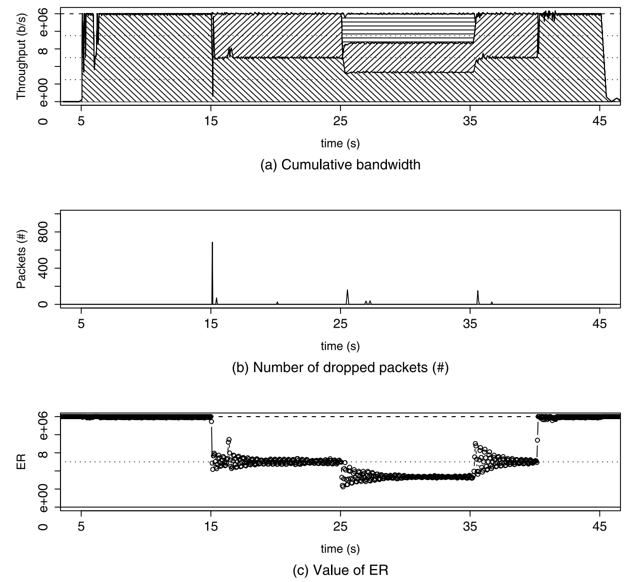


Fig. 6 Numerical example of one-way feedback control.

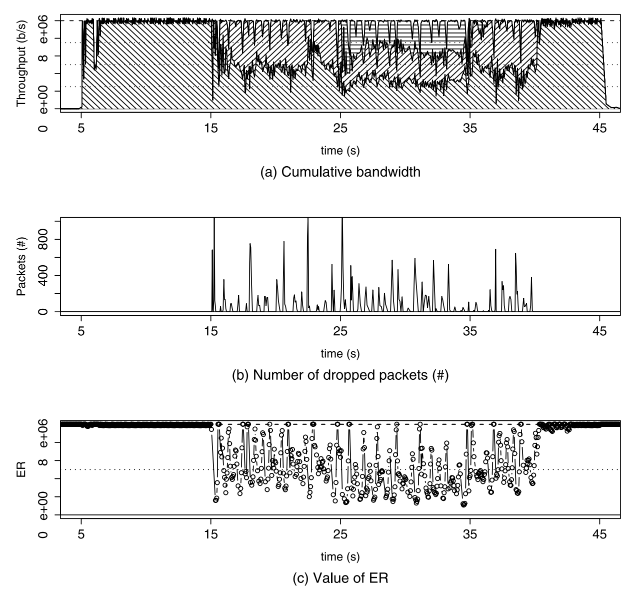


Fig. 7 Numerical example of roundtrip feedback control.

4.3 Impact of One-Way Feedback Control

Before discussing our comparisons, we will briefly discuss the difference between the round-trip and one-way controls. In the simulation scenario, we had three TCP source groups. Paths 1, 2, and 3 started flowing at 5, 10, and 15 s, respectively. Paths 3, 2, and 1 stopped flowing at 25, 30, and 35 s, respectively. The parameter values we used were an α of 0.6, an NI of 50 ms, and a MI of 50 ms. Propagation delays a and b were set to 20 and 30 ms, respectively. We will describe the reason for selecting these parameters in Sect. 4.4.

Figures 6 and 7 show numerical examples of the one-way and round-trip feedback controls. Each figure contains three sub-figures, whose x-axes plot the simulation time in seconds. The y-axes of the first, second, and third sub-figures represent the respective cumulative throughputs of the three paths, the number of dropped packets at the bottleneck router, and the ER value. The cumulative throughput represents how the link is used by the crossing path in the target link. The cumulative line would reach the top of the graph if the whole bandwidth at the link was consumed. Each diagonal region reflects the bandwidth exhausted by each path.

One-way feedback control enables more accurate and faster control of the ER value than is obtained with round-trip control. As we can see from the figure, round-trip control often results in unstable situations when it forces bandwidth to be allocated under TCP flow control. Round-trip feedback control needs the round-trip time of its communication path to notify of the control value and this causes prolonged convergence for controlling the target bandwidth.

4.4 Parameter Selection

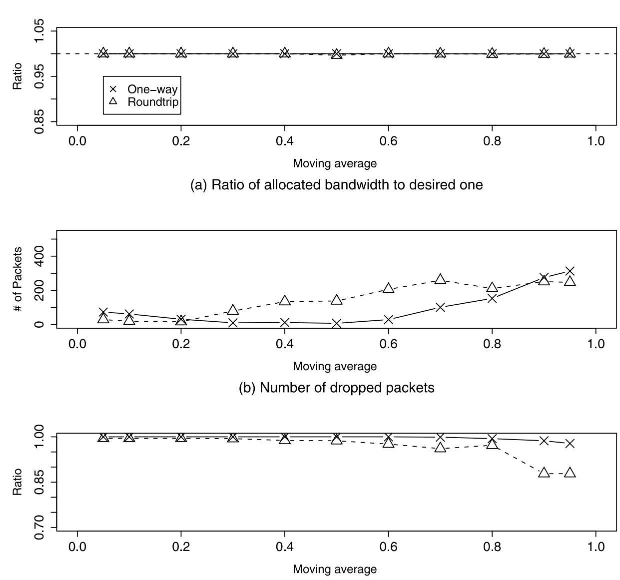
Our proposal has three parameters: α, NI, and MI. ABA depends on these parameters. We investigated what effects they would have and later chose adequate parameters for the experiments. We performed simulation for parameter selection in stable case.

α is the coefficient of the moving average for the core-router measurement. It simply establishes the ER value, which is used to define the traffic threshold smooth. MI and NI are the intervals of notification. MI operates the measurement interval for available bandwidth on each router. NI operates timing to notify of the current ER value. We varied these parameters to comprehensively compare one-way and round-trip feedback control. We investigated how the parameters affected ABA through brief experiments. Unless otherwise stated, the default parameter values in the experiments were an α of 0.6, an NI of 50 ms, and an MI of 50 ms. Propagation delays a and b were both set to 50 ms.

Figure 8 shows the effects of α. The following speed of ER value while changing amount of incoming traffic become slow with small α. A large α increases speed and makes ABA available while the amount of traffic is rapidly changing. However, such sensitivity degrades performance because of over-fitting with micro-bandwidth fluctuations due to TCP control. This degradation is apparent when α is over 0.6 as seen in Fig. 8. Hence, α should be balanced between 0 and 1 according to the degree of bandwidth fluctuation in the target network. Its effects are predictable because it only affects convergence speed in the chain of ER calculations. We only used 0.6 as a default value for α.

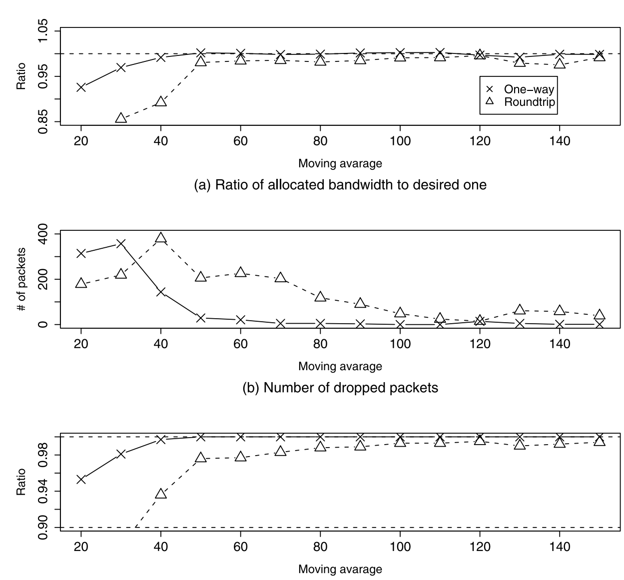
Figure 9 has graphs that show the movement of performance metrics under changing MIs. They show the impact of MIs on performance metrics. Although faster control is required for dynamic traffic control, both feedback controls degrade performance in the figure with short MIs, which means fast monitoring. This phenomenon can be explained by fundamental control theory, which suggests that control after the results of the previous control have been received yields better performance. If the system establishes control before the results are confirmed, the feedback system collapses due to control mismatching. Additionally, long NIs also possibly degrade performance because of large control delays. To avoid collapse and degradation, we need to choose an effective range for MIs and NIs. We only selected effective parameters to control ABA in limited situations because the proposed system includes conflict in the duplicate feedback system between TCP control and ABA control. More detailed analysis of the conflict is beyond the range of this paper. We intend to discuss this analysis in future work.

There are two kinds of points in this control. Operating points that limit incoming traffic are placed on the border router between the external network and the managed network. Monitoring points are placed on core routers inside the managed network. It monitors the current amount of traffic and notifies the operation points of the traffic. Since the operating points are sometimes too far away from monitoring points, control from monitoring to operating is delayed. Control will fail if routers allocate bandwidth before they confirm what effects their previous operation had. One-way feedback control led to instability when the MI was 50 ms or less. Round-trip control led to instability when MI was less than 100 ms. The difference was caused by different control path routes. The monitoring points send control packet directly to operating points. Therefore, the one-way



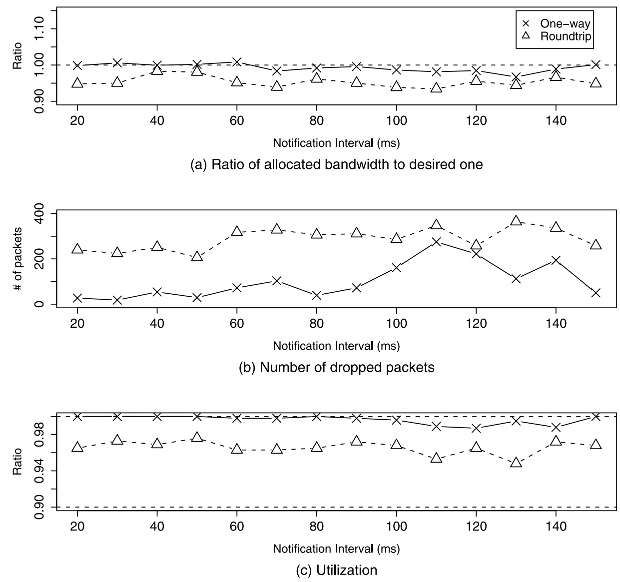
(a) Ratio of allocated bandwidth to desired one
(b) Number of dropped packets
(c) Utilization

Fig. 8 Effect of α.



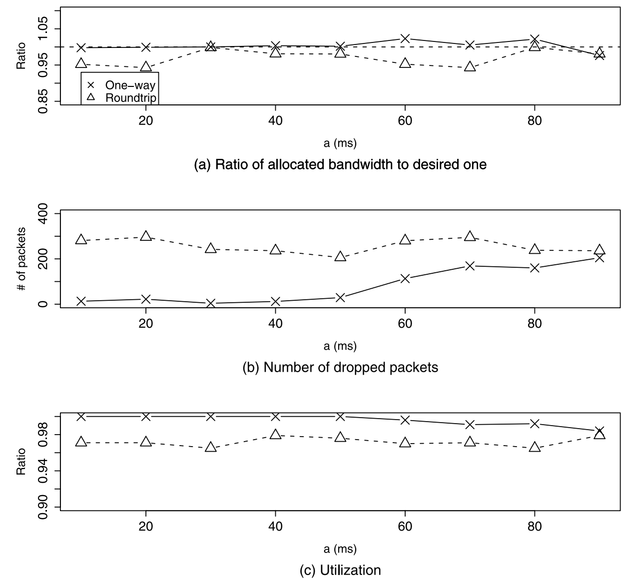
(a) Ratio of allocated bandwidth to desired one
(b) Number of dropped packets
(c) Utilization

Fig. 9 Effect of MI. (NI = 50 ms)



(a) Ratio of allocated bandwidth to desired one
(b) Number of dropped packets
(c) Utilization

Fig. 10 Effect of NI. (MI = 50 ms for one-way feedback control, MI = 100 ms for round-trip one)



(a) Ratio of allocated bandwidth to desired one
(b) Number of dropped packets
(c) Utilization

Fig. 11 Fixed diameter. ($a + b$ = 100 m)

feedback control path is always shorter than that for round-trip feedback.

One-way feedback control has the advantage of a wide range in stability against the changing distances from monitoring and operating. This is because traffic-congestion points often change and the distance between the monitoring and operating points also change in a managed network.

Figure 10 also has graphs of the performance metrics under changing NI, showing its impact. In these experiments, we choose 50 ms as MI for one-way feedback control and 100 ms for round-trip feedback control. These values were chosen as the fastest intervals determined from the results of the previous experiment.

The accuracy of bandwidth allocation declined with large NI and MI. A short propagation time contributed to suppressing traffic fluctuations. As a result, one-way feedback control always provided stable and accurate performance in terms of all three metrics.

From these results, we learned how to determine the control parameters. Quick feedback control could not allocate bandwidths better in all situations. It occasionally deteriorated performance. Slow control can also cause such deterioration. This means the lower and upper limits should be set for control propagation to achieve better performance. One-way feedback control has a wide range of adaptability in traffic control because of its short propagation time.

Our suggestions for the control parameters are that α should be 0.6, and MI and NI should each equal one-half the network diameter. The locations of bottleneck links are sometimes changed in a realistic network. Therefore, we assumed that a congestion point would occur on the center link of a network. This promises better control of ABA in various situations.

4.5 Case with Variable Propagation Delay

As discussed in the previous section, we confirmed faster control with our proposal than that with round-trip control. In this section, we show that this advantage allows high adaptability to feedback bandwidth control even if the congestion point in the target network is drastically changed. To evaluate the advantage, we did two sets of simulations with either fixed or variable diameters. Here, the *diameter* represents the sum of the propagation delay starting from the ingress router and going to the egress router; thus, this is equivalent to $a + b$ in Fig. 5. These environments were constructed to represent typical patterns of network congestion.

In the fixed-diameter case, we set the diameter to 100 ms and varied a and b under the condition that $a + b$ = 100 ms. This indicted performance where the bottleneck links are moved in a fixed network topology. In the variable-diameter case, we fixed a to 20 ms and varied b. This also indicted performance where an amount of traffic, which has various RTTs, is transmitted. In these cases, we optimized both feedback systems by assuming that congestion occurred on a center link in the target communication path.

Figure 11 shows what impact a had on the performance metrics for the fixed-diameter case. We can see that one-way feedback allowed stable performance in terms of all three parameters when a was less than or equal to 50 ms, while round-trip feedback led to unstable performance for all values of a. When a was more than 50 ms, the performance with one-way feedback deteriorated as a increased. When we set a to 90 ms, the performance with one-way feedback was very close to that with round-trip feedback. This is because the propagation delay of control packets from the bottleneck router to the ingress router differed between the two forms of feedback control: the delay for one-way feedback

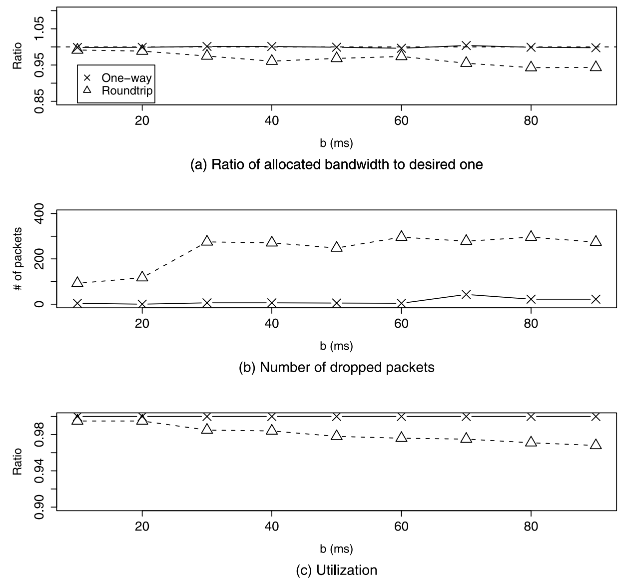


Fig. 12 Variable diameter. (a = 20 ms)

control was a while that for round-trip feedback control was $b + b + a$. In round-trip feedback control, the delay includes round-trip transmission from the bottleneck router to the egress router. One-way feedback control provides superior performance when hotspots in the network often change.

We next considered performance with a variable diameter. Figure 12 shows what impact b had. The figure shows that one-way feedback control is not affected by b increasing although round-trip control clearly is. For small values of b, the round-trip performance with regard to all three metrics is comparable to one-way performance. However, the performance with round-trip feedback control deteriorated significantly with increasing b. The explanation for improved performance when the diameter is fixed also applies in this case.

Let us summarize the above results. With a fixed-diameter topology, one-way feedback control improved utilization for almost all values of a, as well as the bandwidth allocation accuracy when a was less than or equal to 50 ms (equal to the values of NI and MI). With a variable-diameter topology, one-way feedback control always enabled accurate bandwidth allocation as well as close to 100% utilization, while round-trip control was negatively affected by b increasing. As a result, we concluded one-way feedback control has a wide range of adaptability against changing network situations.

4.6 Practical Examples

In the above, we examined the fundamental differences in performance between one-way and round-trip feedback control. The results revealed the superiority of one-way control over round-trip control. One-way control is able to accurately allocate bandwidths in various network situations. Here, we will discuss our confirmation that bandwidth allocation with one-way control works properly under more practical and realistic scenarios where we use either a mixture of different RTTs or a mixture of different weights. For both scenarios, we had three paths, and all flows on paths 1, 2, and 3, respectively, started at 0, 40, and 80 s and then ended at 120, 160, and 200 s.

For the first scenario, we set a to 5, 50, and 100 ms for paths 1, 2, and 3, respectively. Figure 13 shows the throughput and packet loss dynamics for each path, and Table 1 lists the average throughputs and bandwidth ratios for each path,

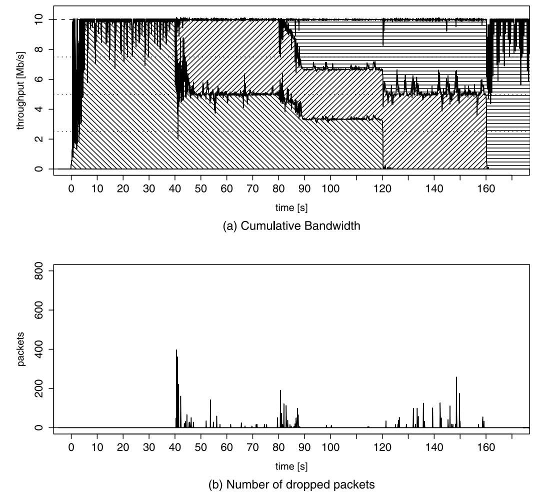


Fig. 13 Numerical examples in case with different RTTs.

Table 1 Throughput comparison in case with different RTTs.

	Number of received packets			Ratio of bandwidth share		
Time (s)	Path1	Path2	Path3	Path1	Path2	Path3
50–80	12541	12459	0	**1.000**	0.993	—
90–120	8372	8405	8223	**1.000**	1.004	0.982
130–160	0	12846	12138	—	**1.000**	0.945

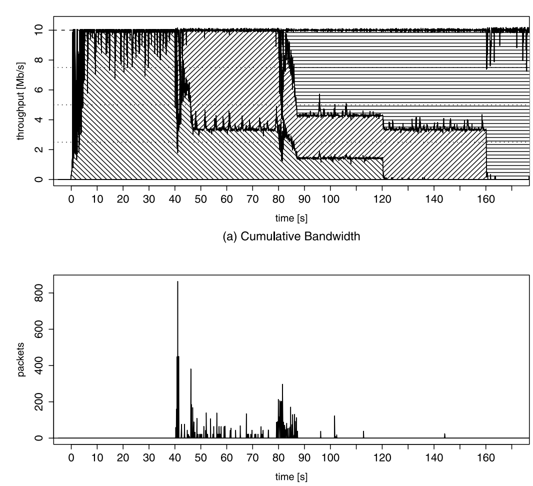


Fig. 14 Numerical examples in case with different weights.

Table 2 Throughput comparison in case with different weights.

	Number of received packets			Ratio of bandwidth share		
Time (s)	Path1	Path2	Path3	Path1	Path2	Path3
50–80	8605	16383	0	**1.000**	1.904	—
90–120	3601	7218	14181	**1.000**	2.004	3.938
130–160	0	8525	16472	—	**2.000**	3.864

normalized by the throughput share of path 1, for each time segment (50, 80 s), (90, 120 s), and (130, 160 s). These results indicate that the ABA control we propose can adapt to different RTT values.

For the second scenario, we set a to 5 ms for paths 1, 2, and 3. In this case, the weights of values varied between paths, being 1 for path 1, 2 for path 2, and 4 for path 3. Figure 14 shows the throughput and packet loss dynamics for all paths, and Table 2 lists the average throughputs and bandwidth ratios for each path. These results mean that our ABA control can also adapt to the heterogeneity of weights.

5. Conclusion

This paper proposed ABA control to provide stable communication in MPLS networks. We showed how to implement a one-way feedback system on an MPLS system. Our proposed mechanism accurately allocates bandwidths based on one-way feedback control. We did simulation experiments, which demonstrated the advantage of our proposal to achieve a rapid convergence time to control bandwidth. In contrast with the results for round-trip feedback control, its rapid bandwidth control was useful for suppressing bandwidth fluctuations. It made dynamic allocation of bandwidth more stable and more precise. Therefore, unfair bandwidth allocation should decrease in networks operated with our proposed method and contribute to allowing ISPs to provide fair allocation in communication services. Moreover, one-way feedback control provides the most rapid method of determining network status from the core node to the edge node. It is possible to apply one-way control to not only ensure fairness but also to other traffic engineering mechanisms such as dynamic re-routing.

Through the experiments, we found degraded performance depends on control interval changes. Short control periods, in particular, also degrade performance. The cause of degradation may be conflict between our ABA control and TCP. In future work, we recognize the analysis of the relationship between both control behavior is important for constructing communication management mechanism inside network which control the whole traffic across the managed network.

References

[1] B. Davie, P. Doolan, and Y. Rekhter, Switching in IP Networks: IP Switching, Tag Switching, and Related Technologies, Morgan Kaufmann, 1998.

[2] M. de Prycker, Asynchronous transfer mode: Solution for broadband ISDN, Third ed., Prentice Hall, 1995.

[3] A. Elwalid, C. Jin, S.H. Low, and I. Widjaja, “MATE: MPLS adaptive traffic engineering,” INFOCOM, pp.1300–1309, 2001.

[4] J. Wang, S. Patek, H. Wang, and J. Liebeherr, “Traffic engineering with AIMD in MPLS networks,” Protocols for High-Speed Networks, pp.192–210, 2002.

[5] C.-L. Lee, C.-W. Chen, and Y.-C. Chen, “Weighted proportional fair rate allocations in a differentiated services network,” IEICE Trans. Commun., vol.E85-B, no.1, pp.116–128, Jan. 2002.

[6] E. Rosen, A. Viswanathan, and R. Callon, “Multiprotocol label switching architecture,” IETF RFC 3031, Jan. 2001.

[7] L. Andersson, P. Doolan, N. Feldman, A. Fredette, and B. Thomas, “LDP specification,” IETF RFC3036, Jan. 2001.

[8] “LBNL network simulator (ns),” available at http://www-nrg.ee.lbl.gov/ns/

Teruaki Yokoyama received his B.E. in computer science from Ritsumeikan University, Kusatsu, in 2000, and his M.E. and Ph.D. degrees in Information Science from the Nara Institute of Science and Technology, Ikoma, in 2002 and 2007. His research interests include algorithms and architectures for connection management on the Internet. He is a member of the Widely Integrated Distributed Environment (WIDE) Project. He is currently an Assistant Professor at the Cyber University, Japan’s first online university.

Katsuyoshi Iida received the B.E., M.E. and Ph.D. degrees in Computer Science and Systems Engineering from Kyushu Institute of Technology (KIT), Iizuka, Japan in 1996, in Information Science from Nara Institute of Science and Technology (NAIST), Ikoma, Japan in 1998, and Computer Science and Systems Engineering from KIT in 2001, respectively. Since Oct. 2000, he was an Assistant Professor in the Graduate School of Information Science, NAIST. Currently, he is an Associate Professor in the Global Scientific Information and Computing Center, Tokyo Institute of Technology, Tokyo, Japan. His research interests include performance evaluation of networking systems, Internet QoS, and mobile networks. He is a member of the WIDE project and IEEE. In 2003, he received the 18th TELECOM System Technology Award, the Telecommunications Advancement Foundation, Japan.

Hiroyuki Koga received B.E., M.E., and Ph.D. degrees in computer science and electronics from Kyushu Institute of Technology, Japan in 1998, 2000, and 2003, respectively. From 2003 to 2004 he was a postdoctoral researcher in the Graduate School of Information Science, Nara Institute of Science and Technology, Japan. From 2004 to 2006 he was a researcher at the Kitakyushu JGN2 Research Center, National Institute of Information and Communications Technology, Japan. Since April 2006 he has been an assistant professor in the Department of Information and Media Sciences, University of Kitakyushu, Japan. His research interests include performance evaluation of computer networks, mobile networks, and communication protocols. He is a member of the IEEE.

Suguru Yamaguchi received the Master of Engineering degree in computer science from Osaka University, Osaka, Japan, in 1988, received the Doctor of Engineering degree in computer science from Osaka University, Osaka Japan, in 1991. He is a professor of Nara Institute of Science and Technology. He has been also a member of WIDE Project, since its creation in 1988, where he has been conducting research on network security system and other advanced networkings for wide area distributed computing environment.