

Simulation of Network Quality with Flow Based Route Setting

Masashi Hashimoto

Abstract— With the consideration of the different types of Quality of Service (QoS) metrics such as delay time, Traffic Engineering (TE) that can smooth Internet traffic, and OpenFlow which offers flexible programmability, flow based route setting is becoming more important. With those flow control schemes, routes can be set by defining routes explicitly or by referring to QoS metrics of links. In order to evaluate traffic and application response behavior within a network with those flow control schemes, we developed a network simulator on OPNET. We introduce an architectural method that allows extension of an existing simulator to permit measurement of the main function “flow base route set” and application response. The simulator treats a flow in an integrated manner at Layer 4 above Layer 3. Since (1) a route can be explicitly designated for each flow and (2) metrics corresponding to more than one QoS can be defined, the different responses of the functions associated with flows and end-end applications can be evaluated at the same time. In this paper, we describe application functions, basic behavior, and simulation examples.

Index Terms— Network simulator, cost metric, shortest path, traffic engineering, packet simulator.

I. INTRODUCTION

NEW network applications and their traffic, such as broadband broadcasting services via IP phones and streaming, have been increasing. In the Internet, delay, hop counts, or inverse value of link capacity is often used as metric and the shortest route with that single metric are widely used. However, it cannot be enough to meet various QoS requirement and mitigate congestion in the network. So, it is becoming more important to set routes flexibly for each network flow.

One reason for this increase in importance is that the quality requirements placed on networks (QoS: Quality of Service) from IP network applications are becoming more diversified. To improve quality, various controls have been introduced including end-end controls in transport or application layers[1],[2], technology that improves service quality with a priority control mechanism installed in node devices within networks[3],[4], and route setting that uses broadband

networks[5],[6]. Considering that broad bandwidth can be realized relatively easily by implementing optical networks, it is important for each service that has QoS requirements to be supported more flexibly[7][8].

Another reason is that traffic must be accommodated in networks efficiently due to the relentless increase in traffic. The technology of Traffic Engineering (TE) disperses traffic across a network and reduces the convergence rate [9],[10]. When traffic flows and network conditions are given, the best traffic accommodation configuration can be acquired by the Linear Programming (LP) approach [11]. It has been suggested that TE be implemented on the basis of one or more metrics [12],[13], but this raises several issues such as the optimum value cannot be acquired or the situation of equal cost multi-paths (ECMP) may occur. Therefore, other than metrics, to realize the gain expected from TE, routes must be set for each flow individually.

Routes can be set individually by using a source routing method of Layer 3 or on a Label Switched Path (LSP) set by Multi-Protocol Label Switch (MPLS), but this type of route setting requires an advanced infrastructure or special optional functions, so it is not used widely. However, the concept called OpenFlow was introduced recently [14]. Also, routers that support OpenFlow have been released [15],[16]. With OpenFlow, flows can be determined from the header information between Layer 2 to Layer 4 (MAC addresses, IP addresses and port numbers) and transferred without concern for layer structures [17]. That makes it easier to set routes individually.

One of the merits of setting routes for each flow individually is that network managers can achieve smooth traffic while users received good application quality, including improvements in TCP application response time and/or reduction in UDP application delay time.

We considered a network simulator[18] that evaluates application response in a network from the end users' point of view. There are some network simulators that are free and open source such as NS2 [19] and commercial variants exist such as OPNET [20], QualNET [21]. The later commercial software is highly trusted. OPNET covers various networks from high-speed wide network to wireless, while QualNET has an advantage particularly in wireless simulations. All are termed packet simulators. They support the acquisition of end-end response attribute, but they originally do not have a function for setting flows as is consider here. Here, we chose OPNET. It is a commercial product but can be used as open source through a

Manuscript received May 7, 2012.

Masashi Hashimoto is with Dept. of Computer Science and Engineering, Faculty of Engineering and Resource Science, Akita University, Akita, 010-8502 Japan. (e-mail: hashimoto@ie.akita-u.ac.jp).

university program. We developed a function on OPNET to handles these flows. In this paper, we explain the developed application function and its operation and also introduce evaluation examples of user application response attributes.

The main contribution of this paper is to propose and demonstrate an architectural method that allows extension of an existing simulator to permit measurement of the main function “flow base route set” and application response. Specifically, we use the port number, which is the header of Layer 4, to treat flow identification, related to headers in the Layer 4 or lower layers, in an integrated manner. The reason to do so is even if an action that considers multiple layers in open flow is performed, in an actual device image, it is equal to considering that flows are all to be input to either port. [22] When we want to identify a header-specific flow below Layer 3, we setup the port number in Layer 4 for the flow.

We describe simulator requirements in Section II, elements and functions of realized network simulator in Section III, operating output examples in Section IV and our conclusions in Section V.

II. SIMULATOR REQUIREMENTS

A. Identifying Flows

We treat all flows as IP layer traffic without concern for interface differences below Layer 3. Here, identifying flows means classifying a flow in Layer 3 and we identify flows with port numbers in Layer 4. That means all flows are identified by transmitting/receiving IP addresses and transmitting/receiving port numbers (socket numbers).

B. Routing with Multiple Metrics

More than one metric such as delay time, reliability or efficiency can be specified for each link. When multiple metrics are specified, a new single “cost” is created by agglomerating the multiple metric and applying a protocol that uses the compound metric [23]. This allows routing is to be determined with one metric. However, here we assume the use of several costs and refer to them as the “cost set.” One cost set defines values associated with delay time, while another is associated with reliability. Our proposed simulator calculates the shortest route for each cost set and determines the gateway for each cost set at each router, or output port for transmission. The Layer 4 port numbers that identify the flows and the corresponding cost sets to be used will be described. With this approach, the metrics used for each flow can be selected. The structure is that the cost set itself depends on only the port number and does not rely on transmitting/receiving IP address. This structure minimizes the increase in complexity that would otherwise occur with increases in routing table number. If we want to select metrics used for identifying flows that include transmitting/receiving IP addresses, we use values that are different from the Layer 4 port number.

Multiple metrics, used as quantities related to QoS, are also to be used for traffic distribution [13] and [24] and high-speed

switching to spare routes in the event of a fault [25] and [26]. In such events, they can be used to carry out topology changes for route control in the IP layer.

C. Each Route Setting Function

We describe here how to set a route for each flow. We describe the node we want to make the flow go through for a particular route. Information of Layer 3 transmitting/receiving addresses that identify the flow, the port number of Layer 4, and the route that the flow goes through can be set on the simulator. Based on this information, each route transfers the flow specified. Unlike II-B, since a flow is defined to include starting IP address, the starting IP address is also referred when the flow is to be set so it can be said that the source routing path is given explicitly.

D. Service and Traffic Generating Function

We provided the simulator with two traffic generating functions. One is associated with application operation such as TCP and HTTP; it makes it possible to designate port numbers. Protocols and other functions/operations can be used as they are. The other is a function that generates background traffic flowing in the network environment; it generates traffic between designated ports and assigns traffic designated port numbers. This function, makes it easier for example, to compare the performance achieved when routes are determined from metrics to that achieved when individual route setting is used.

E. Others

As for flows that are not subjected to identification as per II-B and II-C, any existing protocol (such as OSPF) can be used. We use port numbers defined from outside. They have a function that sets routes for flows that meet flow identification requirements. However, we use original functions provided by the network simulator to handle regular router, server, client, and existing protocols.

III. MODELING VIA NETWORK SIMULATOR

We create the model mentioned in Section II using the network simulator OPNET. The following are the main specifications.

A. Network Elements

Table I shows network elements. Router uses IP address of input packet to identify other port numbers and arranges transfer by using metrics or individual route setting. Client A represents the end-end applications that designate port numbers across the network. Client B designates port number and defines traffic that flows within the network. When traffic is defined by using two Client B entities, one functions as the source while the other works as the sink. We added a function that combines cyclical traffic changes traffic interaction in a time-varying traffic model. Client A and B are both connected to routers. The server is the same as a regular server and supports all applications; it

can, however, designate port numbers to be used. As for the link, Ethernet and other existing links can be used.

TABLE I NETWORK ELEMENTS.

Elements	Function/Operation
Router	Packet transmission. Transfers by referring to port numbers of input packet.
Client A	Defines end-end applications. Defines CS application as related to server. Port numbers can be designated.
Client B	Generates traffic. Functions as source or sink. Port numbers can be designated.
Server	Operation is same as regular server. Port numbers to be used can be designated.
Link	Regular link. Transmits packets.

B. Functions regarding Routing

Table II shows the basic functions and operation regarding routing. The same as OSPF, it carries out shortest path routing, but more than one cost metric can be defined and a routing table is to generated for each cost metric. Each cost metric corresponds to Layer 4 port number of packet. Input packet uses the routing table of its corresponding cost set. As for explicit routing, node to be passed through can be defined by designating the starting client and ending client and port number. Figure 1 shows example behavior of the Router. In this case, three cost sets are provides corresponding to packet of three port numbers. Route has routing tables generated from cost sets and OSPF protocol.

Table II Basic Network Operations.

Function	Operation
Port number identification	Refers to IP address and input packet port number and identifies flows and adopts transmission routes.
Routing table for multiple cost sets	Defines cost set for link. Defines multiple cost sets. Generates routing tables with shortest routes for each cost set.
Function to corresponds Port number to cost set	Describes correspondence of the cost set to be used with input packet port number. If not defined, given (OSPF) protocol is to be used.
Explicit route defining(source routing)	Designates the transmission route of flow.

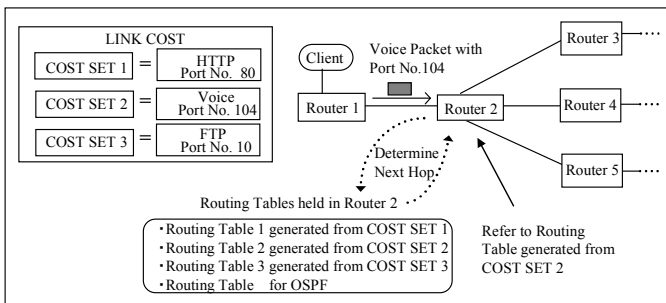


Figure 1 Example Router Behavior under Multiple Cost Sets.

Table III shows the operation for special requirements or irregular setting. We decided to use equal division for equal costs and to re-determine the shortest route after a failure event.

Table III Irregular Network Operations.

Function	Operation
Same cost	Equal division.
Failure event: cost routing	Uses designated cost set and regenerates the shortest route table in the range excluding failure spots. Reconstructs tables while fixing the failure spots.
Failure event: Source routing	Not able to respond. Uses cost or OSPF route when resume after finishing source routing.
Define cost and source routing at the same time.	Source routing comes first.

C. Others

1) Route output function: In order to make sure that the simulator operates properly, we equipped it with a function that acquires and outputs flow routes.

2) Text-based interface: All externally defined in/outputs can be set via text files and we implemented a structure that imports those files within the application. In this way, data can be set easily. Also, they can be used to calculate routes outside of the Linear Programming (LP) calculation. Lists of multiple cost sets, full-mesh traffic and source routing can be defined as text-based files. In LP calculation, topologies consisting of nodes and links are described as data. Thus, topology data are also described as text files, and we developed Network Creator that generates topologies for OPNET from those text files. Figure 2 shows network model generation from text files.

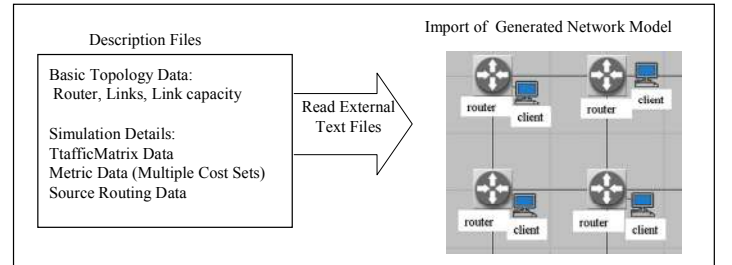


Figure 2 Network Model Generation from Text Files.

IV. SIMULATING OPERATION EXAMPLE

A. Operational Evaluation Requirements for Simulation Simulation Conditions

We set up a different cost set for each port number and check routing operation by selecting a different route and operation that changes application quality in response to route changes. To check such behavior, we considered a cost set of two metrics, the minimum hop count (HOP) and the minimum delay time. As the network topology we used nsfnct [27] shown in Figure 3. It is a model of North America, and each node is an actual city. Since this topology's degree, the ratio between the number of links and the number of nodes, is low

and some distances between nodes with 1 HOP are long, there may be some zones where the minimum HOP number and the minimum delay time are different, and the response speed can be depend on the metric used. As for the delay cost set, we assume that delay time is proportional to distance (actual distance between cities). All links are bidirectional and both directions have the same bandwidth.

As for Client B, we evaluate the simulator's performance on individual route setting. We give full-mesh bidirectional traffic as background traffic, and determine when the traffic is accommodated in the shortest route (as regards delay time) the route that exhibits the TE effect. We evaluate the link usage rate and if the application response improves with the TE effect. Here, we used an even traffic distribution where all full-mesh traffic distributions are the same.

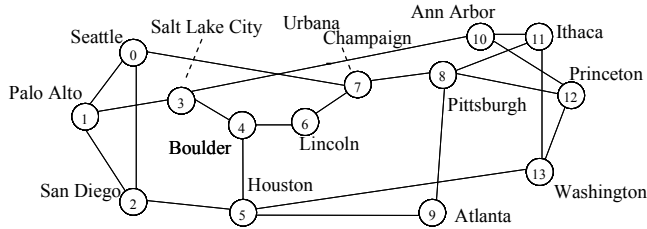


Figure 3 Example network: nsfnet.

B. Basic Output: Topology Image and Route

First, we created text files that described the topology information in Figure 1, and generated topologies with Network Creator. Each router is connected to Client A and B by default. The clients are simply connected to each other and do not generate traffic. Figure 4 is a screen shot of the OPNET-generated topology; it visually represents the network model that connects the routers and Clients A and B. Here, we installed one server for experimentation.

We checked route output operation. Figure 5 shows an example of three HTTP flows. Each port number is different; 80(well known, i.e. OSPF in OPNET), 10011, 10012. 10011 is set to use the number of HOPs as the cost metric and has the same route as that of the well known case. The flow of port number 10012 uses the minimum delay time and has a different route. Two different routes are shown in Figure 6. Here, the simulator output total values of each flow's delay time and the metrics used on the route. Delay time is derived from the physical locations on the network simulator, and the values are different from delay metrics (corresponding distance) defined for the simulator. The upper layer shows route information regarding flows towards the server from the client, while the lower layer shows the opposite flow. By comparing them to the values yielded by external calculations with the designated metrics, we proved that these output routes are the shortest routes.

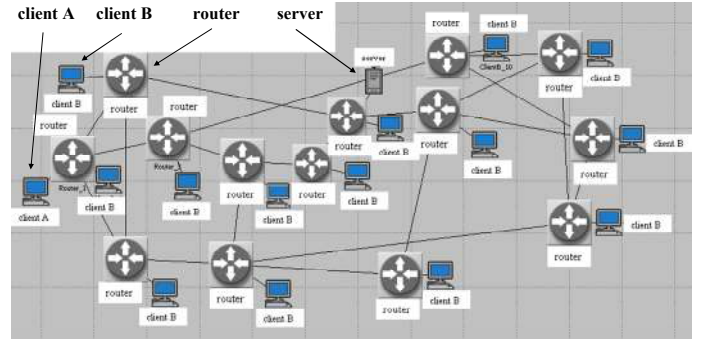


Figure 4 Topology Generated by Network Creator.

		Port No.		Route		Delay		Metric	
Time	Source	Dest	App	Port	Route	Pack	Traffic	Delay	Path Cost
3000	300	405	HTTP	80	1,0,7	102	285600	0.020845	2
3000	300	405	HTTP	10011	1,0,7	66	184800	0.020845	2
3000	300	405	HTTP	10012	1,3,4,6,7	144	403200	0.016015	2976721
3000	405	300	HTTP	80	7,0,1	107	926288	0.021263	2
3000	405	300	HTTP	10011	7,0,1	133	1116712	0.021214	2
3000	405	300	HTTP	10012	7,6,4,3,1	126	1073592	0.016545	2976721

Figure 5 Route Output Screen.

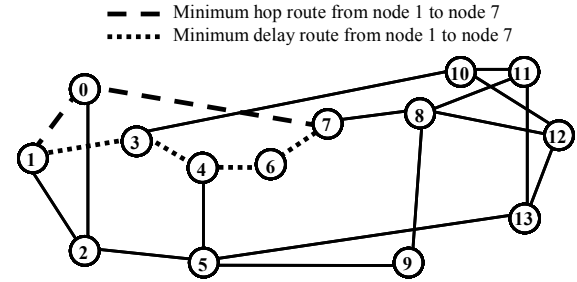


Figure 6 Two Different Routes with Hop and Delay metric.

C. Evaluation of End Application Response Attributes

We evaluated the application response for HTTP and Video as OPNET-standard client-server services. For this evaluation, background traffic is not necessarily required, but we set uniform traffic with the minimum delay metrics. Zones in which services were established, corresponding to the spot shown in the example in IV-B, used different minimum delay route and minimum HOP number route. Using Client A and a server, and designating the port numbers that the application used, we compared the response speed in two situations: the minimum HOP number is used as the metric and the minimum delay time is used as the metric.

1)TCP application: Figure 7 shows an example plot if HTTP response speed. (a) is for the minimum HOP number, while (b) is for the minimum delay time. The horizontal axis is time and application requests are sent regularly. The vertical axis is response time. The figure indicates that (b) is faster. We can see that, when routing is based on the minimum delay time metric, response speed for users is improved.

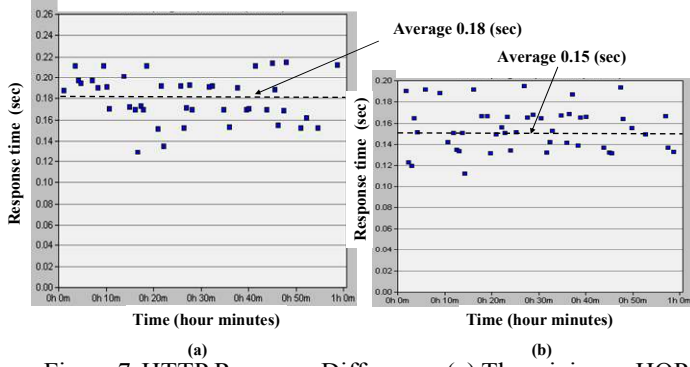


Figure 7 HTTP Response Difference, (a) The minimum HOP number as cost metric, (b) The minimum delay time as cost metric.

2)UDP application: We evaluated the response time of the streaming application when UDP is used. The tendency of delay time itself can be seen in Figure 5, but here, delay time was measured to determine application response. Figure 8 shows typical results. By using the minimum delay time metric, application delay time is decreased and low delay streaming transmission can be realized.

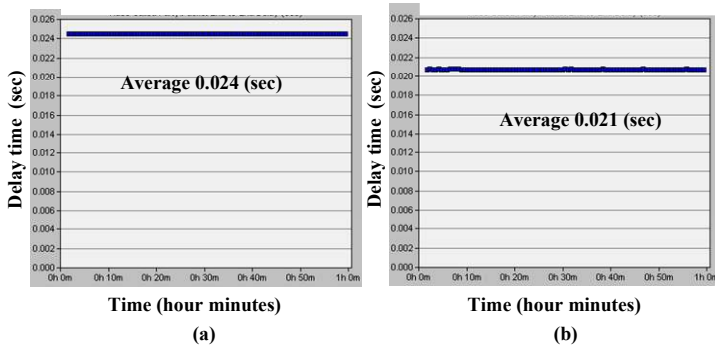


Figure 8 UDP Response Attribute Difference, (a) The minimum HOP number as cost metric, (b) The minimum delay time as cost metric.

3)TE operation with individual route setting: We examined the effect of using TE (individual route setting) in a comparison against minimum delay time metric based routing; the background traffic of IV-A is used as the shortest route and HTTP was taken as the application. The application response times were measured, see Figure 9 (a) and (b). (b), which exhibits smoothed traffic, has faster response. Figure 6 (c) plots the converged rate, the maximum link usage rate within the network. When background traffic is accommodated using the minimum delay time metric, the converged rate is 80%. With TE smoothing, this rate is reduced to 57%. The remaining bandwidth, which is available for other applications, more than doubles. This indicates the value of TE.

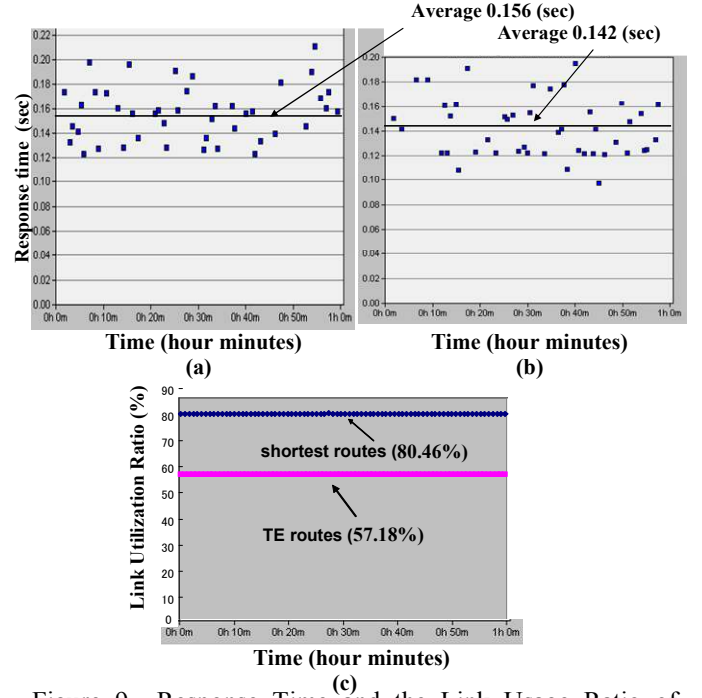


Figure 9 Response Time and the Link Usage Ratio of Individual Route Setting with TE, (a) HTTP response: when the shortest route is set, (b) same: TE used, (c) the maximum link usage ratio (the converged ratio).

V. CONCLUSION

We developed a network simulator that simulates flow behavior and end-end application responses by defining flow routes with a function that sets metrics corresponding to more than one QoS metric and a function that allows routes to be explicitly specified. We proposed and demonstrated a method for modifying the architecture of a simulator so that it can measure the main function of “flow base route set” and application response; our proposal avoids the need to implement a completely new architecture, a requirement imposed by Open flow and multi-metric Networks [7]. We created a model for OPNET and evaluated route setting, metric reference operation, and the consequent impact on application response. We demonstrated that the simulator is a useful tool for evaluating network operation including end-to-end application response with flow-based route setting. While multi-metric networks have been specified by RFC, they have not been deployed yet. However, they may be realized in the field of network virtualization [28], which has been attracting attention recently. For example, they can be realized by relating each of the multiple topologies to a different slice of a virtual network and defining one metric for each slice. This approach is beneficial for network evaluation.

REFERENCES

- [1] J. Postel, IETF “RFC793 TCP”(1981).

- [2] S. Floyd and K. Fall, "Promoting the use of end-to-end congestion control in the Internet", *IEEE/ACM Trans. Networking*, 7(4):458-472, Aug., 1999.
- [3] R. Braden, D. Clark, S. Shenker, IETF "RFC1633 Integrated Services in the Internet Architecture: An Overview,"(1994).
- [4] S. Blake et al., IETF "RFC2475 An Architecture for Differentiated Service"(1988).
- [5] R. Stewart and C. Metz, "SCTP: new transport protocol for TCP/IP," *IEEE Internet Computing*, pp.64-69, 2001.
- [6] M. Hashimoto, T. Murooka, and T. Miyazaki, "A multihoming application flow network using "A-BOX": a fast packet processing node featuring direct traffic monitoring," *IEEE ICC2004*, Vol.2 HS07-7, pp.1253-1257, 2004.
- [7] P. Psanal, S. Mirtorabi, A. Roy, L. Nguyen, and P. Pillay-Esnault, "MT-OSPF: Multi topology (MT) routing in OSPF," IETF, RFC4915, 2007.
- [8] M. Hashimoto, A. Fukuda, and K. Yukimatsu, "Novel Traffic Engineering Scheme Based upon Application Flows for QoS Enhancement", *IT-NEWS(QoS-IP)2008*, pp.167-172, 2008.
- [9] J. Moy, "OSPF Version 2," IETF RFC2328, 1998.
- [10] D. Awduche, J. Malcolm, J. Agogbua, M. O'Dell, and J. McManus, "Requirements for Traffic Engineering Over MPLS," IETF, RFC2702, 1999.
- [11] Y. Wang and Z. Wang, "Explicit Routing Algorithms for Internet Traffic Engineering," *IEEE ICCCN 1999*, pp.582 - 588, 1999.
- [12] Y. Wang, Z. Wang, and L. Zhang, "Internet Traffic Engineering without Full Mesh Overlaying," *IEEE INFOCOM 2001*, pp.565 - 571, 2001.
- [13] B. Fortz and M. Thorup, "Optimizing OSPF/IS-IS Weights in a changing World," *IEEE J. Select. Areas Commun.*, vol.20, No.4, pp.756-767, 2002.
- [14] Open Networking Foundation. (accessed May 2012) [Online]. Available: <https://www.opennetworking.org/>
- [15] Open Networking Foundation. (accessed May 2012) [Online]. Available: <http://www.openflow.org/wp/tag/cisco/>
- [16] Open Networking Foundation. (accessed May 2012) [Online]. Available: <http://www.openflow.org/wp/switch-nec>.
- [17] Open Networking Foundation. OpenFlow White Paper. (accessed May 2012) [Online]. Available: <http://www.openflow.org/wp/documents/>
- [18] Jianli Pan, A Survey of Network Simulation Tools: Current Status and Future Developments (accessed May 2012) [Online]. Available: <http://www.cse.wustl.edu/~jain/cse567-08/ftp/simtools/>
- [19] The Network Simulator – ns. (accessed May 2012) [Online]. Available: http://nslam.isi.edu/nslam/index.php/Main_Page/
- [20] OPNET Technologies. OPNET. (accessed May 2012) [Online]. Available: <http://www.opnet.com/>
- [21] SCALABLE Network Technologies, Inc. QualNet. (accessed May 2012) [Online]. Available: <http://www.qualnet.com/content/products/qualnet/>
- [22] WIKIPEDIA. Enhanced Interior Gateway Routing Protocol. (accessed May 2012) [Online]. Available: <http://wikipedia.org/wiki/EIGRP/>
- [23] L. Liu, T. Tsuritani, I. Morita, G. Hongxiang, and J. Wu, "OpenFlow-based wavelength path control in transparent optical networks: A proof-of-concept demonstration," *Optical Communication (ECOC)*, 2011 37th European Conference, Tu.5.K.2, pp.1-3, Sept, 2011.
- [24] M. Hashimoto, T. Ishizaki, "Multi-Topology Traffic Engineering with Metrics not based on ECMP," *Cyber Journals: Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Telecommunications (JSAT)*, pp.26-33, Apr. 2012.
- [25] A. Kvalbein, A.F. Hansen, T. Cicic, C. Gjessing, and O. Lysne, "Multiple Routing Configurations for Fast IP Network Recovery," *IEEE/ACM Transactions on Networking*, Vol.17, No. 2, pp. 473 - 486, Apr., 2009.
- [26] S. Kamamura, T. Miyamura, Y. Uematsu, and K. Shiimoto, "Scalable Backup Configurations Creation for IP Fast Reroute," *IEICE Trans. Commun*, vol.E94-B, No.1, pp.109-117, Jan., 2011.
- [27] X. Niu, W.-D. Zhong, G. Shen, and T. H. Cheng, "Connection establishment of label switched paths in IP/MPLS over optical networks," *Photonic Network Communications*, vol. 6, pp. 33-41, 2003.
- [28] A. Takahara, "The Future Carrier Networks: Its Vision and Architecture," *IEICE Trans. COMMUN.*, Vol E93-B, No.3 pp.450-453, 2010.

from Kyoto University, Kyoto, Japan in 1984, respectively. He joined Nippon Telegram and Telephone Public Corporation (presently, NTT), 1984. From then, he has been engaged in the study and development of optical subscriber transmission system, and conducted research in optical parallel signal processing from 1987. He received a doctoral degree from Waseda University in 1995. He joined the Akita University, Akita, Japan, in April 2005. His present research interests involve traffic engineering and application flow control which suits to current and future Internet age. He is a member of IEICE, IPSJ of Japan and IEEE.

Masashi Hashimoto is an Associate Professor at Akita University, Akita, Japan. He received the B.E. degree in electrical engineering from Waseda University, Tokyo, Japan in 1982 and M.E. degree in electrical engineering