An Inter-Domain Path Computation Scheme Adaptive to Traffic Load in Domains

Nagao OGINO\(^{1a}\) and Hajime NAKAMURA\(^{1}\), Members

SUMMARY The establishment of inter-domain traffic engineered paths is a requisite to accomplishing an end-to-end bandwidth guarantee and end-to-end resource optimization. Though the inter-domain paths must be reliable, it is difficult to compute suitable backup inter-domain paths in advance when the traffic engineering information is not disclosed outside of each domain. This means that the inter-domain path computation must satisfy the severe requirement of path establishment delay, since all inter-domain paths traversing the links in failure need to be computed after the failure occurs. Though several inter-domain path computation schemes have been proposed, their relative characteristics remain unknown. First, this paper classifies the conventional inter-domain path computation schemes into two types, i.e. end-to-end and per-domain schemes, and compares their performances under various traffic loads. Based on results of the comparisons, this paper proposes an adaptive inter-domain path computation scheme that can satisfy the severe requirement of the path establishment delay. In this scheme, the domain sequence from the source node to the destination node is divided into multiple sub-domain sequences according to the traffic load in each domain. The end-to-end path computation scheme is applied to the sub-domain sequences under heavy traffic loads, while the per-domain path computation scheme is applied to those under normal traffic loads. The simulation results show that the proposed scheme can adaptively satisfy the requirement for the path establishment delay while it maintains the optimality of path computation, even if the traffic load applied to each domain changes.

key words: inter-domain path computation, adaptation to traffic load, path establishment delay, optimality of path computation, computer simulation

1. Introduction

The establishment of inter-domain traffic engineered paths is a requisite to accomplishing an end-to-end bandwidth guarantee and end-to-end resource optimization [1]–[3]. Signaling protocol and routing protocol extensions to establish inter-domain label-switched paths have been standardized in the IETF [4]–[6]. Since inter-domain path computation is complex and the node responsible for the path computation may have limited visibility in the multi-domain network, PCE-based (path computation element-based) architecture and related protocols are being standardized [7]–[11]. In the PCE-based architecture, each domain only includes the PCE responsible for path computation within the domain. The inter-domain path computation can be achieved through cooperation between multiple PCEs. When the inter-domain corresponds to the inter-provider, PCE in each domain can also hide the results of its path computation from the other domains [12].

The inter-domain paths must be reliable, and a backup inter-domain path is necessary to recover a working inter-domain path from any failure [13]–[15]. Considering single SRLG (shared risk link group) failure, the backup inter-domain path should be SRLG-disjoint with the protected working inter-domain path. The simultaneous computation of the working and backup inter-domain paths is realizable if each SRLG only includes links in the same domain [16]. However, it is difficult when an SRLG includes links in different domains, and the SRLG information in each domain is not disclosed from a security perspective. In this case, the backup inter-domain path that is SRLG-disjoint to the protected working inter-domain path can only be computed after the working path actually has a failure. This means that all inter-domain paths traversing the links in failure must be recomputed and reestablished avoiding the links in failure when an SRLG failure occurs. Thus, the inter-domain path computation should be sufficiently fast and must satisfy the severe requirement of the path establishment delay.

Though several inter-domain path computation schemes have been proposed [17]–[19], their relative characteristics remain unknown. Network operators must select appropriate schemes that satisfy their requirements of the inter-domain path establishment delay, according to the traffic loads applied to their networks. This paper aims to resolve this problem in network operation. First, this paper classifies inter-domain path computation schemes into two types, i.e. end-to-end and per-domain schemes, and compares their performances under various traffic loads. Based on results of the comparisons, this paper proposes an inter-domain path computation scheme that adapts to the traffic load in each domain. In this scheme, the domain sequence from the source node to the destination node is divided into multiple sub-domain sequences according to the traffic load in each domain. The end-to-end path computation scheme is applied to the sub-domain sequences under heavy traffic loads, while the per-domain path computation scheme is applied to those under normal traffic loads. By adopting the proposed scheme, the requirement for the path establishment delay can be satisfied while the optimality of path computation is maintained, even if the traffic load applied to each domain changes.

Section 2 of this paper proposes an adaptive inter-domain path computation scheme following an explanation of the conventional schemes. Section 3 evaluates the effectiveness of the proposed inter-domain path computation scheme through a computer simulation. Finally, Sect. 4 con-
2. Inter-Domain Path Computation Schemes

In this paper, a domain refers to a collection of network elements within path computation responsibility. A domain can be an IGP (interior gateway protocol) area, an AS (autonomous system), or a provider network. Inter-domain advertisements of traffic engineering information face problems of scalability. Moreover, inter-domain advertisements are also problematic from a security perspective, when the inter-domain corresponds to the inter-provider. Thus, each domain includes a PCE (path computation element), and each PCE is only responsible for path computation within the corresponding domain [7]–[11]. Each PCE gathers the traffic engineering information using the IGP-TE (IGP-traffic engineering) [20], [21] or from the management system within the corresponding domain.

2.1 Conventional Inter-Domain Path Computation Schemes

First, the sequence of domains that the inter-domain path should traverse is determined based on the policy in the source node. Inter-domain path computation can be performed either before or during the establishment of inter-domain paths. In the former case, the source node obtains information concerning the appropriate inter-domain route in advance and specifies that inter-domain route explicitly within the signaling message for path establishment. In the latter case, the source node only specifies the sequence of domains that the inter-domain path should traverse within the signaling message. The computation of each segment of the inter-domain route is triggered by the signaling message and performed within each related domain. In this paper, inter-domain path computation schemes corresponding to the former case are called an end-to-end scheme and those corresponding to the latter case are called a per-domain scheme.

2.1.1 End-to-End Path Computation Scheme

Figure 1(a) is a conceptual diagram illustrating the end-to-end scheme. Figure 1(a) also shows the sequence of path computation request/reply messages and signaling messages in the end-to-end scheme [6], [9]. The source node first requests end-to-end path computation to the PCE within the same domain prior to path establishment. The PCE receiving the path computation request cooperates with multiple PCEs corresponding to the domain sequence specified by the source node, and computes the optimum route toward the destination node. Each PCE on the domain sequence executes the inter-domain path computation according to the request from either the source node or the neighboring upstream PCE.

The backward recursive procedure is proposed as an algorithm to compute the optimum inter-domain route [18].

This algorithm is based on an extension of the distributed Bellman-Ford algorithm [22]. Each PCE computes the optimum idle routes from the source node toward the respective ingress border nodes within the next domain, and notifies the next PCE of the computed results. For example, the PCE-1 computes the optimum idle routes from the source node S toward the ingress border nodes Bi21 through Bi2n. The PCE-1 holds the optimum routes information and only reports the costs of the optimum routes using the path computation request message sent to the PCE-2. The PCE-2 computes the route segments composing the optimum idle inter-domain routes from the source node S toward the ingress border nodes Bi31 through Bi3n based on the reported route costs. The PCE-2 holds the route segments information and only reports the costs of the optimum routes using the path computation request message sent to the next PCE.

Thus, the idle inter-domain route with minimum cost is recursively computed from the source node toward the destination node. Once the optimum idle inter-domain route toward the destination node is computed, each PCE notifies an ingress border node on the optimum inter-domain route using the path computation response message. Each PCE can select one of the route segments composing the optimum inter-domain route, based on the reported ingress border node information. When the path computation succeeds, the inter-domain path can be established with certainty by means of this scheme.
### 2.1.2 Per-Domain Path Computation Scheme

Figure 1(b) is a conceptual diagram illustrating the per-domain scheme. Figure 1(b) also shows an example of the sequence of path computation request/reply messages and signaling messages in the per-domain scheme [6],[9]. The source node only requests intra-domain path computation to the PCE within the same domain and starts establishing the inter-domain path with a signaling message that specifies the sequence of domains that the inter-domain path should traverse. When each ingress border node receives the signaling message, it requests intra-domain path computation to the PCE within the same domain. Each PCE computes the optimum idle route toward the ingress border nodes in the next domain.

Based on the result of path computation, the inter-domain path is established toward an ingress border node in the next domain. If it is known that no idle route exists within the next domain, the inter-domain path is re-established toward another ingress border node. This process is called crank-back [23]. The loss rate of path requests can be reduced by executing crank-back. Subsequently, the newly selected ingress border node requests intra-domain path computation to the PCE within the next domain. In the per-domain scheme, the end-to-end optimality of the inter-domain path is not guaranteed, while the optimum idle route is independently computed within each domain.

In Fig. 1(b), the source node S requests the intra-domain path computation to the PCE-1. The PCE-1 computes the optimum idle intra-domain routes from the source node S to the respective ingress border nodes Bi21 through Bi2n. The source node S selects and establishes the intra-domain path toward the ingress border node Bi2n. Then, the ingress border node Bi2n requests the intra-domain path computation to the PCE-2. However, it is known in the PCE-2 that no idle route exists from the ingress border node Bi2n, and the intra-domain path toward the ingress border node Bi2n is released. Next, the source node S establishes the intra-domain path toward the ingress border node Bi21, and the ingress border node Bi21 requests the intra-domain path computation to the PCE-2. Now, an intra-domain path from the ingress border node Bi21 is successfully established within the Domain-2.

### 2.2 Proposed Inter-Domain Path Computation Scheme

An inter-domain path computation scheme adaptive to the traffic load in each domain is proposed. The proposed scheme also assumes that the sequence of domains is given in advance according to the policy in the source node. In this scheme, the domain sequence from the source node to the destination node is divided into multiple sub-domain sequences according to the traffic load in each domain. The appropriate path computation scheme is applied to each sub-domain sequence based on its traffic load.

As shown in Fig. 2, the traffic load in each domain is measured in the PCE within the domain. For example, the approximated traffic load can be deduced from the arrival rate of path computation requests for the PCE. When each PCE is stateful, the approximated traffic load can be deduced from the number of inter-domain paths accommodated in each domain [10]. Each PCE notifies all the neighboring PCEs of the measurement results of the traffic load. The fluctuation in the traffic load applied to each domain is supposed to be sufficiently slow compared to the delay time for reporting the measured traffic load. Thus, each PCE can correctly recognize the traffic load in the next downstream domain when the PCE receives the inter-domain path computation request. Therefore, each PCE can judge the appropriate path computation scheme for the next downstream domain based on the reported traffic load in the next domain. If the appropriate scheme for the next domain is different from that in the considered domain, the sub-domain sequence terminates in the considered domain.

When the traffic load is heavy, the non-existence of an idle inter-domain route can be confirmed prior to path establishment in the end-to-end scheme. However, the required number of crank-backs rapidly increases under heavy traffic loads in the per-domain scheme, even though this scheme only requires simple intra-domain path computation under normal traffic loads. To satisfy the requirement for the path establishment delay, application of the per-domain scheme is proposed for the sub-domain sequence under normal traffic loads, while the end-to-end scheme is proposed for the sub-domain sequence under heavy traffic loads. The per-domain scheme can be applied to each domain involved in the sub-domain sequence in the usual manner. However, the end-to-end scheme applied to the sub-domain sequence needs to compute the optimum route segments toward multiple ingress border nodes within the succeeding domain. This enables crank-back within the sub-domain sequence to which the end-to-end scheme is applied.
1 through \(m\), \(m + 1\) through \(n\), and \(n + 1\) through \(N\) respectively. In the proposed path computation scheme, the end-to-end scheme is applied to the first and the last sub-domain sequences, while the per-domain scheme is applied to each domain involved in the middle sub-domain sequence.

When crank-back is performed in a sub-domain sequence to which the end-to-end scheme is applied, the inter-domain path needs to be re-established between multiple domains. Aiming to reduce the number of domains where the inter-domain path is re-established, the computed route segments should include a common route that is as long as possible from the starting ingress border node, though it slightly sacrifices the optimality of route segments. For example, crank-back is required in the first sub-domain sequence if no idle route can be found within the domain \(m + 1\) in Fig. 3. Then, the end-to-end scheme applied to the first sub-domain sequence should compute the route segments with a common route that is as long as possible from the source node \(S\). In Fig. 3, the computed route segments include a common route from the source node toward the ingress border node within the domain \(m\). This ingress border node within the domain \(m\) may be selected based on its route cost from the source node or the number of ingress border nodes involved in the domain \(m + 1\) and toward which an idle route exists.

3. Performance Evaluation of Inter-Domain Path Computation Schemes

In this section, the effectiveness of the proposed inter-domain path computation scheme is verified through an exhaustive computer simulation.

3.1 Evaluated Multi-Domain Network

Figure 4 shows the evaluated multi-domain network, which consists of \(N\) domains. The first domain includes 5 source nodes and 5 egress border nodes, while the last domain includes 5 ingress border nodes and 5 destination nodes. Each transit domain includes 5 ingress border nodes and 5 egress border nodes. All the nodes within a domain are inter-connected by a bidirectional intra-domain route, all of which are assumed to share no sub-route. A pair of egress and ingress border nodes is connected by a unidirectional inter-domain link. An intra-domain route can accommodate 10 inter-domain paths per a direction, while its cost is uniformly distributed between 0.0 and 100.0. An inter-domain link can accommodate 50 inter-domain paths, while its cost can be regarded as 0.0 since all the inter-domain paths inevitably traverse \(N - 1\) inter-domain links.

In this paper, the loss rate of path requests, the optimality of path computation, and the path establishment delay have been selected to evaluate the service level performances of the inter-domain path computation schemes. The purpose of inter-domain path computation involves seeking the minimum cost inter-domain route with sufficient idle bandwidth. Therefore, the optimality of path computation indicates the costs of the computed inter-domain paths. The path establishment delay is important when many inter-domain path requests occur simultaneously due to restoration from a link failure. When a link failure occurs, all inter-domain paths traversing the failed link must be recomputed and reestablished instantaneously.

These performances are evaluated and averaged in terms of 10 different multi-domain networks with a given number of domains and varying intra-domain route costs. The arrival rate of path requests is kept identical among all the pairs of source and destination nodes. Each performance is evaluated as the arrival rate of path requests between a pair of source and destination nodes \((A)\) changes. The delay time for notifying the neighboring PCEs of the traffic load is disregarded, assuming that this delay time is sufficiently small compared to the variation speed in the arrival rate of path requests. The arrival process of path requests is random, and the duration of the established paths follows a negative exponential distribution, with an average value of 1.0. The simulation time is set sufficiently long so that each evaluated value has a 95% confidence interval whose size is less than 1% of the entire value.

3.2 Evaluation of Conventional Schemes

As a preliminary, the conventional end-to-end and per-domain schemes are evaluated. The per-domain scheme is evaluated by varying the permitted number of crank-backs per domain \((Cb)\). Figure 5(a) shows the loss rate of path requests when the number of domains \(N\) changes. As shown in Fig. 5(a), the loss rate in the per-domain scheme without crank-backs \((Cb = 0)\) is considerably large. However, the loss rate in the per-domain scheme with an unrestricted number of permitted crank-backs \((Cb = 4)\) is almost identi-
cal to that in the end-to-end scheme. When an idle route exists in a given domain sequence, that idle route is almost certainly obtained in the per-domain scheme with unrestricted crank-backs. As shown in Fig. 5(a), the loss rate of path requests increases as the number of traversed domains increases. This tendency is remarkable in the per-domain schemes with a restricted number of crank-backs.

Figure 5(b) shows the average cost of established paths, i.e., the optimality of path computation schemes, when the number of domains \( N \) changes. As shown in Fig. 5(b), the average cost of the established paths increases in proportion to the number of traversed domains in both the schemes. The average path costs in the end-to-end scheme are always less than the average cost of inter-domain paths that only traverse a single intra-domain route at each domain \((= 50.0 \times N)\). This indicates that an idle intra-domain route that transits the other border nodes from an ingress border node toward an egress border node is hardly ever selected because each ingress border node within a domain is selected equally in the end-to-end scheme.

The average path cost in the per-domain scheme increases considerably compared with that in the end-to-end scheme. Particularly, the increase in the average cost cannot be ignored when the applied traffic load is large. The average cost exceeds the average cost of inter-domain paths that only traverse a single intra-domain route at each domain \((= 50.0 \times N)\), indicating that an alternative intra-domain route that transits the other border nodes is often selected. This is because only a particular ingress border node tends to be selected within a domain and thus the intra-domain routes directly connecting with that selected ingress border node and the egress border nodes rapidly become busy.

Figure 5(c) shows the delay of inter-domain path establishment when the number of domains \( N \) changes. The path establishment delay consists of the path computation time and the message transfer delay. Two types of messages are considered, namely, the path computation request-reply messages and the signaling messages to establish and release the paths. The vertical axis in Fig. 5(c) indicates the delay time when the path computation time to seek the idle minimum cost route between a pair of ingress and egress border nodes within a domain is presumed to be 1.0 msec. The transfer delay of intra-domain path computation request or reply messages is assumed to be 1.0 msec, and that of inter-domain path computation request or reply messages is assumed to be 5.0 msec. The transfer delay of signaling messages to establish or release the path segments within a domain is presumed to be 10.0 msec.

When the loss rate of path requests can be disregarded, the path computation time (msec) and the message transfer delay (msec) in the end-to-end scheme are given by \( 5.0 + 25.0 \times (N-2) + 5.0 \) and \( 2.0 + 10.0 \times (N-1) + 20.0 \times N \), respectively. In the end-to-end scheme, the path computation finishes immediately and no path establishment is performed when no idle route can be found within the transit domain. Therefore, the path establishment delay is reduced due to the increase in the applied traffic load, since the loss
rate of path requests increases.

When the applied traffic load is normal, the path computation time in the per-domain scheme is small because only the route segments from the selected ingress border nodes are sought within each domain. The message transfer delay in the per-domain scheme is also small since interdomain transfer of path computation request/reply messages is unnecessary. If no crank-back is required, the path computation time (msec) and the message transfer delay (msec) in the per-domain scheme are given by $5.0 \times (N - 1) + 1.0$ and $2.0 \times N + 20.0 \times N$, respectively. As the applied traffic load increases, more crank-backs are required in the per-domain scheme to maintain the loss rate of path requests at almost the same level as in the end-to-end scheme. Therefore, both the path computation time and the message transfer delay increase as the applied traffic load increases. In the per-domain scheme, path establishment delay increases due to the increase in the applied traffic load.

In conclusion, the end-to-end scheme is suitable from the viewpoint of both optimality and path establishment delay under a heavy traffic load. In contrast, the per-domain scheme is more advantageous from the viewpoint of path establishment delay when the applied traffic load is normal.

3.3 Evaluation of Proposed Scheme

The proposed inter-domain path computation scheme is evaluated in terms of the multi-domain network shown in Fig. 4. In this subsection, the number of domains $N$ is fixed at 6 and the domain sequence is divided into two sub-domain sequences each of which is composed of three domains. In the simulation, the path requests for the inter-domain paths traversing only one of the two sub-domain sequences also arrive in the multi-domain network. In this case, the egress border nodes within Domain-3 become the destination nodes, or the ingress border nodes within Domain-4 become the source nodes. The arrival rates of path requests for both the inter-domain paths traversing only a sub-domain sequence and those traversing both the sub-domain sequences are adjusted, such that the arrival rate of path requests in either of the two sub-domain sequences can be kept constant. Performances of the proposed scheme are evaluated in terms of only the inter-domain paths traversing both the sub-domain sequences.

3.3.1 Comparison with End-to-End Scheme

Figure 6 compares the proposed inter-domain path computation scheme to the conventional end-to-end scheme. As shown in Fig. 6(a), the arrival rate of path requests in either of the two sub-domain sequences is kept at 6.0 and that in the other sub-domain sequence varies. The horizontal axis in Figs. 6(b) and 6(c) indicates this varying arrival rate of path requests between a pair of source and destination nodes ($A$). As shown in Fig. 6(a), the combined scheme indicates that the end-to-end scheme is applied to the sub-domain sequence with a constant arrival rate of path requests and the per-domain scheme is applied to the sub-domain sequence with a varying arrival rate of path requests. In contrast, the end-to-end scheme and the per-domain scheme indicate that the end-to-end and per-domain schemes are applied to all the domains. The loss rate of path requests becomes almost identical between the three schemes.

Figure 6(b) shows the average cost of established paths in the three schemes. The average cost in the combined scheme is larger than that in the end-to-end scheme. However, sufficient optimality of path computation can be ob-
tained by the combined scheme when the varying arrival rate of path requests is not so large. This means that the per-domain scheme can be applied to the sub-domain sequence under normal traffic load. When the varying arrival rate of path requests increases considerably, degradation in the optimality of path computation can be avoided by adopting the end-to-end scheme in all the domains.

Comparing the two cases of the combined scheme where the end-to-end scheme is applied to Domains-1 through -3 and Domains-4 through -6, the average cost becomes smaller in the former case. This is because the effectiveness of the end-to-end scheme is far greater in the first sub-domain sequence where the optimum idle route from the source node toward one of multiple ingress border nodes within Domain-4 is sought.

Figure 6(c) shows the path establishment delay in the three schemes. Because the arrival rate of path requests in one of the sub-domain sequences is kept large, the path establishment delay in the combined scheme is always smaller than that in the per-domain scheme. When the varying arrival rate of path requests \( A \) is considerably large, the path establishment delay in the end-to-end scheme becomes the smallest. However, the path establishment delay in the end-to-end scheme increases as the varying traffic load decreases. In contrast, the path establishment delay in the combined scheme is reduced because the per-domain scheme is adopted in the sub-domain sequence to which the varying traffic load is applied.

Comparing the two cases of the combined scheme, the path establishment delay becomes smaller when the varying traffic load is relatively small and applied to the second sub-domain sequence. In this case, only a small number of crank-backs results from the varying light traffic load applied to the second sub-domain sequence. In contrast, the path establishment delay becomes larger when the varying traffic load is relatively large and applied to the second sub-domain sequence. This is because the required number of crank-backs increases due to the varying heavy traffic load applied to the second sub-domain sequence. When a constant heavy traffic load is applied to the second sub-domain sequence, the increase in the required number of crank-backs can be reduced owing to the end-to-end scheme adopted in the second sub-domain sequence.

It is difficult to know the actual requirement for the path establishment delay, since it is determined by the network operator. The end-to-end scheme is desirable from the viewpoint of path optimality as long as the requirement for the path establishment delay is satisfied. This means that the end-to-end scheme is always desirable when the requirement for the path establishment delay is not severe. For Fig. 6, the end-to-end scheme is always advantageous if the requirement for the path establishment delay is less than 300 msec. However, the end-to-end scheme cannot satisfy the severe requirement for the path establishment delay as the applied traffic load decreases. The per-domain scheme should be adopted in the sub-domain sequence to which the normal traffic load is applied. If the path establishment delay is required to be less than 250 msec, this requirement can be satisfied and the path computation optimality can also be maintained by the combined scheme when the varying arrival rate of path requests \( A \) is smaller than about 5.5 in the case of Fig. 6.

The proposed scheme corresponds to the end-to-end scheme when a heavy traffic load is applied to all the domains. In contrast, the proposed scheme is identical to the combined scheme when the traffic load applied to a part of the domains is normal. Thus, the proposed scheme can adaptively satisfy the severe requirement for the path establishment delay when the traffic load applied to each domain changes. When the requirement for the path establishment delay is less than 250 msec, the proposed scheme can satisfy the requirement by adopting the end-to-end scheme in the sub-domain sequence where the arrival rate of path requests is larger than about 5.5 and adopting the per-domain scheme in the sub-domain sequence where the arrival rate of path requests is smaller than about 5.5.

3.3.2 Comparison with Per-Domain Scheme

Figure 7 compares the proposed inter-domain path computation scheme to the conventional per-domain scheme. As shown in Fig. 7(a), the arrival rate of path requests in either of the two sub-domain sequences is kept at 4.0 and that in the other sub-domain sequence varies. The horizontal axis in Figs. 7(b) and 7(c) indicates this varying arrival rate of path requests between a pair of source and destination nodes \( A \). As shown in Fig. 7(a), the combined scheme indicates that the per-domain scheme is applied to the sub-domain sequence with a constant arrival rate of path requests and the end-to-end scheme is applied to the sub-domain sequence with a varying arrival rate of path requests. In contrast, the end-to-end scheme and the per-domain scheme indicate that the end-to-end and per-domain schemes are applied to all the domains respectively. The loss rate of path requests becomes almost identical between the three schemes.

Figure 7(b) shows the average cost of established paths in the three schemes. The average cost in the combined scheme is larger than in the end-to-end scheme and smaller than in the per-domain scheme, since the combined scheme is a combination of both the end-to-end and per-domain schemes. As shown in Fig. 7(b), optimality in the per-domain scheme is remarkably reduced when the varying arrival rate of path requests \( A \) increases considerably. Therefore, the end-to-end scheme is hopefully applied to the sub-domain sequence under heavy traffic load as in the combined scheme. Degradation in the optimality of path computation can be avoided by adopting the combined scheme.

Comparing two cases of the combined scheme where the end-to-end scheme is applied to Domains-1 through -3 and Domains-4 through -6 respectively, the average cost becomes smaller in the former case. This result is attributable to the same reason shown in Fig. 6(b). Comparing the two cases of the per-domain scheme, the average cost becomes larger when the traffic load in the second sub-domain se-
This means that traffic load in the second sub-domain sequence has a greater influence on the increase in the average cost, compared with that in the first sub-domain sequence. This is because the required number of crank-backs increases as the traffic load in the succeeding sub-domain increases.

In both the combined and per-domain schemes, the path establishment delay becomes larger if traffic load in the second sub-domain sequence is larger than that in the first sub-domain sequence. This means that traffic load in the second sub-domain sequence has a greater influence on the increase in the path establishment delay, compared with that in the first sub-domain sequence. This is because the required number of crank-backs depends primarily on the traffic load in the succeeding sub-domain sequence or domain.

In conclusion, the proposed scheme can satisfy the requirement for the path establishment delay while maintaining the optimality of path computation, owing to adoption of the end-to-end scheme in the domains where the applied traffic load exceeds the threshold given in advance and adoption of the per-domain scheme in the domains where the applied traffic load is less than the threshold. The precise value of threshold depends on the structure of networks and the arrival pattern of path requests. If the requirement for the path establishment delay is less than 250 msec, threshold on the arrival rate of path requests should be set about 5.5 in the cases of Figs. 6 and 7.

4. Conclusions

Network operators must select appropriate inter-domain path computation schemes to satisfy the severe requirements for the path establishment delay, even if the traffic load applied to each domain changes. This paper classified
inter-domain path computation schemes into end-to-end and per-domain schemes and compared their performances under various traffic loads. Based on the results of comparison, this paper proposed an inter-domain path computation scheme adaptive to the traffic load in each domain. In this scheme, the end-to-end path computation scheme is applied to the domains under heavy traffic load more than a threshold given in advance, while the per-domain path computation scheme is applied to those under normal traffic load less than the threshold. The service performances in the proposed scheme were evaluated through an exhaustive computer simulation. The results of evaluation revealed that the proposed scheme can adaptively satisfy the requirement for the path establishment delay while maintaining the optimality of path computation, even when the traffic load applied to each domain changes.

Acknowledgments

The authors would like to thank Dr. Akiba, President & CEO, and Dr. Suzuki, Executive Director, of KDDI R&D Laboratories Inc., for their encouragement throughout the study.

References


Nagao Ogino received his B.E., M.E., and Dr.E. degrees from the University of Tokyo, in 1977, 1979 and 1982, respectively. He joined the Research and Development Laboratories of Kokusai Denshin Denwa Co. Ltd. (currently KDDI) in 1982, and has been engaged in research on ATM networks, Intelligent Networks, and telecommunication software engineering. He was a supervisor at ATR Adaptive Communications Research Laboratories from 1996 until 2000, and was engaged in research on multi-agent based adaptive systems and market-based resource allocation schemes. He is currently a senior research engineer of Communications Network Planning Laboratory at KDDI R&D Laboratories Inc. and a guest associate professor at Electro-Communications University. His current interests include traffic engineering in multi-domain networks and platform technology for context-aware services.

Hajime Nakamura received his B.E., M.E. and Ph.D. degrees from Waseda University, Tokyo in 1988, 1990, and 2002, respectively. He joined the Research and Development Laboratories of Kokusai Denshin Denwa Company Ltd. (currently KDDI) in 1990, and has been engaged in research on communications network planning. His current research interests include the architecture of next-generation telecommunications networks. He is currently a senior manager of the Communications Network Planning Laboratory of KDDI R&D Laboratories, Inc. He received the Young Engineer Award of the IEICE in 1996, Best Paper Awards of the IEICE in 1998 and 2002, Information Network Research Awards of the IEICE in 2002 and 2006, and Best Paper Award of APSITT 2005. He is an IEEE member.