Electronic Preprint for Journal of Information Processing Vol.21 No.2

Regular Paper

LISP-based Application-Layer Multicasting System for a Content Distribution Network

Hiroshi Yamamoto[1,a)] Katsuyuki Yamazaki[1]

Received: June 28, 2012, Accepted: December 7, 2012

Abstract: A content distribution network (CDN) where the information provider can distribute copies of contents to a group of cache servers is a very useful solution in various on-line services. An application-layer multicasting (ALM) system is a candidate technology for constructing the CDN, and can be achieved by utilizing a Locator/Identifier Separation Protocol (LISP) which is actively discussed in IETF. A mapping system which manages relationship between each multicast group and the group members (i.e., cache servers) is a core component of the system, but the centralized system requires costly resources for handling a large-scale CDN. In this study, we propose a new mapping system for the LISP-based application-layer multicasting system using distributed cloud computing technologies. The proposed system utilizes a distributed hash table (DHT)-based network consisting of a large number of LISP routers to manage the membership information of multicast groups, and shortens the start-up time needed for newly-arrived multicast members to start communicating with other members. This paper considers the performance of the proposed system by using a realistic and a large-scale computer simulation and clarifies that the mapping system can halve the start-up time compared with the simple DHT-based system.

Keywords: LISP, Multicast, CDN, DHT, location-aware node selection, network coordinate system

1. Introduction

A content distribution network (CDN) where the information provider can distribute copies of contents to a group of cache servers is a very useful solution in various on-line services [7], [18]. For example, the CDN can improve the satisfaction of viewers of video on demand services (e.g., YouTube [24]) and users of file sharing services (e.g., Dropbox [9]) because the users can retrieve their desired contents from the nearest cache server in the shortest time. Furthermore, the system can increase scalability of the contents distribution system by transferring the access load on a centralized server to many cache servers dispersed over the Internet.

IP multicast is a candidate technology for building a global and public content distribution network where one can simultaneously distribute any type of content to a group of end-users. However, it is not yet widely available through the Internet. This is because many information service providers (ISP) restrict transmissions of IP multicast packets due to their policy, and it is very difficult for IP multicast to accommodate reliable data transmissions through TCP.

On the other hand, Locator/Identifier Separation Protocol (LISP) is actively discussed in the Internet Engineering Task Force (IETF) as a next generation network protocol which offers multihoming and mobility without changing existing protocol stacks on the network equipment [11]. By using a mapping system which manages the relationship between an IP address (EID) of a multicast group which corresponds to each content distribution and IP addresses (RLOCs) of LISP routers (i.e., cache servers) to which the users connect to obtain their desired contents, a global application-layer multicasting (ALM) can be achieved. The ALM is very suitable for constructing the CDN because it transmits contents through IP unicast which is permitted by all ISPs, and supports TCP connections. In LISP Working Group of IETF, discussion about collaborations of LISP and Multicast has just started [4]. The mapping system can be implemented by using a high performance server, but the centralized system requires costly resources for handling a large-scale CDN. Furthermore, in order to reduce the stress of waiting contents acquisition, the system should be able to update the membership information immediately when the new user joins the CDN.

Therefore, in this study, we propose a new decentralized mapping system consisting of a large number of LISP routers dispersed over the Internet [23]. In order to manage a large amount of membership information of multicast groups by LISP routers in a completely decentralized manner, the proposed system utilizes a distributed hash table (DHT)-based technology which is used in cloud computing [1], [16]. Furthermore, in order to minimize the start-up time required for locating/updating the membership information of a selected multicast group and for notifying the group members of the updated information, we also propose three methods to work with the system. As the first method, each LISP router establishes logical connections to the closest neighbors based on their proximity so as to reduce query forwarding latencies on the DHT-based network. The second method decides a LISP router which is close to all members of each multicast group, and replicates the membership information on the selected LISP router. Furthermore, the third method derives coor-

[1] Nagaoka University of Technology, Nagaoka, Niigata 940–2188, Japan
[a)] hiroyama@nagaokaut.ac.jp

dinates of LISP routers on a pre-defined network coordinate space so that the latencies between any pair of them can be estimated very quickly by calculating the distance between the coordinates.

The rest of this paper is organized as follows. In Section 2, the existing technologies related with this study are described. Section 3 presents the proposed decentralized mapping system. Section 4 introduces an evaluation model and performance measures of the computer simulation, and Section 5 gives the simulation results of the proposed system. Finally, the conclusions are presented in Section 6.

2. Related Works

This section presents a LISP-based application-layer multicasting system. Furthermore, we introduce decentralized network technologies that can be utilized to resolve main problems of the system.

2.1 LISP-based Application-Layer Multicasting System

Locator/Identifier Separation Protocol (LISP) is a next generation network protocol which offers multihoming and mobility benefits without changing the protocol stacks of both end-devices and routers on the Internet. It is actively discussed in the LISP Working Group (WG) of Internet Engineering Task Force (IETF) [11]. LISP separates IP addresses into two new numbers, End Point Identifiers (EIDs) and Routing Locators (RLOCs). RLOC is assigned to a boundary point (i.e., LISP router) of a network site, and used for forwarding packets through the Internet. EID is used for numbering an end-device, and is allocated from an EID address block associated with the site where the device is located. The LISP router has functions for encapsulating packets originated by devices using EIDs for transport across the Internet where packets are routed by using RLOCs. By dynamically changing relationships between EIDs and RLOCs on a mapping system according to conditions or locations of end-devices, multihoming and mobility can be easily achieved.

The LISP WG of IETF has finished discussing functions for accommodating Protocol Independent Multicast (PIM) on the LISP architecture [12], and has just started discussions about a new application-layer multicasting (ALM) system based on LISP [4]. The mapping system between EIDs and RLOCs is a core component of the global ALM system. **Figure 1** presents an overview of the system. As shown in this figure, the system manages the relationship between an IP address (EID) of a multicast group and the IP addresses (RLOCs) of LISP routers to which its group members (i.e., end-users) are connecting. When receiving IP packets addressed with the multicast EID, a LISP router obtains RLOCs related with the EID from the mapping system, and forwards packets toward the RLOCs by unicast. The membership of the multicast group is cached on the LISP router in order to transmit subsequent packets.

Each LISP router manages two forwarding tables, inter-forwarding table and intra-forwarding table as shown in **Fig. 2**. The inter-forwarding table is built based on the membership information managed on the mapping system, and records RLOCs of other LISP routers joining the same multicast group. When the LISP router receives the multicast packets from the multicast-supported network, it identifies other LISP routers corresponding to the multicast group by referring to the table, and forwards the packets toward the routers. The LISP router simply transmits packets to others in the same multicast group, hence there is no multicast loop. On the other hand, the intra-forwarding table is used to distribute the multicast packets to end-users in the multicast-supported network where the LISP router belongs. In Fig. 2, the table manages IP addresses of end-users in the same multicast group, but the structure of the table depends on a multicast routing protocol utilized in the network.

Figures 3 and **4** show the sequence of the membership update on the LISP router and the mapping system when the new member joins the multicast group, respectively. When receiving a multicast join request from an end-user, the LISP router first checks whether the IP address (EID) of the multicast group has already been registered in the intra forwarding table. If not, the table creates a new entry for the multicast EID and records the IP address of the user on the entry. Next, the LISP router searches an entry of the EID on the inter forwarding table. If the entry does not exist in the table, the membership update request for joining the multicast group of the EID is forwarded to the mapping system. After that, the LISP router waits for the membership information of the multicast group from the mapping system, and adds the information to the inter forwarding table. Here, when the publisher

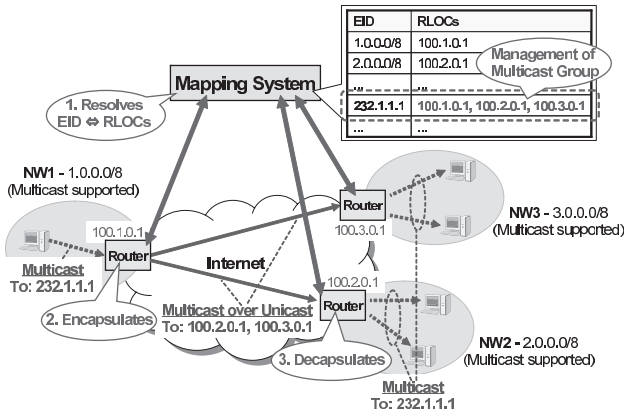

Fig. 1 LISP-based application-layer multicasting system.

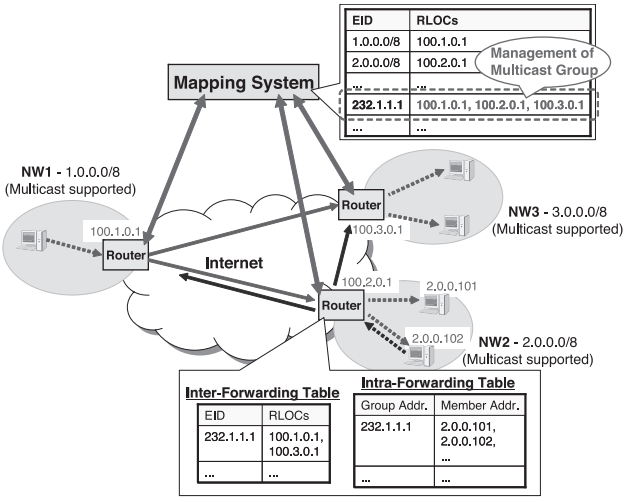

Fig. 2 Forwarding table of LISP-based multicasting system.

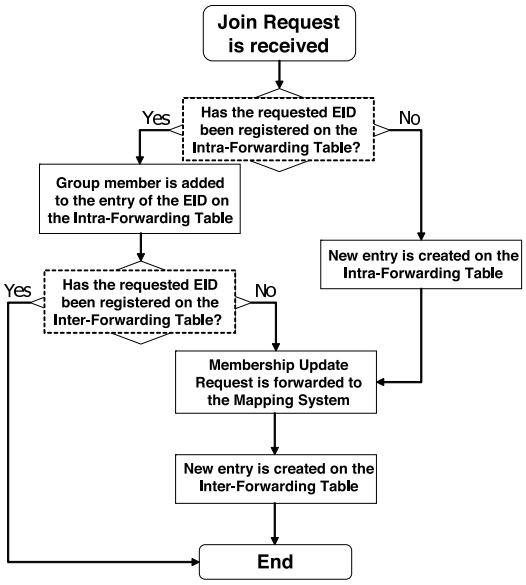

Fig. 3 Sequence of membership update in LISP router.

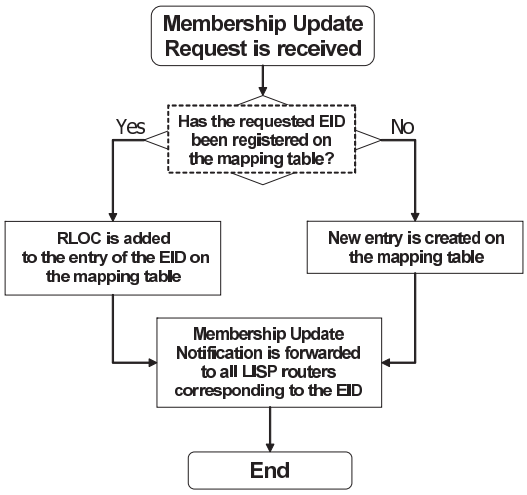

Fig. 4 Sequence of membership update in mapping system.

starts to distribute the content, it should transmit the join request to the LISP router so that the mapping system creates an entry of the multicast group corresponding to the content. Furthermore, when the group member list of an entry in the intra-forwarding table becomes empty due to leave requests from end-users, the LISP router deletes the entry and sends the membership update request to the mapping server so as to leave the multicast group. For membership updates to the multicast group, the mapping system adds/deletes the RLOC of the LISP router to/from the mapping table, and notifies the updated group members of the updated membership information.

For sending the membership update request to the mapping system, the LISP router can use a "Map-Register" message which is defined in the LISP standard drafts to register the relationship between the RLOC and the associated EID [11]. The mapping system updates the mapping table by receiving the message on which the RLOC of the LISP router and the requested multicast address (EID) are recorded. By setting a pre-determined flag to the unused field, the message can be identified as the update request of the multicast group. In addition, a "Map-Notify" message which is included in the drafts can be used for the mapping system to notify the LISP routers of the membership update. Like the "Map-Register" message, the "Map-Notify" message format has an unused area where the mapping system can set the flag.

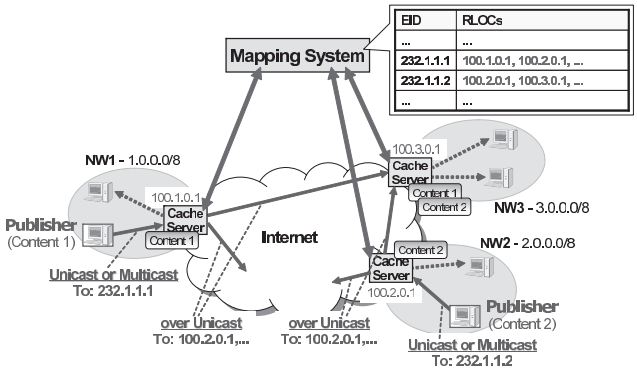

Fig. 5 LISP-based content distribution network.

By utilizing the LISP-based ALM system, a global content distribution network (CDN) can be easily achieved. In the system, the mapping system manages locations of cache servers (i.e., LISP routers) corresponding to each content (i.e., multicast group) as shown in **Fig. 5**. As a result, the content which is transmitted to the multicast address is copied on the cache servers to which the end-users connect to retrieve it.

In order to achieve the global application-layer multicasting, a method of encapsulating packets between distinct sites has been standardized in IETF so far (e.g., GRE: Generic Routing Encapsulation [10]). However, the method does not include a management system of the overlay network, and hence requires an administrator to understand the structure of the network topology and to apply an appropriate setting to each router. In contrast, in the LISP-based system, update of the overlay network structure can be automatically propagated to related LISP routers from the mapping system. Hence, the multicast group can be dynamically/automatically configured. Furthermore, in order to provide the global CDN service, the service operator has to strictly manage members of each multicast group. The mapping system of the LISP-based system enables the operator to control entries of forwarding tables in the LISP routers. Therefore, the LISP can be a candidate technology for building an ALM infrastructure for the global CDN service. Here, the application-layer multicasting system includes two functions, group management and distribution topology optimization. The LISP mainly focuses on the management function of the multicast group, but the construction algorithm of the multicast topology can be implemented in the LISP-based system as a function of the mapping system. In a future study, we will integrate the multicast topology optimization algorithm into our proposed mapping system.

The mapping system can be implemented by using a high-performance server as shown in Figs. 1 and 5, but we utilize three decentralized network technologies for building a large-scale CDN. First, a distributed hash table (DHT)-based network presented in Section 2.2 is used to build a decentralized mapping system. Second, a location-aware node selection framework shown in Section 2.3 is utilized to decide where the multicast membership information should be replicated in order to shorten

start-up time of the new content distribution. Third, a network coordinate system shown in Section 2.4 allows the nodes to quickly estimate latency between any pair of them when the node runs the above-mentioned location-aware node selection.

2.2 Distributed Hash Table

The distributed hash table (DHT)-based network is a distributed system constructed by a large number of computers (nodes) without any centralized control [1], [16]. In this network, the nodes are organized into a structured graph that maps data keys to a node. The structured graph enables the users to discover the data item corresponding to the given key in a short time.

There are many implementations of the network [19], [20], [21], of which Chord [21] is a well-known implementation. Each node and each data item are assigned unique ID and key by hashing their identifiers (e.g., IP address, filename), respectively, and are then mapped onto a one-dimensional identifier space. In the identifier space, the node with identifier id_i is responsible for managing the data whose key is within id_{i-1} and id_i, and is referred to as a *successor node* of the data (id_i is the identifier of a node having the i-th smallest identifier in the space).

Each node maintains the routing table with up to $\log_2 N$ other nodes (N is the total number of nodes), and each routing table entry includes both the Chord identifier and the location (e.g., IP address) of the relevant node. The routing table with node id_i contains the entry corresponding to the successor nodes of $id_i + 2^{j-1}$. By transmitting lookup queries to the node with the ID closest to the target key, Chord efficiently locates the successor node of the user's requesting data on average in $O(\log_2 N)$ hops of the query forwarding.

2.3 Network Location-aware Node Selection Framework

Network location-aware node selection framework selects the closest node to the given target based on network location in a large-scale distributed environment. Meridian [22] is a lightweight and scalable framework that forms a loosely-structured overlay network. The node keeps track of a small, fixed number of other nodes and organizes the neighbor list, recording the locations of the nodes. The neighbor list is updated by using a scalable gossip protocol, which notifies other nodes of the memberships in the system.

When the node receives a "closest node discovery to the target T" query, it determines its latency d to T, and probes its neighbors to determine their distance to the target. The query is forwarded to the neighbor closest to the target, and the process will continue until no closer neighbor is discovered.

Meridian requires relatively modest state management and processing of the nodes as mentioned above and can efficiently discover the closest node to the target on average in $O(\log N)$ hops of query forwarding.

2.4 Network Coordinate System

Network coordinate system is a latency prediction and management system, which maps computers to a pre-defined geometric space where an actual latency between computers can be estimated by calculating the distance between a pair of coordinates. The term "coordinate" here means the position of a computer on the geometric space. For example, if the geometric space is two dimensional (X, Y), the coordinate of computer i is (x_i, y_i). Therefore, the system enables computers to obtain latencies between any pairs of them without directly measuring the latencies by using an active measurement method such as "Ping."

There are many centralized or decentralized algorithms that decide accurate coordinates of computers on the geometric space [8]. Vivaldi is a famous decentralized algorithm, and can be easily adopted to a decentralized overlay network [6]. In the algorithm, each computer obtains the actual distance to a randomly selected target computer by using the active measurement tool, and then calculates the estimated latency from the coordinates. If the estimated latency is larger than the actual one, the coordinates of the computer on the geometric space move toward the target so as to minimize the difference between the actual and estimated ones. Otherwise, the coordinates of the computer move away from the target.

By repeatedly adjusting the coordinates of computers as mentioned above, the accurate coordinates of computers can be determined.

3. Proposed Decentralized Mapping System

The goal of this study is to design a scalable mapping system which is responsible for managing relationships between IP addresses (EIDs) of multicast groups and IP addresses (RLOCs) of LISP routers in network sites where the subscribers of the contents (i.e., end-users) are located. In the system, the end-users express their interest by generating a join request which specifies an EID of a desired multicast group (e.g., a desired content). The system should have enough resources to manage a large amount of multicast membership information, and to notify the LISP routers (i.e., cache servers) of the updated information in real time when a publisher starts to distribute new content or a new subscriber joins the group.

To satisfy these requirements, the proposed mapping system is built by interconnecting a large number of LISP routers dispersed over the Internet. The proposed system is equipped with the completely distributed technology used in cloud computing that decides and locates the LISP routers responsible for managing the membership information of the multicast group.

3.1 Assumed Network Configuration

In the proposed system, a popular implementation of the DHT-based network, Chord, is selected for building the mapping system [21]. **Figure 6** shows the network configuration of the mapping system. As shown in this figure, each LISP router is assigned a unique identifier (Router ID) by hashing its location information (for example, IP address), which is mapped to a large circular identifier space. Furthermore, each membership information is also assigned a unique key (Group ID) by hashing an EID of the multicast group.

When publishing/retrieving content, an end-user generates a join request corresponding to the desired content distribution, and transmits it to the nearest LISP router. The LISP router generates a membership update request, and records a Group ID, an EID

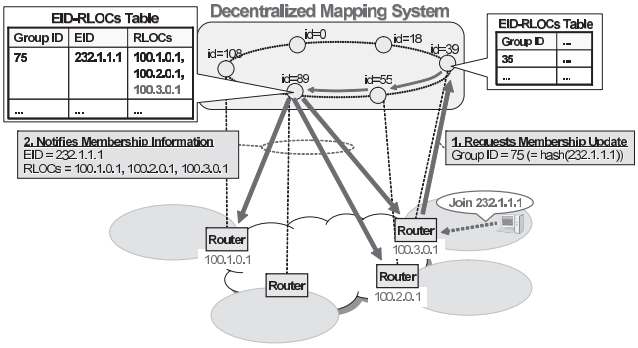

Fig. 6 Assumed network configuration.

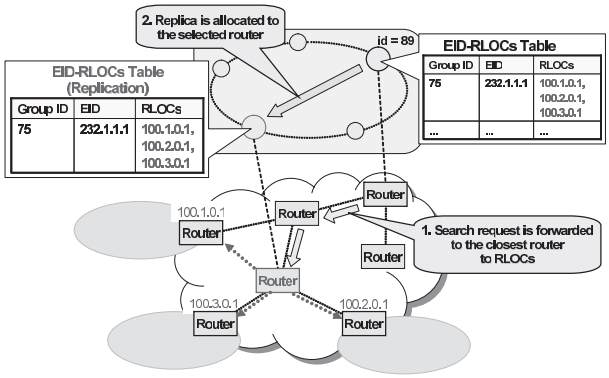

Fig. 7 Proposed mapping router selection.

```
1. MappingRouter_decision{
2.    min_latency = inf
3.    min_neighbor = null
4.    latency_guide = this.EstimatedNotificationLatency
5.    for (i in neighbor_list){
6.        latency = i.EstimatedNotificationLatency
7.        if (latency < min_latency){
8.            min_latency = latency
9.            min_neighbor = i
10.       }
11.   }
12. }
13. if (latency_guide < min_latency){
14.   return this
15. }
16. else{
17.   return min_neighbor.MappingRouter_decision
18. }
19.}
```

Fig. 8 Algorithm of proposed mapping router selection.

```
1. EstimatedNotificationLatency{
2.    max_latency = 0
3.    for (i in locator_set){
4.       if (max_latency < latency_between_this_and_router_i){
5.          max_latency = latency_between_this_and_router_i
6.       }
7.    }
8.    return max_latency
9. }
```

Fig. 9 Algorithm of notification latency estimation.

of the multicast group and its RLOC on it. After that, the LISP router chooses one neighbor whose Router ID is larger than the Group ID and is the closest to that from the routing table, and forwards the request to the selected neighbor. This process is repeated until the request arrives at the successor LISP router of the Group ID or the router which stores a replication of the membership information of the multicast group.

The successor router has the role of managing membership information related with the Group ID on the EID-RLOCs table. If the router has already had membership information related to the Group ID recorded on the received request, it adds a new RLOC to the information. Or else, the router creates a new entry on the table, and registers the relationship between an EID and a RLOC to it. And then, the router notifies all multicast members associated with the Group ID of the updated membership information. After that, if the proposed system uses a proximity-aware mapping router selection, the algorithm for locating a LISP router whose distances to all members of the multicast group are small is invoked as described in Section 3.3.

3.2 Proposed Proximity-aware Neighbor Selection

In order to shorten the start-up time needed for newly-arrived users to start sending/receiving the contents, the proposed system is equipped with a proximity-aware neighbor selection method [5], [14]. The method selects neighbors of each LISP router on Chord based on latencies between routers so as to minimize the lookup time of the desired membership information. In the method, candidates of the j-th neighbor of a LISP router with Router ID of id_i refer to the first m routers in the identifier space range $id_i + 2^j$ to $id_i + 2^{j+1} - 1$.

And then, the LISP router measures the distance to each candidate neighbors by using an active measurement tool such as Ping, and selects the lowest-latency router from the candidates as the j-th neighbor in the routing table.

3.3 Proposed Proximity-aware Mapping Router Selection

In order to further shorten the start-up time, a new proximity-aware mapping router selection method is proposed. The proposed method decides where membership information of each multicast group should be replicated so as to minimize the time needed for notifying all multicast members of updated membership information as shown in **Fig. 7**.

The proposed method is based on the existing network location-aware node selection framework, called Meridian [22]. Each LISP router keeps track of a small, fixed number of other routers selected randomly from a wide area of the Internet and organizes a neighbor list recording their locations. Note that, the neighbor list is constructed in addition to the routing table of the network configuration.

As described in the previous section, after the LISP router managing the membership information accepts the membership update request, it generates a mapping router search request in order to decide a router (named Mapping Router) which stores a replication of the information. Here, the request includes RLOCs of all members of the multicast group.

The router, which received the request, first calculates a maximum latency *latency_guide* toward the multicast group members on line 4 in **Fig. 8**. The algorithm of the maximum latency derivation is presented in **Fig. 9**. Next, as shown on lines 5–6 in Fig. 8,

all neighbors of the router are requested to derive their maximum latencies. As a result, the router decides the neighbor whose maximum latency is the smallest on lines 7–9. Finally, if no closer neighbor is detected than the router, the closest router currently discovered is chosen as a mapping router (lines 13–14). Otherwise, the request is forwarded to the closest neighbor, and it will repeat the process (lines 16–17).

The membership information of the multicast group is replicated on the new mapping router, and the replica stored on the previous mapping router is deleted. Furthermore, the route to the mapping router is recorded on a routing table of routers whose routing tables include the successor router of the multicast group so that the membership update request arrives at the mapping router.

As mentioned in Section 2.3, the main objective of Meridian is to locate the closest node to the given target. In addition, an application of Meridian, Central Leader Election, has also been proposed in Ref. [22], which enables the users to locate a centrally situated node with respect to a set of given targets. Each node in the application requests its neighbors to determine their average distances to the set of targets, and then selects the neighbor whose average distance is the smallest.

In order to locate the closest router to all members of the multicast group, our selection method utilizes a procedure of the application. However, unlike the application, each router requests its neighbors to measure their maximum distances to the members, and selects the neighbor whose maximum distance is the smallest.

3.4 Proposed Network Coordinate-based Latency Estimation

In the above-mentioned selection method, the LISP router must request each neighbor to measure the distances to all members of the multicast group when receiving the search request of the mapping router. If the latencies between LISP routers were obtained by using an active measurement tool, a replication delay needed for locating a new mapping router and for updating routing tables of routers related with the membership information may increase. Therefore, in order to minimize the replication delay, we leverage a network coordinate system, called Vivaldi [6], for estimating the latencies without using the active measurement tool.

All LISP routers periodically measure their distances to others, and calculate their own coordinates on the pre-defined geometric space based on the measurement results by using the Vivaldi algorithm. The algorithm adjusts the coordinates so as to minimize the difference between the actual latency and an estimated one which is derived by calculating the distance between a pair of coordinates as described in Section 2.4. In our proposed system, the coordinates of routers has three dimensions (X, Y, Height) and the distance d_{ij} between routers i and j is calculated by the following equation because it has been clarified that the coordinate space achieves more accurate latency estimation in Ref. [6].

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} + h_i + h_j. \quad (1)$$

In this equation, (x_i, y_i, h_i) and (x_j, y_j, h_j) are coordinates of router i and j, respectively. As in earlier research [2], [3], we use a virtual Height which represents the component of latency a computer experiences in all its paths due to its Internet access link.

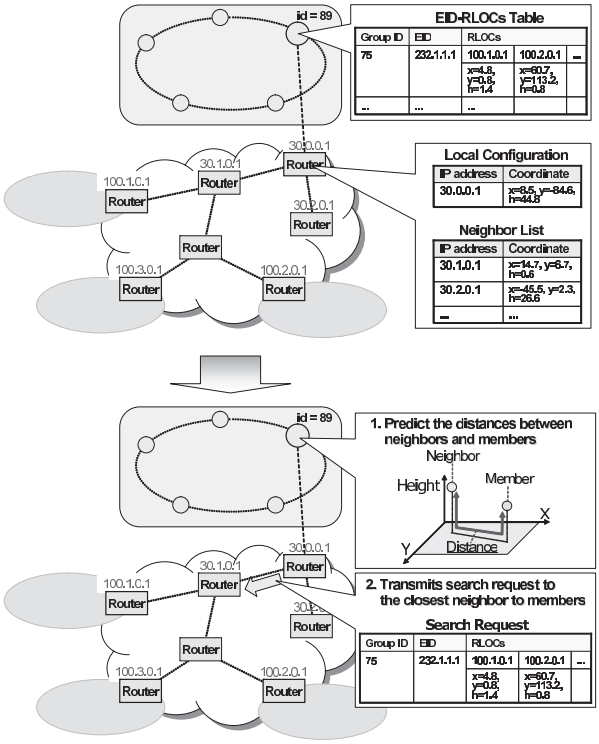

Fig. 10 Proposed network coordinate-based latency estimation.

As shown in **Fig. 10**, each LISP router manages its own coordinates, and periodically exchanges it with its neighbors registered on its neighbor list built for the mapping router selection. In addition, when a LISP router participates in a multicast group, the coordinates of the LISP router is recorded on the membership update request and is transmitted to a router which manages the multicast group. Namely, coordinates of all multicast members are stored on the LISP router that manages the membership information of the multicast group.

When the router accepts the membership update request, it generates the search request including RLOCs and coordinates of all members of the multicast group, and starts to locate the optimal mapping router. Therefore, when receiving the search query, the LISP router can estimate the maximum distance between each neighbor and the members from the coordinates of both neighbors and the members recorded on the search query.

As a result, the proposed method can decrease the time required for locating an optimal mapping router by reducing additional measurement time.

4. Evaluation Model and Performance Measures

The performance of the proposed decentralized EIDs/RLOCs mapping system is investigated by using computer simulations that assume a realistic network.

4.1 Evaluation Model for Proposed Mapping System

In order to evaluate the effectiveness of the proposed system, latencies between a large number of routers deployed on the Internet should be prepared. In this study, we utilize the "King

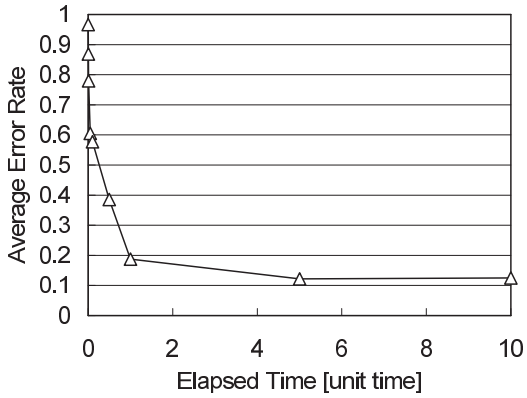

Fig. 11 Average error rate vs. elapsed time.

data set" that contains measurements of latencies between a set of DNS servers [13]. The latencies are measured by the King tool which estimates round-trip time (RTT) between two arbitrary DNS servers by using recursive DNS queries. First, the tool sends the query to the first DNS server for resolving the name of node that the server is managing. Next, another query is sent to the first server but is routed to the second DNS server by requesting a lookup of a name that the second server is responsible for. The RTT between two DNS servers can be found as the difference in RTTs of these two queries.

An example of the King data set including latencies between all pairs of 462 servers has been released [15]. The data set captures the realistic characteristics of latencies between nodes on a large-scale distributed system. The latencies on the data set are utilized as latencies between LISP routers in the computer simulations. Here, we assume that multiple end-users are connecting to the LISP router (i.e., a cache server) through wired/wireless connections, and positions of the users are the same as the LISP router to which they are connecting.

Furthermore, the routers attempt to derive their optimal coordinates on the geometric space by using Vivaldi algorithm when adopting the proposed coordinate-based latency estimation method described in Section 3.4. In the evaluation, we use a measure of "elapsed time" to express the running time of the proposed system. Here, the elapsed time of 1 [unit time] is defined as a running time needed until measurements of latencies between all routers are completed. For example, 5 [unit time] indicates the time interval where every router calculates their distances to all others five times.

Figure 11 shows the average error rate of predicted latencies between all pairs of routers based on the coordinates. The error rate is calculated according to Eq. (2).

$$error\_rate = \frac{|Latency_A - Latency_E|}{Latency_A}. \quad (2)$$

where $Latency_E$ and $Latency_A$ are defined as the estimated latency and the actual one, respectively. As shown in this figure, the average error rate decreases to 0.12 as the elapsed time increases.

4.2 Evaluation Model for Scalability Study

For the distributed system, evaluation of scalability for a large number of nodes is an important topic. However, the data set of 462 nodes presented in Section 4.1 is not sufficient for the evaluation.

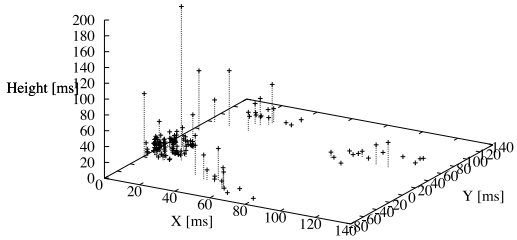

Fig. 12 2-D + height coordinates of 462 DNS servers.

Therefore, in order to expand the network model, the 462 nodes are mapped into two-dimensions with a height network coordinate space by using the Vivaldi algorithm as shown in **Fig. 12**. Here, we run the Vivaldi algorithm for 10 [unit time] because the error rate of the coordinates becomes the minimum value when the elapsed time is 10 [unit time] in Fig. 11. After that, the distributions of the three sub-coordinates (X, Y, Height) were calculated so that up to 50,000 coordinates were generated according to the distributions.

In the evaluation model, we assume an ideal condition where the error rate of the estimated latency from the coordinates becomes 0, and the actual latencies are modeled between all pairs of a very large number of LISP routers from the coordinates.

4.3 Performance Measures

In this research, the effectiveness of handling end-users' join requests to multicast groups on a large-scale distributed CDN is considered. As a measure of the effectiveness, we consider the start-up time needed for a LISP router to locate the router managing the membership information of the requested multicast group and to notify all LISP routers related with the group of the updated membership when a newly joined end-user is connecting. Furthermore, we evaluate a replication delay which indicates the additional time needed for locating a new mapping router and for updating routing tables of routers related with the membership information. The routes toward the mapping router become unstable due to updates of routing tables during the replication delay.

5. Simulation Results

First, the effectiveness of each proposed method of the mapping system are evaluated by using the evaluation model of Section 4.1. Next, we clarify the fundamental scalability of the system on the ideal evaluation model where the estimated latency from the coordinates has no error as described in Section 4.2.

5.1 Performance of Proposed Method

In this evaluation, the number of multicast groups is set to 10 percent of the number of LISP routers, and each LISP router randomly selects one multicast group and generates one membership update request.

First, the effectiveness of the proximity-aware neighbor selection method is evaluated without using other methods. **Figure 13** shows the effect of the number m of candidate neighbors on the start-up time when the elapsed time is 1 [unit time]. As shown

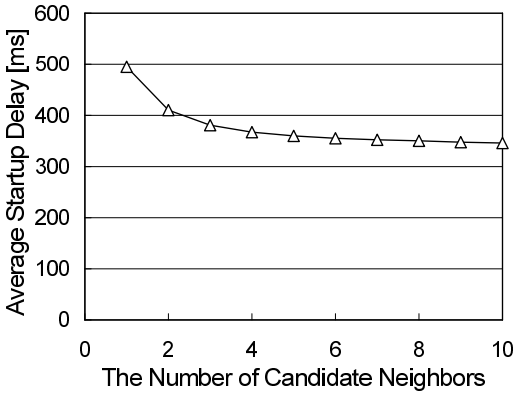

Fig. 13 Start-up time vs. the number of candidate neighbors.

in this figure, the start-up time becomes 500 [ms] when the number of candidate neighbors is one (i.e., when the system does not use the proximity-aware neighbor selection), and the start-up time decreases as the number of them increases. This is because each LISP router can select a closer neighbor from more candidates, and the time needed for the router to transmit the query to the neighbor can be reduced.

However, if the number of candidate neighbors increases more than 5, the start-up time does not decrease so much. On the contrary, a large number of them may increase the time interval required for rewiring logical connections between routers on the DHT-based network under churn. Therefore, the optimal number m of candidates (e.g., $m = 5$) should be carefully selected based on the network conditions.

Next, effectiveness of other methods (i.e., mapping router selection and coordinate-based latency estimation) on the proposed mapping system is evaluated. Here, occurrences of network failures (e.g., churns) on the DHT are not considered, hence the number of candidate neighbors is set to a large value (i.e., $m = 10$) in order to maximize potential of the proximity neighbor selection.

Figures 14 and **15** present variations of the start-up time and the replication delay with the increase in the elapsed time, respectively. As shown in Fig. 14, the proposed system with the mapping router selection (Mapping System + PNS + MRS) markedly decrease the start-up time compared with the system without that (Mapping System + PNS). This is because the mapping router selection replicates membership information of each multicast group on a LISP router which can finish notifying all members of an update of the membership in the shortest time. However, in Fig. 15, it costs more than 1 [s] to replicate the membership information on the closest router to members.

On the other hand, the use of the network coordinate-based latency estimation (Mapping System + PNS + MRS + CLE) can decrease the replication delay to less than 200 milliseconds without increasing the start-up time except when the coordinates of LISP routers have not been accurate yet (i.e., when the elapsed time is very small).

The LISP routers do not change their positions dynamically so that the coordinates do not change so much after having converged to their actual values. Therefore, we can conclude that the proposed system with all three proposed methods achieves excellent start-up time with acceptable replication costs.

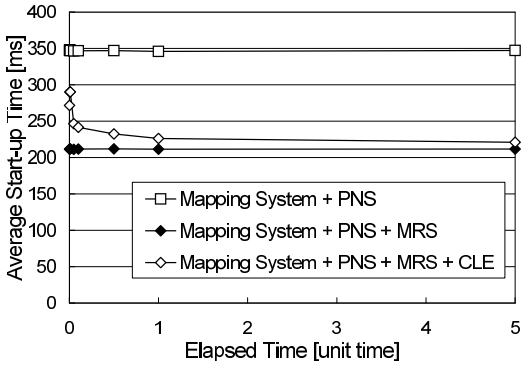

Fig. 14 Start-up time vs. elapsed time.

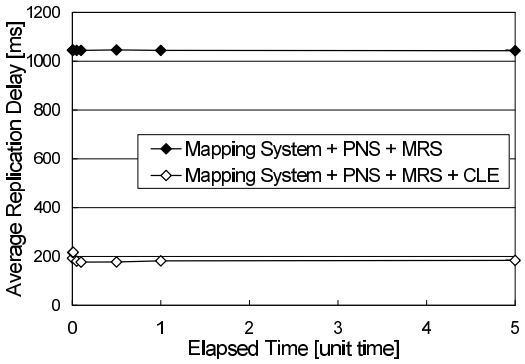

Fig. 15 Replication delay vs. elapsed time.

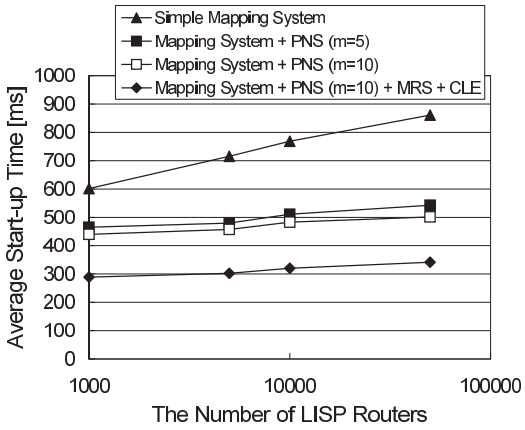

Fig. 16 Start-up time vs. the number of LISP routers.

5.2 Scalability Study

In order to clarify scalability of the proposed decentralized mapping system, we evaluate the effect of the number of LISP routers, which ranges from 1,000 to 50,000, on the performance measures. In this evaluation, the measures are start-up time and replication delay as explained in Section 4.3 as the same as the evaluation in Section 5.1. In this evaluation, the number of multicast groups is set to 1 percent of the number of LISP routers, and each LISP router randomly selects one multicast group and generates one membership update request.

Figure 16 shows the effect of the number of LISP routers on the average start-up time. As shown in this figure, if the mapping system does not utilize any proposed method (Simple Mapping System), the start-up time increases to 900 [ms] with the increase in the number of LISP routers. On the other hand, by utilizing a proximity-aware neighbor selection (Mapping System + PNS), a

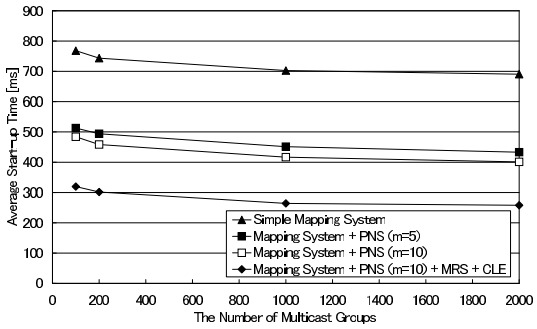

Fig. 17 Average start-up time vs. the number of multicast groups.

mapping router of the multicast membership information can be located in shorter time. In addition, with the increase in the number m of candidate neighbor, the proposed system decreases the start-up time to 500 [ms] even when the number of LISP routers is 50,000.

The proximity-aware mapping router selection and the coordinate-based latency estimation (Mapping System + PNS + MRS + CLE) further improves the start-up time. Note that the start-up time of the system with the latency estimation is the same as that without the latency estimation because the estimated latencies from the coordinates are assumed to have no error in this evaluation. As shown in Fig. 16, the system with the proposed methods achieves excellent start-up time in less than 350 [ms].

In addition, **Fig. 17** shows the effect of the number of multicast groups on the average start-up time when the number of LISP routers is 10,000. As shown in this figure, the start-up time of the proposed mapping system gradually decreases as the number of multicast groups increases. This is because the system can quickly notify all members of the updated membership information when the number of group members in each multicast group becomes small (i.e., the number of multicast group becomes large). Therefore, we conclude that the proposed system can achieve scalability for a large number of multicast groups.

Next, we evaluate the performance (Mapping System + PNS + MRS + CLE) of the proposed system on a more realistic pattern of join requests (i.e., requests of a large number of devices concentrated on a few multicast groups). Here, the number of all LISP routers is set to 50,000, the number of groups is set to 500, and the number m of candidate neighbors is set to 10. The existing research has clarified that the access pattern of contents on the Internet can be modeled by Zipf's law [17]. Therefore, we evaluate the effect of the exponent s of the Zipf's law shown in Eq. (3) on the performance measures. In the computer simulation, a serial number ($k = 1, 2, \ldots$) is assigned to each multicast group, and the multicast group with the smaller serial number k is subscribed by a larger number of end-users.

$$f(k, s, N) = \frac{1/k^s}{\sum_{n=1}^{N} 1/n^s}. \quad (3)$$

where f is the frequency at which the group with the serial number k is selected by the end-user, and N is the number of groups. With the increase in s, the highly ranked groups (i.e., highly ranked contents) become more popular.

Figure 18 shows the relationship between the start-up time and the exponent of Zipf's law. When smaller number of groups are selected by a large number of members, the mapping router needs more time to notify all group members of the membership information. However, the proposed system with two methods achieves the excellent start-up time even when only highly ranked groups are selected.

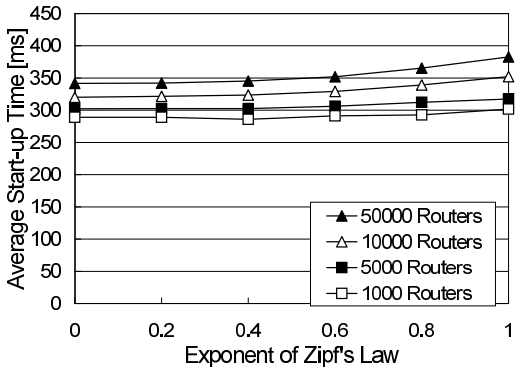

Fig. 18 Average start-up time vs. exponent of Zipf's law.

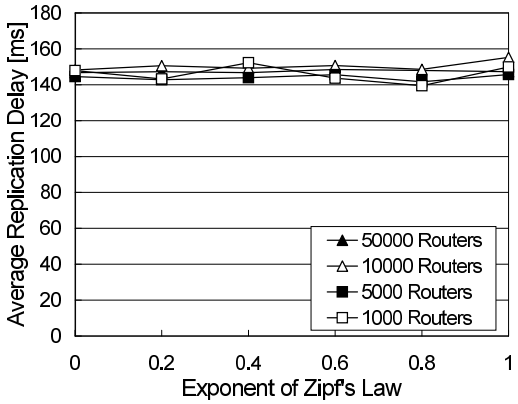

Fig. 19 Average replication delay vs. exponent of Zipf's law.

Furthermore, the effect of the exponent of Zipf's law on the replication delay of the proposed system (Mapping System + PNS + MRS + CLE) is shown in **Fig. 19**. As shown in this figure, the time needed for the system to replicate membership information on a new mapping router is only less than 160 [ms] regardless of not only the concentration of join requests on the multicast groups but also the number of LISP routers.

On the other hand, the decentralized mapping system generates more traffic of control messages than the centralized system, which may congest the Internet and/or may prevent the LISP routers from achieving their efficient operations. Therefore, in order to confirm scalability of our proposed system, we evaluate the amount of traffic of control messages generated by the system.

In this evaluation, we considered the amount of control messages including both membership update requests for locating the LISP router which manages the requested Group ID (see Section 3.1) and mapping router search requests for deciding the router which stores the replication of the membership information (see Section 3.3). The number of control messages for constructing the DHT-based network of LISP routers is not counted in the evaluation. This is because the DHT-based network can be constructed before starting to manage the membership information of the multicast groups. In addition, the messages for both manag-

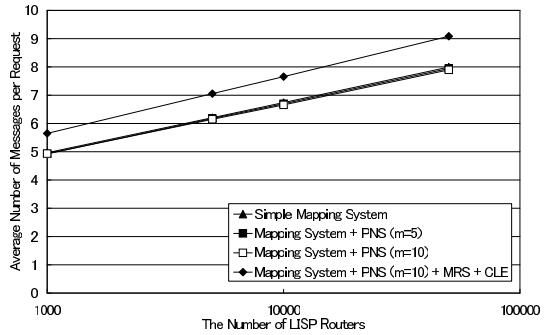

Fig. 20 Average number of control messages vs. the number of LISP routers.

ing a loosely-structured network of the proximity-aware mapping router selection and deriving coordinates of LISP routers on the pre-determined geometric space are ignored in the evaluation on the same score.

Figure 20 shows the relationship between the number of control messages and the number of LISP routers. As shown in this figure, as the number of LISP router increases, the number of control messages generated per membership update request gradually increases also on the mapping system without our proposed methods (Simple Mapping System). However, it does not exceed 8 even if the number of LISP routers is 50,000. The use of the proximity-aware neighbor selection method (Mapping System + PNS) does not affect the amount of traffic because it does not significantly change the topology of the DHT-based system. In contrast, by utilizing the proximity-aware mapping router selection and the coordinate-based latency estimation (Mapping System + PNS + MRS + CLE), the number of messages increases little. This is because the system generates additional traffic in order to establish routes to the mapping router storing replicas of the membership information, as explained in Section 3.3.

Here, if the coordinate-based latency estimation is not used, the system should generate a large amount of traffic for measuring network latencies between LISP routers. The number of neighbors of each LISP router is about 40, hence at least 40 messages should be generated per mapping router search request. Therefore, in terms of the traffic load, the system with the coordinate-based latency estimation method outperforms that case without the method. From the evaluation, we can conclude that the use of all methods does not generate a large amount of control messages.

6. Conclusions

In this paper, a new decentralized EID/RLOCs mapping system for LISP-based application-layer multicasting system has been proposed. The advantage of cloud computing technologies has been used to build a completely decentralized mapping system that is composed by a large number of LISP routers.

The proposed system has utilized the distributed hash table (DHT)-based network to manage and resolve the relationship between an IP address (EID) of each multicast group and IP addresses (RLOCs) of LISP routers. The proposed system has also used a neighbor selection method which enables LISP routers to choose lower-latency neighbors based on the proximity between the routers in order to shorten the start-up time needed for the newly-arrived multicast members to start communicating with other members. Furthermore, we have proposed a method which quickly replicates the membership information on a router which is close to all group members.

The performance of the proposed system has been investigated by using a realistic computer simulation. Through the evaluation, we have clarified that the proposed mapping system with the proposed methods can achieve excellent start-up time in less than 250 [ms] with decreasing replication delay of the membership information to 200 [ms]. Meanwhile the simple DHT-based system has increased the start-up time to more than 1 [s]. Furthermore, even in a large-scale network where there are 50,000 LISP routers, the start-up time of the proposed system has not exceeded 350 [ms].

In a future study, we will consider more scalable systems by considering the trade-off between the start-up time and the scalability (i.e., degree of load concentration). Furthermore, practicality of a large-scale flat name space of the DHT-based mapping system of LISP on not only the CDN but also ICN/CCN (Information/Content Centric Networking) will be evaluated.

Acknowledgments This study was supported in part by KDDI Foundation, Research Grant Program.

References

[1] Androutsellis-Theotokis, S. and Spinellis, D.: A Survey of Peer-to-Peer Content Distribution Technologies, *ACM Computing Surveys*, Vol.36, No.4, pp.335–371 (2004).
[2] Agarwal, S. and Lorch, J.R.: Matchmaking for Online Games and Other Latency-Sensitive P2P Systems, *Proc. ACM SIGCOMM 2009*, pp.315–326 (Aug. 2009).
[3] Chen, Y., Xiong, Y., Shi, X., Zhu, J., Deng, B. and Li, X.: Pharos: Accurate and Decentralised Network Coordinate System, *IET Communications*, Vol.3, No.4, pp.539–548 (2009).
[4] Coras, F., Cabellos, A., Domingo, J., Maino, F. and Farinacci, D.: Lcast: LISP-based Single-Source Inter-Domain Multicast, *LISP WG*, *IETF-83* (Mar. 2012), available from ⟨http://www.ietf.org/proceedings/83/slides/slides-83-lisp-5.pdf⟩.
[5] Dabek, F., Li, J., Sit, E., Robertson, J., Kaashoek, M.F. and Morris, R.: Designing a DHT for low latency and high throughput, *Proc. NSDI'04*, Vol.1 (Mar. 2004).
[6] Dabek, F., Cox, R., Kaashoek, F. and Morris, R.: Vivaldi: A Decentralized Network Coordinate System, *Proc. ACM SIGCOMM 2004*, pp.15–26 (Aug. 2004).
[7] Dilley, J., Maggs, B., Parikh, J., Prokop, H., Sitaraman, R. and Weihl, B.: Globally Distributed Content Delivery, *IEEE Internet Computing*, Vol.6, No.5, pp.50–58 (2002).
[8] Donnet, B., Gueye, B. and Kaafar, M.A.: A Survey on Network Coordinates Systems, Design, and Security, *IEEE Communications Surveys and Tutorials*, Vol.12, No.4, Forth Quarter (2010).
[9] Dropbox, available from ⟨http://www.dropbox.com/⟩.
[10] Farinacci, D. and Meyer, D.: Generic Routing Encapsulation (GRE), *Network WG*, IETF (Mar. 2000).
[11] Farinacci, D., Fuller, V., Meyer, D. and Lewis, D.: Locator/ID Separation Protocol (LISP), draft-ietf-lisp-13, *LISP WG*, IETF (June 2011).
[12] Farinacci, D., Meyer, D., Zwiebel, J. and Venaas, S.: LISP for Multicast Environments, draft-ietf-lisp-multicast-06, *LISP WG*, IETF (June 2011).
[13] Gummadi, K.P., Saroiu, S. and Gribble, S.D.: King: Estimating Latency between Arbitrary Internet End Hosts, *Proc. SIGCOMM IMW 2002*, pp.5–18 (Nov. 2002).
[14] Gummadi, K., Gummadi, R., Gribble, S., Ratnasamy, S., Shenker, S. and Stoica, I.: The impact of DHT routing geometry on resilience and proximity, *Proc. ACM SIGCOMM 2003*, pp.381–394 (Aug. 2003).
[15] King data set, available from ⟨http://pdos.csail.mit.edu/p2psim/kingdata/⟩ (2004).
[16] Lua, E.K., Crowcroft, J., Pias, M., Sharma, R. and Lim, S.: A Survey and Comparison of Peer-to-Peer Overlay Network Schemes, *IEEE Communications Surveys and Tutorials*, Vol.7, No.2, pp.72–93 (2005).
[17] Newman, M.E.J.: Power laws, Pareto distributions and Zipf's law, *Contemporary Physics*, Vol.46, No.5 (2005).

[18] Pierre, G. and Steen, M.: Globule: A Collaborative Content Delivery Network, *IEEE Communications Magazine*, Vol.44, No.8, pp.127–133 (2006).
[19] Ratnasamy, S., Francis, P., Handley, M., Karp, R. and Shenker, S.: A Scalable Content Addressable Network, *Proc. ACM SIGCOMM 2001*, pp.161–172 (Aug. 2001).
[20] Rowstron, A. and Druschel, P.: Pastry: Scalable, Decentralized Object Location and Routing for Large-scale Peer-to-Peer Systems, *Proc. 18th IFIP/ACM International Conference on Distributed Systems Platforms* (*Middleware 2001*), pp.329–350 (Nov. 2001).
[21] Stoica, I., Morris, R., Liben-Nowell, D., Karger, D.R., Kaashoek, M.F., Dabek, F. and Balakrishnan, H.: Chord: A Scalable Peer-to-Peer Lookup Protocol for Internet Applications, *ACM/IEEE Trans. Networking*, Vol.11, No.1, pp.17–32 (2003).
[22] Wong, B., Slivkins, A. and Sirer, E.G.: Meridian: A Lightweight Network Location Service Without Virtual Coordinates, *ACM SIGCOMM Computer Communication Review*, Vol.35, No.4, pp.85–96 (2005).
[23] Yamamoto, H. and Yamazaki, K.: LISP-based Information Multicasting System using Location-aware P2P Network Technologies, *Proc. IEEE Consumer Communications and Networking Conference* (*CCNC2012*), pp.660–664 (Jan. 2012).
[24] YouTube, available from ⟨http://www.youtube.com/⟩.

Hiroshi Yamamoto received his M.E. and D.E. degrees from the Kyushu Institute of Technology, Iizuka, Japan in 2003 and 2006, respectively. From April 2006 to March 2010, he worked at Fujitsu Laboratories Ltd., Kawasaki, Japan. Since April 2010, he has been an Assistant Professor in the Department of Electrical Engineering, Nagaoka University of Technology. His research interests include computer networks, distributed applications, and networked services. He is a member of IEICE and IEEE.

Katsuyuki Yamazaki received his B.E. and D.E. degrees from University of Electro-communications and Kyushu Institute of Technology in ’80 and ’01, respectively. At KDD Co. Ltd., he had been engaged in R&D and international standardization of ISDN, S.S. No.7, ATM networks, L2 networks, IP networks, mobile and ubiquitous networks, etc., and was responsible for the R&D strategy of KDDI R&D Labs. He is currently a Professor of Nagaoka University of Technology.