PAPER *Special Section on Network Virtualization and Network Softwarization for Diverse 5G Services*

Shortcut Creation for MeNW in the Consideration of Topological Structure and Message Exchanged

Masahiro JIBIKI[†a)], ***Member and*** **Suyong EUM**[††b)], ***Nonmember***

SUMMARY This article proposes a method to improve the performance of Message Exchange Network (MeNW) which is modern data distribution network incorporating the search and obtain mechanism. We explore an idea of shortcut creation which can be widely adapted to a topological structure of various network applications. We first define a metric called Efficiency Coefficient (EC) that quantifies the performance enhancement by a shortcut creation. In the design of EC, we consider not only diameter of the topology but also the amount of messages exchanged in the network. Then, we theoretically analyze the creation of a single optimal shortcut in the system based on the performance metric. The simulation results show that the shortcut by the proposed method reduces the network resource to further 30% compared with conventional approaches.

key words: *topological structure, message exchange, metric, shortcut, resource optimization*

1. Introduction

All forms of data continue to grow its heterogeneity and number. Thus, the role of a network has been shifted from communication between two end users to data delivery, especially large data file for content delivery and many to many/frequent communication caused by machine to machine (M2M) or SNS*. This trend has been demanding the networks to continuously provide users with enormous number of data, which results in the emergence of various network technologies/architectures; Information-Centric Networking (ICN) [1]–[3], Content Delivery Network (CDN) [4], Cloud Based Networking (CBN) [5], and Internet of Things (IoT) [6]. The kernel of the design in such network architectures lies in two important operational principles, namely search and obtain. Technically speaking, the former is known as name resolution process which maps the name of data to its location information, e.g. IP address. Then, a user request can be forwarded to the destination which holds the data. The latter involves delivering the data to the requester, which generally consumes more network resource than the former. Thus, these mechanisms should be designed carefully in order to provide fundamental performance improvement for the modern network architectures. Here, we call various object exchanged in these network systems as message, and define an architectural framework as *Message Exchange Networks* (*MeNW*), which incorporates highly optimized search and obtain mechanism.

In this article, we aim to optimize the performance of a MeNW by creating a shortcut on the topology of the system. The topology design is the core of such a system design from the perspective of message delivery time and network resource occupancy. Also, a shortcut creation is a fundamental approach to achieve the performance improvement natively. We initially define a performance metric called *Efficiency Coefficient* (*EC*) to quantify the efficiency of a shortcut in the system. Since we deal with communication networks, in the design of EC, we consider not only diameter of the topology but also the amount of messages exchanged in the network. Based on the metric, we theoretically analyse the creation of a single optimal shortcut in the system. The analysis leads us to propose a method that creates a shortcut, which enables us to enhance the performance of MeNW. Simulation studies show that a shortcut creation based on the proposed method can reduce the network resource by average 30% further comparing to conventional approaches.

This paper is organized as follows. In Sect. 2, we review its related researches. In Sect. 3, we propose a performance metric called EC. Then, we theoretically analyse a creation of shortcut in a general topology, and propose a method to create shortcuts inspired by the analysis in Sect. 4. In Sect 5, we evaluate the method through simulation. Finally, we conclude this article in Sect. 6.

2. Related Works

There have been various attempts in the development of a network architecture which supports efficient data distributions: from traditional peer to peer (P2P) networks to information centric networks (ICN) considered as a future network architecture for efficient data dissemination. While P2P was introduced to get operated on top of IP networks, ICN initially aimed to replace current IP networks to achieve the goal of efficient data dissemination. Due to the reason, one main concern of P2P is how to create an efficient overlay logical topology over IP networks: low diameter or low latency. Similarly, ICN tends to adapt an overlay approach to deal with its early deployment issue.

Manuscript received April 11, 2018.
Manuscript revised July 24, 2018.
Manuscript publicized September 20, 2018.
[†]The author is with National Institute of Information and Communications Technology, Koganei-shi, 184-8795 Japan.
[††]The author is with Graduate School of Information Science and Technology, Osaka University, Suita-shi, 565-0871 Japan.
a) E-mail: jibiki@nict.go.jp
b) E-mail: suyong@ist.osaka-u.ac.jp
DOI: 10.1587/transcom.2018NVP0004

*According to [7], around 29 billion connected devices are forecasted by 2022, of which around 18 billion will be related to IoT, and video traffic will be 82% of all consumer Internet traffic by 2021, up from 73% in 2016 [8].

Traditionally overlay logical topologies have been constructed based on Distributed Hash Table (DHT) mechanism such as CoDoNS [9], SFR [10], and LISP-DHT [11]. More recently, these approaches work on the design of topological structure to reduce the latency such as Multilevel DHT (MDHT) [12], [13] and Hierarchical SkipNet (HSkip) [14]. PSIRP [15] has chosen Hierarchical Chord [16] among the many available overlay designs. For a general network architecture, Kostas et al proposed hierarchical Pastry [17]. The process of building a topology in DHT systems can be considered as the process of creating shortcuts. For instance, when a node joins to the system in Chord [18], it connects to the destination nodes which are a certain number of hops away from the joining node, e.g. $2^s|_{s=0\dots\log_2^N}$ (N: total number of nodes in the system). In a broad sense, a shortcut creation can be understood as a process of creating a small distance topology. However, the conventional approaches mainly consider a distance metric to build a static topology without considering actual message exchanges in the system, which results in a non-optimal solution. In a communication network, a few popular traffic flows dominate the total network traffic and the popularity fluctuates. Thus, a dynamic approach to create shortcuts considering not only a network distance but also actual message exchanges in the system is highly demanded.

In [19], the authors introduced "path caching with expiration" which creates shortcuts in a topology, and the connection disappears after a time threshold. Individual peers store routes to other peers learned while handing forwarding requests. Then, one of the peers learned is selected as a destination peer of a shortcut in terms of the reduction of hop counts.

In [20], the authors proposed three different types of a shortcut: local hint cache, path hint cache, and global hint cache. The local hint cache can be considered as an enlarged successor list, which includes connections to its immediate neighbors. Path hint cache extends "path caching with expiration" proposed in [19] in a way that the selection of a shortcut destination is based on the reduction of overall latency rather than simple hop counts. The global hint cache aims to reach every peer in a topology approximately within two-hops. Local hint cache has a successor list which includes around 1000 neighbors' peers.

In [21]–[23], the authors considered the reduction of the stretch factor in a graph. Without complete routing information, traffic between two nodes tends to follow other than a shortest path. Their approaches aim for the reduction of the calculation time or space of routing table in terms of decreasing the stretch factor.

In summary, the previous approaches mainly focused on the reduction of hop counts and delay, which are purely based on topological structure without considering other performance metrics such as traffic demand. However, we believe that shortcuts should be created considering not only the reduction of delay but also traffic demand simultaneously to use network resources more efficiently. For this reason, we propose a method to create shortcuts considering the both as performance metrics.

3. EC (Efficiency Coefficient): Performance Metric for MeNW

In this section, by considering not only the distance on the topology or the amount of messages exchanged, but also the combination of both, we discuss about a shortcut creation which improves the efficiency of the network resources in the whole MeNW.

3.1 The Definition and Example of EC

We define new metric *EC* (Efficiency Coefficient) that shows efficiency of shortcut:

$$EC = (Reduced\ Distance) \times (Amount\ of\ Message) \quad (1)$$

By combining the distance on the topology with the amount of messages exchanged, *EC* represents an improvement of the efficiency that a shortcut brings into the network regarded as a communication system. In Fig. 1, consider a path from $node_0$ to $node_n$ (henceforth, $path_{0,n}$), and the nodes on this path are represented as $node_i$ ($1 \le i \le n-1$). In the example, $node_0$ creates a shortcut to $node_i$. Among subgraphs linked to $node_i$, the subgraph which does not include the nodes on $path_{0,n}$ is represented as $subgraph_i$. Here, let a_i be the amount of messages that $node_0$ exchanges with the nodes in $subgraph_i$ (i.e. the shaded area) via $node_i$. Fig. 1 shows one branch subgraph, but in a general network there are multiple subgraphs. In such a case, the amount of messages in the subgraphs is aggregated and can be modeled as the amount of messages to one virtual subgraph (Fig. 2).

The distance from $node_0$ to $node_i$ is i hops, and the hops reduced by the shortcut are $i-1$ since the original distance i is reduced to one. That is, the shortcut saves the network resources by $O(i-1)$ for each message exchanged. Consequently, in the case where $node_0$ creates a shortcut to $node_i$ and exchanges messages with a node in $subgraph_i$ via this shortcut (Fig. 1), the following network resources are saved and this is the EC value for $subgraph_i$ given by the shortcut:

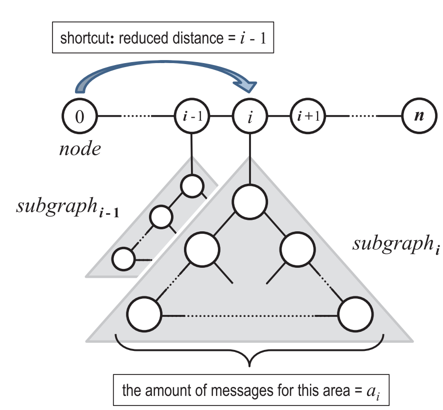

Fig. 1 Illustration of the efficiency coefficient (EC).

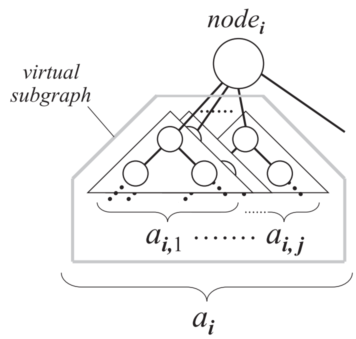

Fig. 2 Modeling the amount of messages in multiple subgraphs (general network topology).

$$EC_i^{(i)} = (i-1) \cdot a_i \quad (2)$$
(superscript: subgraph, subscript: shortcut)

Moreover, $node_0$ exchanges messages with the other $subgraph_{i-1}$ via this shortcut, and calculates the EC value for $subgraph_{i-1}$ (i.e. $EC_i^{(i-1)}$) different from Eq. (2). Thus, the final EC value in the case of creating a shortcut to $node_i$ (i.e. EC_i) is $\sum_{t=1}^{n-1} EC_i^{(t)}$. Thereafter, the optimal end node of a shortcut from $node_0$ is determined by comparing all these final EC values calculated while changing the end point.

Each node needs to record the amount of message a_i for each $subgraph_i$ in order to calculate the EC value. This can cause scalable problems. However, the amount of message (e.g. a_i in Fig. 2) can be described by a power-law distribution [24]. In other words, a few popular contents dominate the traffic in a communication network [8]. In this sense, recording only popular contents will be enough instead of tracking all messages, which resolves the scalability issue.

3.2 The Control Scheme and Utility Based on EC

First, we consider the control scheme based on *EC*. In order to decide the optimal end node of a shortcut by *EC*, each node needs to recognize the topology where a message is exchanged. On the Internet, the topology for all routes is obtained by a routing protocol. However, since the network like DHT, for example, can not obtain all topology in the network, it builds its own logical network and locally decides the route for each exchange message. In the case of the MeNW like the former, it is enough to notify the start node the information of the end node (e.g. node ID, address, etc.; henceforth, the same) as the control information exchanged between the nodes. In the case of the MeNW like the latter, it is necessary to add the information of each transfer node to the message exchanged and notify the start node to this list as well, so that each node can recognize the topology. This is a large overhead in control. However, since it is generally expected that the frequency of the logical topology change is few, it is not necessary to create a list of transfer nodes for each message exchanged. For example, sampling and recording may be performed within the allowable range of control overhead as follows;

1. The start node adds its own information to the exchange message timely (e.g. probabilistic selection of messages to start exchanging, etc.).
2. Each transfer node adds its own information if the node information is included in the received message, otherwise it forwards as it is.
3. The end node notifies the start node its own information and the node list as the control information, if this list is included in the message exchanged (otherwise it does not notify).

Next, we consider the utility of a shortcut creation based on *EC*. The utility that can be expected for a shortcut is to reduce the time required for communication and the amount of used network resources by reducing the number of hops required for communication. In particular, the latter can be expected to resolve a bottleneck of the network resources. One of the resolution for a bottleneck is traffic engineering (TE) [25]. In TE, not only topology but also resources utilization situation is required as the control information exchanged between the nodes. By adding this to the message exchanged and extending the definition of *EC*, TE based on *EC* is also possible. However, because the usage situation of the resources fluctuates drastically more than the topology, it may be difficult in some cases by this method alone. Even in such case, a shortcut creation based on *EC* can be operated simultaneously with TE that manage resources more directly (e.g. as operations in different layers like DHT + lower network). Consequently, by using these properly according to the situation, we think that a bottleneck problem can be settled.

As the method of resolving the bottleneck, there is a cache system used for ICN, CDN etc. in addition to the above TE [26]. Here, in order to clarify the relation between a shortcut based on *EC* and a cache, if we consider again the meaning of a shortcut, it is considered that a shortcut creation is to move its start node to the end of the shortcut virtually. That is, if the start node of message exchanged is regarded as a server and each end node is regarded as a client, it can be said that a shortcut creation using *EC* is the server migration to the location which can efficiently deal with each client access. Thus, the server can determine the efficient cache location by calculating the EC value for only the node of Case 1 in Sect. 4.1 (i.e. in this case, the cache is placed on the route from the server to the client). Thereafter, when the server again performs this calculation, its target becomes only the client that does not hit the above cache (that is, in the previous calculation, non-target node corresponding to Case 2). Therefore, by repeating this process, each cache is efficiently located within the network based on *EC*, we think that reduction of network load can be expected.

In the next section, we describe the method for efficiently determining the node with optimal EC value. This node is the best end node (destination) of a shortcut.

4. Theoretical Analysis: End Node of a Shortcut that Maximizes the EC Value

In this section, we analyze how to determine the end node of a shortcut, which results in the maximization of the EC value defined in Eq. (1). We prove there is only one maximum EC value in a creation of a shortcut, and determine its end node that leads to the maximum EC value. Due to this proposal to create a shortcut in MeNW, the optimal performance of MeNW can be achieved.

4.1 Reduction of Distance by a Shortcut

First, we analyze reduction of distance by a shortcut. This result is used for calculation of the maximum EC value in the next section.

The topology of the network we target in the paper is an undirected graph. In the following, a set of links constituting a route from a node to another node is called a path, and a path between $node_i$ and $node_j$ is defined as $path_{i,j}$. Furthermore, a path which consists of three or more nodes (i.e. a link set on the route from $node_i$ via $node_j$ to $node_k$) is defined as $path_{i,j,k} = path_{i,j} \| path_{j,k}$ if needed. Especially, when $path_{i,j}$ is a shortest path, it is defined as $path^{(s)}_{i,j}$. A length of $path_{i,j}$ is defined as $|path_{i,j}|$. Let take up a shortest path tree with $node_0$ as root, in the following, we consider the case to create a shortcut from $node_0$ to $node_i$ on this shortest path tree†. Hereafter, a path before the shortcut creation and after the creation is expressed as "$path$" and "$\widehat{path}$", respectively. Similarly, the distance is regarded as the number of hops (i.e. path length), and we consider the amount of change in a hop count. When the shortcut is created, the distances between $node_0$ and all nodes connected via $node_i$ changes. These nodes whose distance from $node_0$ change are classified into the following two cases;

- Case 1: The node whose new shortest path does not pass through the old shortest path between $node_0$ and $node_i$ $((path^{(s)}_{0,i} \cap \widehat{path}^{(s)}_{0,x}) = \emptyset)$.
- Case 2: The node whose new shortest path passes through the old shortest path between $node_0$ and $node_i$ $((path^{(s)}_{0,i} \cap \widehat{path}^{(s)}_{0,x}) \neq \emptyset)$.

The $node_x$ which belongs to Case 1/2 respectively and $path^{(s)}_{0,i}$, $\widehat{path}^{(s)}_{0,x}$ are shown in Fig. 3.

First, about the node which belongs to Case 1, we prove the basic propositions used in subsequent analysis, regarding reduction of the distance by shortcut.

Proposition I:
Consider $node_x$ which satisfies the following conditions.

†Since the component nodes of a large system, e.g. DHT, do not have the global view of the system, each node can only create an incomplete shortest path tree. The influence by the difference among both is evaluated with Sect. 5.2.

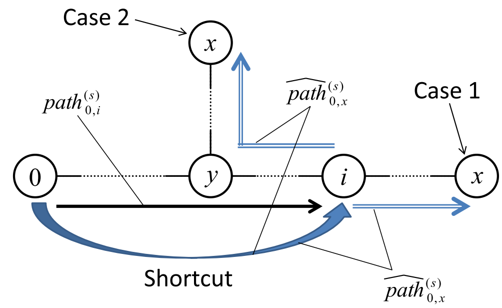

Fig. 3 Classification of $node_x$s whose distance from $node_0$ change by a shortcut creation.

$$\widehat{path}^{(s)}_{0,i} \subset \widehat{path}^{(s)}_{0,x}$$

The following proposition holds.

$$|path^{(s)}_{0,x}| - |\widehat{path}^{(s)}_{0,x}| \leq |path^{(s)}_{0,i}| - 1$$

Proof:
Assume that the following equation holds.

$$|path^{(s)}_{0,x}| - |\widehat{path}^{(s)}_{0,x}| > |path^{(s)}_{0,i}| - 1 \tag{3}$$

About $\widehat{path}^{(s)}_{0,x}$ (i.e. $\widehat{path}^{(s)}_{0,i,x}$), the following equation holds.

$$|\widehat{path}^{(s)}_{0,x}| = |\widehat{path}^{(s)}_{0,i}| + |\widehat{path}^{(s)}_{i,x}| = 1 + |\widehat{path}^{(s)}_{i,x}| \tag{4}$$

Next, consider $path_{0,x} = path^{(s)}_{0,i} \| path^{(s)}_{i,x}$, the following equation holds.

$$|path_{0,x}| = |path^{(s)}_{0,i}| + |path^{(s)}_{i,x}| \tag{5}$$

By substituting Eq. (4) and Eq. (5) into Eq. (3), the following equation holds.

$$|\widehat{path}^{(s)}_{i,x}| < |path^{(s)}_{i,x}| + (|path^{(s)}_{0,x}| - |path_{0,x}|) \tag{6}$$

Here, since $path^{(s)}_{0,x}$ is a shortest path and $path_{0,x}$ is not a shortest path, the following equation holds.

$$|path^{(s)}_{0,x}| - |path_{0,x}| \leq 0 \tag{7}$$

From Eq. (6) and Eq. (7),

$$|\widehat{path}^{(s)}_{i,x}| < |path^{(s)}_{i,x}|$$

Therefore, the shortest path between $node_i$ and $node_x$ changes by the creation of the shortcut. Here, since the added path is only $\widehat{path}^{(s)}_{0,i}$, $\widehat{path}^{(s)}_{i,x}$ which changed contains $\widehat{path}^{(s)}_{0,i}$ surely. Consequently, the following relation holds.

$$\begin{aligned}\widehat{path}^{(s)}_{0,x} &= \widehat{path}^{(s)}_{0,i} \| \widehat{path}^{(s)}_{i,x} \\ &= \widehat{path}^{(s)}_{0,i} \| (\widehat{path}^{(s)}_{i,0} \| \widehat{path}^{(s)}_{0,i} \| \widehat{path}^{(s)}_{i,x})\end{aligned}$$

From the above, there is a loop in the shortest path $\widehat{path}^{(s)}_{0,x}$, and this is contradictory to the definition of a shortest path. □

Next, about the node which belongs to Case 2, we prove similarly the basic propositions used in subsequent analysis, regarding reduction of the distance by shortcut.

Proposition II:
Consider $node_x$ and $node_y$ which satisfy the following conditions.

$$node_y \in (path^{(s)}_{0,i} \cap \widehat{path}^{(s)}_{0,x}) \quad (8)$$

$$\widehat{path}^{(s)}_{0,i} \subset \widehat{path}^{(s)}_{0,y} \subset \widehat{path}^{(s)}_{0,x} \quad (9)$$

The following equation holds.

$$|path^{(s)}_{0,x}| - |\widehat{path}^{(s)}_{0,x}| \leq 2 \cdot |path^{(s)}_{0,y}| - |path^{(s)}_{0,i}| - 1$$

Proof: Assume that the following equation holds.

$$|path^{(s)}_{0,x}| - |\widehat{path}^{(s)}_{0,x}| > 2 \cdot |path^{(s)}_{0,y}| - |path^{(s)}_{0,i}| - 1 \quad (10)$$

About $\widehat{path}^{(s)}_{0,x}$ (i.e. $\widehat{path}^{(s)}_{0,i,y,x}$), the following equation holds.

$$\begin{aligned}|\widehat{path}^{(s)}_{0,x}| &= |\widehat{path}^{(s)}_{0,i}| + |\widehat{path}^{(s)}_{i,y}| + |\widehat{path}^{(s)}_{y,x}| \\ &= 1 + |\widehat{path}^{(s)}_{i,y}| + |\widehat{path}^{(s)}_{y,x}| \quad (11)\end{aligned}$$

Next, consider $path_{0,x} = path^{(s)}_{0,y} \parallel path^{(s)}_{y,x}$ and $path^{(s)}_{0,i}$. Since $path^{(s)}_{0,i} = path^{(s)}_{0,y} \parallel path^{(s)}_{y,i}$ from Cond. (8), the following equations hold.

$$\left.\begin{aligned}|path_{0,x}| &= |path^{(s)}_{0,y}| + |path^{(s)}_{y,x}| \\ |path^{(s)}_{0,i}| &= |path^{(s)}_{0,y}| + |path^{(s)}_{y,i}|\end{aligned}\right\} \quad (12)$$

From Eq. (12),

$$\begin{aligned}&2 \cdot |path^{(s)}_{0,y}| \\ &\quad = |path_{0,x}| + |path^{(s)}_{0,i}| - |path^{(s)}_{y,x}| - |path^{(s)}_{y,i}| \quad (13)\end{aligned}$$

By substituting Eq. (11) and Eq. (13) into Eq. (10), the following equation holds†.

$$\begin{aligned}&|\widehat{path}^{(s)}_{i,y}| + |\widehat{path}^{(s)}_{y,x}| \\ &\quad < |path^{(s)}_{i,y}| + |path^{(s)}_{y,x}| + (|path^{(s)}_{0,x}| - |path_{0,x}|) \quad (14)\end{aligned}$$

Here, since $path^{(s)}_{0,x}$ is a shortest path and $path_{0,x}$ is not a shortest path, the following equation holds.

$$|path^{(s)}_{0,x}| - |path_{0,x}| \leq 0 \quad (15)$$

From Eq. (14) and Eq. (15),

$$|\widehat{path}^{(s)}_{i,y}| + |\widehat{path}^{(s)}_{y,x}| < |path^{(s)}_{i,y}| + |path^{(s)}_{y,x}| \quad (16)$$

† $\because |path^{(s)}_{y,i}| = |path^{(s)}_{i,y}|$

Therefore, the shortest paths between $node_i$ and $node_y$ or between $node_y$ and $node_x$ change by the creation of the shortcut i.e. from Cond. (9), $\widehat{path}^{(s)}_{i,x}$ changes. Here, since the added path is only $\widehat{path}^{(s)}_{0,i}$, $\widehat{path}^{(s)}_{i,x}$ which changed contains $\widehat{path}^{(s)}_{0,i}$ surely. Consequently, the following relation holds.

$$\begin{aligned}\widehat{path}^{(s)}_{0,x} &= \widehat{path}^{(s)}_{0,i} \parallel \widehat{path}^{(s)}_{i,x} \\ &= \widehat{path}^{(s)}_{0,i} \parallel (\widehat{path}^{(s)}_{i,0} \parallel \widehat{path}^{(s)}_{0,i} \parallel \widehat{path}^{(s)}_{i,x})\end{aligned}$$

From the above, there is a loop in the shortest path $\widehat{path}^{(s)}_{0,x}$, and this is contradictory to the definition of a shortest path. □

4.2 The Node with the Maximum EC Value

In this section, we analyze about the end node of a shortcut that maximizes the EC value.

Generally, a multipath exists between nodes in an arbitrary topology. For example, in Fig. 4, although $node_{[(n+1)/2]}$ and $node_{[(n+1)/2]+1}$ adjoin, both are end nodes of different shortest paths $path_{0,1,[(n+1)/2]}$ and $path_{0,n,[(n+1)/2]+1}$ with $node_0$ as root (i.e. $|path_{0,1,[(n+1)/2]}| < |path_{0,n,[(n+1)/2]}|$). If a shortcut is created to $node_i$ in this graph, some nodes on $path_{0,n,[(n+1)/2]}$ can be reached with less number of hops through the new shortest path including $node_i$ (i.e. the dotted path in Fig. 4). However, not only the length of the path, but also the amount of the message is involved in the calculation of the EC value. Consequently, in the case where a shortcut is created from $node_0$ to $node_i$, for simplicity, calculation of the EC value for determining the end node of the shortcut is aimed at $node_x$ which satisfies Eq. (17),

$$(path^{(s)}_{0,i} \subset path^{(s)}_{0,x}) \cup (path^{(s)}_{0,x} \subset path^{(s)}_{0,i}) \quad (17)$$

from the following reasons.

- The node that satisfies Eq. (17) can obtain the maximum reduction by a shortcut, among hop count reduction ranges indicated from **Proposition I/II**.
- If the amount of messages to the $node'$ on $path_{0,n,[(n+1)/2]+1}$ is smaller than the node on

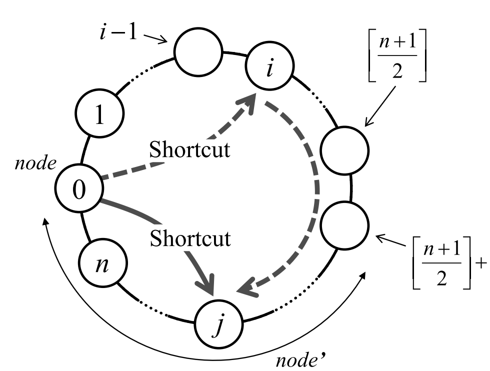

Fig. 4 Schematic illustration of two shortest paths ($path_{0,1,[(n+1)/2]}$ and $path_{0,n,[(n+1)/2]+1}$) with $node_0$ as root

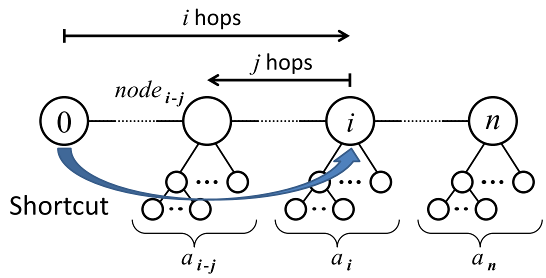

Fig. 5 Schematic illustration of a shortcut creation.

$path_{0,1,[(n+1)/2]}$, even if the *node′* is omitted, there is no large effect on calculation of the EC value.

- If the amount of messages to the *node′* on $path_{0,n,[(n+1)/2]+1}$ is more than the node on $path_{0,1,[(n+1)/2]}$, since it is more effective to directly create a shortcut to $node'_j$ on $path_{0,n,[(n+1)/2]+1}$, we do not include the *node′* in this calculation and calculate anew the case of the shortcut creation to $node'_j$.

The node branching from $path^{(s)}_{0,i}$ also can be considered in the same way.

If we create a shortcut between $node_0$ and $node_i$, since $node_i$ is i hops away from $node_0$, we can reduce the actual distance by $i-1$ hops (Fig. 5). In this case, there are two groups of nodes that benefit by the shortcut from $node_0$ to $node_i$: one group is called *inner-nodes* that are nearer than $node_i$, and the other is called *outer-nodes* that are further than $node_i$. From **Proposition I**, the shortcut always reduces the distance between $node_0$ and any *outer-nodes* which satisfy Eq. (15) by $i-1$ hops. However, not all *inner-nodes* benefit from the shortcut, and so we still more consider the case of a shortcut to *inner-nodes* to analyze its benefit. Let us consider $node_{i-j}$ among *inner-nodes* that is j hops away from $node_i$. There are two cases to reach from $node_0$ to $node_{i-j}$: one is the path via the shortcut with the distance of $1+j$ hops, and the other is the path which directly reaches to $node_{i-j}$ (not via the shortcut) with the distance of $i-j$. The shortcut is valid only if Eq. (18) is satisfied ($\because 1+j \leqq i-j$).

$$j \leqq \left[\frac{i-1}{2}\right] \tag{18}$$

In addition, we define another parameter to analyze the benefit of the shortcut creation (Fig. 5).

a_i: total number of requests from $node_0$ to the node further than $node_i$ except for $node_{i+1}$.

Based on the definition and description above, the following derives the Efficiency Coefficient (EC).

Once a shortcut is created from node $node_0$ to $node_i$ (Fig. 5), it reduces the distance by $i-1$ hops for any request from $node_0$ to *outer-nodes*. In addition, the shortcut also reduces the distance by $i-2j-1$ hops, as derived **Proposition II**, for requests from $node_0$ to $node_{i-j}$ on *inner-nodes*. Thus, EC_i is defined as follows:

$$EC_i = \sum_{k=i}^{n}(i-1)\cdot a_k + \sum_{j=1}^{[(i-1)/2]}(i-2j-1)\cdot a_{i-j}$$

Then, EC_{i+1} (i.e. the case of the shortcut creation to $node_{i+1}$) is defined sequentially as follows:

$$EC_{i+1} = \sum_{k=i+1}^{n} i\cdot(a_k) + \sum_{j=1}^{[i/2]}(i-2j)\cdot a_{i-j+1}$$

Therefore, the difference is as follows:

$$\begin{aligned} &EC_i - EC_{i+1} \\ &= i\cdot a_i - \sum_{k=i}^{n}(a_k) - (i-2)\cdot a_i + \sum_{j=1}^{[(i-1)/2]}(a_{i-j}) \\ &= \sum_{j=0}^{[(i-1)/2]}(a_{i-j}) - \sum_{k=i+1}^{n}(a_k) = \sum_{k=i-[(i-1)/2]}^{i}(a_k) - \sum_{k=i+1}^{n}(a_k) \end{aligned} \tag{19}$$

Next, we prove the basic proposition used in subsequent analysis.

Proposition III:
Consider a positive numerical sequence s_t ($s_t > 0$). There is at most one x which satisfies the following equation.

$$\sum_{t=p}^{q-x} s_t = \sum_{t=q-x+1}^{r} s_t \tag{20}$$

Proof:
Consider an integer sequence x_i which satisfies Eq. (20). Here, assume that plural terms exist in x_i. Let x_j, x_k ($x_j < x_k$) be terms in x_i, and $x_j = x_k - \bar{x}$ ($\bar{x} \geq 1$) is substituted for Eq. (20)

$$\sum_{t=p}^{q-x_k+\bar{x}} s_t = \sum_{t=q-x_k+\bar{x}+1}^{r} s_t$$

From the above,

$$\sum_{t=p}^{q-x_k} s_t + \sum_{t=q-x_k+1}^{q-x_k+\bar{x}} s_t = \sum_{t=q-x_k+1}^{r} s_t - \sum_{t=q-x_k+1}^{q-x_k+\bar{x}} s_t \tag{21}$$

From Eq. (20) ($x = x_k$) and Eq. (21), the following equation holds.

$$\sum_{t=q-x_k+1}^{q-x_k+\bar{x}} s_t = 0$$

The above is contradictory to the definition of s_t ($s_t > 0$). Therefore, plural x does not satisfy Eq. (20). Finally, consider the case where s_t is constant (ex. $s_t = u > 0$). In this case, for example, if $q = p + \mathrm{A}$ and $r = p + 2\mathrm{B} + 1$ ($\mathrm{A} > \mathrm{B}$), then $x = \mathrm{A} - \mathrm{B}$ satisfies Eq. (20). Consequently, the proposition is proved. □

From **Proposition III**, a numerical sequence S_q

$$S_q = \sum_{t=p}^{q} s_t - \sum_{t=q+1}^{r} s_t$$

becomes one of following cases according to p and r:

- only one q satisfies $S_q = 0$.
- always $S_q < 0$ or $S_q > 0$

Therefore, from the above argument and Eq. (19), EC has only global maximum value when i satisfies Eq. (22).

$$\sum_{k=i-[(i-1)/2]}^{i} (a_k) = \sum_{k=i+1}^{n} (a_k) \quad (22)$$

Otherwise, Eq. (19) is a monotonic function, and so this case also has only one maximum value.

From this analysis, the EC value is a function of the parameter i that represents the location of the destination node. That is, as the parameter i changes, EC value becomes one of following cases:

- Having one global maximum value
- Increasing monotonously
- Decreasing monotonously

From these, the number of change from positive to negative is only at most one time in the case where the difference of EC changes. Consequently, process of calculating the maximum EC can be closed when the amount of its change becomes negative. Moreover, Eq. (21) is derived without considering special conditions of the network topology and the distribution of requests a_i. Therefore, in the case where a shortcut is considered from a given node, we can scalably determine one shortcut that maximizes the EC value considering not only the types of topology but also the amount of messages exchanged in the network by sequentially checking the signs of the differences in each value.

5. Evaluations

This section provides simulation results that evaluate the effect of the proposed shortcut creation to MeNW. In the following, the evaluation is carried out from next three points,

- Topological reduction
- Efficiency improvement of network resource
- Feasibility

and verifies how the shortcut creation based on EC affects topological structure of the network as well as improves the efficiency of the network resources usage. Finally, the feasibility of the proposal is discussed.

5.1 MeNW for Evaluation

MeNW used for evaluation constitutes a tree topology consisting of $2^{L+1} - 1$ nodes, where L is the depth of the tree. Also assume that each node holds the topology information of the entire network. The kinds of messages exchanged in the whole network are 2^L, and these kinds are identified by numbers from 0 to $2^L - 1$. In addition, the message number corresponds one-to-one with the destination of the message. As the distribution of the appearance frequency according to the kind of message (i.e. the destination of the message), we use the normal distribution generated by the Box-Muller transform [27]. However, in the Box-Muller transform, since the standard normal distribution N(0, 1^2) is generated from the uniform distribution (0, 1), we made two modifications in this evaluation.

a) About the variance, we made the following modification. In order to set the final value range of the generated normal distribution random number to [0, $2^L - 1$] (i.e. equal to the kind of message), first, it is necessary to limit its range to $[-\frac{2^L-1}{2}, \frac{2^L-1}{2}]$. We used the uniform random number "(random()% 2^L) / ($2^L \cdot 1.0$)"† instead of a uniform distribution (0, 1), and modified so that the max normal distribution value generated for the min uniform random value $\frac{1}{2^L \cdot 1.0}$ other than 0 is $\frac{2^L-1}{2}$. Specifically, we calculated the coefficient "a" which transforms stochastic variable adaptively for each evaluation, and adjusted the value range of the generated normal distribution random number to be $[-\frac{2^L-1}{2}, \frac{2^L-1}{2}]$ (variance = $a^2 \cdot 1^2$).

b) About the mean, after changing the value range of the normal distribution random number by the above coefficient "a", we modified it so that the appearance frequency of the message number $\frac{2^L-1}{2}$ was the maximum (mean = $\frac{2^L-1}{2}$).

By the modification of a) and b), the range of the final value range of the generated normal distribution random number is set to [0, $2^L - 1$].

In this evaluation, $2^L \cdot 20$ messages are generated according to the scale L of the network and exchanged. Each of these messages are assigned the above normal distribution random number as the message number (i.e. the kind of message). Since the message number corresponds one-to-one with the destination of the message, that is, the distribution of the destination of the message follows a normal distribution. Each node in MeNW records frequent appearing messages and their destinations while forwarding these. After that, each node starts calculating the EC value along the path to the recorded nodes, and creates a shortcut. According to the evaluation, we change the scale of its environment by varying the value of L.

5.2 Topological Reduction

Figure 6 plots the efficiency coefficients versus the network distances to confirm the goodness of the metric "efficiency coefficient" as an indicator of topological reduction. In this evaluation, we first define one leaf node of the initial tree as the reference node. The reference node creates a message, and sends it to the destination node which is other leaf node. Then, the reference node records top 2% of the most popular messages, and starts calculating the EC value. Sub-

†since *random()* used in this evaluation generates an integer random value between from 0 to $2^{32} - 1$, the kind of uniform random number generated from the above equation is finite.

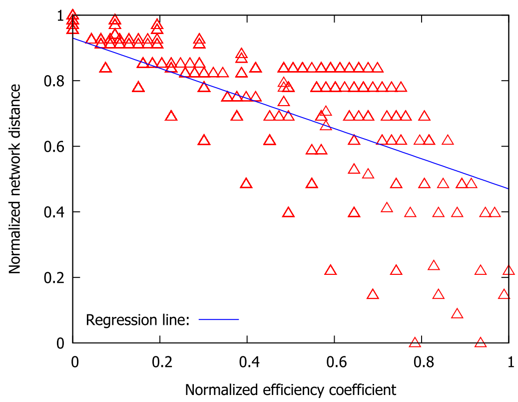

Fig. 6 Network distances vs. efficiency coefficients: correlation coefficient is −0.7268 (NW size = 2047).

sequently, when a shortcut is created to each node from the reference node, we calculate the network distance, e.g. average shortest path. The network distances and their corresponding efficiency coefficients are normalized and plotted in the figure. The correlation coefficient between two data sets is ≈ −0.73, which shows a strong correlation. It justifies us to use the EC value as the metric for the analysis of optimum shortcut destination.

Next, we evaluate the efficiency of the shortcut determined by the proposed method. As stated in the footnote in Sect. 4.1, in a large-scale network, each node cannot be aware of all shortcuts to other nodes, so there may be unknown shortest paths. Therefore, we compare the difference between the number of hops, (a) when messages are sent to the next node recognized as the shortest path by each node itself, and (b) when messages are sent to next node on the truth shortest path. The difference between both is called "stretch factor" generally. For example, in landmark routing, since triangle routing aiming at reduction of route information is carried out, the stretch factor is generally 3 as the standard values [21]–[23]. In this evaluation, all routes other than via a shortcut are known, and only the influence of unknown shortcut that other nodes create is evaluated, which is different from general triangle routing. Figure 7 shows the result. The stretch factor of the proposed method is almost 1.2 even when the scale of the network is increased. Generally, as a shortcut increases in the network, the probability of passing through unknown shortcut (but, the shortcut that each node itself determines as optimal) also increases in the case (a) above. That is, even if (a) is selected, the path through which the message is actually passed is close to (b). Therefore, we think that the increase of the stretch factor caused by the proposal is still applicable for practical use.

5.3 Efficiency Improvement of Network Resource

In this section, we evaluate the efficiency of the network resources in the whole MeNW that the shortcut based on the optimal EC value brings. First, we prepare two kind of the network; one including the shortcut created by Path Hint Cache (*HC*) [20], the other including the shortcut created by the optimum EC value (*EC*), respectively. After that, we measure the length of the path through which all messages pass in these networks, and compare the approaches of *EC* with *HC*. The same number of shortcuts is created for *EC* and *HC* cases. Figure 8 shows the result. In utilization of the network resources, *EC* is on average 30% better than *HC* even when the scale of the network is increased. *HC* creates the shortcut directly to the node with high possibility for destination of a message. However, *EC* determines the end node of the shortcut considering not only one node with high possibility but also the frequency of messages addressed to its neighbors (i.e. calculating the EC value). As a result, it can be said that *EC* is higher in efficiency per one shortcut.

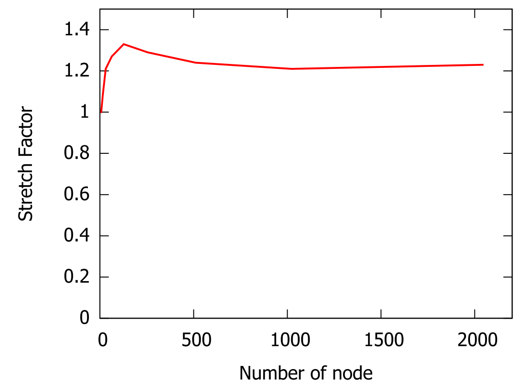

Fig. 7 Transition of stretch factor according to change of network size.

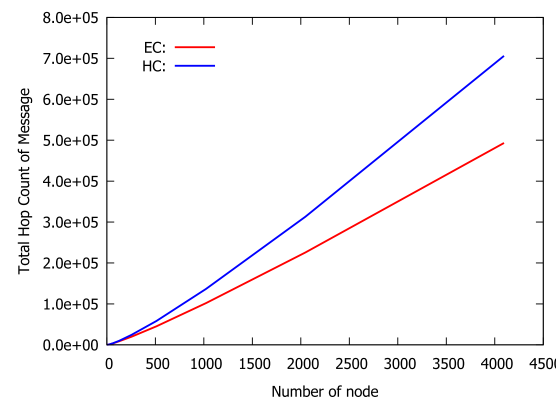

Fig. 8 Transition of network resource utilization according to change of network size.

5.4 Feasibility

Finally, we evaluate feasibility of our proposal. In the calculation of the EC value, each node is necessary to record the number of messages for each destination. As the scale of the network grows (i.e. as contents or IoT objects types increase), scalability problems can be caused, so in practice only a few popular messages can be recorded. In the following, we compare the difference (the number of hop) between true EC optimum node and calculated one according to change of rank size which records popular messages.

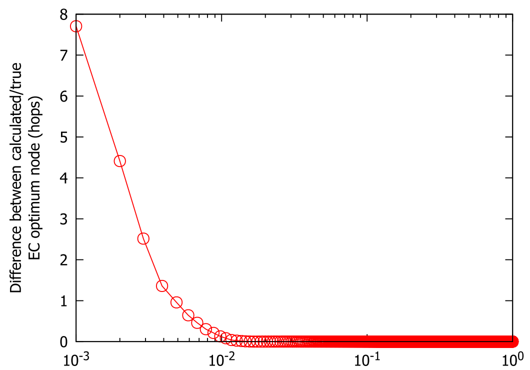

Fig. 9 Difference between true/calculated EC optimum node according to change of rank size (the kind of message: 1024).

Figure 9 shows the result. This result indicates that if size of a rank is about 0.3% of all message type (i.e. contents or IoT objects type), the error from optimum value is kept within 3 hops. However, this sample size may not be enough as compared with the amount of these expected in the future. We think this issue as follows. Generally, since the distribution of popularity follows a power law rather than a normal distribution, size of a rank can be still small [8], [24]. Moreover, since the latest server is equipped with TB (2^{40}) class memory, it is expected that the list of the popular contents following a power law is practically processed in the usual server.

6. Conclusions

The usage forms of network continue to grow its heterogeneity and number. This trend has been demanding the numerous network applications/services to continuously provide enormous number of data, and these have suffered from high latency and inefficient use of network resource. We believe that a proper management of shortcuts in the system can be a promising solution to deal with such problems.

We initially defined the performance metric called *Efficiency Coefficient* (*EC*). Since we deal with communication networks, in the design of EC, we consider not only diameter of the topology but also the amount of messages exchanged in the network. Based on the metric, we theoretically analysed the creation of a single optimal shortcut in the system. The analysis leads us to propose a method that creates a shortcut, which enables us to enhance the performance. Simulation studies show that a shortcut creation based on the proposed method can reduce the network resource by average 30% further comparing to conventional approaches.

References

[1] V. Jacobson, D.K. Smetters, J.D. Thornton, M.F. Plass, N.H. Briggs, and R.L. Braynard, “Networking named content,” Proc. 5th International Conference on Emerging Networking Experiments and Technologies (ACM CoNEXT’09), pp.1–12, Rome, Itary, Dec. 2009.

[2] C. Dannewitz, D. Kutscher, B. Ohlman, S. Farrell, B. Ahlgren, and H. Karl, “Network of information (NetInf) - An information-centric networking architecture,” Comput. Commun., vol.36 no.7, pp.721–735, April 2013.

[3] S. Eum, K. Nakauchi, M. Murata, Y. Shoji, and N. Nishinaga, “CATT: Potential based routing with content caching for ICN,” Proc. 2nd edition of the ICN Workshop on Information-Centric Networking, pp.49–54, Helsink, Finland, Aug. 2012.

[4] A. Pathan and R. Buyya, “A taxonomy and survey of content delivery networks,” Technical Report, GRIDS-TR-2007-4, Grid Computing and Distributed Systems Laboratory, The University of Melbourne, Australia, Feb. 2007.

[5] B. Wang, Z. Qi, R. Ma, H. Guan, and A.V. Vasilakos, “A survey on data center networking for cloud computing,” Computer Networks, vol.91, pp.528–547, Nov. 2015.

[6] L. Atzori, A. Iera, and G. Morabito, “The Internet of things: A survey,” Computer Networks, vol.54, no.15, pp.2787–2805, Oct. 2010.

[7] “Internet of Things forecast,” https://www.ericsson.com/en/mobility-report/internet-of-things-forecast (Online Available)

[8] “Cisco Visual Networking Index: Forecast and Methodology, 2016-2021,” https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/complete-white-paper-c11-481360.pdf, June 2017.

[9] V. Ramasubramanian and E.G. Sirer, “The design and implementation of a next generation name service for the Internet,” Proc. SIGCOMM 2004, pp.331–342, Portland, Oregon, USA, Aug. 2004.

[10] M. Walfish, H. Balakrishnan, and S. Shenker, “Untangling the web from DNS,” Proc. 1st Conference on Symposium on Networked Systems Design and Implementation, vol.1, San Francisco, California, USA, March 2004.

[11] L. Mathy and L. Iannone, “LISP-DHT: Towards a DHT to map identifiers onto locators,” Proc. 4th International Conference on Emerging Networking Experiments and Technologies (ACM CoNEXT 2008), Madrid, Spain, Dec. 2008.

[12] M. D’Ambrosio, C. Dannewitz, H. Karl, and V. Vercellone, “MDHT: A hierarchical name resolution service for information-centric networks,” Proc. ACM SIGCOMM workshop on Information-Centric Networking, Toronto, Canada, Aug. 2011.

[13] H. Liu, X. De Foy, and D. Zhang, “A multi-level DHT routing framework with aggregation,” Proc. 2nd edition of the ICN Workshop on Information-Centric Networking, pp.43–48, Helsink, Finland, Aug. 2012.

[14] C. Dannewitza, M. DAmbrosiob, and V. Vercelloneb, “Hierarchical DHT-based name resolution for information centric networks,” Comput. Commun., vol.36, no.7, pp.736–749, 2013.

[15] M. Ain, et al., “D2.3 - Architecture Definition, component Descriptions, and Requirements,” Deliverable, PSIRP 7th FP EU-funded project, Feb. 2009.

[16] J. Rajahalme, M. Särelä, K. Visala, and J. Riihijärvi, “On name-based inter-domain routing,” Computer Networks, vol.55, no.4, pp.975–986, 2011.

[17] K.V. Katsaros, N. Fotiou, X. Vasilakos, C.N. Ververidis, C. Tsilopoulos, G. Xylomenos, and G.C. Polyzos, “On inter-domain name resolution for information-centric networks,” Proc. 11th International IFIP TC 6 Conference on Networking - Volume Part I, Prague, Czech Republic, May 2012.

[18] I. Stoica, R. Morris, D. Karger, M.F. Kaashoek, and H. Balakrishnan, “Chord: A scalable peer-to-peer lookup service for internet applications,” Proc. SIGCOMM 2001, pp.149–160, San Diego, California, USA, Aug. 2001.

[19] M. Roussopoulos and M. Baker, “CUP: Controlled update propagation in peer-to-peer networks,” Proc. General Track: 2003 USENIX Annual Technical Conference, San Antonio, Texas, USA, June 2003.

[20] K. Tati and G.M. Voelker, “ShortCuts: Using soft state to improve DHT routing,” Web Content Caching and Distribution: 9th International Workshop, Beijing, China, Oct. 2004.

[21] M. Farshi, P. Giannopoulos, and J. Gudmundsson, “Finding the best

shortcut in a geometric network," Proc. Twenty-first Annual Symposium on Computational Geometry, Pisa, Italy, June 2005.
[22] M. Farshi, P. Giannopoulos, and J. Gudmundsson, "Improving the stretch factor of a geometric network by edge augmentation," SIAM J. Comput., vol.38, no.1, pp.226–240, 2008.
[23] M. Enachescu, M. Wang, and A. Goel, "Reducing maximum stretch in compact routing," 27th IEEE International Conference on Computer Communications, Joint Conference of the IEEE Computer and Communications Societies, Phoenix, AZ, USA, March 2008.
[24] M. Cha, H. Kwak, P. Rodriguez, Y.-Y. Ahn, and S. Moon, "Analyzing the video popularity characteristics of large-scale user generated content systems," IEEE/ACM Trans. Netw. (TON), vol.17, no.5, pp.1357–1370, Oct. 2009.
[25] N. Wang, K.H. Ho, G. Pavlou, and M. Howarth, "An overview of routing optimization for internet traffic engineering," IEEE Commun. Surveys Tuts., vol.10, no.1, pp.36–56, First Quarter 2008.
[26] M. Zhang, H. Luo, and H. Zhang, "A survey of caching mechanisms in information-centric networking," IEEE Commun. Surveys Tuts., vol.17, no.3, pp.1473–1499, Third Quarter 2015.
[27] G.E.P. Box and M.E. Muller, "A Note on the generation of random normal deviates," Ann. Math. Statist., vol.29, no.2, pp.610–611, 1958.

Masahiro Jibiki received the PhD degree in systems management from the University of Tsukuba, Japan, in 2003. In 1992, he joined NEC Corporation and was a researcher in the Central Research Laboratories until 2011. From 2006 to 2009, he was also a visiting professor at the University of Wakayama, Japan. Currently, he is an expert researcher in the National Institute of Information and Communications Technology. His research interests include networking, software science, and mathematical model.

Suyong Eum received B.E. in Electronics Engineering from MyongJi University, Korea and D.E. in Electrical and Computer Engineering Science from RMIT University, Australia in 2007. After his postdoctoral fellowship in Osaka University, Japan, he had been working for the National Institute of Information and Communications Technology (NICT) as a senior researcher. He is now an Associate Professor with Graduate School of Information Science and Technology, Osaka University. His current research interests include information centric networking, complex network theory, and network optimization.