IEEE TRANSACTIONS ON
EMERGING TOPICS
IN COMPUTING

Received 15 November 2013; revised 5 March 2014; accepted 11 March 2014. Date of publication 11 June 2014;
date of current version 30 October 2014.

Digital Object Identifier 10.1109/TETC.2014.2330517

An Overlay-Based Data Mining Architecture Tolerant to Physical Network Disruptions

KATSUYA SUTO[1], HIROKI NISHIYAMA[1], NEI KATO[1], KIMIHIRO MIZUTANI[2], OSAMU AKASHI[2], AND ATSUSHI TAKAHARA[2]

[1]Graduate School of Information Sciences, Tohoku University, Sendai 980-8579, Japan
[2]NTT Corporation, Yokosuka 239-0847, Japan
CORRESPONDING AUTHOR: K. SUTO (suto@it.ecei.tohoku.ac.jp)

ABSTRACT Management scheme for highly scalable big data mining has not been well studied in spite of the fact that big data mining provides many valuable and important information for us. An overlay-based parallel data mining architecture, which executes fully distributed data management and processing by employing the overlay network, can achieve high scalability. However, the overlay-based parallel mining architecture is not capable of providing data mining services in case of the physical network disruption that is caused by router/communication line breakdowns because numerous nodes are removed from the overlay network. To cope with this issue, this paper proposes an overlay network construction scheme based on node location in physical network, and a distributed task allocation scheme using overlay network technology. The numerical analysis indicates that the proposed schemes considerably outperform the conventional schemes in terms of service availability against physical network disruption.

INDEX TERMS Big data mining, neighbor selection, overlay network, physical network disruption, service availability, task allocation.

I. INTRODUCTION

With the rapid development of the information and communication technologies, ''big data'' has been generated from various aspects, such as online transactions, logs, search queries, health records, social networking information, science data, and so forth. It is widely recognized that big data mining is a key component that is required for the actualization of smart society [1]. Since big data comprises various types of data (such as e-mail, social media, video, and sensor data), the big data mining becomes exceedingly complex. Additionally, the big data mining needs to output result data expeditiously in response to the real time demand [2]. Therefore, conventional data mining employs parallel data mining architectures such as MapReduce [3] and Hadoop [4] to fulfill these requirements.

In those architectures, the data processing is executed by distinct nodes (called processing nodes) but system management task is served by a master node. While such a centralized management scheme simplifies the design and implementation, this scheme lacks scalability because the centralized management overseen by a master node may decrease the system performance when the number of nodes increases [5]. Additionally, since the master node is a single point of failure, the service availability can dramatically decrease when the master node ceases to function. From these reasons, scalability and service availability are critical issues for parallel data mining architecture.

As a remedy for improving scalability, an overlay-based parallel data mining architecture has been proposed. Since all the nodes execute both management and processing functions by using overlay network, this architecture can balance the management load. Additionally, this architecture achieves higher service availability against the breakdown of master node because it keeps providing the data mining until overlay network is disrupted.

However, this architecture cannot ensure the service availability against physical network disruption (e.g., router breakdown due to hardware trouble or DDoS attacks) [6]. The physical network disruption does not only lead to the cease of function of the damaged router but also disrupts the communications of the servers, which are connected with the damaged router. In other words, numerous nodes are removed

from the overlay network by the physical network disruption. This results in an emergence of isolated nodes in the overlay network and decreases the service availability of the data mining.

To deal with the above-mentioned problem, we propose an overlay-based parallel data mining architecture that is tolerant to physical network disruption. Our proposed architecture is designed based on the integration of overlay and physical networks. This paper proposes an overlay network topology for maximizing the connectivity against server breakdowns. Furthermore, overlay network construction and task allocation schemes are proposed for maximizing the service availability against physical network disruption.

The remainder of this paper is organized as follows. The relevant research works on parallel data mining architecture are surveyed in Section II. In Section III, we present our envisioned overlay-based parallel data mining architecture. Section IV numerically analyzes the service availability of the conventional and proposed overlay-based parallel data mining architecture. Section V evaluates the effect of physical network disruption on service availability and verifies the effectiveness of the proposed architecture through numerical calculation. Finally, concluding remarks are provided in Section VI.

II. AN OVERVIEW OF PARALLEL DATA MINING ARCHITECTURE

In this section, we introduce the conventional parallel data mining architecture based on the centralized management mechanism. Then we describe the existing works that aim to improve the service availability, followed by the shortcomings of these existing schemes. Moreover, we describe an overlay-based parallel data mining architecture that can overcome the weakness of the conventional architecture.

A. CONVENTIONAL PARALLEL DATA MINING ARCHITECTURE

MapReduce is the most popular architecture for parallel big data mining [4], [7]. In MapReduce, servers are classified into two types of nodes, i.e., a single master node and multiple processing nodes. While the master node schedules mapping and reduction processes and manages file name space operations (i.e., open, close, and rename), the processing nodes store data and execute mapping and reduction processes.

When a data processing request is injected, the master node partitions the task into some data blocks, which are distributed to distinct processing nodes. Then, each processing node (called mapper) performs the mapping process, which classifies a large amount of information and picks out the information required for the next process. After the mapping process, the master node selects a reducer, which performs the reduction process, from mappers. The reducer integrates the information extracted in the mapping process and outputs the analyzed results.

Since mapping and reduction processes are executed in distributed manner, MapRecue can execute the data mining at the speed proportional to the number of servers. Additionally, valuable existing works conducted in [8]–[12] developed high-performance parallel data mining architectures in terms of processing speed, network resource efficiency, computational resource efficiency, and energy efficiency. Despite the significant advantages, those architectures still suffer from server breakdowns because the success probability of data mining decreases when the servers fail due to hardware troubles or software bugs [13], [14].

To cope with this issue, the common MapReduce architecture (e.g., current Hadoop [15]) replicates each data block and distributes the replicated ones to distinct nodes, which increases the service availability against server breakdowns. Additionally, current Hadoop utilizes multiple master nodes mechanism to increase service availability against the breakdown of master node. However, it is difficult to ensure the service availability under real environment since the optimal numbers of replications and master nodes depend on the probability and scale of breakdowns.

The works [16] and [17] proposed processing scheduling technique that can shorten execution time of the data mining under failure-prone environment. However, because these works assume the scale of server breakdowns is small, the capability of data mining is dramatically decreased when a larger scale of breakdown, such as physical network disruption, occurs. Therefore, a parallel data mining architecture that is tolerant to physical network disruption is absolutely imperative to provide future ''ubiquitous big data mining service''.

B. OVERLAY-BASED PARALLEL DATA MINING ARCHITECTURE

Overlay-based parallel data mining is one of architectures that improve the service availability against server breakdowns [18]–[20]. In this architecture, all the servers execute both management and processing functions. The overlay network is constructed by all servers and utilized to find processing nodes, similar to the master nodes in the conventional architecture. This architecture can keep providing the service even if some nodes are removed from the overlay network.

Fig. 1 shows an example of mapping and reduction processes in the overlay-based parallel data mining architecture. When a data processing request is injected, a node that received the request (node A in the Fig. 1) executes a reception function by using the overlay network. In other words, the node finds mappers by using flooding message, where mappers are randomly selected (nodes B, C, and D in the Fig. 1). Then, a mapper that initially finished the mapping process (node D in the Fig. 1) becomes a reducer, and it requests to other mappers to transmit the processed data to itself, where the request message can be forwarded by using flooding scheme. After receiving the processed data from mappers, the reducer executes the reduction process and outputs the analyzed result.

In this architecture, since the connectivity of overlay network dramatically affects the service availability of

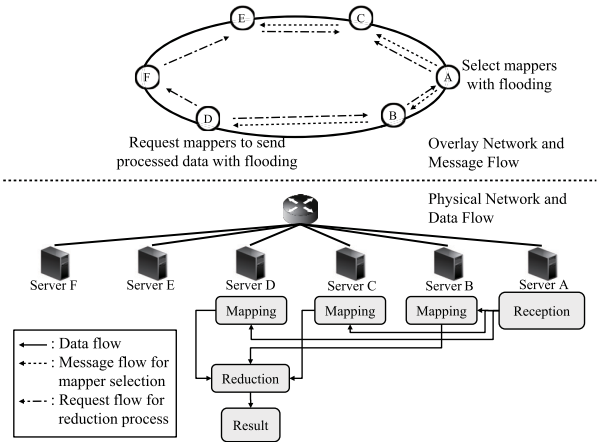

FIGURE 1. Mapping and reduction processes in overlay-based parallel data mining architecture.

data mining, there are numerous works, which tackled the connectivity issue from the various viewpoints, i.e., context-aware, graph theory based, and complex network theory based overlay network construction schemes [21]–[24]. These works make overlay networks that are tolerant to small-scale server breakdowns but do not consider the large-scale server breakdowns, i.e., physical network disruption. Therefore, this paper develops an overlay-based parallel data mining architecture that is tolerant to physical network disruption so that data mining is available at anytime, anywhere.

III. ENVISIONED OVERLAY-BASED PARALLEL DATA MINING ARCHITECTURE

In this section, we propose a novel overlay-based parallel data mining architecture to improve the service availability against server breakdowns and physical network disruption by utilizing physical network information. First, we introduce an overlay network topology following a bimodal degree distribution for maximizing connectivity against server breakdowns. Then, we propose a neighbor selection scheme in order to improve the number of available nodes after physical network disruption occurs. Furthermore, a task allocation scheme, which succeeds in data mining under physical network disruption, is proposed.

A. OVERLAY NETWORK TOPOLOGY BASED ON BIMODAL DEGREE DISTRIBUTION

We introduce an optimal overlay network topology that is tolerant to server breakdowns caused by hardware troubles and DDoS attacks. While hardware troubles cause random removal of nodes from the overlay network regardless of the degree of nodes, DDoS attacks remove higher degree nodes since malicious attackers attempt to disrupt the overlay network. To achieve high connectivity against both hardware troubles and DDoS attacks, this paper focuses on overlay network following a bimodal degree distribution.

In the bimodal degree distribution, there exist two types of nodes, i.e., Super Nodes (SNs) with higher degree k_s and Leaf Nodes (LNs) with lower degree k_l. Thus, the value of bimodal degree distribution, p_k^*, is expressed with the number of nodes, N, the number of SNs, N_s, the number of LNs, N_l, and the average degree, $\langle k \rangle$, as follows.

$$p_k^* = \begin{cases} N_s/N, & \text{if } k = k_s = \sqrt{\langle k \rangle N}, \\ N_l/N, & \text{if } k = k_l = \langle k \rangle, \\ 0, & \text{otherwise}, \end{cases} \tag{1}$$

where N_s and N_l are defined with an optimal node ratio, r, for maximizing connectivity against server breakdowns as follows.

$$N_s = rN, \tag{2}$$

$$N_l = (1 - r)N. \tag{3}$$

According to the work of T. Tanizawa *et al.* [25], r is formulated with the average degree $\langle k \rangle$, as the following equations.

$$r = \left(\frac{A^2}{\langle k \rangle N} \right)^{\frac{3}{4}},$$
$$A = \left\{ \frac{2\langle k \rangle^2 (\langle k \rangle - 1)^2}{2\langle k \rangle - 1} \right\}^{\frac{1}{3}}. \tag{4}$$

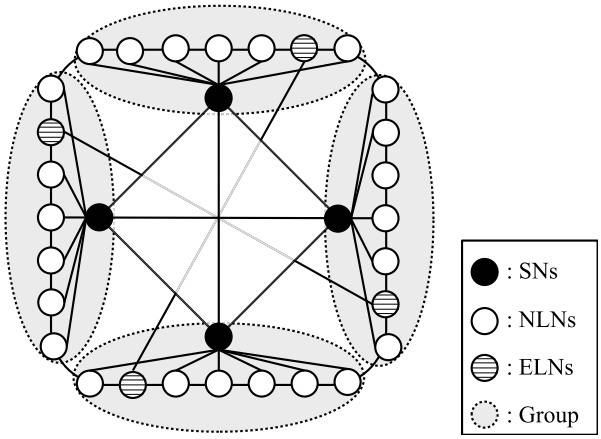

FIGURE 2. An optimal network topology which is tolerant to server breakdowns.

Based on the optimal bimodal degree distribution, an optimal network topology for maximizing connectivity against server breakdowns has been developed [24]. Fig. 2 shows an optimal network topology when $\langle k \rangle$ is nearly equal to 3. The LNs are classified into Normal Leaf Nodes (NLNs), which connect with an SN, and Extra Leaf Nodes (ELNs), which do not connect with an SN but connect with other ELNs. This topology is divided into multiple smaller groups. Each group consists of an SN, NLNs, and ELNs, where each group has the same number of nodes. Each SN connects with all other SNs to construct a complete graph since SNs transmit a number of messages to other SNs. Additionally, each SN connects to NLNs of its group corresponding to its degree. NLNs and ELNs connect with others to construct a ring topology, where the NLNs only connect to the SN that is located in the same group and the ELNs are evenly located in the overlay network.

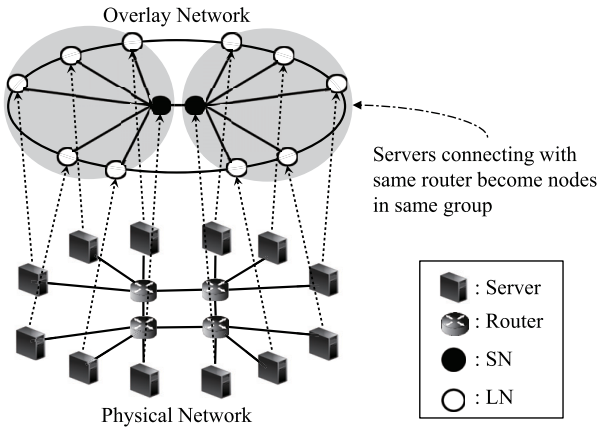

FIGURE 3. An overlay network constructed based on the proposed neighbor selection principle.

B. PHYSICAL NETWORK AWARE NEIGHBOR SELECTION

Since the neighbor selection scheme affects the connectivity of overlay network, we propose a neighbor selection scheme to construct an overlay network that achieves higher connectivity against physical network disruption. The physical network disruption has a specific characteristic such as ''locality'', i.e., servers connecting with the malfunctioning router are removed from the overlay network. Therefore, it is desirable that the servers that are connecting with the same router (or located in the same area) in the physical network become neighboring nodes (or belong to same group) in the overlay network, as shown in Fig. 3. In this neighbor selection principle, most of the links of the removed nodes are also the links to other removed nodes. In other words, this scheme can achieve higher connectivity against the physical network disruption because there remain a lot of links between the surviving nodes.

In order to construct an overlay network based on our neighbor selection principle and optimal network topology, we propose two procedures: (*i*) node joining procedure and (*ii*) network maintenance procedure. The node joining procedure is autonomously executed by a newly joined node (NJN) to distribute the management load to all nodes. In the node joining procedure, an NJN selects the servers that are located in the same area in the physical network as neighboring nodes in the overlay network. On the other hand, SNs periodically execute the network maintenance procedure, which reconstructs network to keep the optimal network topology.

Procedure 1 shows the node joining procedure. First, an NJN receives an inter-group list, L_{inter}, from an SN to know the information that is required for deciding its Affiliation Group (AG). L_{inter} shared between SNs contains the following information: the average degree of the network, the SN's addresses, and the number of nodes in each group. The NJN performs PING-PONG mechanism to all SNs to know the hop count to each SN, where the NJN can know the IP addresses of SNs from L_{inter}. Then, the NJN selects an adequate group g, in which an SN has the minimum hop count from the NJN on physical network, as AG. After joining group g, the NJN receives the intra-group list, L^g_{intra}, from the SN of group g. L^g_{intra} is shared between nodes in the group g, and it contains the address and degree of each node in the group.

Procedure 1 Node Joining Procedure

1: Get inter-group list L_{inter}
2: Select group g as Affiliation Group (AG)
3: Join group g
4: Get intra-group list L^g_{intra} which is shared in group g
5: Establish connection with nodes of group g
6: Get intra-group list $L^{g'}_{\text{intra}}$ which is shared in group g' which is at diagonally-cornered of group g
7: Establish connection with LNs of group g'
8: Update L^g_{intra} and $L^{g'}_{\text{intra}}$

Procedure 2 Network Maintenance Procedure

1: Update the inter-group list L_{inter}
2: **if** $N_{\text{s}} < N'_{\text{s}}$ **then**
3: Group construction process
4: **end if**
5: **if** $\epsilon < \delta$ **then**
6: Regrouping process
7: **end if**

The NJN attempts to establish connection with nodes of group g in order to follow our neighbor selection principle. First, the NJN establishes connection with LNs to construct ring topology. The NJN performs the PING-PONG mechanism to all LNs found in L^g_{intra} in order to find a candidate for the neighboring LN, which has the minimum hop count from the NJN on physical network. Then, the NJN randomly selects a link between the candidate and the neighboring node of the candidate, and inserts itself into the middle of the link. In other words, the NJN breaks the existing link and creates new links between itself and each node that is involved. Additionally, the NJN checks the degree of the SN of its AG and connects to the SN if the degree of the SN is not fully filled.

Moreover, the NJN attempts to establish connections with LNs of group g', which is at diagonally-cornered of group g, as long as the degree of the NJN is lower than the average degree. First, the NJN gets the intra-group list, $L^{g'}_{\text{intra}}$, from the SN of group g' and finds candidates that are LNs having the lower degree than the average degree in group g'. Then, the NJN establishes connections to the candidates. Finally, the NJN updates L^g_{intra} and $L^{g'}_{\text{intra}}$, and transmits them to nodes of each group.

On the other hand, after receiving the updated intra-group list, L^g_{intra}, the SN of group g (referred to as ''ESN'') executes the network maintenance procedure as shown in Procedure 2. First, the ESN updates L_{inter} based on L^g_{intra}, and transmits it to other SNs. Then, the ESN calculates the ideal number of SNs, N'_{s}, by using (2) with the average degree and the total number of nodes found in L_{inter}. When the current number of

SNs, N_s, is lower than the ideal number of SNs, N'_s, the ESN executes a group construction process.

Group construction process – The objective of this process is to construct new groups with the increase in the number of nodes. First, the ESN selects a node from the biggest group as a new SN. The ideal degree of the newly created SN is calculated according to (1). Then, the ESN constructs a new group by dividing the AG of the newly created SN into two smaller groups evenly.

However, since these newly created groups have fewer nodes than other groups, the size of each group is required to be the same. Therefore, the ESN calculates the difference between the largest group size and the smallest group size, δ, where the number of nodes in each group can be found in L_{inter}. If each group size becomes disproportionate, i.e., $\epsilon < \delta$, the ESN executes a regrouping process, where ϵ indicates the disproportion threshold. The network manager configures the value of ϵ, arbitrarily. While the larger value of ϵ decreases the number of regrouping process executions but increases the possibility of having groups with different sizes, the smaller value of ϵ increases the number of regrouping process executions, which results in an unstable network.

Regrouping process – In this process, the ESN restructures the groups so that the size of each group is the same as others. First, the ESN calculates the average group size, G_{ave}. The ESN changes the AG of some nodes so that the size of smaller groups approximates G_{ave}, where the nodes that change the AG are selected in order of nodes that are near the smaller groups on overlay network. If the selected nodes connect to the SN of the prior AG, the ESN breaks the existing link to the SN of prior AG and creates new link between the selected node and the SN of new AG.

C. PHYSICAL NETWORK AWARE TASK ALLOCATION

While the proposed neighbor selection scheme achieves higher connectivity of the overlay network and increases the number of available nodes against physical network disruption, the overlay-based data mining architecture fails to output processing result when all mappers that have same data block are removed. Therefore, in the remainder of this section, we propose a task allocation scheme that can still mine data even when physical network disruption occurs. Since the nodes in the same area are removed by physical network disruption from the overlay network that is constructed based on the proposed neighbor selection scheme, it is clearly understood that choosing nodes of the farthest groups as mappers ensures the existence of at least one redundant data. Therefore, our task allocation principle distributes each replicated data block to distinct nodes in diagonally-cornered groups.

A Reception Node (RN) that received the request from a client starts a task allocation procedure shown in Procedure 3. Here, the notations used in task allocation procedure are summarized in Table 1. First, the RN receives L_{inter} from the SN of its AG to know the number of groups, N_{groups}, and constructs a group list, $G = \{1, 2, \ldots, N_{\text{groups}}\}$. Then, the

Procedure 3 Task Allocation Procedure

1: Given: $G, B, D, T_1, \ldots, T_B$
2: **for** $1 \leq i \leq B$ **do**
3: RP(D, G, T_i)
4: **end for**

TABLE 1. A list of notations used in task allocation procedure.

G	Group list
B	Number of partitioned data blocks
D	Number of replicas of each data block
T_i	Partitioned data block i

Function 1 RP(d, G, T)

1: Select a group $g \in G$ in random manner
2: Select a mapper m from group g
3: Allocate data block T to m
4: $G \leftarrow G - \{g\}$ and $d \leftarrow d - 1$
5: **if** $d > 0$ **then**
6: Select a mapper m' belonging to the group g' which is at diagonally-cornered of group g
7: Allocate data block T to m'
8: $G \leftarrow G - \{g'\}$ and $d \leftarrow d - 1$
9: **if** $d > 0$ **then**
10: Request m' to execute RP(d, G, T)
11: **end if**
12: **end if**

RN partitions a task into B data blocks, $T_1, \ldots, T_B$, where the number of data blocks, B, and the number of replicas of each data block, D, are configured by network managers. Subsequently, the RN executes the function RP(D, G, T_i) on each data block T_i.

The RN allocates data block T with d replications by our task allocation principle, as shown in Function 1. First, the RN selects a group g from the group list, G, in random manner. Then, the RN finds a node m in the selected group g as a mapper by using "flooding scheme" and allocates data block T to the mapper m. Since our task allocation principle selects mappers from different groups, the RN removes the group g from G with decrement of d. Additionally, the RN selects other mapper when the data replication has not been completed (i.e., $d > 0$). The RN selects mapper m' that belongs to the group g', which is at diagonally-cornered of the group g and allocates data block T to the mapper m'. Similarly to initial mapper selection, the RN removes the group g' from G with decrement of d. If more replications are required to allocate, the RN requests the mapper m' to execute the function RP(d, G, T), which will be continued until the value of d becomes 0.

IV. SERVICE AVAILABILITY ANALYSIS

In this section, we mathematically analyze the service availability of overlay-based parallel data mining architecture after

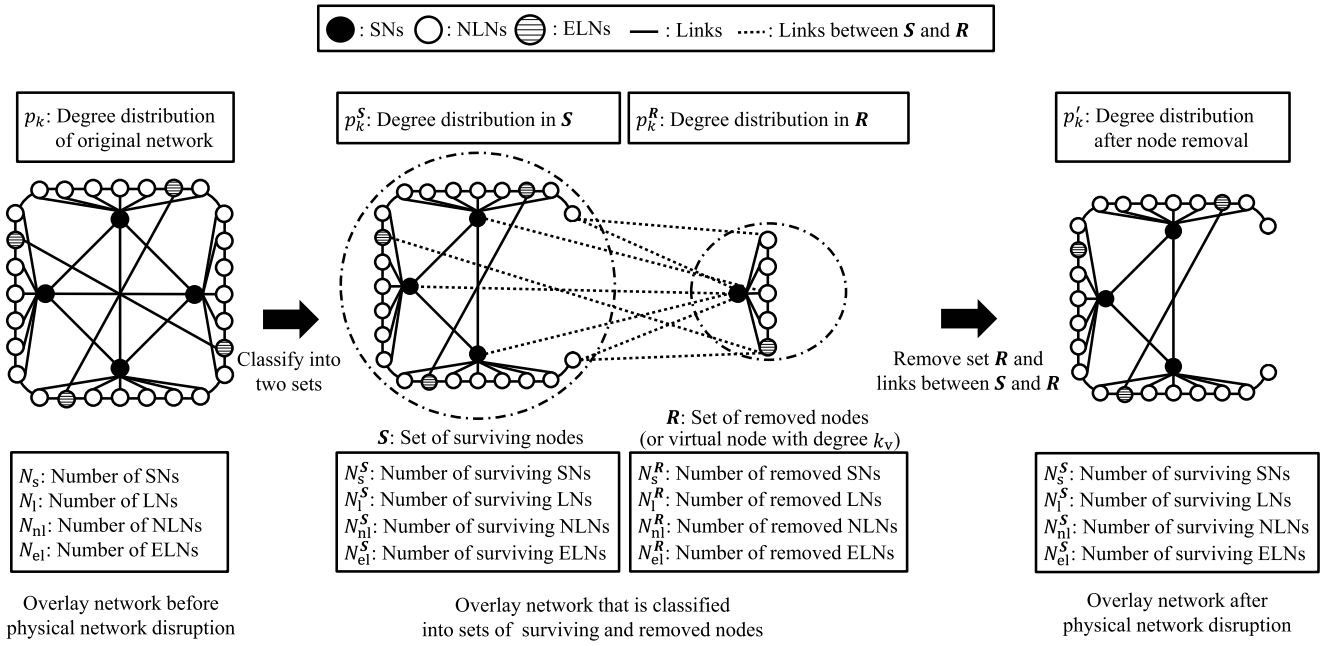

FIGURE 4. An analytical model of the impact of physical network disruption on overlay network (in case of the proposed neighbor selection), and notations defined in each situation.

physical network disruption. The service availability can be derived by the following steps: (*i*) modeling of probability that a node is removed from overlay network by physical network disruption, (*ii*) formulation of the giant cluster ratio of overlay network after physical network disruption, (*iii*) formulation of the probability that a task is successfully processed. In our analysis, the degree distribution is used to model the network topology for the simplicity of mathematical analysis.

A. NODE REMOVAL PROBABILITY

The physical network disruption causes numerous nodes to be removed from the overlay network and the probability that a node is removed differs depending on neighbor selection schemes. Therefore, we model the node removal probability in overlay networks that is constructed based on the existing and proposed neighbor selection schemes, respectively. Here, we define the node removal probability, f_k, which denotes the probability that a node with degree k will be removed from the overlay network.

In the existing neighbor selection scheme, since a newly joining node randomly selects neighboring nodes regardless of the degree of nodes in overlay network and location of nodes in physical network, the physical network disruption causes nodes to be randomly removed from the overlay network regardless of the degree and location. Therefore, the node removal probability in the overlay network that is constructed in random manner, f_k^{rand}, is expressed with the number of removed nodes, $N^{\boldsymbol{R}}$, as follows.

$$f_k^{\mathrm{rand}} = \frac{N^{\boldsymbol{R}}}{N}. \tag{5}$$

On the other hand, in the proposed neighbor selection scheme, a newly joining node selects nodes that belong to the same segment in physical network, as neighboring nodes in the overlay network. Therefore, physical network disruption removes a cluster, which is composed of the removed nodes, from the overlay network. Fig. 4 shows the analytical model of the impact of physical network disruption on the overlay network that is constructed based on the proposed neighbor selection scheme. In order to simplify the analysis, the original overlay network can be classified into sets of surviving and removed nodes, $\boldsymbol{S}$ and $\boldsymbol{R}$, respectively. Here, we can assume that the set of removed nodes, $\boldsymbol{R}$, is a virtual node with degree k_{v}. Therefore, the node removal probability in the overlay network that is constructed based on the proposed neighbor selection, f_k^{pro}, can be defined as follows.

$$f_k^{\mathrm{pro}} = \begin{cases} 1, & \text{if } k = k_{\mathrm{v}}, \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

Due to this virtualization, the degree distribution of the original overlay network, p_k, which is described as (1), gets altered. Let $N_{\mathrm{s}}^{\boldsymbol{S}}$ and $N_{\mathrm{l}}^{\boldsymbol{S}}$ denote the numbers of surviving SNs and LNs, where the total number of surviving nodes, $N^{\boldsymbol{S}}$, is expressed as the sum of $N_{\mathrm{s}}^{\boldsymbol{S}}$ and $N_{\mathrm{l}}^{\boldsymbol{S}}$. Therefore, the degree distribution of the altered overlay network, $\tilde{p}_k$, can be expressed as follows.

$$\tilde{p}_k = \begin{cases} 1/(N^{\boldsymbol{S}}+1), & \text{if } k = k_{\mathrm{v}}, \\ N_{\mathrm{s}}^{\boldsymbol{S}}/(N^{\boldsymbol{S}}+1), & \text{if } k = k_{\mathrm{s}}, \\ N_{\mathrm{l}}^{\boldsymbol{S}}/(N^{\boldsymbol{S}}+1), & \text{if } k = k_{\mathrm{l}}, \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

Moreover, supposed that the degree of leaf nodes is 3 and the number of removed nodes is less than half of the total number of nodes in the original overlay network, k_{v} is expressed as the sum of links between the $\boldsymbol{S}$ and $\boldsymbol{R}$, which are classified into four kinds of links as follows: the links between the removed SNs and surviving SNs, the links between the removed SNs and surviving NLNs, the links between the removed ELNs and surviving ELNs, and the links between the removed LNs and surviving LNs, which construct a ring topology. Therefore, k_{v} is formulated as follows.

$$k_{\mathrm{v}} = N_{\mathrm{s}}^{\boldsymbol{S}} N_{\mathrm{s}}^{\boldsymbol{R}} + \left[N_{\mathrm{s}}^{\boldsymbol{R}}\{k_{\mathrm{s}} - (N_{\mathrm{s}} - 1)\} - N_{\mathrm{nl}}^{\boldsymbol{R}}\right] + N_{\mathrm{el}}^{\boldsymbol{R}} + 2. \tag{8}$$

B. GIANT CLUSTER RATIO

Since the node removal affects the degree distribution of overlay network, we derive the degree distribution after physical network disruption, p'_k, by using the node removal probabilities, which are modeled in previous subsection. p'_k can be expressed as the sum of the probability that nodes with degree i become nodes with degree k after nodes are removed, where $k \leq i$. The nodes in $\boldsymbol{R}$ and links between the $\boldsymbol{S}$ and $\boldsymbol{R}$ are removed by nodes removal probability, f_k (e.g., f_k^{pro} in case of the proposed neighbor selection scheme). Thus, p'_k can be formulated as follows.

$$p'_k = \begin{cases} \sum_{i=0} (f_i)^i p_i^{\boldsymbol{S}}, & \text{if } k = 0, \\ \sum_{i=k} \binom{i}{k} (f_i)^{i-k} (1-f_i)^k p_i^{\boldsymbol{S}}, & \text{otherwise}, \end{cases} \tag{9}$$

where $p_i^{\boldsymbol{S}}$ is the degree distribution in $\boldsymbol{S}$ before the links between the $\boldsymbol{S}$ and $\boldsymbol{R}$ are removed, and is decided as follows.

$$p_i^{\boldsymbol{S}} = \frac{(1-f_i)p_i}{1 - \sum_j f_j p_j}. \tag{10}$$

A cluster that has maximum number of nodes after physical network disruption is referred to as the "giant cluster". The giant cluster ratio, G_{c}, is defined with the number of nodes in a giant cluster, N_{gc}, and the total number of nodes in the original overlay network, N, as follows.

$$G_{\mathrm{c}} = \frac{N_{\mathrm{gc}}}{N}. \tag{11}$$

Considering G_{c} from different perspective, it is expressed with the ratio of the number of nodes that do not belong to the smaller clusters to the total number of nodes in the original overlay network [26]. The smaller clusters can be classified into the clusters that have a single node (shortly referred to as the "single-node clusters") and the clusters that have more than one nodes (shortly referred to as the "multiple-nodes clusters"). The ratio of the nodes that belong to the single-node clusters is expressed as the probability that exists nodes with degree 0 after physical network disruption, p'_0. Additionally, the ratio of the nodes that belong to multiple-nodes clusters is expressed as $\sum_{k=1}^{\infty} p'_k (u_k)^{k-1}$, where u_k denotes the average probability that a link connected to a node with degree k leads to another node that does not belong to the giant cluster. Therefore, G_{c} is formulated as follows.

$$G_{\mathrm{c}} = 1 - p'_0 - \sum_{k=1}^{\infty} p'_k (u_k)^{k-1}. \tag{12}$$

C. SUCCESS PROBABILITY OF DATA MINING

In the parallel processing architecture, each task is partitioned into some data blocks, which are replicated and distributed to distinct mappers, and a reducer successfully executes reduction process if it receives at least one replication of each data block from the mappers. Therefore, the probability that a data mining task is successfully processed, P_{success}, is decreased by the node removal due to physical network disruption. P_{success} is expressed with the number of partitioned data blocks, B, the number of replicas, D, and the probability that there exists a node that has replication i of partitioned data block j in giant cluster, $a_{i,j}$, as follows.

$$P_{\mathrm{success}} = \prod_i^B \left\{ 1 - \prod_j^D (1 - a_{i,j}) \right\}. \tag{13}$$

Moreover, in case of the random task allocation scheme, the mappers are randomly selected regardless of their location. Therefore, the probability that there exists a node that has replication i of data block j in giant cluster, $a_{i,j}^{\mathrm{rand}}$, can be expressed with giant component ratio as follows.

$$a_{i,j}^{\mathrm{rand}} = G_{\mathrm{c}}. \tag{14}$$

For simplicity, the proposed task allocation scheme can be explained in the case of two replicas. In that case, two replicated data blocks are allocated to the mappers belonging to two different diagonally-cornered groups. Let Q be the total number of nodes in the groups locating between the groups having the two mappers. It is easy to see that if the number of removed nodes by physical network disruption is less than $Q + 2$, at least one of the mappers will survive in the overlay network. Therefore, $a_{1,j}^{\mathrm{pro}} = 1$ or $a_{2,j}^{\mathrm{pro}} = 1$ if $N^{\boldsymbol{R}} < Q + 2$.

V. ASSESSMENT OF PHYSICAL NETWORK DISRUPTION EFFECT ON SERVICE AVAILABILITY

In this section, we aim to investigate the effect of physical network disruption on the service availability of data mining. Additionally, we confirm the effectiveness of our proposed architecture in comparison with existing architecture that are designed without considering physical network, i.e., neighboring nodes are randomly selected and data blocks are distributed in a random manner. In this evaluation, we show the number of available nodes and number of tasks that are successfully processed in order to verify the effectiveness of the proposed neighbor selection and task allocation schemes, respectively. Mathematical expressions in previous section are used for our performance evaluation.

We suppose that the physical network follows power-law degree distribution, which is a well known fact, and its topology is a tree structure, where the number of nodes including

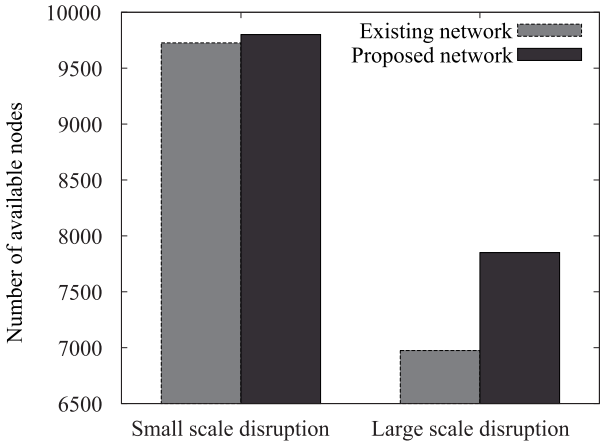

FIGURE 5. **The number of available nodes in different physical network disruption scenarios.**

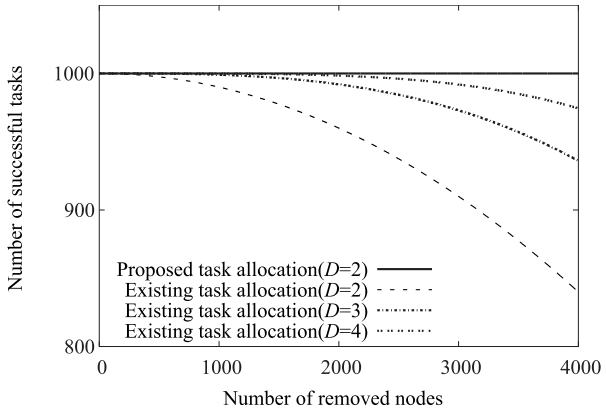

FIGURE 6. **Impact of the number of removed nodes on the number of successful tasks.**

servers and routers is set to 10^4. The overlay network is constructed by all nodes and follows the bimodal degree distribution, where the degree of leaf nodes is set to 3. We suppose that a processing task is partitioned into 5 data blocks and the total number of processing tasks is 10^3. We evaluate the performance of data processing after different types of physical network disruptions occur.

A. PERFORMANCE COMPARISON OF NEIGHBOR SELECTION SCHEMES

In order to verify the effectiveness of the proposed neighbor selection scheme, we evaluate the number of available nodes after a physical network disruption occurs in two overlay networks, i.e., (i) overlay network that is constructed based on the proposed neighbor selection scheme (shortly referred to as the ''proposed network''), and (ii) overlay network that is constructed based on the random neighbor selection scheme (shortly referred to as the ''existing network''). We suppose two kinds of physical network disruption, i.e., small scale disruption (where approximately 2% of nodes are removed from overlay network) and large-scale disruption (where approximately 20% of nodes are removed from overlay network).

Fig. 5 depicts the number of available nodes in different physical network disruption scenarios. While the number of available nodes in the existing network represents the lower value, the proposed network achieves maximum number of available nodes regardless of the physical network disruption scenarios. This is because the proposed overlay network is not disrupted since the removed nodes are located in the same area. Moreover, the proposed network attains much better performance when the number of removed nodes increases. Therefore, we can confirm the effectiveness of the proposed neighbor selection scheme.

B. PERFORMANCE COMPARISON OF TASK ALLOCATION SCHEMES

In the remainder of this section, we verify the effectiveness of the proposed task allocation scheme by comparison of existing scheme in terms of the number of successful task after physical network disruption. While the proposed task allocation replicates 2 times, the number of replicas is set to either 2, 3, or 4 in existing task allocation. In the both cases, the overlay network is constructed based on the proposed neighbor selection scheme.

Fig. 6 demonstrates the number of successful tasks when the number of removed nodes by physical network disruption is varied from 0 to 4000. The existing task allocation scheme falls to an extremely low availability with a progressive increase of number of removed nodes even if the number of replicas increases. On the other hand, the proposed task allocation scheme achieves 100% success probability of data mining with minimum replications regardless of the number of removed nodes because it ensures existence of the nodes that have a data block for processing in giant cluster. It can be concluded that the overlay-based data mining architecture with the proposed neighbor selection and task allocation schemes can execute big data mining with higher success rate and lower processing cost.

VI. CONCLUSION

An overlay-based data mining architecture, which fully distributes management and processing functions by using overlay network technologies, can potentially provide scalable data mining in large-scale network. However, due to physical network disruption, this architecture dramatically decreases service availability of data mining. To solve this problem, we proposed neighbor selection and task allocation schemes based on integration of the overlay and physical networks. In order to improve the success probability of data mining against physical network disruption, our neighbor selection scheme constructs overlay network based on node location in physical network and our task allocation scheme selects nodes from different diagonally-cornered groups in the overlay network as mappers. Moreover, the results obtained from the numerical analysis demonstrated the effectiveness of our proposed schemes in terms of significant improvement in the service availability. Thus, our proposed schemes can be

considered to make big data mining available under the network environment where physical network disruption occurs.

REFERENCES

[1] N. Elmqvist and P. Irani, "Ubiquitous analytics: Interacting with big data anywhere, anytime," *IEEE Computer Magazine*, vol. 46, no. 4, pp. 86–89, Apr. 2013.
[2] F. Gebara, H. Hofstee, J. Hayes, and A. Hylick, "Big data text-oriented benchmark creation for hadoop," *IBM Journal of Research and Development*, vol. 57, no. 3/4, pp. 10:1–10:6, May-Jul. 2013.
[3] J. Dean and S. Ghemawat, "MapReduce: Simplified data processing on large clusters," in *Proc. of 6th Symposium on Operating Systems Design and Implementation*, San Francisco, USA, Dec. 2004, pp. 137–150.
[4] "Hadoop [Online]." Available: http://hadoop.apache.org/.
[5] K. McKusick and S. Quinlan, "GFS: Evolution on fast-forward," *ACM Queue Magazine*, vol. 7, no. 7, pp. 1–11, Aug. 2009.
[6] K. Suto, H. Nishiyama, X. S. Shen, and N. Kato, "Designing P2P networks tolerant to attacks and faults based on bimodal degree distribution," *Journal of Communications*, vol. 7, no. 8, pp. 587–595, Aug. 2012.
[7] S. Ghemawat, H. Gobioff, and S.-T. Leung, "The google file system," in *Proc. of 19th Symposium on Operating Systems Principles*, New York, USA, Oct. 2003, pp. 29–43.
[8] A. Verma, L. Cherkasova, and R. H. Campbell, "Orchestrating an ensemble of MapReduce jobs for minimizing their makespan," *IEEE Transactions on Dependable and Secure Computing*, vol. 10, no. 5, pp. 314–327, Sept.-Oct. 2013.
[9] M. Cardosa, A. Singh, H. Pucha, and A. Chandra, "Exploiting spatio-temporal tradeoffs for energy-aware MapReduce in the cloud," *IEEE Transactions on Computers*, vol. 61, no. 12, pp. 1737–1751, Dec. 2012.
[10] Y. Zhang, Q. Gao, L. Gao, and C. Wang, "Priter: A distributed framework for prioritizing iterative computations," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 9, pp. 1884–1893, Sept. 2013.
[11] D. Warneke and O. Kao, "Exploiting dynamic resource allocation for efficient parallel data processing in the cloud," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 6, pp. 985–997, Jun. 2011.
[12] M. Asahara, S. Nakadai, and T. Araki, "LoadAtomizer: A locality and I/O load aware task scheduler for MapReduce," in *Proc. of IEEE 4th International Conference on Cloud Computing Technology and Science*, Taipei, Dec. 2012, pp. 317–324.
[13] A. Rabkin and R. H. Katz, "How hadoop clusters break," *IEEE Software Magazine*, vol. 30, no. 4, pp. 88–94, Jul.-Aug. 2013.
[14] H. Jin, X. Yang, X.-H. Sun, and I. Raicu, "Large-scale distributed systems at google: Current systems and future directions," in *Keynote Speech at the 3rd ACM SIGOPS International Workshop Large Scale Distributed Systems and Middleware*, Montana, USA, OCT. 2009.
[15] "HDFS Fereration [Online]." Available: http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-hdfs/Federation.html.
[16] Q. Zheng, "Improving MapReduce fault tolerance in the cloud," in *Proc. of IEEE International Symposium on Parallel & Distributed Processing, Workshop and Phd Forum*, Atlanta, USA, Apr. 2010, pp. 1–6.
[17] J. Dean, "Adapt: Availability-aware MapReduce data placement for non-dedicated distributed computing," in *Proc. of 32nd IEEE International Conference on Distributed Computing Systems*, Macau, Jun. 2012, pp. 516–525.
[18] H.-C. Hsiao, H.-Y. Chung, H. Shen, and Y.-C. Chao, "Load rebalancing for distributed file systems in clouds," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 5, pp. 951–962, May 2013.
[19] F. Azzedin, "Towards a scalable HDFS architecture," in *Proc. of International Conference on Collaboration Technologies and Systems*, San Diego, USA, May 2013, pp. 155–161.
[20] J. Zhang, G. Wu, X. Hu, and X. Wu, "A distributed cache for hadoop distributed file system in real-time cloud services," in *Proc. of ACM/IEEE 13th International Conference on Grid Computing*, Beijing, China, Sept. 2012, pp. 12–21.
[21] Z. Yao, X. Wang, D. Leonard, and D. Loguinov, "Node isolation model and age-based neighbor selection in unstructured P2P networks," *IEEE/ACM Transactions on Networking*, vol. 17, no. 1, pp. 144–157, Feb. 2009.
[22] W. Xiao, M. He, and H. Liang, "Cayleyccc: A robust P2P overlay network with simple routing and small-world," *Academy Publisher Journal of Networks*, vol. 6, no. 9, pp. 1247–1253, Sept. 2011.
[23] P. Flocchini, A. Nayak, S. Member, and M. Xie, "Enhancing peer-to-peer systems through redundancy," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 1, pp. 15–24, Jan. 2007.
[24] K. Suto, H. Nishiyama, N. Kato, T. Nakachi, T. Fujii, and A. Takahara, "THUP: A P2P network robust to churn and dos attack based on bimodal degree distribution," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 9, pp. 247–256, Sept. 2013.
[25] T. Tanizawa, G. Paul, R. Cohen, S. Havlin, and H. E. Stanley, "Optimization of network robustness to waves of targeted and random attacks," *Physical Review E*, vol. 71, no. 4, Apr. 2005.
[26] M. E. J. Newman, "Assortative mixing in networks," *Physical Review Letters*, vol. 89, no. 208701, May 2002.

KATSUYA SUTO received the M.S. degree in information science from the Graduate School of Information Sciences, Tohoku University, Sendai, Japan, in 2013, where he is currently pursuing the Ph.D. degree. His research interests are in the areas of cloud computing, parallel data processing, and network robustness. He was a recipient of the prestigious Deans Award from Tohoku University in 2013, the Best Paper Award at the IEEE 79th Vehicular Technology Conference in Spring 2013, and the Institute of Electronics, Information and Communication Engineers (IEICE) Communications Society's Academic Encouragement Award. He is a Student Member of IEICE.

HIROKI NISHIYAMA is an Associate Professor with the Graduate School of Information Sciences, Tohoku University, Sendai, Japan, where he received the M.S. and Ph.D. degrees in information science in 2007 and 2008, respectively. He has authored more than 100 peer-reviewed papers, including many high-quality publications in prestigious IEEE journals and conferences. He was a recipient of the Best Paper Awards from many international conferences, including the IEEE flagship events such as the IEEE Global Communications Conference in 2013 (GLOBECOM'13), GLOBECOM'10, and the IEEE Wireless Communications and Networking Conference in 2012 (WCNC'12). He was also a recipient of the IEEE Communications Society's Asia-Pacific Board Outstanding Young Researcher Award in 2013, the Institute of Electronics, Information and Communication Engineers (IEICE) Communications Society's Academic Encouragement Award in 2011, and the FUNAI Foundation's Research Incentive Award for Information Technology in 2009. He has served as the Co-chair for Cognitive Radio and Networks Symposium of the 2015 IEEE International Conference on Communications (ICC'15), the Co-Chair for Selected Areas in Communications Symposium of the IEEE ICC'14, an Associate Editor of the *IEEE Transactions on Vehicular Technology*, an Associate Editor of the journal of *Peer-to-Peer Networking and Applications* (Springer), and the Secretary of the IEEE ComSoc Sendai Chapter. His research interests cover a wide range of areas, including satellite communications, unmanned aircraft system networks, wireless and mobile networks, ad hoc and sensor networks, green networking, and network security. One of his outstanding achievements is relay-by-smartphone, which makes it possible to share information among many people by using only WiFi functionality of smartphones. He is a member of IEICE.

NEI KATO received the bachelor's degree from Tokyo Polytechnic University, Tokyo, Japan, in 1986, the M.S. and Ph.D. degrees in information engineering from Tohoku University, Sendai, Japan, in 1988 and 1991, respectively. He was promoted to Full Professor at the Graduate School of Information Sciences, Tohoku University, in 2003. He currently serves as a Member-at-Large on the Board of Governors and the IEEE Communications Society, the Chair of the IEEE Ad Hoc and Sensor Networks Technical Committee, the Chair of the IEEE ComSoc Sendai Chapter, the Associate Editor-in-Chief of the *IEEE Internet of Things Journal*, an Area Editor of the *IEEE Transactions on Vehicular Technology*, and an Editor of the *IEEE Wireless Communications Magazine* and the *IEEE Network Magazine*. He has served as the Chair of the IEEE ComSoc Satellite and Space Communications Technical Committee (2010–2012). His awards include the Minoru Ishida Foundation Research Encouragement Prize (2003), the Distinguished Contributions to Satellite Communications Award from the IEEE ComSoc, Satellite and Space Communications Technical Committee (2005), the FUNAI Information Science Award (2007), the TELCOM System Technology Award from the Foundation for Electrical Communications Diffusion (2008), the Institute of Electronics, Information and Communication Engineers (IEICE) Network System Research Award (2009), the IEICE Satellite Communications Research Award (2011), the KDDI Foundation Excellent Research Award (2012), the IEICE Communications Society Distinguished Service Award (2012), five Best Paper Awards from the IEEE GLOBECOM/WCNC/VTC, and the IEICE Communications Society Best Paper Award (2012). In addition to his academic activities, he also serves on the expert committee of the Telecommunications Council, Ministry of Internal Affairs and Communications, and as the Chairperson of ITU-R SG4 and SG7, Japan. He is a fellow of the IEICE.

KIMIHIRO MIZUTANI is a Researcher with NTT Network Innovation Labs, Tokyo, Japan. He received the M.S. degree in information system from the Nara Institute of Science and Technology, Ikoma, Japan, in 2010. His research interest is future Internet architecture. He was a recipient of the Best Student Paper Award from the International Conference on Communication Systems and Application in 2010, and the research awards from the IEEE Computer Society and the Information Processing Society of Japan and the Institute of Electronics, Information and Communication Engineers (IEICE) in 2010 and 2013, respectively. He is a member of IEICE and the IEEE Communication Society.

OSAMU AKASHI received the B.Sc. and M.Sc. degrees in information science and the Ph.D. degree in mathematical and computing sciences from the Tokyo Institute of Technology, Tokyo, Japan, in 1987, 1989, and 2001, respectively. He joined NTT Software Laboratories, Tokyo, in 1989, and is a Senior Research Scientist with NTT Network Innovation Laboratories, Tokyo. His research interests are in the areas of distributed systems, multiagent systems, and network architectures. He is a member of the Association for Computing Machinery, the Institute of Electronics, Information and Communication Engineers, the IEEE Computer Society and the Information Processing Society of Japan, and the Japan Society for Software Science and Technology.

ATSUSHI TAKAHARA received the B.S., M.S., and Dr. Eng. degrees from the Tokyo Institute of Technology, Tokyo, Japan, in 1983, 1985, and 1988, respectively. He joined NTT LSI Laboratories, Tokyo, in 1988, where he has been involved in formal methods of VLSI design, reconfigurable architectures, and IP processing. From 2003 to 2008, he was the Director of the Visual Communications Division at the Department of Service Development and Operations, NTT Bizlink Inc., Tokyo, where he was involved in developing and operating an IP-based visual communication service. From 2008 to 2011, he was the Executive Manager of the Media Innovation Laboratory at NTT Network Innovation Laboratories, Tokyo, where he has been the Director of NTT Network Innovation Laboratories since 2011. His current research interests are in IP networking for real-time communication applications and IP infrastructure technologies. He is a member of the Association for Computing Machinery, the Institute of Electronics, Information and Communication Engineers, and the IEEE Computer Society and the Information Processing Society of Japan.