IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 22 August 2022, accepted 14 September 2022, date of publication 30 September 2022,
date of current version 10 October 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3211066

RESEARCH ARTICLE

New Neighbor Selection Method for Blockchain Network With Multiple Regions

HIROSHI MATSUURA[1], **YOSHINORI GOTO**[2], **AND HIDEHIRO SAO**[3]
[1]NTT Network Service System Laboratories, Tokyo 180-8585, Japan
[2]NTT Advanced Technology Corporation, Tokyo 180-0006, Japan
[3]Nippon Telegraph and Telephone West Corporation, Miyakojima-ku, Osaka 534-0024, Japan

Corresponding author: Hiroshi Matsuura (hiroshi.matsuura.gt@hco.ntt.co.jp)

ABSTRACT One of the features of the Bitcoin network today is that its participating nodes are located in different regions of the world. Generally, inter-region data transmission is relatively time-consuming compared with intra-region data transmission. Thus, an important challenge for a blockchain network is shortening the block propagation time in order to reduce forks and maintain fairness, i.e., similar mining durations, for all miners. Previous methods have tried to increase the block propagation speed at the expense of imposing a higher burden on each node and a higher risk of eclipse attack. This paper proposes a new neighbor selection method that is based on only the neighbor's regional information and assumes that a node has a relatively small number of neighbors located outside its region. By using this simple method, the distribution of blocks throughout the network becomes faster and the random neighbor-selection nature of the blockchain network is kept intact; thus, risk of an eclipse attack is low. This paper also proposes a block propagation model over multiple regions for exploring the theoretical reasons for the effectiveness of the proposed neighbor selection method. Finally, it examines a migration scenario to the proposed method from the default neighbor selection method implemented in Bitcoin nodes, and evaluates the migration effects in various sizes of networks.

INDEX TERMS Blockchain, fork, P2P, neighbor selection, propagation delay, eclipse attack.

I. INTRODUCTION

Blockchain was proposed [1] to prevent the double spending problem [2] in P2P (peer-to-peer) networks. Since then, blockchain technology has been used for financial purposes, most famously in Bitcoin [3], [4] and Ethereum [5]. In addition, blockchain technology is being considered for other purposes, such as cloud storage [6], Internet of Things (IoT) [7], [8], [9], healthcare [10], [11], and law enforcement [12], [13], [14]. The International Telecommunication Union (ITU) has summarized the various blockchain use cases, which include information and communications technology (ICT), arts, culture, entertainment, and supply chain management [15]. Consequent with the surge in these demands, blockchain service platforms such as Hyperledger Fabric [16] and NutBaaS [17] have been proposed.

A feature of a blockchain network is its decentralized verification mechanism, wherein the participating nodes can be located in various regions of the globe. In addition, while new blockchain networks start with a smaller number of nodes (e.g., around 500), the Bitcoin network, for example, currently has grown to have more than 10,000 nodes. For such circumstances, this paper proposes a neighbor selection method that can speed up block propagation in blockchain networks of various sizes with multiple regions.

One of the important effects of speeding up block propagation is reduction of the fork rate. Fork models [18], [19] assume that the forking probability decreases as the block propagation time throughout the network is reduced. Another effect of faster block propagation is that the miner nodes in a blockchain network can be made to have similar mining start times. The miner who created the previous block has the earliest mining start time for the next block and the miner who is the latest in receiving the block in the network has the

The associate editor coordinating the review of this manuscript and approving it for publication was Barbara Guidi.

latest mining start time; thus, reducing the block propagation time is beneficial for maintaining similar mining durations, i.e., fairness, among miners.

In the default configuration, each Bitcoin node picks eight nodes randomly as its outbound neighbors [20], but this random selection may lead to inappropriate neighbors being chosen for fast block propagation. Relay networks like Falcon [21] and FABRE [22] can make a network much faster than a peer-to-peer network by using relay gateways. However, in terms of fairness to miners, the miners near the relay gateways have the advantage of spreading their newly mined blocks faster than miners farther away.

A neighbor selected from inside a region is defined as an inside neighbor and a neighbor selected from outside a region is defined as an outside neighbor. In [23], we proposed each node selects its neighbors randomly from inside and outside the region to which it belongs. Because the neighbor selection is random, it is resilient against eclipse attacks. In addition, the only restriction on neighbor selection is fewer number of outside neighbors for each node, so its computational burden is light. This paper extends this neighbor selection method. The following summarizes the work it presents.

- The number of outside neighbors for each node is set to one or two, which is much smaller than that of the total random neighbor selection. This feature of having fewer outside neighbors helps to shorten block propagation times not only in each region, but also in outside regions. To demonstrate this effect, this paper analyzes a block propagation wave model in detail; the theoretical equations based on this model prove that setting one or two outside neighbors to each node speeds up block propagation throughout the world.
- Simulation results show that the theoretically-proved best number of outside neighbors also gives the fastest block propagation speed in each network with 500–6500 nodes. In addition, the proposed method with fewer outside neighbors is superior to previous methods in terms of the block propagation speed in each network. We analyzed the superiority of the proposed method based on the simulation results and the theoretical equations proposed in this paper.
- This paper also describes a migration scenario from total random neighbor selection, which is the default method for Bitcoin nodes, to the proposed method. For this purpose, the ratio of the nodes using the proposed method is gradually increased from 0% to 100%. It is found that there is no network where the average block propagation speed is degraded because of adoption of the proposed method. Rather, even a small infusion of nodes implementing the proposed method significantly reduces the block propagation times.

The remainder of this paper is as follows. Section II shows related works of this study, Section III provides background information, while Section IV presents the proposed neighbor selection method, the new block propagation model with multiple regions, and its theoretical advantages over previous methods. Section V describes the evaluation results using a simulator and discusses the applicability of the proposed method to blockchain networks. Section VI concludes this paper.

II. RELATED WORKS

Many researchers have tried to shorten the block propagation time in a blockchain network by periodically replacing the neighbors of each node with nodes that have faster block propagation speeds (which is called the Aoki method) [24], wider bandwidth [25], or shorter physical distances [26], [27], [28]. These neighbor selection methods, however, have two drawbacks. First, they are vulnerable to eclipse attacks [29], [30], [31]. That is, if an attacker aims to attack a target node, the attack nodes will send blocks quickly, or the attacker will place its nodes near the target node. In these ways, the attacker can allocate its malicious nodes as the neighbors of the target node without difficulty. Second, if the block propagation speed or the bandwidth of a node is used for the neighbor selection criterion, each node has to monitor all of its neighbors all the time, so its computational burden of neighbor selection will increase.

Besides, neighbors can be chosen on the basis of their mining power and their node degree [32]. The cited paper recommended that each node should choose neighbors with larger mining power or with higher node degree, but information on these features is not exchanged in the current blockchain protocols. Furthermore, the information may not be accurate because some nodes in a blockchain network may have an incentive to send false information to get other nodes to connect to them.

There is another approach that uses a minimum spanning tree (MST) for the neighbor selection [33]. In that method, a leader node in a blockchain network creates an MST in the network by taking all the network edge costs into consideration. However, it takes a long time for a leader node to gather all the necessary information for creating an MST; it is also time-consuming to notify all the other nodes of the created MST especially when there are many nodes in the network. In addition, there is only one route between two nodes on an MST, meaning that a created block may not be delivered to some nodes if one of the edges of the MST has a problem. The method proposed in this paper, on the other hand, does not seek the optimal block propagation time by considering all the edge information in the world. Rather, it focuses on making the most effective use of the local IP addresses of the neighbor candidates for faster block propagation. Thus, in the following sections, we compare the proposed method with other neighbor selection methods that use local neighbor information rather than network-wide information.

Note that one of the important characteristics of a public blockchain network is the high level of anonymity of the participating nodes. There are previous studies on identifying users in the Bitcoin network by linking their IP addresses with their transactions [34]. However, it is difficult to identify the real IP addresses of the Bitcoin users if they use a VPN or TOR, i.e., the Onion routing project. In these cases, other Bitcoin nodes can recognize the disclosed VPN servers and TOR exit routers, but the real IP addresses of the users are concealed from them. In this paper, it is assumed that each node utilizes only the disclosed IP addresses in the blockchain network for selecting its neighbors, even if their real IP addresses are different from the disclosed ones. This is because these disclosed nodes are the closest gateways in the blockchain network for receiving or sending blocks to the users.

Note that some permissioned private blockchains are used within only one region [35], [36]. The proposed neighbor selection method is not for such private blockchains; it is rather for public permissionless blockchains in which the participating nodes are located in multiple regions, such as the Bitcoin and Ethereum networks.

III. RESEARCH BACKGROUND

A. BLOCKCHAIN PROTOCOL

The blocks are distributed in the blockchain network and stored in a ledger that is possessed by all the blockchain nodes. Miners compete to find a nonce, i.e., an arbitrary number to be used just once in a cryptographic communication, that can create a satisfactory hash value for the new block by using the hash value of the previous block. In this way, a miner's ability to generate a new block is determined by the power of its computer, so this block generation process is called a proof of work (PoW) system. Any node in a blockchain network can become a miner, so we will use miners and nodes interchangeably in the rest of the paper.

After a new block is created, the miner broadcasts the block over the network through its outbound neighbor nodes. The nodes that receive the incoming block verify it and add it to their local copy of the blockchain. The block transmission protocol between two nodes is shown in Fig. 1. First, node_1 sends an *inv* message to its outbound neighbor node_2 after the verification process. If node_2 does not have the block, it sends a *getdata* message to node_1 to request the block. A block can be up to 1 MB, while an *inv* message has only a few bytes. Therefore, by using this protocol, unnecessary transmissions of blocks containing a large amount of data are avoided.

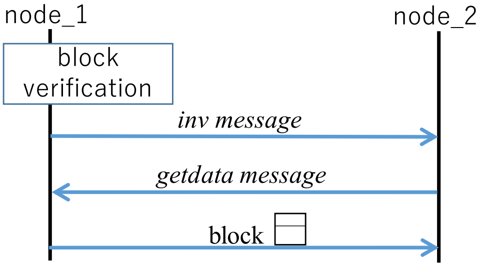


FIGURE 1. Block transmission protocol between neighbors.

B. BLOCKCHAIN P2P NETWORK

Blockchain nodes transmit transactions/blocks to their outbound neighbors through the transmission control protocol (TCP)/ Internet protocol (IP). Each node has *new* and *tried* tables to store the IP addresses of blockchain nodes in the same network. The *new* table is initially set using information on a domain name system (DNS) server when the node first joins the network. Nodes update their *new* table periodically, reflecting recent neighbor information by using *addr* messages. The *tried* table maintains the IP addresses of neighbors to which the node had a connection.

In the default configuration [20], a Bitcoin node picks eight nodes from the tables randomly for setting its outbound neighbors, to which blocks or transactions are broadcast. This paper calls this outbound neighbor selection method total random selection. There are also inbound neighbors, which are selected in response to the neighbor addition requests by other nodes. One node can have at most 125 neighbors.

Its P2P nature, however, makes a blockchain network vulnerable to various malicious attacks, such as distributed denial of service (DDoS) attack, eclipse attack, and sybil attack [37]. Among them, eclipse attacks are sometimes facilitated by an inappropriate neighbor selection. Fig. 2 shows an example of an eclipse attack on a blockchain network. The target node is surrounded by seven malicious nodes out of eight outbound neighbors. These malicious nodes can manipulate the blocks received from the target node and send the manipulated blocks to the network through their outbound neighbor nodes. These malicious nodes can also stop propagating the block, so that the target node becomes isolated from the network. In these ways, an eclipse attack can ruin the target node in the blockchain network. In this example, outbound neighbors are used for the attack, but inbound neighbors can be also used, by their not sending important information, such as transactions and newly mined blocks, to the target nodes.

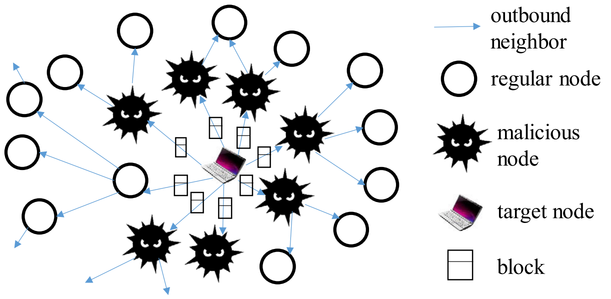


FIGURE 2. Eclipse attack example in blockchain network.

The previous neighbor selection methods have narrow criteria for choosing neighbors for each node, such as block propagation speed, bandwidth, and physical distance between the neighbor and the node. In these cases, the selected neighbors can be easily replaced with malicious nodes for an eclipse attack. The TendrilStaller attack [38] is a kind of eclipse attack that replaces high-bandwidth mode neighbors of the target node with malicious nodes in the Bitcoin network.

IV. PROPOSED NEIGHBOR SELECTION METHOD

A. DESCRIPTION OF METHOD AND ITS MERITS

Fig. 3 shows the data structure and procedure each node uses to choose its outbound neighbors and inbound neighbors. In this example, the upper limit of inside neighbors is six, and the upper limit of outside neighbors for each node is two. As explained in Sect. III, each blockchain node downloads information on the IP addresses of the network from a DNS server and it stores this information in the *new* table. ICANN (Internet Corporation for Assigned Name and Numbers) [39] allocates IP addresses region by region. Thus, from the IP address of a node in the *new* table, blockchain node A can determine if it belongs to the same region as node A. In the figure, there are two outside neighbors (F, K) and six inside neighbors (C, M, D, H, FJ, G) in the outbound neighbor list, while there are one outside neighbor (E) and five inside neighbors (Y, IG, H, MH, KI) in the inbound neighbor list.

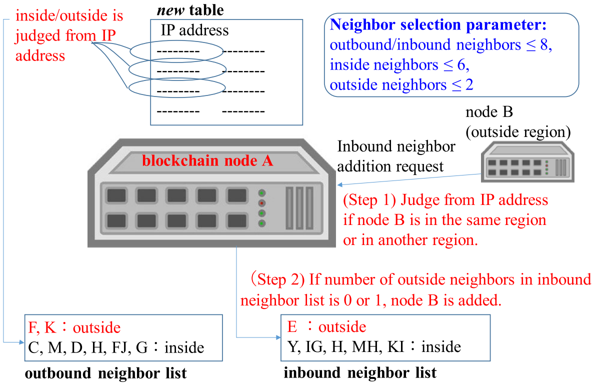

inside/outside is judged from IP address
new table
IP address
-------- --------
-------- --------
-------- --------
-------- --------
Neighbor selection parameter:
outbound/inbound neighbors ≤ 8,
inside neighbors ≤ 6,
outside neighbors ≤ 2
node B
(outside region)
Inbound neighbor addition request
blockchain node A
(Step 1) Judge from IP address if node B is in the same region or in another region.
（Step 2) If number of outside neighbors in inbound neighbor list is 0 or 1, node B is added.
F, K：outside
C, M, D, H, FJ, G：inside
outbound neighbor list
E ：outside
Y, IG, H, MH, KI：inside
inbound neighbor list

FIGURE 3. **Proposed neighbor selection implementation in each node.**

Bitcoin nodes are grouped into six regions: North America, Europe, South America, Asia Pacific, Japan, and Australia. According to data provided by the SimBlock simulator, in 2019, North America had 33.16% of the total number of Bitcoin nodes, Europe 49.98%, South America 0.9%, Asia Pacific 11.77%, Japan 2.24 %, and Australia 1.95% [40]. Under these conditions, when outbound neighbors are randomly selected, at least half of them will be outside neighbors on average. In contrast, the proposed method determines a smaller upper limit of outside neighbors compared with the total random selection method. For instance, in Fig. 3, the upper limit of the outside neighbors is set to two, so one or two neighbors in the outbound neighbor list are outside node A's region.

In the proposed neighbor selection method, the condition on the inbound neighbors can optionally be set to be the same as that for the outbound neighbors. That is, the upper limit of inside neighbors and the upper limit of outside neighbors can be set for inbound neighbors in the same way as for outbound neighbors. In Fig. 3, if blockchain node A receives an inbound neighbor addition request from node B, it judges if node B is located in the same or an outside region from its IP address. Then, if node B is located in an outside region and if there is one or no outside neighbor in the inbound neighbor list, node B is added to the list. Moreover, if node B is located in the same region and if there are less than six inside neighbors in the inbound neighbor list, node B is added to the list.

In this way, except for the upper limit of inside/outside neighbors, the neighbors of each node are randomly selected.

The previous methods have narrow neighbor selection criteria that makes easy to attack the target node, such as faster block propagation, wider bandwidth, and shorter distance. Here, if the criterion is faster block propagation, an eclipse attacker can place its attack nodes next to the target node and send it only the most recently created blocks. If the criterion is wider bandwidth, the attack nodes can allocate a large bandwidth when sending blocks to the target node, and if the criterion is shorter distance, the attack nodes should be located as close as possible to the target node. By taking advantage of its neighbor selection criteria, the proposed method can avoid the risks of these eclipse attacks.

However, it is conceivable that an eclipse attacker could put its attack nodes in the same region as the targeted node and thereby exploit the feature that the proposed method sets each node to have more inside neighbors. The probability that all six inside neighbors of a target node using the proposed method are replaced with attack nodes can be calculated as

$$P_{rep(6,6)} = \prod_{i=0}^{5} \frac{N_A - i}{N_R - i}, \tag{1}$$

where N_A is the number of attack nodes and N_R is the total number of nodes, except for the target node, in the region where the target node is located. $P_{rep(6,6)}$ in (1) is obtained because the target node happens to select an attack node as a neighbor six times in a row, and the probability of selecting $(i+1)$th attack node is $\frac{N_A - i}{N_R - i}$.

In the same way, the probability that x out of six inside neighbors are replaced with attack nodes is calculated as

$$P_{rep(6,x)} = {}_6C_x \frac{\prod_{i=0}^{x} (N_A - i) \prod_{j=0}^{5-x} (N_R - N_A - j))}{\prod_{k=0}^{5} (N_R - k)}. \tag{2}$$

$P_{rep(6,x)}$ in (2) is obtained because the probability of selecting the $(i+1)$th attack node in the $(k+1)$th inside neighbor selection is $\frac{N_A - i}{N_R - k}$, and the probability of selecting the $(j+1)$th non-attack node in the $(k+1)$th inside neighbor selection is $\frac{N_R - N_A - j}{N_R - k}$, and there are ${}_6C_x$ different combinations depending on when (from 1^{st} to 6^{th} inside neighbor selection) these x attack nodes are selected as neighbors by the target node.

Here, we define $P_{rep(6,x>n)} = \sum_{x=n}^{5} P_{rep}(6, x+1)$ as the probability that the target node has more than n of its inside neighbors replaced with attack nodes, which is equivalent to saying that more than n out of 8 of its neighbors are replaced with attack nodes. Fig. 4 plots $P_{rep(6,x>n)}$ $(0 \leq n \leq 5)$ versus the number of attack nodes $(100 \leq N_A \leq 1000)$ in a region with 2001 nodes, which means $N_R = 2000$. As shown, if

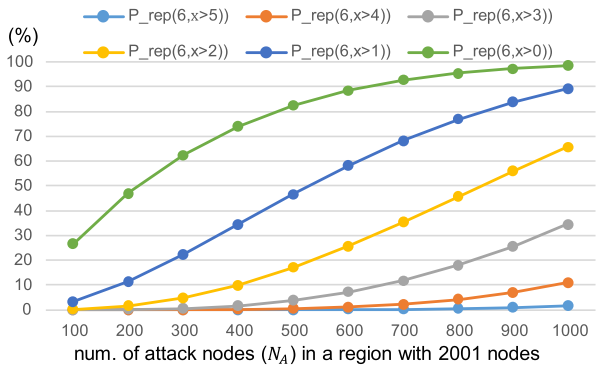


FIGURE 4. Values of $P_{rep(6,x>n)}$ based on attack node ratios.

$N_A = 100$, $P_{rep(6,x>2)}$ is 0.21%; i.e., there is almost no chance that the target node has more than two of its neighbors replaced with attack nodes if the target node uses the proposed method. $P_{rep(6,x>1)}$ is about 3%, and $P_{rep(6,x>0)}$ is about 26.5%; thus, it is unlikely that even one node will be replaced with an attack node if $N_A \leq 100$. As N_A increases to 500 or more, $P_{rep(6,x>0)}$ becomes more than 80%, so the target node is likely to have at least one attack node as its neighbor. However, $P_{rep(6,x>3)}$ remains relatively low, at about 34%, even when N_A is 1000, i.e., half of the region nodes are attack nodes. That is to say, even if half of the region nodes are attack nodes, it is unlikely that half or more of the target node's neighbors are attack nodes.

On the other hand, if each node in a region has a narrow criterion like those above for selecting its six neighbors, and the attack nodes know the criterion, it is inevitable that the target node will have five or more of its neighbors replaced with the attack nodes, even if the number of attack nodes is less than 100 in the same scale of network with 2001 nodes. Thus, when the number of participating nodes is large, an eclipse attacker will find it much more difficult to attack a target node that uses the proposed neighbor selection method than a target node using the other neighbor selection methods.

Moreover, the inside/outside judgement for each node can be done without imposing a substantial computational burden. This is because, except for the neighbor selection based on the IP addresses of the neighbor candidates, there is no extra burden imposed on each node. In particular, if each node separates IP addresses of the nodes located in the same region from the other IP addresses in the *new* table, the burden of choosing eight outbound neighbors is the same as that of choosing eight outbound neighbors in the random selection method. This is because each node can randomly select six IP addresses from the same region and select two IP addresses from the other regions.

On the other hand, if each node has to acquire the neighbor candidates' computational power, distance, or available bandwidth, it has to allocate $O(N)$ memory space, where N is the number of nodes in the network, to evaluate its neighbor candidates. This is because every node in the network can become a neighbor of any other node; thus, all possible neighbor candidates should be evaluated using a determined criterion. In addition, each node has to update the ranking of the neighbor candidates whenever the criterion value in each neighbor candidate changes, of which the corresponding time complexity is $O(\log(N))$ if each node uses a binary search tree for managing the rank of its neighbor candidates. In contrast, each node implementing the proposed method does not have to allocate this additional $O(N)$ memory space and does not have to use its CPU power for ranking its neighbor candidates.

B. THEORETICAL ANALYSES USING BLOCK PROPAGATION MODEL

1) MOTIVATION AND PROPOSAL

Table 1 shows the matrix of average transmission delays among the six regions in the Bitcoin network in 2019 according to the SimBlock simulator. Table 2 shows the matrix of ping transmission times between cities located in the six regions, as measured on an Internet site [41] in July, 2022.

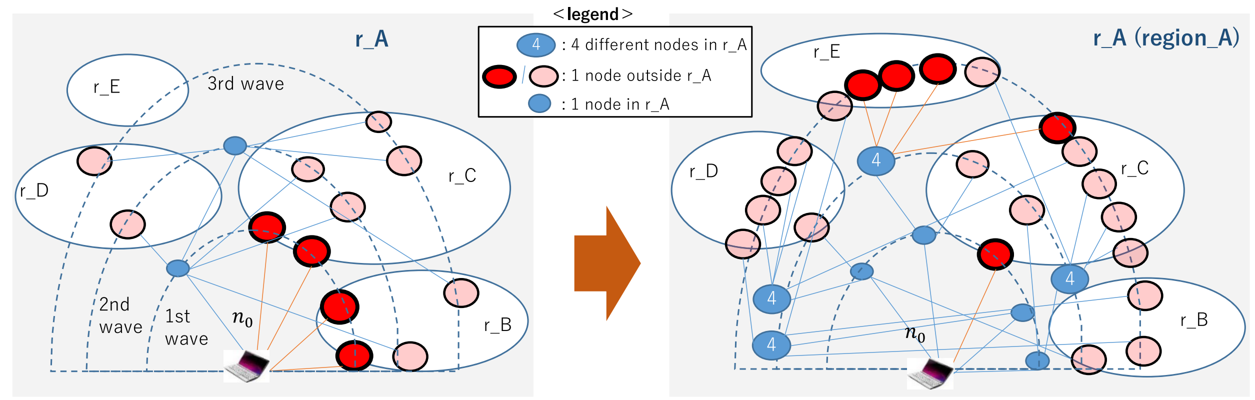
TABLE 1. Transmission delay matrix in bitcoin NW in 2019 (unit: ms).

	North America	Europe	South America	Asia-Pacific	Japan	Australia
North America	32	124	184	198	151	189
Europe	124	11	227	237	252	294
South America	184	227	88	325	301	322
Asia-Pacific	198	237	325	85	58	198
Japan	151	252	301	58	12	126
Australia	189	294	322	198	126	16

TABLE 2. Ping transmission time matrix among 6 regions in 2022 (unit: ms).

destination source	North America (Dallas)	Europe (Paris)	South America (Santiago)	Asia-Pacific (Hong Kong)	Japan (Tokyo)	Australia Sydney)
North America (New York)	44	73	126	227	177	208
Europe (Frankfurt)	125	10	208	209	216	295
South America (Bogota)	96	159	67	260	213	264
Asia-Pacific (Seoul)	183	278	292	60	34	181
Japan (Osaka)	144	248	262	58	8	195
Australia (Melbourne)	211	239	325	230	138	13

From Table 1, the Bitcoin inter-region transmission delays were on average 5.2 times larger than the inner-region transmission delays in 2019. From Table 2, the inter-region ping transmission times were on average 5.9 times longer than the inner-region ping transmission times. Generally, these ping times are proportional to the distance between the source and destination cities, but we also found that if the distances between source/destination pairs are similar, pairs within a



(a) Total random selection : 80% outside neighbors
→Only max 4 outside nodes receive block from r_A in each wave

(b) Proposed method : 20% outside neighbors
→Max 4^{n-1} outside nodes receive block from r_A in *nth* wave

FIGURE 5. Concept of proposed method: Fewer outside neighbors contribute faster block propagation in and outside region.

region tend to have a shorter transmission time. For example, the distance between Los Angeles and New York is about 3936 km, while the distance between Dallas in the US and Bogota in Columbia is about 3909 km, which is a little shorter. The ping transmission time between Los Angeles and New York (73 ms) was, however, about 24% shorter than that between Bogota and Dallas. Therefore, we consider it is beneficial to reduce the number of outside neighbors for each node so that the inter-region transmissions are reduced.

To show the effectiveness of having fewer outside neighbors for each node, let us consider a network with five different regions (r_A $\sim$ r_E) as shown in Fig. 5. Here, each node has five outbound neighbors. In the figure, n_0 indicates the miner that has created the new block in r_A, and the block is sent to its outbound neighbors in the first wave. In the second wave, the nodes that received the block in the first wave send it to their outbound neighbors.

In Fig. 5(a), each node uses the total random selection method, so it has just one inside neighbor and four outside neighbors on average; i.e., at most four outside neighbors can receive the block directly from r_A. Here, r_E has not received the block directly from r_A by the third wave. Note that indirect block reception whereby some of the nodes in r_E receive the block from nodes via another region except for r_A is not considered in Fig. 4, as it would entail multiple inter-region transmissions taking much longer for the block to reach r_E. In addition, there is only one inside neighbor for each node in (a), so it takes a long time until all the nodes in r_A can receive the block if there are many nodes in r_A.

In Fig. 5(b), on the other hand, there is just one outside neighbor, so the maximum number of nodes (4^{n-1}) can receive the block in the nth wave. In the third wave, the number of outside nodes receiving the block directly from r_A is much larger than in (a). In addition, the maximum number of inside nodes receiving the block in r_A is 4^n in the nth wave, so the block propagations in each region are much faster than in (a).

As can be seen in Fig. 5, the proposed method offers faster block propagation that is independent of the block size. Thus, having fewer outside neighbors not only speeds up block propagation but also small-sized transactions over a network. We suppose, therefore, that the proposed neighbor selection method enhances the decentralized nature of a blockchain network by enabling all the nodes in the network to share the same transactions and blocks in a faster manner regardless of their regions.

2) ANALYSES USING BLOCK PROPAGATION MODEL WITH MULTIPLE REGIONS

Fig. 6 shows the block propagation wave model in a region based on [43], starting from the miner labeled n_0. In the figure, each blue arrow shows a block transmission between nodes and each red arrow shows an *inv* message sent to a node to which the block is not delivered because the node has already received the block. In the following, $|W_k|_{in}$ is defined as the expected number of nodes that receive the block created by miner n_0 in the kth wave in the region. It is calculated as

$$|W_k|_{in} = \sum\nolimits_{m=1}^{|W_{k-1}|_{in}} |W_{k(m)}|_{add\_i}, \tag{3}$$

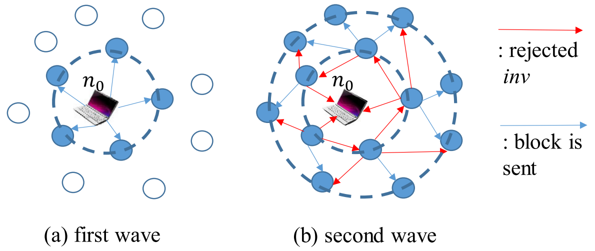
where $|W_{k(m)}|_{add\_i}$ is the expected number of nodes receiving the block from the mth node in the kth wave in the region. In the $(k-1)$th wave, $|W_{k-1}|_{in}$ nodes are expected to receive the block. These nodes send the block to their inside outbound neighbors in the kth wave in the region.

Denoting the number of inside outbound neighbors as A, $|W_{k(m)}|_{add\_i}$ can be expressed as

$$|W_{k(m)}|_{add\_i} = Ap_{f_{k(m)}}, \tag{4}$$

where $p_{f_{k(m)}}$ is the block forwarding probability from the mth node in the kth wave in the region. For the mth node in the kth wave, $p_{f_{k(m)}}$ for each inside neighbor is expressed as

$$p_{f_{k(m)}} = \frac{N_{in} - 1 - \sum_{j=0}^{k-1} |W_j|_{in} - \sum_{j=0}^{m-1} |W_{k(j)}|_{add\_i}}{N_{in} - 1}, \tag{5}$$



(a) first wave (b) second wave

FIGURE 6. Block propagation wave model in a region.

where N_{in} is the number of nodes in the region. The value of the denominator in (5), $N_{in} - 1$, indicates the number of nodes except for n_0 in the region. The value of the numerator indicates the expected number of nodes that have not received the block before the mth node in the kth wave sends it. $\sum_{j=0}^{k-1} |W_j|_{in}$ indicates the expected number of nodes having received the block by the $(k-1)$th wave, and $\sum_{j=0}^{m-1} |W_{k(j)}|_{add\_i}$ indicates the expected number of nodes having received the block in the kth wave before the mth node sends the block to its outbound neighbors. (5) indicates that as the number of waves and the number of nodes having sent the block become larger, the block forwarding probability in the region goes down.

Now let us analyze the block propagation wave model with multiple regions in detail. Fig. 7 shows a block propagation wave model from one region to its outside regions. The block created by miner n_0 is propagated to the five outside regions. Except for R_1, in all the outside regions, at least one node has received the block by the third wave directly from the region where n_0 is located. As such, it is critical to send the block to at least one node in each region with fewer waves from n_0 without passing through multiple regions in order to propagate the block in a swift manner. As is discussed above, it is not helpful to count the nodes receiving the block that has passed through multiple regions because of the long transmission delay.

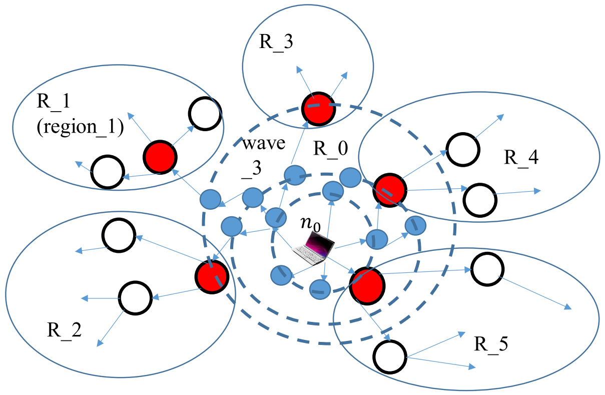


FIGURE 7. Wave model of block propagation to outside regions.

$|W_k|_{out}$ is the expected number of nodes in the outside regions that receive the block created by n_0 in the kth wave directly from the region where n_0 is located; it is calculated as

$$|W_k|_{out} = \sum_{m=1}^{|W_{k-1}|_{in}} |W_{k(m)}|_{add\_o}, \tag{6}$$

where $|W_{k(m)}|_{add\_o}$ is the expected number of nodes receiving the block from the mth node in the kth wave outside the region. In the $(k-1)$th wave, $|W_{k-1}|_{in}$ nodes are expected to receive the block inside the region, and these nodes send the block to their outside outbound neighbors in the kth wave.

If the number of outside outbound neighbors is B, $|W_{k(m)}|_{add\_o}$ can be expressed as

$$|W_{k(m)}|_{add\_o} = Bp_{g_{k(m)}}, \tag{7}$$

where $p_{g_{k(m)}}$ is the block forwarding probability from the mth node in the kth wave outside the region. This is because for the mth block sender in the kth wave, there are B outside outbound neighbors. $p_{g_{k(m)}}$ is calculated as

$$p_{g_{k(m)}} = \frac{N_{out} - \sum_{j=0}^{k-1} |W_j|_{out} - \sum_{j=0}^{m-1} |W_{k(j)}|_{add\_o}}{N_{out}}, \tag{8}$$

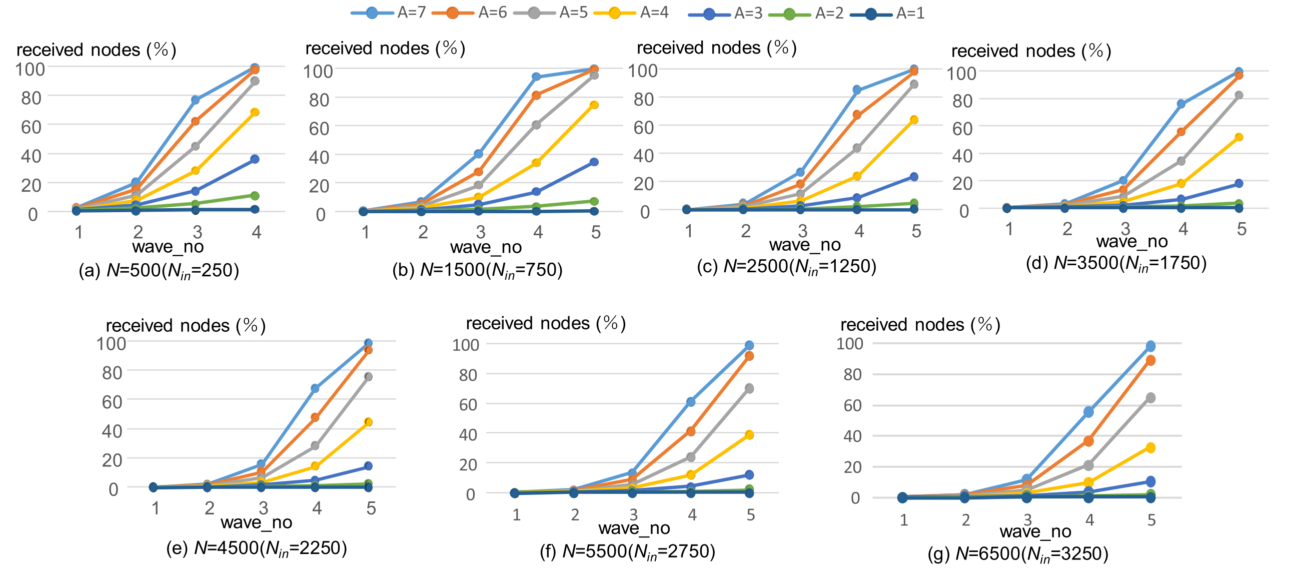
where N_{out} is the number of nodes outside the region. The numerator indicates the expected number of outside nodes that have not received the block before the mth node in the kth wave sends it. $\sum_{j=0}^{k-1} |W_j|_{out}$ indicates the expected number of outside nodes having received the block by the $(k-1)$th wave, and $\sum_{j=0}^{m-1} |W_{k(j)}|_{add\_o}$ indicates the expected number of outside nodes having received the block in the kth wave before the mth node sends the block to its outbound neighbors.

Like $p_{f_{k(m)}}$ in (5), it is clear that $p_{g_{k(m)}}$ decreases as the number of waves and the number of nodes having sent the block become larger.

3) THEORETICAL NUMBER OF WAVES NECESSARY FOR BLOCK PROPAGATION IN/OUTSIDE REGION

Equations (3) and (4) indicate that having more inside outbound neighbors, i.e., a larger A, is beneficial for increasing the nodes receiving the block in the region for each wave, while (6) and (7) indicate that larger values of both inside and outside outbound neighbors, A and B, are beneficial for increasing the nodes receiving the block outside the region for each wave.

Therefore, we compared the block propagation times between networks with nodes having 32 outbound neighbors ($A+B=32$) and networks with nodes having eight outbound neighbors ($A+B=8$) [23]. We found that the networks with nodes having eight outbound neighbors had the shorter block propagation times, though theoretically the condition: $A+B=32$ has more nodes receiving the block inside and outside the region in each wave. In other words, this result shows if each node has a large number of outbound neighbors, it takes longer for each node to forward a block to all of its neighbors; thus, the block propagation times throughout the network suffer from the individual nodes' broadcast overhead, as is discussed in [26].



(a) N=500(N_{in}=250)

(b) N=1500(N_{in}=750)

(c) N=2500(N_{in}=1250)

(d) N=3500(N_{in}=1750)

(e) N=4500(N_{in}=2250)

(f) N=5500(N_{in}=2750)

(g) N=6500(N_{in}=3250)

FIGURE 8. **Percentage of inside nodes that are expected to receive block by each wave.**

On the other hand, a previous study [42] showed that in order to minimize the fork rate in a network, it is desirable that $\log N < A+B \leq \lceil N/100 \rceil$, where N is the total number of nodes in the network and $\lceil\rceil$ denotes the ceiling function. It also showed that in a network with 500 nodes, $A+B=5$ is most effective choice with which to minimize the fork rate. In this study, we examined networks having between 500 and 6500 nodes and found that, except for the network with 500 nodes, $A+B=8$ satisfies the above condition. Under the condition $A+B=5$, Fig. 5 compares the proposed method with the total random selection method. In addition, asking the individual nodes to change the number of their outbound neighbors depending on the number of network nodes is difficult in a public blockchain network. For these reasons, each node has eight outbound neighbors.

In the following, the expected number of nodes receiving the created block inside and outside the region is visualized on the basis of equations (3) – (8) in networks with multiple regions. The number of regions in a network and the number of nodes in each region are based on the Bitcoin network in 2019.

Fig. 8 shows the theoretically calculated ratio of nodes that are expected to receive a block inside the region where the block is created. The networks have between 500 and 6500 nodes. The expected number of nodes receiving the block in the kth wave, $|W_k|_{in}$, is calculated using (3), while N_{in} in (5) is set to half the number of the network nodes, as Europe had 49.98% nodes in the Bitcoin network in 2019. At that time, Europe likely had the longest block propagation times in the Bitcoin network, thus it is important to find the necessary number of the block propagation waves in the region.

From Fig. 8, it is clear that $A=7$ and $A=6$ have an advantage over the other values. For these choices of number of inside outbound neighbors, block propagation almost finishes by the 4th wave in the network with 500 nodes and by the 5th wave in the other networks. In the last wave, the gap between $A=7$ and $A=6$ is less than 4.6% on average, so there are many cases in which these values give the same number of waves for a block to propagate in a region.

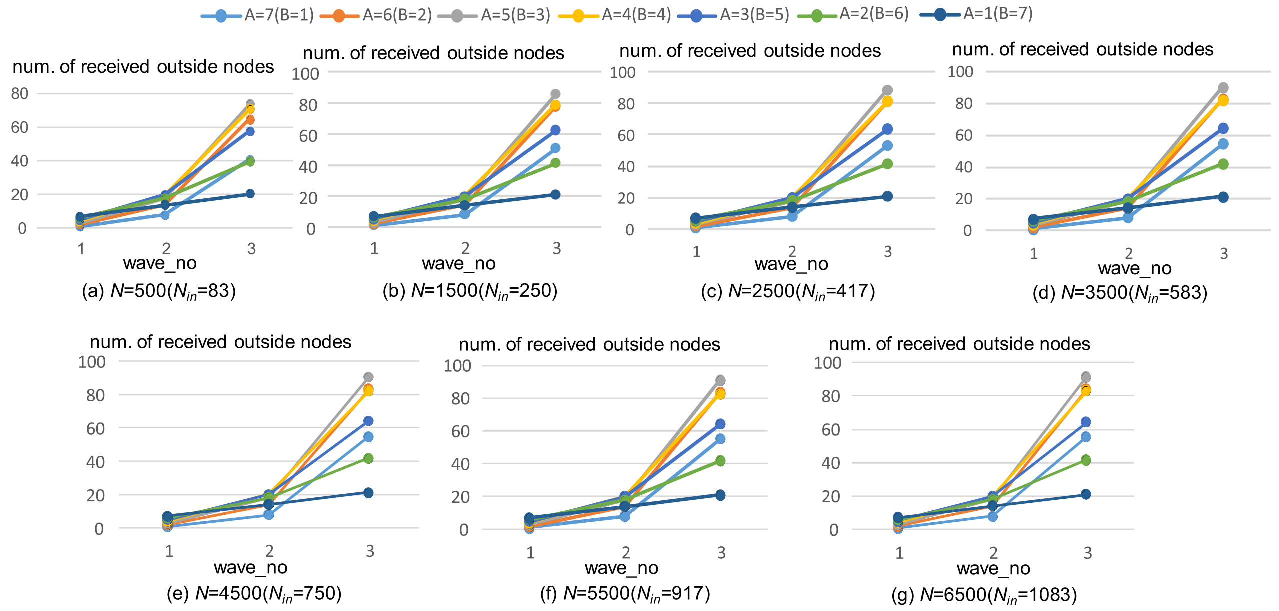
Fig. 9 shows the theoretical number of outside nodes that are expected to receive a block created inside a region by the third wave in each network. The expected number of outside nodes receiving the block in the kth wave, $|W_k|_{out}$, is calculated using (6). N_{in} is set to 1/6th of the total network nodes in each network, so N_{out} in (8) is 5/6th of the total network nodes. When the number of waves, k, is 5, $|W_k|_{out}$ grows proportionally larger as A is increased. However, as long as $k < 4$, $A=5$, and hence $B=3$, is the best setting with which to increase the number of the outside nodes receiving the block by the third wave.

As discussed above, an important factor in reducing the block propagation time throughout a network is for all of its regions to receive at least one block from the originating region in fewer waves. The probability whereby the kth block-receiving outside node does not belong to the target region p_k is calculated as

$$p_k = \frac{N_{out} - k - 1 - N_{targetR}}{N_{out}}. \tag{9}$$

Here, $N_{targetR}$ is the number of nodes located in the target region. Among the N_{out} outside nodes, there are $(N_{out} - k - 1)$ nodes that have not received the block. Moreover, nodes located in the target region should not be chosen as the kth node.

The probability of the k outside nodes that have received the block and do not belong to the target region, $p_{k\_out\_targetR}$,



(a) N=500(N_{in}=83)
(b) N=1500(N_{in}=250)
(c) N=2500(N_{in}=417)
(d) N=3500(N_{in}=583)
(e) N=4500(N_{in}=750)
(f) N=5500(N_{in}=917)
(g) N=6500(N_{in}=1083)

FIGURE 9. **Number of outside nodes that are expected to receive block by each wave.**

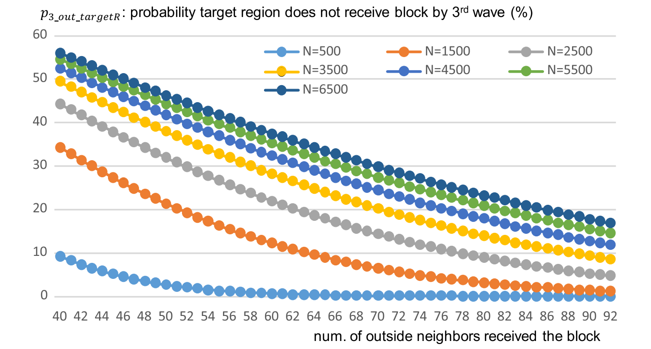


FIGURE 10. **Theoretical probability of block not reaching all outside regions.**

is calculated as

$$p_{k\_out\_targetR} = \prod_{j=1}^{k} p_j. \qquad (10)$$

We obtain this probability because the non-target-region node selection with probability p_j has to be continued until the kth node selection.

For each network in Fig. 9 $N_{targetR}$ is set as $N_{targetR} = 0.009N$, because it corresponds to the lowest node ratio in the Bitcoin network in 2019, i.e., that of the South America region. In other words, South America is considered to be the bottleneck region for propagating a block throughout the world. Fig. 10 shows the theoretical values of $p_{3\_out\_targetR}$ in the seven networks.

The results in Fig. 9 and Fig. 10 show that, in each network, all of the outside regions can receive the created block from its region of origin within the first three or four waves if **A** is set to 4 or more. This is because the theoretical number of nodes receiving the block outside the region by the fourth wave exceeds 200 if $A > 3$, and $p_{4\_out\_targetR}$ is nearly 0% when the number of outside nodes reaches 200. Therefore, delivering the block propagation to all the outside regions by the third wave in each network is important for shortening the propagation time.

When $A = 7$, and hence $B = 1$, many fewer outside nodes are expected to receive the block by the third wave than when $2 < A < 7$, even though $A = 7$ is the most efficient choice in terms of the block propagation within a region (Fig. 8). In the case of the network with 6500 nodes in Fig. 9(g), the expected number of outside nodes receiving the block by the third wave is 55.5 when $A = 7$, and $p_{3\_out\_targetR}$ is about 41% (Fig. 10), meaning there is a 41% chance that the block will not have propagated to all the regions. On the other hand, the expected number of outside nodes receiving the block is 91.3 when $A = 5$ and $p_{3\_out\_targetR}$ is about 17%. Thus, the probability of a block propagating to all the regions by the third wave is much larger than in the case with $A = 7$. This difference is large in the networks with many nodes. Even in the network with 1500 nodes, the expected number of outside nodes is 50.8 with $A = 7$ ($p_{3\_out\_targetR}$ is about 20%), whereas it is 85.8 with $A = 5$ ($p_{3\_out\_targetR}$ is about 2%).

In the network with 500 nodes, however, the expected number of outside nodes is 40.8 with $A = 7$, and $p_{3\_out\_targetR}$ is about 8%. Thus, the probability of the block being delivered to all the regions is about 92%. In other words, $A = 7$ is appropriate for a relatively smaller network with around 500 nodes considering its higher effectiveness for the block propagation in a region as demonstrated in Fig. 8.

On average, $p_{3\_out\_targetR}$ is only 2.8% larger when $A = 6$ compared with $A = 5$. This small disadvantage is outweighed by the large gap in terms of the block receiving ratio inside a region for each wave, as shown in Fig. 8. For example, after the fourth wave in each network, the average expected block receiving ratio inside the region is 19% higher with $A = 6$ than with $A = 5$.

In summary, this theoretical analysis shows that $A = 7$ ($B = 1$) and $A = 6$ ($B = 2$) are appropriate outbound neighbor selection parameters for a faster block propagation in a network with multiple regions. In the case of a small network with around 500 nodes, $A = 7$ is more appropriate than $A = 6$, whereas in a larger network with 1500 or more nodes, $A = 6$ is the most appropriate choice.

In addition, our assumption of targeting the blocks coming directly from the region in which they were created is appropriate. This is because a block directly propagates to all the regions in the first four waves, whereas a block coming via multiple regions generally comes later than the fifth wave, because inter-region block transmission takes 5.2 times longer than inner-region block transmission, as shown in Table 1.

V. EVALUATION

We evaluated our neighbor selection method as follows. First, we determined the optimal numbers of outbound inside and outside neighbors for each network by using the SimBlock simulator. Second, we compared our method with the total random selection method and the Aoki method [24] in terms of block propagation time throughout the network using the simulator. Third, we examined a migration scenario wherein the ratio of nodes using the proposed neighbor selection method to those using the total random selection method is gradually increased.

A. SIMULATION ENVIRONMENT

SimBlock can be used to simulate multi-region blockchain networks under the same conditions as Bitcoin in 2015 or in 2019. We chose the 2019 Bitcoin network and varied the number of nodes between 500 and 6500. In particular, the nodes were randomly located in one of the six regions with the same probability as in the 2019 network, and the transmission delays were as in Table 1. PoW was used as the consensus algorithm, and each node was set to be able to be a miner, but the computer power for mining on each node was set randomly. The number of outbound neighbors was set to 8, the block size was set to 535 KB or 1MB, and the block creation interval was set to 10 minutes.

B. SIMULATION OF OPTIMAL NUMBER OF INSIDE/OUTSIDE NEIGHBORS FOR EACH NODE

The optimal numbers of outbound inside neighbors, A, and outside neighbors, B, for each node in various sizes of network were theoretically analyzed under the condition of eight outbound neighbors examined in the previous section. Here, we evaluated the parameters of the proposed methods by using SimBlock. The inbound option, which determines the upper limit of inside/outside inbound neighbors, was not used in the evaluation, and the maximum number of inbound neighbors for each node was set to 30.

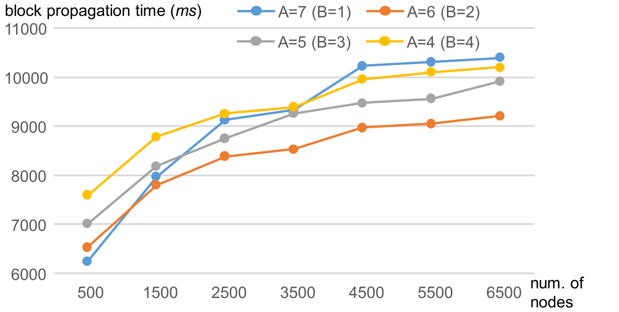


FIGURE 11. Block propagation times with different as.

Fig. 11 shows the block propagation times of the proposed method with different values of A and B. The main-chain height was set to 200, so one simulation ended after the 201st block of the main chain was delivered throughout the network. The block size was set to 535KB in this subsection. Each plot in the figure indicates the average block propagation time to all the nodes in the network. That is, if a mined block failed to be distributed to a node in the network, the block propagation time was not included in the average value. To make the values trustworthy, we ran the simulation three times under the same conditions and took the average of the three runs.

As in the previous section, $A = 7$ turned out to be the most effective for the network with 500 nodes, while $A = 6$ was most effective for the networks with 1500 or more nodes. It is an important factor to reduce the number of block propagation waves for delivering the block to all the outside regions. In the network with 500 nodes, it was highly likely for the block to reach all the outside regions by the third wave when $A = 7$. However, $A = 7$ ended up having the longest block propagation time in the networks with 3500 or more nodes in Fig. 11, because the fourth wave was more necessary for these networks with $A = 7$.

Instead, $A = 6$ had the shortest block propagation time for the networks with 1500 or more nodes. It had a high probability to deliver a block to all the outside regions by the third wave. Even though $A = 5$ is theoretically a bit faster in the sense of propagating the block to the outside regions by the third wave as shown in Fig. 9, this small advantage was surpassed by the much faster block propagation speed in each region with $A = 6$ in Fig. 11.

C. BLOCK PROPAGATION TIMES OF PREVIOUS AND PROPOSED METHODS

We compared the block propagation times of the proposed method with those of the total random selection method and the Aoki method. To evaluate the block size effects, we changed the block size between 535KB and 1MB. The inbound option, in which the upper limit of the inbound neighbors is set to 8 as shown in Fig. 3, was also evaluated. Except for the proposed method with the inbound option, the maximum number of inbound neighbors for each node was set to 30. For the proposed method, $B = 1$ outside neighbor was set for the network with 500 nodes and $B = 2$ was set for the other networks; these values were theoretically and quantitatively proved to be efficient in the preceding sections.

1) AOKI METHOD

The Aoki method aims to minimize the block propagation time in a multi-region environment. It chooses an outbound neighbor for each node on the basis of the block propagation time through that neighbor. Each block has a time stamp in which the block creation time is stored. Thus, if a node receives a block from an inbound neighbor (M), the node can calculate the block propagation time starting from the block creation (T_{block}) and ending with the *inv* message received time (T_{inv}). The node considers the propagation time as the score of the neighbor M ($SCORE_M$):

$$SCORE_M = T_{inv} - T_{block}. \tag{11}$$

If the node has already received *inv* messages from M, $SCORE_M$ is defined as

$$SCORE_M = (1 - P)\, SCORE_M + P(T_{inv} - T_{block}), \tag{12}$$

where P is a parameter in the range [0,1]. Neighbor candidates having a smaller $SCORE_M$ are chosen as the outbound neighbors. Aoki and Shudo [24] found that $P = 0.3$ has the shortest block propagation time. In addition, to prevent eclipse attacks from happening, at least one neighbor should be chosen randomly for each node. Under this restriction, it is claimed to be best to replace 7 out of 8 outbound neighbors in accordance with the scores defined in (12). The outbound neighbor renewal interval was set to 10 received blocks for each node.

After referring to the evaluation results in [24], we set $P = 0.3$ and chose 7 neighbors on the basis of (12) in 10 received blocks for each node. The Aoki method is optimized for high block propagation speed at the expense of a heavy burden on each node and vulnerability to eclipse attacks. The results of the following experiments show that the proposed method can propagate a block throughout a network faster than the Aoki method can.

2) BLOCK PROPAGATION TIME

Fig. 12 shows the 535KB block propagation times for each method under the condition $A + B = 8$. In this figure, "proposed (outbound)" indicates the proposed method without the inbound option, whereas "proposed (out/inbound)" indicates the proposed method with the inbound option. The results clearly show that the proposed method is far superior to the total random selection method and the Aoki method for all network conditions. Each node in the total random selection method has five or more outside neighbors on average, so it is equivalent to $A \leq 3$ and $B \geq 5$. Under this condition, block propagation in and outside the region becomes slower, as shown in Figs 8 – 10.

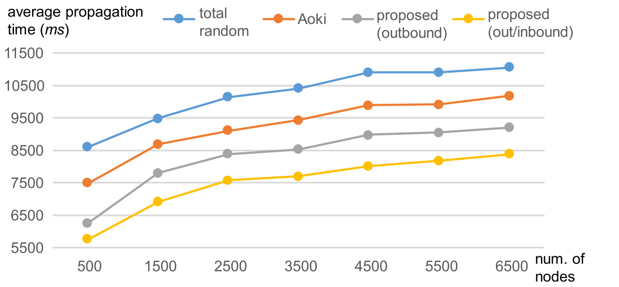


FIGURE 12. Comparison of 535KB block propagation times with different methods.

The Aoki method updates the outbound neighbors by prioritizing the propagation speed every ten block receipts, so the propagation times to most nodes in a network are considered to be very close to the optimal values. Looking at Fig. 12, however, it is clear that the Aoki method prevents some nodes from receiving blocks quickly. Equation (12) can be used to explain the reason. (12) determines the new outbound neighbors of each node every 10*th* block arrival, and a low-scoring node at that point tends to be selected as a new neighbor by many nodes as their outbound neighbor. Other nodes, however, may suffer in this situation because their inbound neighbors are reduced by the high concentration of connections to the low-scoring node, and as a result, they may receive blocks more slowly.

In Fig. 12, the inbound option has a clear effect on all the networks, and the upper limit of the inbound neighbors helps to prevent some nodes from having much fewer inbound neighbors compared with the other nodes. If there are some nodes with much fewer inbound neighbors, there are more miners who cannot propagate their newly mined blocks to the nodes within a small number of waves, and this lengthens the block propagation time.

This phenomenon can be explained in terms of the block propagation speed in each region. The block propagation probability, $p_{f_{k(m)}}$, in the block-created region with the inbound option is

$$p_{f_{k(m)}} = \frac{N_{in} - 1 - \sum_{j=0}^{k-1} |W_j|_{in} - \sum_{j=0}^{m-1} |W_{k(j)}|_{add\_i}}{N_{in} - 1 - N_{in\_full}}, \tag{13}$$

in which N_{in_full} indicates the number of inside nodes that have received the block (before the mth node in the kth wave sends it) up to the maximum limit of inside nodes, which is determined to be 7 in a network with 500 nodes or 6 with 1500 or more nodes. The number of these inside nodes is subtracted in the denominator of (13), because there is no chance for them to receive the block from inside the region because of the limitation determined by the inbound option. Therefore, $p_{f_{k(m)}}$ with the inbound option becomes larger than that in (5), which is assumed to have no inbound node restriction on each node. In particular, the gap between $p_{f_{k(m)}}$ in (13) and in (5) widens as k grows because N_{in_full} increases with k.

This theory is also applicable to outside the block-created region. Here, the block propagation probability, $p_{g_{k(m)}}$, outside the block-created region with the inbound option is

$$p_{g_{k(m)}} = \frac{N_{out} - \sum_{j=0}^{k-1} |W_j|_{out} - \sum_{j=0}^{m-1} |W_{k(j)}|_{add\_o}}{N_{out} - N_{out\_full}}, \qquad (14)$$

in which N_{out_full} indicates the number of outside nodes that have received the block up to the maximum limit of outside nodes, which is determined to be 1 in a network with 500 nodes or 2 with 1500 or more nodes. The number of these outside nodes is subtracted in the denominator of (14), because there is no chance for these nodes to receive the block from outside the region due to the limitation determined by the inbound option. Therefore, $p_{g_{k(m)}}$ with the inbound option becomes larger than that in (8), which is assumed to have no inbound node restriction on each node. In addition, the gap between $p_{g_{k(m)}}$ in (14) and in (8) widens as k grows because N_{out_full} increases with k. In this way, even from a theoretical point view, it is beneficial to take the inbound option in the proposed method.

Fig. 13 shows the 1MB block propagation times for each method, with which all the conditions except for the block size were set to those in Fig. 12. Proposed (outbound) and proposed (out/inbound) had shorter block propagation times than the other two methods in every network condition, though the gap between proposed (outbound) and the Aoki method narrowed when the number of network nodes was increased to 4500 or more, compared with Fig. 12. We found that, for a block size of 1MB, choosing the neighbors with smaller block propagation times in the Aoki method was more effective, because the block propagation times between two different nodes in a network were increased due to the larger block size.

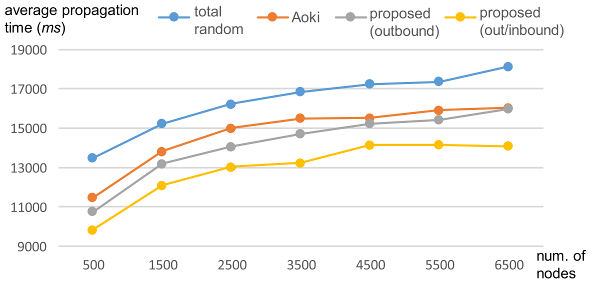


FIGURE 13. Comparison of 1MB block propagation times with different methods.

In Fig. 13, however, the inbound option had a clear effect on all the networks like in Fig. 12. This is due to the higher block forwarding probabilities in and outside the block-created region, as shown in (13) and (14). Because of these enhanced block forwarding probabilities, the number of block propagation waves needed to reach all the nodes in a network was reduced, and this effect became clearer when a 1MB block was used. We analyze this is because each wave took more time due to the larger block size.

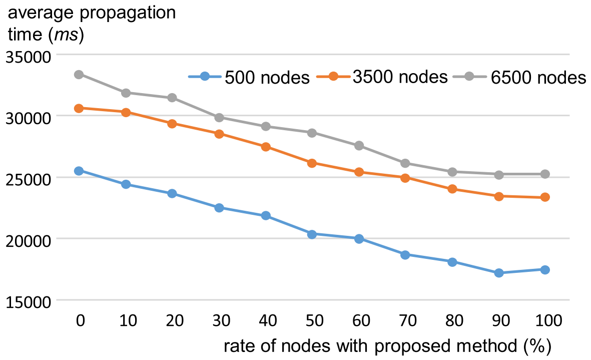


FIGURE 14. Block propagation time versus ratio of nodes adopting proposed method.

The effect of the inbound option became clearer especially when the number of nodes in the network was large (Fig. 13). This phenomenon can be explained by comparing (13) with (5) and comparing (14) with (8). In (13), the denominator, $N_{in}-1-N_{in\_full}$, becomes larger as N_{in} increases. Thus, $p_{f_{k(m)}}$ grows as N_{in_full} approaches $N_{in}-1$. Without the inbound option, the denominator is fixed to $N_{in}-1$, as shown in (5); thus, the gap between $p_{f_{k(m)}}$ in (5) and (13) becomes larger as N_{in} grows. This theory is also applicable to $p_{g_{k(m)}}$ in (8) and (14): the gap between $p_{g_{k(m)}}$ in (8) and (14) becomes larger as N_{out} grows.

D. SCENARIO OF MIGRATION TO PROPOSED NEIGHBOR SELECTION METHOD

Even though the theoretical analyses and simulation results presented above have proven the superiority of the proposed method to the previous methods, we cannot force every node in the blockchain to adopt it. Therefore, in this subsection we propose a migration scenario from a network with nodes using the total random selection method, which is the default setting for a Bitcoin node, to one with nodes using the proposed method with the inbound option. During the migration phase, there is a hybrid network situation with two different types of node, each adopting its own neighbor selection method. We evaluated this situation by changing the ratio of nodes implementing our method.

The networks had 500, 3500, or 6500 nodes and block size was set to 535KB; the rest of the evaluation conditions were the same as those mentioned in subsections *A* and *B* above. Fig. 14 shows the average block propagation times versus ratio of the nodes adopting the proposed method in each network. The plots are the average values of three evaluations. It can be seen that even with 10% of the nodes adopting the proposed method, the average propagation time was reduced by 3.3%; with 90% adoption, the reduction rate was as much as 26.9%, even higher than when all nodes use the proposed method.

The largest reduction rate with 90% adoption of the proposed method was for the network with 500 nodes. In this case, the reduction rate was 33.3%, which was about 1% higher than when all the nodes use the proposed method.

This phenomenon can be explained using Figs. 9 and 10. In Fig. 9(a), the expected number of outside nodes receiving the block by the third wave is 40.8 when $A = 7$, and this value means that the ratio for South America to receive the block is about 92% from the data in Fig. 10. However, if there are a small number of nodes that have more outside neighbors, the expected number of outside nodes by the third wave can be increased to more than 41. We suppose that this is why South America's block receiving rate by the third wave was increased, and the average block propagation time was reduced. Of course, we have to consider the negative effect on the block propagation time in each region because of the 10% of nodes with the total random selection method, but from this result, it is clear that this negative effect was offset by the dominant 90% of nodes adopting the proposed method.

Another interesting phenomenon we observe in this evaluation is that, in the network with 500 nodes, there is a strong tendency for a miner adopting the proposed method to propagate the blocks it creates faster to all the other nodes in the network. This advantage becomes apparent when the ratio of the nodes adopting the proposed method is lower, such as less than 50%. This phenomenon is illustrated in Fig. 15, which plots average propagation time versus adoption rate of the proposed method: miners with the proposed method propagated the created block faster than miners with the total random selection method. When 10% and 20% of the miners used the proposed method; their propagation speeds were about 10% faster than those of the miners using the total random selection method.

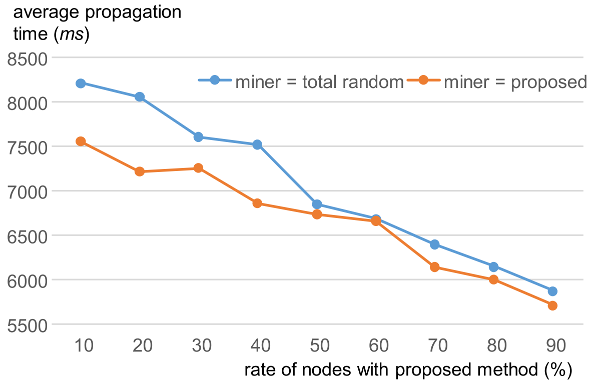


FIGURE 15. Average block propagation time versus proportion of miners adopting proposed method.

We performed this analysis on a small network with 500 nodes, so setting the miner's number of inside neighbors to a large number, i.e. 7, was beneficial for propagating the block inside the region faster. This is especially important for a network with a smaller rate of nodes adopting the proposed method because many nodes in each region have few inside neighbors and they have thus less power for propagating the block in the region at a fast speed.

From this evaluation, we can say that gradually infusing the proposed method into a network does not have any harmful effect and it leads to a significant improvement in block propagation speed even when only 10% of the nodes adopt it. In addition, this beneficial effect should encourage new nodes participating in a small blockchain network to use the proposed method, because the blocks created by the new nodes will be propagated to all the other nodes faster.

VI. CONCLUSION

We proposed a new neighbor selection method for a blockchain network that enhances the block propagation speed throughout the network and fairness among miners. In addition, because of its random neighbor selection, it is robust against eclipse attacks, and the burden incurred by the neighbor selection is light because it uses only the IP addresses of the neighbor candidates. We also proposed a block propagation wave model with multiple regions that clarifies the optimal numbers of inside/outside neighbors for propagating a block throughout a network in relation to the number of nodes in that network.

In the simulation experiment, we showed the theoretically optimal number of outside neighbors for each node also gave the shortest block propagation time in each network with 500–6500 nodes. We also demonstrated the applicability of the proposed method to blockchain networks by showing that its block propagation times are much shorter than those of the total random selection method and the Aoki method in networks with 500–6500 nodes. The inbound option was also evaluated and it was shown to be effective at shortening the block propagation times. Equations (3) – (8), (13), and (14) clarified the superiority of the proposed method and the inbound option in a theoretical manner.

We also examined a scenario depicting migration from the total random neighbor selection method to the proposed method. We found that the block propagation speed significantly increases even when a small proportion of nodes in the blockchain network adopt our method. We also found that a miner adopting the proposed method will have an advantage in propagating blocks it creates faster throughout a small network with 500 nodes.

REFERENCES

[1] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," Tech. Rep., 2008. [Online]. Available: https://www.bitcoin.org/bitcoin.pdf

[2] G. O. Karame, E. Androulaki, and S. Capkun, "Two Bitcoins at the price of one? Double-spending attacks on fast payments in Bitcoin," *IACR Cryptol. ePrint Arch.*, 2012. [Online]. Available: https://eprint.iacr.org/2012/248.pdf

[3] M. Campbell, *Bitcoin and Beyond: Cryptocurrencies, Blockchains, and Global Governance*. Evanston, IL, USA: Routledge, 2018.

[4] M. Jin, X. Chen, and S.-J. Lin, "Reducing the bandwidth of block propagation in bitcoin network with erasure coding," *IEEE Access*, vol. 7, pp. 175606–175613, 2019.

[5] D. Reijsbergen, S. Sridhar, B. Monnot, S. Leonardos, S. Skoulakis, and G. Piliouras, "Transaction fees on a honeymoon: Ethereum's EIP-1559 one month later," in *Proc. IEEE Int. Conf. Blockchain (Blockchain)*, Dec. 2021, pp. 196–204.

[6] S. Wilkinson, T. Boshevski, J. Brandoff, and V. Buterin. (Dec. 2014). *Storj a Peer-to-Peer Cloud Storage Network*. [Online]. Available: https://www.storj.io/storj2014.pdf

[7] A. Dorri, S. S. Kanhere, and R. Jurdak, "Blockchain in Internet of Things: Challenges and solutions," 2016, *arXiv:1608.05187*.

[8] H. F. Atlam, A. Alenezi, M. O. Alassafi, and G. Wills, "Blockchain with Internet of Things: Benefits, challenges, and future directions," *Int. J. Intell. Syst. Appl.*, vol. 10, no. 6, pp. 40–48, 2018.

[9] A. Banafa, "IoT and blockchain convergence: Benefits and challenges," *IEEE Internet Things Newslett.*, Jan. 2017.

[10] W. J. Gordon and C. Catalini, "Blockchain technology for healthcare: Facilitating the transition to patient-driven interoperability," *Comput. Struct. Biotechnol. J.*, vol. 16, pp. 224–230, 2018.

[11] K. Rabah, "Challenges and opportunities for blockchain powered healthcare systems: A review," *Mara Res. J. Med. Health Sci.*, vol. 1, no. 1, pp. 45–52, 2017.

[12] M. Sajjad, M. Nasir, K. Muhammad, S. Khan, Z. Jan, A. K. Sangaiah, M. Elhoseny, and S. W. Baik, "Raspberry pi assisted face recognition framework for enhanced law-enforcement services in smart cities," *Future Gener. Comput. Syst.*, vol. 108, pp. 995–1007, Jul. 2020.

[13] K. E. C. Levy, "Book-smart, not street-smart: blockchain-based smart contracts and the social workings of law," *Engaging Sci., Technol., Soc.*, vol. 3, pp. 1–15, Feb. 2017.

[14] C. M. Christopher, "The bridging model: Exploring the roles of trust and enforcement in banking, bitccoin, and blockchain," *Nevada Law J.*, vol. 17, no. 1, pp. 139–180, 2016.

[15] *Distributed Ledger Technology Use Cases*, ITU-T, FG DLT D2.1, 2019.

[16] E. Androulaki, A. Barger, V. Bortnikov, C. Cachin, K. Christidis, A. DeCaro, D. Enyeart, C. Ferris, G. Laventman, and Y. Manevich, "Hyperledger fabric: A distributed operating system for permissioned blockchains," in *Proc. 13th EuroSys Conf.*, Apr. 2018, p. 30.

[17] W. Zheng, Z. Zheng, X. Chen, K. Dai, P. Li, and R. Chen, "NutBaaS: A blockchain-as-a-service platform," *IEEE Access*, vol. 7, pp. 134422–134433, 2019.

[18] J. Misic, V. B. Misic, X. Chang, S. G. Motlagh, and M. Z. Ali, "Block delivery time in bitcoin distribution network," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2019, pp. 1–7.

[19] Y. Shahsavari, K. Zhang, and C. Talhi, "A theoretical model for fork analysis in the bitcoin network," in *Proc. IEEE Int. Conf. Blockchain (Blockchain)*, Jul. 2019, pp. 237–244.

[20] *Bitcoin Core Integration/Staging Tree*. Accessed: Jul. 19, 2022. [Online]. Available: https://github.com/bitcoin/bitcoin

[21] *Falcon—A Fast Bitcoin Backbone*. Accessed: Jul. 19, 2022. [Online]. Available: https://www.reddit.com/r/btc/comments/4n62bo/falcon_a_fast_bitcoin_backbone/

[22] *What is FABRA*. Accessed: Jul. 19, 2022. [Online]. Available: https://www.bitcoinfibre.org/

[23] H. Matsuura, Y. Goto, and H. Sao, "Region-based neighbor selection in blockchain networks," in *Proc. IEEE Int. Conf. Blockchain (Blockchain)*, Dec. 2021, pp. 21–28.

[24] Y. Aoki and K. Shudo, "Proximity neighbor selection in blockchain networks," in *Proc. IEEE Int. Conf. Blockchain (Blockchain)*, Jul. 2019, pp. 52–58.

[25] K. Wang and H. S. Kim, "FastChain: Scaling blockchain system with informed neighbor selection," in *Proc. IEEE Int. Conf. Blockchain (Blockchain)*, Jul. 2019, pp. 376–383.

[26] A. Sudhan and M. J. Nene, "Peer selection techniques for enhanced transaction propagation in bitcoin peer-to-peer network," in *Proc. 2nd Int. Conf. Intell. Comput. Control Syst. (ICICCS)*, Jun. 2018, pp. 679–684.

[27] S. Park, S. Im, Y. Seol, and J. Paek, "Nodes in the bitcoin network: Comparative measurement study and survey," *IEEE Access*, vol. 7, pp. 57009–57022, 2019.

[28] W. Bi, H. Yang, and M. Zheng, "An accelerated method for message propagation in blockchain networks," 2018, *arXiv:1809.00455*.

[29] E. Heilman, A. Kendler, A. Zohar, and S. Goldberg, "Eclipse attacks on Bitcoin's peer-to-peer network," in *Proc. USENIX Secur. Symp.*, 2015, pp. 129–144.

[30] A. E. Yves-Christian, B. Hammi, A. Serhrouchni, and H. Labiod, "Total eclipse: How to completely isolate a bitcoin peer," in *Proc. 3rd Int. Conf. Secur. Smart Cities, Ind. Control Syst. Commun. (SSIC)*, Oct. 2018, pp. 1–7.

[31] M. Tran, I. Choi, G. J. Moon, A. V. Vu, and M. S. Kang, "A stealthier partitioning attack against bitcoin peer-to-peer network," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2020, pp. 894–909.

[32] S. Jiang and J. Wu, "Taming propagation delay and fork rate in bitcoin mining network," in *Proc. IEEE Int. Conf. Blockchain (Blockchain)*, Dec. 2021, pp. 314–320.

[33] H. Baniata, A. Anaqreh, and A. Kertesz, "DONS: Dynamic optimized neighbor selection for smart blockchain networks," *Future Gener. Comput. Syst.*, vol. 130, pp. 75–90, May 2022.

[34] P. L. Juhász, J. Stéger, D. Kondor, and G. Vattay, "A Bayesian approach to identify bitcoin users," *PLoS ONE*, vol. 13, no. 12, Dec. 2018, Art. no. e0207000.

[35] K. Ntolkeras, H. Sharif, S. D. Salmasi, and W. Knottenbelt, "Performance analysis of a hyperledger iroha blockchain framework used in the U.K. livestock industry," in *Proc. IEEE Int. Conf. Blockchain (Blockchain)*, Dec. 2021, pp. 456–461.

[36] P. Mittal, A. Walthall, P. Cui, A. Skjellum, and U. Guin, "A blockchain-based contactless delivery system for addressing COVID-19 and other pandemics," in *Proc. IEEE Int. Conf. Blockchain (Blockchain)*, Dec. 2021, pp. 1–6.

[37] A. Soni and S. Maheshwari, "A survey of attacks on the bitcoin system," in *Proc. IEEE Int. Students' Conf. Electr., Electron. Comput. Sci. (SCEECS)*, Feb. 2018, pp. 1–5.

[38] M. Walck, K. Wang, and H. S. Kim, "TendrilStaller: Block delay attack in bitcoin," in *Proc. IEEE Int. Conf. Blockchain (Blockchain)*, Jul. 2019, pp. 1–9.

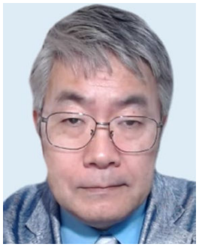
[39] *ICANN*. Accessed: Jul. 15, 2022. [Online]. Available: https://www.icann.org/

[40] Y. Aoki, K. Otsuki, T. Kaneko, R. Banno, and K. Shudo, "SimBlock: A blockchain simulator," in *Proc. 2nd Workshop Cryprocurrencies Blockchains Distrib. Syst. (CryBlock)*, Apr. 2019, pp. 325–329.

[41] *Global Ping Statistics*. Accessed: Jul. 19, 2022. [Online]. Available: https://wondernetwork.com/pings

[42] A. Kertesz and H. Baniata, "Consistency analysis of distributed ledgers in fog-enhanced blockchains," in *Proc. Int. Eur. Conf. Parallel Distrib. Comput. (Euro-Par)*, vol. 27, 2021, pp. 393–404.

[43] Y. Shahsavari, K. Zhang, and C. Talhi, "A theoretical model for block propagation analysis in bitcoin network," *IEEE Trans. Eng. Manag.*, vol. 69, no. 4, pp. 1459–1476, Aug. 2022.



HIROSHI MATSUURA received the B.E. degree from Kyushu University, Fukuoka, Japan, in 1989. He joined the NTT Laboratories, in 1989, after which he worked on network design and management. He joined the NTT West Research and Development Center, in 1999, where he was engaged in managing storage area networks. He returned to the NTT Laboratories, in 2003, where he developed GMPLS routing architecture and studied the Steiner tree problem, k shortest path algorithms, and maximizing the lifetime of data-gathering sensor trees in WSNs. His current research interest includes optimizing routing in blockchain networks.

YOSHINORI GOTO received the B.E. and M.E. degrees in applied physics from Tohoku University, Japan, in 1992 and 1994, respectively. He is currently a Senior Engineer working at NTT Advanced Technology Corporation. He joined NTT, in 1994, and was engaged in the development of video transmission system over optical access network and STB for IPTV. He is studying network technologies for future networks. He has also promoted the standardization works in ITU, IOWN-GF, ETSI, APT, TTC, and oneM2M.

HIDEHIRO SAO received the B.E. degree from Ehime University, Ehime, Japan, in 1996. He joined NTT, in 1996, and began researching and standardizing signal systems between exchanges and between exchanges and terminals. After engaging in the launch of internet access services and the establishment of a framework for the maintenance and operation of IP services, he was assigned as the Manager to oversee all engineering operations. In 2019, he was promoted the commercialization of network-related research and development. He is currently engaged as the Manager responsible for the telecommunication facilities (Osaka, Western Japan).