# ITF: A Blockchain System with Incentivized Transaction Forwarding

Jiarui Zhang* and Yaodong Huang†
*Department of Electrical and Computer Engineering, Stony Brook University, Stony Brook, NY 11794, USA
jiarui.zhang.2@stonybook.edu
†College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China
yaodong.huang@outlook.com

***Abstract*—The blockchain is introduced as a safe and decentralized technology widely used in cryptocurrencies. It provides a distributed and disintermediation system to securely process and store transactions between peer devices. Traditionally, every transaction in the blockchain is broadcasted throughout the network, which leaves huge computational and communicational overhead to nodes. Nodes may refuse to forward transactions, thereby hindering the consensus of the blockchain. In this paper, we design a blockchain system with Incentive Transaction Forwarding (ITF). ITF allows nodes to share the revenue from transaction fees as the incentive for transaction forwarding. We propose a mechanism keeping the topology updated for computing incentive allocations. We develop an incentive allocation algorithm to distribute revenue among nodes that forward transactions. We analyze the security of ITF and prove that nodes cannot get unfair advantages in our system by common attacks. Extensive simulations show that our system can have fair incentive allocations for relay nodes and against several attacks from adversaries.**

***Index Terms*—Blockchain, Incentive allocation, Transaction forwarding, Security**

## I. Introduction

With the emerging high-speed and low latency networking systems such as 5G and Wi-Fi 6 [1], an increasing number of devices, such as smartphones, IoT devices, and connected vehicles are revolutionizing our daily lives through interchanging a massive amount of data. These devices possess sufficient capabilities for sensing, computing, and storing. With such capabilities, users can create and sell data and services to others, which raises the demand for efficient and secure micropayment systems.

Recently, blockchain [2] is introduced as a safe and decentralized technology for secure transactions and has been widely used in cryptocurrencies. It provides a safe ledger to store the history of transactions, the credits of participants, and other information. The blockchain is a distributed system with many security features for micro-payments. First, it is easy for nodes to restore and verify the block content. Second, a blockchain is designed to have resistance to modify its records. Third, it is a distributed system and ensures transaction security without centralized authorities.

Broadcasting all transactions is important to the blockchain. To achieve consensus of the blockchain content over different nodes, all transactions are broadcasted to all nodes. The broadcast process depends on the forwarding of relay nodes in the blockchain network, which is altruistic. Massive amounts of transactions bring huge computational and communicational overhead to nodes because nodes are required to voluntarily forward these transactions to other nodes without any compensation. Nodes may delay or even refuse to forward transactions because it is not beneficial [3]. The delaying and dropping of transactions put a threat to the data security and consensus of the blockchain system.

To further encourage nodes to forward transactions and compensate their costs on forwarding, we consider distributing revenue among relay nodes as a solution, and we call it incentive allocation. The incentive allocation process gives a part of transaction fees as the revenue to relay nodes. There are several challenges of incentive allocation. First, since nodes need to consume a lot of resources to forward data, we need to measure the contribution of each node in the process of transaction forwarding and distribute revenue among nodes based on their contribution. Second, we measure the contribution of nodes in transaction forwarding according to the network topology, and the network topology changes constantly. We need to dynamically adjust the allocation in a network with constantly changing topologies. Third, distributing revenue to relay nodes may create security loopholes for adversaries who try to attack the system to obtain more revenue. We need to ensure adversaries cannot obtain unfair profits over honest nodes.

In this paper, we propose a new blockchain system ITF, which supports relay nodes to obtain revenue in the blockchain system. We propose a mechanism to detect and maintain connecting and disconnecting events in a network and process the changes of network participants. We design algorithms to distribute transaction fees to relay nodes. We also analyze several common attacks and prove the algorithms ensure adversaries cannot obtain unfair profits under these attacks. Extensive simulations show that our proposed system distributes fair revenue to relay nodes and has resistance to multiple types of attacks.

The main contributions of this paper are in the following.

- We propose a new blockchain system ITF focusing on secure incentive allocations for relay nodes over blockchain networks.
- We design a mechanism keeping the network topology updated to support incentive allocations for relay nodes.

- We propose two algorithms to share revenue over nodes that forward transactions. The first algorithm gives a reduction of the network topology to decrease the number of links when computing the incentive allocation. The second algorithm computes the revenue for each node.
- In the perspective of security, we analyze our new blockchain system with some common attacks to show that it can ensure adversaries cannot obtain unfair profits over the revenue sharing.
- We conduct extensive simulations on the incentive allocation and the protection against different attacks. The results show that our application of the proposed incentive allocation algorithm in the ITF can fairly distribute revenue to honest nodes, and resist several types of attacks for unfair profits.

The rest of this paper is organized as follows. In Section II, we discuss some related work. In Section III, we address the incentive allocation problem and propose the overall system model. In Section IV, we discuss the blockchain structure of the ITF blockchain system. In Section V, we propose our incentive allocation algorithms. In Section VI, we discuss possible types of attacks and analyze the security of the system. In Section VII, we evaluate the incentive allocation and the performance against different malicious attacks of ITF. Finally, we conclude the paper and discuss future work in Section VIII.

## II. Related Work

Blockchain technology was first introduced in 2008 by Satoshi Nakamoto [2] and has been one of the most popular Peer-to-Peer (P2P) structures. A blockchain consists of blocks, each of which has a unique authentication method to confirm a unique order and integrity. Since the blockchain is often built for untrusted parties, the fairness of blockchain is important and has been well studied in multiple aspects. Most of the existing studies on the fairness of blockchain mainly focus on mining and rewarding mechanisms. Eyal and Sirer [4] proved that users can gain more profits than deserved by strategic mining in Bitcoin. Pass and Shi [5] proposed their Fruitchain design to improve the fairness of mining. Amoussou-Guenou et al. [6] fully defined the fairness in Tendermint-core blockchains and propose a modification of the Tendermint rewarding to match their proposed fairness.

In P2P environments, participants are consumers who require services from others, and they are also providers who provide services to others in the network. To keep a balance between consumers and providers in P2P environments, incentive allocation algorithms are developed. The incentive allocation algorithms often encourage information exchanges, service supports, and collaborative works. Most of the traditional incentive allocation algorithms are based on economics and game theory. Feldman et al. [7] proposed several incentive techniques in the P2P systems to optimize levels of cooperation. Yan et al. [8] gave an autonomous compensation game model to encourage data exchange in a spatial restricted P2P system. In dynamic and distributed P2P environments, He et al. proposed an incentive mechanism based on the blockchain [9]. Their mechanism rewards intermediate nodes on the path from the sender to the receiver and imports the blockchain system to resist attacks. However, the incentive allocation algorithms applicable in P2P environments may not apply to blockchain environments. For instance, Buttyan et al. [10] proposed barter-based schemes such that users can trace the historical records of other users and prevent several attacks, but users are not supposed to have long historical records in blockchain environments. Thus, incentive schemes for blockchain applications need to be more sophisticated for the blockchain. Researches which are based on reputation [11] [12] deploy reputation systems to measure contributions and incentives, but they need centralized certificate authorities to avoid whitewashing attacks [13] or Sybil attacks [14].

Incentive allocations in the blockchain still have a lot of room for improvement. The bitcoin lightning network [15] encourages users to conduct off-chain transactions through channels, and the transaction senders pay fees to the intermediate nodes that establish the channels. As a result, most transactions become off-chain transactions, thereby saving on-chain space and increasing transaction throughput. The current research focus of the bitcoin lightning network is to improve the transmission efficiency [16] and maximize the profit of a single node [17], [18]. Sharding [19] is another technology to improve the throughput of blockchain. It divides the network into smaller committees, each of which can work independently. The current incentive scheme of the sharding is to improve the reputation of the nodes that perform well, and the nodes with higher reputations have higher probabilities to be selected as leaders of the committee [20], [21]. The reputation method is mainly to improve the security of the sharding by optimizing the selection of the committee, rather than an incentive allocation strategy. Meanwhile, blockchain technologies provide a kind of new possibilities to resolve the security and privacy problems in the Internet of Things (IoT) [22] and edge computing [23] environments. Ding et al. [24] studied a specialized incentive mechanism for the blockchain to attract IoT devices to purchase computational power from edge servers to participate in the mining process. Huang et al. [25] proposed a hybrid system for edge computing environments and analyzed the incentive allocation algorithm in the system. Most of these works are based on their specific application scenarios, thus generalized algorithms still have room for development.

There are some works on the incentive for relay nodes in blockchain systems. The techniques and environments used in these works are diverse. Babaioff et al. [3] addressed that there is no motivation for nodes to broadcast transactions, and proposed a reward scheme that motivates nodes to propagate transactions. Their scheme is based on the assumption that the network is modeled as a forest of $d$-ary directed trees, each of them of height $H$. Abraham et al. [26] designed a signed propagation chain solution for their proposed blockchain called Solidus to motivate information propagation. The signed propagation chain requires support from the committee of

Solidus. Moreover, the chain requires all transactions to be continuously encrypted according to the propagation path. This results in completely different versions of the same transaction processed by all nodes, thus additional overhead may be required during decryption and synchronization. Ersoy et al. [27], [28] found a formula that distributes the transaction fee among propagating nodes. Their formula includes a critical parameter related to the network topology, and the choice of this parameter is an open problem. Li et al. [29] proposed CreditCoin to incentivize vehicles to forward announcements in a blockchain-based network. Because their solution relies on the trusted authority and a cloud server, it is not centralized and cannot be applied in general networks. Wang et al. [30] proposed a solution to a UTXO-based blockchain network. Because the solution assumes that all nodes in the network only propagate transactions to a certain miner, their solution cannot be applied when the miner is uncertain. In addition, their solution requires a large number of additional transactions for each transaction to distribute revenue to all nodes on the path.

## III. Problem Formulation and System Model

In this section, we discuss the background of incentive allocations for relay nodes. We further propose corresponding formulations and models to solve the problem.

### *A. Background and Goal of Incentive Allocation*

In traditional blockchain systems, block generators receive revenue for processing and authorizing transactions. The revenue includes two parts. First, to encourage nodes to maintain the blockchain, the blockchain system gives revenue to the block generators. Second, to encourage the block generator to authorize a transaction, the transaction proposer associates a transaction fee along with the transaction. The block generator receives the fee after authorizing the transaction.

In addition to the block generators, it is necessary to provide revenue to relay nodes, because relay nodes contribute to blockchain systems. Transactions need to be broadcast to all nodes in the network of the blockchain so that nodes can reach a consensus. To effectively and reliably broadcast transactions by the general flooding algorithm, each relay node needs to forward transactions to its neighbors. Thus, forwarding nodes play a critical role in transaction broadcasting and blockchain consensus. However, nodes may have the motivation to refuse forwarding transactions for their own benefit. Babaioff et al. [3] pointed out that a node may hold a transaction to prevent others from knowing it, thus increasing the probability of obtaining the corresponding transaction fee in the future blocks. Meanwhile, there is no penalty for those nodes that do not forward transactions in the current blockchain systems. For instance, in the current Bitcoin P2P network [31], the protocol requires a node to forward at least one transaction through a link within 30 minutes to keep the link connected. Nodes can only forward a small portion of transactions, thereby reducing bandwidth consumption while maintaining links with peers. This results in low forwarding efficiency in the network.

For these reasons, we allocate revenue to relay nodes and incentivize them to forward transactions. We call this process incentive allocation. To achieve our goal, we propose a new blockchain system that supports the incentive allocation in Section IV. We describe the incentive allocation algorithm in Section V and analyze the security in Section VI.

### *B. Revenue for Mining and Forwarding*

In a real blockchain, we need to consider the cost-to-income ratio, that is, the unit revenue of successful mining divided by the unit cost needs to be greater than the unit revenue of successful forwarding divided by the unit cost. In the case that nodes can get revenue from mining and forwarding, nodes may stop mining if the forwarding cost-to-income ratio is better than the mining cost-to-income ratio. For sake of simplicity, we do not discuss the cost of mining and forwarding in this paper. We discuss the revenue in the following. In this case, nodes are not motivated to generate new blocks, thus the security of the blockchain is compromised.

To avoid this threat, the mining revenue must be equal to or greater than the forwarding revenue. The source of the mining revenue includes the transaction fees and the system revenue for new blocks, while the source of the forwarding revenue only includes the transaction fees. If the transaction fees for relay nodes are less than 50% of the total transaction fees, the mining revenue will always be higher than the forwarding revenue. Nodes will keep mining, because the mining revenue from transaction fees is no less than the forwarding revenue shared by all nodes, and the blockchain also provides revenue for successfully mining new blocks.

### *C. Blockchain Network Topology*

It is feasible to record the network topology in the blockchain, based on the following two facts. First, although the number of wallets in the blockchain can be very large, the number of relay nodes is limited. As the largest blockchain network, the Bitcoin network has over 74 million wallets by July 2021 [32] and around 12 thousand nodes [33]. Thus, most blockchain wallets do not create relay nodes, and the number of relay nodes allows us to compute the topology of the network. Second, previous works [34], [35] show that the number of links from a specific node is limited. Therefore, the total number of links is limited and can be stored.

We use the blockchain to maintain the network topology. The network topology is denoted as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ indicates the set of nodes as a vertex set, and $\mathcal{E}$ indicates the set of links between different nodes as an edge set. There are two types of nodes in $\mathcal{G}$, wallet nodes and relay nodes. Wallet nodes do not forward transactions. Thus, wallet nodes have no link between each other, while relay nodes can have links to any other nodes.

### *D. Change of Links*

The change of links in the blockchain includes connecting events and disconnecting events.

*1) Connecting events:* Two nodes connect through a new link when they can directly contact each other. Connecting events happen when new nodes join or nodes try to improve their connectivity. To set up a new link, the pair of connected nodes both propose the link to announce that the node is established. A new link becomes valid once the pair of connected nodes both send the new link messages.

Similar to transaction proposers paying transaction fees to block generators, nodes that generate new links also pay fees to block generators. Two nodes connected through a link need to pay fees for establishing the link. It serves two purposes. First, it encourages nodes to maintain existing links instead of frequently replacing links to enhance the stability of the network. Second, it can prevent malicious nodes from filling a large number of connecting events in the block to achieve the effect of denial-of-service (DoS) attacks. Therefore, a reasonable payment method must not only enable honest nodes to gain benefits while maintaining links but also prevent the network from being attacked by DoS attacks. The specific amount of the payment and the ratio of the payment by both nodes are beyond the scope of this paper.

*2) Disconnecting events:* A node disconnects from an existing link when it loses contact with the other node. Disconnecting events happen because of reasons such as the leaving of nodes, restrictions of the network bandwidth, or power outages. In general, the node that causes a disconnecting event does not broadcast specific messages to indicate it proactively. Thus, a link becomes invalid when any endpoint detects and proposes the loss of the link.

Note that links can be declared invalid unilaterally. Disabling links can save network bandwidth and power consumption of nodes, thus nodes may proactively shut down some links. However, disabling links may also decrease the connectivity of the network. It is necessary to ensure that nodes cannot gain extra profits by disabling any active links to avoid malicious disconnecting events. We will prove that nodes cannot gain extra profits by disabling any links in Section V.

### *E. Change of Nodes*

The change of nodes in the blockchain network includes nodes joining and leaving the network. The blockchain does not record the specific message indicating that a new node has joined. However, other nodes can detect new nodes. Once a new address appears in links, nodes will know that a new node with the new address has joined the network, and then the new node will be added into $\mathcal{V}$. Temporary disconnection of nodes is common in the network, and nodes will not send specific announcements to indicate that they have left the network. Therefore, we do not discuss nodes leaving the network in this paper.

### *F. Activated Time of Nodes*

Although nodes are not removed from the node set $\mathcal{V}$, most nodes do not participate in the blockchain continuously. A study in Ethereum [34] shows that most nodes do not transfer funds frequently. We consider that most nodes that do not transfer funds frequently are inactive. Compared with such inactive nodes, nodes who frequently participate in transactions are more actively involved in transaction forwarding. Thus, we focus on active nodes from $\mathcal{V}$. We introduce a property which is called activated time to measure the active level of nodes.

**Activated time:** The activated time of a node is the index of the latest block that includes a transaction where the node is the payer or payee.

## IV. Blockchain Design

In this section, we discuss the design of the ITF blockchain system. We propose a specialized structure to maintain network topology and incentive allocations.

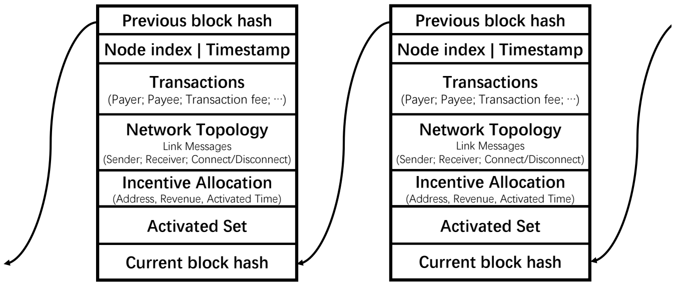

Fig. 1. An overview of the ITF block

### *A. ITF Blockchain Structure*

Fig. 1 provides an overview of the ITF blockchain structure. ITF records verification information and transactions, which is similar to traditional blockchains based on Nakamoto consensus [2]. Besides, ITF has two additional fields: network topology field and incentive allocation field. The network topology and incentive allocations can be computed according to the data stored in the above fields, in which all nodes reach a consensus.

*1) Network topology field:* The network topology field records the network topology information, including connecting and disconnecting messages. A link between two nodes will be valid if both endpoints of this link broadcast connecting messages, and blocks will record them when receiving the connecting messages from both nodes. Any endpoint of a link can propose a disconnecting message, and blocks will record the disconnecting message when receiving it from either node of the link.

*2) Incentive allocation field:* The incentive allocation field records the final result of incentive allocation. To indicate the result of incentive allocation, the block records the address, revenue, and activated time of each node that receives revenue for relay nodes in the block. The block generator computes the incentive allocation for all transactions in the current block and stores it in the block.

If the block does not record the result of incentive allocation correctly, it will not be approved by nodes. Nodes will not acknowledge transactions and incentive allocation in unapproved blocks. It motivates the block generator to maintain the network topology field and incentive allocation field.

### B. Network Topology Maintenance

*1) Network topology consensus in the blockchain:* Ideally, all nodes can reach a consensus on the network following the time order of all network changes. Unfortunately, it is difficult to strictly follow the time order, because time synchronization is hard among all nodes. Even if there is an unambiguous time stamp system to confirm the exact time, nodes are still unable to guarantee that all of the network changes happen chronologically due to inevitable network delay. Thus, ITF uses the index of the block as the signal of applying network topology changes. Each block has a unique index and confirmed previous blocks. Each block and its previous blocks record confirmed network topology changes. These confirmed network topology changes construct a confirmed network. More specifically, for a blockchain $\mathcal{B} = \{B_1, B_2, ..., B_n\}$ with the latest block $B_n$, once an index $i$ where $i \in [1, n]$ is selected, there will exist a list of corresponding unambiguous network topology changes which includes all changes from $B_1$ to $B_i$. In this case, all nodes can reach the network topology consensus and the blockchain consensus together. To avoid block generators selecting connecting and disconnecting events in the blocks maliciously, topology changes in the current block have no impact on incentive allocations in the current block. Incentive allocations in block $B_n$ apply the network topology which is accumulated by network topology changes from $B_1$ to $B_{n-1}$.

*2) Storage consumption:* In general, the number of connecting events does not exceed the number of transactions in the long term, because the purpose of connecting links in the network is to help transmit transactions. Establishing a link is to facilitate the transfer of multiple transactions between two nodes. If the number of connecting events is greater than the number of transactions, there must exist redundant links. Since connecting events require payments, nodes will not establish too many redundant links. Meanwhile, the number of disconnecting events cannot exceed the number of connecting events. A node will not propose disconnecting events unnecessarily for the potential future use of links.

A connecting event message only needs to include basic information such as the addresses and signatures of both nodes, which consumes fewer resources than a transaction. A disconnecting event message includes similar information and has a similar low cost. In practice, the consumption of the network topology field will be much smaller than the storage consumption of transactions.

### C. Incentive Allocation Maintenance

*1) Incentive allocation storage:* As Fig. 1 shows, the incentive allocation field records three entries for each node that receives revenue.

- Address: The wallet address of the node.
- Revenue: The amount of revenue received by the node.
- Activated time: The activated time of the node.

The revenue is added to the address of each corresponding node automatically.

*2) Restricting the maximum number of nodes in incentive allocations:* As we mentioned above, the number of wallets in blockchain can be large. Therefore, restricting the maximum number of nodes in incentive allocations is necessary. We propose the following strategy to limit the maximum number of nodes when computing incentive allocations.

The main idea of our strategy is to allocate revenue for active nodes in the network. The node with more recent activated time is considered more active. A set of nodes is selected to receive the revenue by ordering activated times of all nodes. We call the set **activated set**. To avoid block generators selecting transactions in the blocks maliciously, the changes of the activated set in the current block cannot affect incentive allocations in the current block. According to the common prefix property, $k$ is a parameter[1] to make sure that honest nodes can confirm blocks until $B_{n-k}$, and attackers cannot modify those blocks. Therefore, transactions in the block $B_n$ give revenue to nodes in the activated set in the block $B_{n-k}$. In this way, the block generator cannot manipulate the incentive allocation by modifying the activated set.

## V. Incentive Allocation Algorithm

In this section, we propose a novel incentive allocation algorithm to distribute revenue to relay nodes. We discuss the goal of the allocation, propose the algorithm, and provide detailed analyses.

### A. Goals of the Incentive Allocation Algorithm

We list the following goals of the incentive allocation algorithm.

- The algorithm allocates revenue by contributions of relay nodes when broadcasting transactions.
- Nodes cannot increase their revenue by disconnecting.
- The computational time complexity of the algorithm is acceptable to most nodes.

### B. Incentive Allocation Algorithm and Analysis

The input of the algorithm consists of a set of transactions $\mathcal{T}$, nodes in the activated set $\mathcal{V}'$ and their links $\mathcal{E}'$. Transaction $t$ in $\mathcal{T}$ consists of three parts, i.e., $t = (s, q, w)$. $s$ denotes the payer of transaction $t$ which starts to broadcast the transaction to the network. $q$ denotes the payee of transaction $t$. $w$ denotes the transaction fee.

We now discuss a reduction of the graph $\mathcal{G}' = (\mathcal{V}', \mathcal{E}')$. according to the transaction broadcast process. When node $s$ broadcasts a transaction $T$ to the network, the transaction is sent to all nodes who directly connect to $s$. Each relay node will forward the transaction once it receives the transaction. It implies that nodes may forward the transaction if it receives the transaction through the shortest paths from $s$. Links that belong to the shortest paths are sufficient to forward the transaction to

[1] $k$ is 6 in Bitcoin.

all nodes in the shortest time. A transaction forwarding process over one of these links is called a **sufficient forwarding**. Since the shortest paths contribute to all of the sufficient forwarding processes, the framework of broadcast contribution is composed of the shortest paths. This fact allows us to simplify the calculation of the incentive allocation through the reduction of the graph. Thus, we propose a reduction that nodes can ignore links that are not a part of the shortest paths from $s$ when computing the incentive allocation of the transaction. This reduction keeps the broadcast process efficient because a node always receives the transaction from the shortest paths from $s$. This reduction also removes links that are less involved in the broadcast process of the transaction. We propose Algorithm 1 to apply the reduction to the graph.

TABLE I
NOTATIONS USED IN ALGORITHMS

| | |
|---|---|
| $\mathcal{TG}$ | Result graph after applying reduction algorithm |
| $d_i$ | The length of the shortest path from the payer $s$ to node $i$. Node $i$ is located at level $d_i$ |
| $p_i$ | The outdegree of node $i$ in $\mathcal{TG}$ |
| $r_{d_i}$ | The ratio of level $d_i$'s revenue to level $M-1$'s revenue |
| $c_{d_i}$ | The total number of nodes at level $d_i$ |
| $g_{d_i}$ | The total number of outdegrees of all nodes at level $d_i$ |
| $M$ | The maximum value of $d_i$ |
| $S$ | The total amount of $r_{d_i}$ for all possible $d_i$ |
| $a_i$ | The number of revenue to node $i$ |
| $\mathcal{D}$ | The set of $d_i$ |
| $(i,j)$ | A directed link between node $i$ and $j$ |

**Algorithm 1** Graph Reduction Algorithm

```
Input: $\mathcal{G}' = (\mathcal{V}', \mathcal{E}'), t = (s, q, w)$
Output: $\mathcal{TG} = (\mathcal{TV}, \mathcal{TE}), \mathcal{D} = \{d_i | i \in \mathcal{TV}\}$
$\mathcal{TV} = \mathcal{V}'$, $\mathcal{TE} = \emptyset$
for all $i \in \mathcal{V}'$ do
    Obtain $d_i$ by building a BFS tree from $s$
end for
$\mathcal{D} = \{d_i | i \in \mathcal{V}'\}$
for all $(i,j) \in \mathcal{E}'$ do
    if $d_j = d_i + 1$ then
        Append $(i,j)$ to $\mathcal{TE}$
    end if
end for
```

The computational time complexity of this algorithm is $O(|\mathcal{V}'| + |\mathcal{E}'|)$, which is the same as the time complexity of building a BFS tree.

We use **level** to denote a node set that includes all nodes that are the same distance from $s$. For example, level $i$ includes all nodes with distance $i$ from $s$. Any link in $\mathcal{TE}$ connects a node at level $n$ to another node at level $n+1$. We call the payer $s$ is the root in the graph $\mathcal{TG}$ since $d_s = 0$. Any simple path from the root to any node $i$ is a projection of one of the shortest paths from the payer $s$ to the node $i$ in the original graph $\mathcal{G}$. As a special case, when a node $i$ cannot reach $s$, $d_i = \infty$.

Next, we propose an algorithm for distributing revenue. Our algorithm includes mainly three steps. First, the algorithm collects necessary information of levels and nodes for computing the following steps. Second, the algorithm computes the total revenue of each level according to the network topology to ensure that the revenue of any node will not increase after the node increases its level. Third, the algorithm counts the number of links between each node and its next level to determine the revenue of each node. The detailed algorithm is presented in Algorithm 2.

**Algorithm 2** Incentive Allocation Algorithm

```
Input: $\mathcal{TG} = (\mathcal{TV}, \mathcal{TE}), w, \mathcal{D} = \{d_i, i \in \mathcal{TV}\}$
Output: $\mathcal{A} = \{a_i, i \in \mathcal{TV}\}$
$\forall n \in [0, |\mathcal{TV}| - 1]$, initialize $c_n = 0, r_n = 0, g_n = 0$
$\forall i \in \mathcal{TV}$, initialize $p_i = 0, a_i = 0$
for all $(i,j) \in \mathcal{TE}$ do
    $p_i = p_i + 1$
end for
for all $i \in \mathcal{TV}$ do
    $c_{d_i} = c_{d_i} + 1$
    $g_{d_i} = g_{d_i} + p_i$
end for
Find maximize $M$ s.t. $c_M \neq 0$
$r_{M-1} = 1$
for $n$ from $M-2$ to 1 do
    $r_n = r_{n+1}((c_n - 1)c_{n+1} + 1)/2$
    $S = S + r_n$
end for
for all $i \in \mathcal{TV}$ do
    $a_i = (p_i r_{d_i} w)/(g_{d_i} S)$
end for
```

In Algorithm 2, $\frac{r_{d_i} w}{S}$ is the actual revenue distributed to level $d_i$. $\frac{p_i}{g_{d_i}}$ is the ratio of the links from node $i$ to the total links from level $d_i$. We derive $a_i$ by $\frac{p_i r_{d_i} w}{g_{d_i} S}$.

We now prove that nodes cannot get more revenue by unilaterally disconnecting.

**Lemma 1.** *Given the graph $\mathcal{TG}$ and a transaction, node $i$ can only keep or increase the shortest path from the transaction payer $s$ by disconnecting $i$ from other nodes.*

*Proof.* Assume that after removing link $(i,j)$ from $\mathcal{TG}$, $d_i$ is decreased to $d_i'$, $d_i' < d_i$. There exists a path $P = (u_1, u_2, ..., u_{d_i'})$ from $s$ to $i$, satisfying $u_1 = s$, and link $(u_x, u_{x+1})$ exists for each integer $x \in [1, d_i' - 1]$. The path $P$ is still available before removing link $(i,j)$, thus $d_i \leq d_i'$. It contradicts the assumption $d_i' < d_i$.

Node $i$ can keep $d_i$ unchanged by keeping all its links unchanged. In addition, node $i$ can increase $d_i$ by disconnecting links. For example, when $i$ disconnects all links, $d_i = \infty$. □

**Theorem 2.** *For a specific transaction, node $i$ cannot get more revenue by disconnecting it from other nodes, while other nodes keep their shortest paths.*

*Proof.* From Lemma 1, node $i$ can only keep or increase its level $d_i$. First, we discuss the case that node $i$ keeps its level $d_i$. Our algorithm computes ratio $r_{d_i}$ for level $d_i$, then gives node $i$ ratio $\frac{p_i}{g_{d_i}}$ of total revenue of level $d_i$. $r_{d_i}$ is independent of the number of links. If node $i$ disconnects nodes at level $d_i - 1$, $\frac{p_i}{g_{d_i}}$ will not change. If node $i$ disconnects nodes at level $d_i + 1$, $\frac{p_i}{g_{d_i}}$ will decrease. Thus, node $i$ cannot get more revenue by disconnecting it from other nodes, while all nodes keeping their shortest paths.

Next, we discuss the maximum revenue that node $i$ can get when $d_i$ increases. We consider that node $i$ moves its level $d_i$ to $d_i + 1$. Every node at level $d_i$ has at most one link to each node at level $d_i + 1$, thus each node has at most $c_{d_i+1}$ links to level $d_i + 1$. There are at most $(c_{d_i} - 1)c_{d_i+1}$ links at this level except links of node $i$. Therefore, any node $i$ at level $d_i$ receives at least $\frac{1}{(c_{d_i}-1)c_{d_i+1}+1}$ of total revenue at level $d_i$. When $d_i = M - 1$, moving to the next level cannot get any revenue since $r_{d_i+1} = r_M = 0$. Otherwise, level $d_i + 2$ exists when $d_i < M - 1$. There are at least $c_{d_i+2}$ links from level $d_i+1$ to level $d_i+2$. Before node $i$ moves to level $d_i+1$, there exists at least one link from a node at level $d_i+1$ to each node at level $d_i+2$. Node $i$ has at most $c_{d_i+2}$ links to nodes at level $d_i+2$. Node $i$ can get at most $\frac{1}{2}$ of revenue from level $d_i+1$. Since $r_{d_i} = \frac{r_{d_i+1}((c_{d_i}-1)c_{d_i+1}+1)}{2}$, we have $\frac{r_{d_i}}{(c_{d_i}-1)c_{d_i+1}+1} \geqslant \frac{r_{d_i+1}}{2}$, therefore node $i$ always receives revenue from level $d_i$ no less than revenue from level $d_i + 1$.

As a result, for a specific transaction, node $i$ cannot get more revenue by disconnecting it from other nodes, while other nodes keep their shortest paths. □

With the knowledge in the proof of Theorem 2, we explain Algorithm 2 line by line. Line 1 and 2 initialize the variables. Lines 3 to 5 compute the outdegree of each node in $\mathcal{TG}$, which is also the number of links to the next level. Lines 6 to 9 computes the total number of nodes at each level, and the total number of links from these nodes to the next level. Line 10 finds the deepest level $M$, and sets the multiplier $r_{M-1}$ to 1. Level $M$ cannot receive revenue since the nodes do not forward the transaction. Nodes at level $M-1$ only serve nodes at level $M$. For convenience, the multiplier of the $M-1$ level $r_{M-1}$ with the least income is set to 1. Lines 11 to 14 establish a relationship between the revenue of each two adjacent level, that is, $r_n = r_{n+1}((c_n - 1)c_{n+1} + 1)/2$. The reason we set the multiplier is discussed in the proof of Theorem 2. $S$ is the total of the multiplier for each level, which means each level $d_i$ finally receives $r_{d_i}/S$ of total revenue $w$. Lines 15 to 17 compute the final revenue $a_i$ to each node $i$. $a_i$ consists two parts, $r_{d_i}w/S$ is the total revenue to level $d_i$, and $p_i/g_{d_i}$ is the rate that $i$ receives from the total revenue of level $d_i$.

The computational time complexity of this algorithm is $O(|\mathcal{TV}| + |\mathcal{TE}|)$. $|\mathcal{TV}| = |\mathcal{V}|$ and $|\mathcal{TE}| < |\mathcal{E}|$, and the total complexity of Algorithm 1 and 2 is $O(|\mathcal{V}| + |\mathcal{E}|)$.

Our incentive allocation algorithm matches the goals defined in this section. The algorithm allocates revenue based on the contributions of relay nodes when broadcasting transactions. Theorem 2 proves that nodes cannot increase their revenue by disconnecting. The algorithm with linear computational time complexity is acceptable to most nodes. It is worth mentioning that wallet nodes cannot obtain revenue. The wallet nodes are not connected with each other, and the relay nodes connected with the wallet nodes always receive the transaction first. Therefore, wallet nodes have no connection to the next level and cannot obtain revenue.

## VI. Attack Resistance

In this section, we analyze possible attacks to show that the ITF blockchain and the incentive allocation algorithm are secure against potential hazards.

### A. Attacking Models

We assume that all nodes are selfish and rational. No node intends to tamper with the system without extra profits. Because the ITF blockchain inherits mining parts and mechanisms from Bitcoin [2], resistances of classical attacks like double-spending [36] are retained. We propose and analyze two potential attacks against the ITF system.

*1) Sybil attack:* A Sybil attacker creates a large number of pseudonymous nodes and obtains extra profits through these nodes [14].

*2) Activated set attack:* The activated set attack is a special attack against the ITF system. In the ITF system, a node receives revenue only if the activated set includes it. When an adverse node is not in the activated set, it sends a transaction immediately to join the activated set and receives revenue.

### B. Safety Analysis

Without loss of generality, we assume there is one adversary that controls multiple adverse nodes. The primary target of the adversary is to increase the revenue from incentive allocations.

Essentially, the adversary attempts to obtain extra profits by changing the graph composed of nodes in the activated set. The adversary can create two kinds of fake identities, including pseudonymous nodes and fake links. They can use these fake identities to change the network topology and further manipulate incentive allocations. The adversary claims those identities but does not maintain these fake identities, thus fake identities cannot provide services. Theoretically, the adversary can have an infinite number of pseudonymous nodes, and most of them have no links. The adversary adds a node by adding links to the node with no links initially and removes a node by removing all the links of the node. Thus, we can reduce the operations of nodes to the operations of links. Moreover, the adversary cannot obtain extra profits by removing links as we prove in Section V-B. Therefore, we can simplify all the operations of the adversary into adding fake links.

After adding links, the adversary may obtain extra profits in the following two cases. First, our incentive allocation algorithm distributes different total revenue to different levels. Since changing the shortest path means changing the level, the adversary obtains different revenue by improving the shortest paths. Second, since the total revenue of each level is fixed, the adversary obtains extra profits by increasing the density of adverse nodes at specific levels.

*1) Changing shortest paths:* To shorten the shortest paths, the adversary needs to declare fake links between nodes whose original shortest path gap is at least 2. We first discuss fake links that connect an honest node and an adverse node. Once an honest node receives a transaction from any link, it can estimate the delivery time from all links by public topology knowledge. Fake links cannot deliver transactions in time.

Thus, honest nodes can detect such fake links and disconnect from the corresponding adverse nodes.

Then we discuss the adversary adding fake links between adverse nodes. In general, there exists at least one honest node whose shortest path is improved by a link with an adverse node. This honest node is supposed to receive information earlier through this link. Since there exists a fake link between adverse nodes, the honest node cannot receive transactions in the expected time. The honest node will disconnect from links cannot forward transactions in time, although these links are not fake. As a result, the adversary will lose the revenue for serving this honest node.

As a special case, we consider none of the shortest paths between any two honest nodes is improved. It implies that the adversary cannot obtain extra profits from transactions from honest nodes. The adversary can only obtain extra profits from transactions of adverse nodes. Note that most of the transaction fees are paid to honest nodes because honest nodes always have most of the computing power to be block generators. Therefore, the adversary cannot obtain extra profits from honest nodes to have extra profits over incentive allocation.

*2) Dominating the activated set:* To dominate the activated set, the adversary creates transactions between its adverse nodes, then these nodes join the activated set and start to receive revenue. However, in order for block generators to approve the transactions, the adversary needs to pay a large number of transaction fees for such transactions. Before the adversary dominates the activated set, most of the revenue is paid to honest nodes. After the adversary dominates the activated set, honest nodes will join the activated set because of new transactions. Thus, the adversary has to make transactions and pay transaction fees periodically to keep dominating the activated set. We give evaluations in Section VII to show that the adversary cannot obtain extra profits in this case.

## VII. Performance Evaluation

In this section, we provide the results of the performance of the ITF blockchain. We show the distribution of the incentive allocation. We also conduct attacks to test the security of the ITF blockchain. We write code to simulate all nodes, and they operate the same blockchain. All code is written in C++.

### *A. Distribution of Incentive Allocation*

To show the contribution of nodes in forwarding processes, we define **sufficient forwarding** in Section V-B. Theoretically, nodes that make more sufficient forwarding contribute more during the forwarding processes and may receive more revenue. Therefore, we compare the sufficient forwarding times of nodes and the revenue they receive.

We generate the test network by algorithms in [37]. This algorithm generates a realistic model, incorporating hierarchy and redundancy. The number of links of each node varies from 4 to 60. The range of the number of links is sufficient to show the relationship between the number of links and the revenue of nodes, and there are enough samples for each number of links in the generated graph.

The test network has 10000 nodes. Each node broadcasts a transaction once to join the activated set. We compute the final revenue and the number of sufficient forwarding times for each node. The activated set includes all nodes initially, thus all nodes can receive transaction fees for relay nodes. We assume that all honest nodes broadcast transactions with the same amount of transaction fees, which is called the **standard transaction fee**. Assume that a node costs $f$ transaction fees and receives $u$ revenue. $f$ includes transaction fees for broadcasting transactions, while $u$ includes revenue for relay nodes and block generators. We define the **profit rate** as $(u - f)/f_0$, where $f_0$ is the standard transaction fee. In this test, $f = f_0$, because each node broadcasts a transaction once. We assume that each node has the same computing power, thus each node has the same probability to become a block generator, and all nodes will receive the same proportion of transaction fees for block generators. To show the impact of the transaction fees for relay nodes, we set the transaction fees for relay nodes to the maximum, which is 50%.

Fig. 2(a)(b) shows the distribution of the profit rate and the number of sufficient forwarding times. The profit rate increases when the number of links increases. We further process data to find the relationship between the profit rate, the number of sufficient forwarding times, and the number of links. We are interested in the average profit rate per sufficient forwarding and we call it **average unit profit rate** in the following discussion. Fig. 2(c) shows that the average unit profit rate increases when the number of links increases. Note that nodes with negative profit rates pay revenue to nodes with positive profit rates in the network. Most nodes with less than 22 links have an average unit profit rate of less than zero, and nodes with more than 22 links have an average unit profit rate of more than zero. It implies that a node with more links receives more profits with each sufficient forwarding. The line is not stable when the number of nodes is greater than 50 because the number of samples with such a number of links is not enough.

We derive a relationship between the number of links and the average profit rate per sufficient forwarding. We compute the average unit profit rate divided by the number of links. The dotted line in Fig. 2(c) shows it. The dotted line keeps increasing for a small number of links, and slowly increasing. As we can see, the average unit profit rate per link is stable at zero when the number of links reaches a threshold. Because the average unit profit rate is accumulated by the contribution of all links, we can conclude that a node receives revenue linearly as the number of links increases.

As a result, the incentive allocation algorithm distributes revenue to nodes based on their contributions. It encourages nodes to improve the connectivity of the system by setting up links that support sufficient forwarding.

### *B. Sybil Attack*

To test the resistance to the Sybil attack we mentioned in Section VI-A, we generate the test network by Watts-Strogatz model [38]. This model generates networks that reveal the

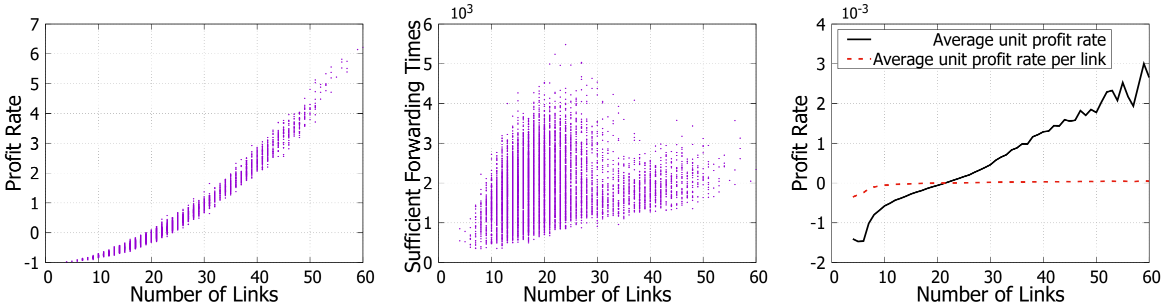

Fig. 2. The simulation result of the generated network with 10000 nodes. (a) shows the distribution of the number of links and the profit rate. (b) shows the distribution of the number of links and the number of sufficient forwarding times. (c) shows the average unit profit rate and the average unit profit rate per link for each number of links.

properties of some real communication networks. Compared with the algorithm in [37], the distribution of the link number of each node in the graph generated by this algorithm is more concentrated, which is clearer for observing the impact of the attack.

We now analyze the cost and profit of the Sybil attack for simulation design. The cost of the Sybil attack is for pseudonymous nodes created by the adversary to join the activated set. Each pseudonymous node needs to pay a transaction fee to join the activated set. Since the creation of a pseudonymous node cannot improve the computing power of the adversary, the adversary cannot receive more revenue for block generators. Therefore, there are two possibilities for increasing profit, including reducing costs and creating pseudonymous nodes to obtain more revenue for relay nodes. Based on the composition of cost and profit, the simulations will compare the profit of the adversary when it creates different numbers of pseudonymous nodes and pays different amounts of transaction fees for pseudonymous nodes.

The test network includes 1000 nodes. We randomly choose a node as an adverse node, and it creates multiple pseudonymous nodes. These pseudonymous nodes and the adverse node construct a complete graph. Each node broadcasts one transaction, and each pseudonymous node also broadcasts one transaction to join the activated set. The activated set includes all nodes initially, thus all nodes can receive transaction fees for relay nodes. We assume that all honest nodes broadcast transactions with the standard transaction fee $f_0$. The adversary pays $f$ for each of its pseudonymous nodes and the adverse node. When the adversary pays $yf_0$ for each of $x$ pseudonymous nodes, it totally pays $f = xyf_0$. We assume each node has the same computing power, thus each node has the same probability to be the block generator. Pseudonymous nodes cannot be the block generator, because the computing power of the adversary is limited. The profit $u$ of the adversary is the total revenue of the adverse node and all pseudonymous nodes. We compute the total profit rate $(u - f)/f_0$. The adversary obtains extra profits when the profit rate is positive. To show the resistance of our algorithm to the Sybil attack, we maximize the revenue of the adversary. Thus, block generators obtain 50% of transaction fees, and relay nodes obtain another 50%.

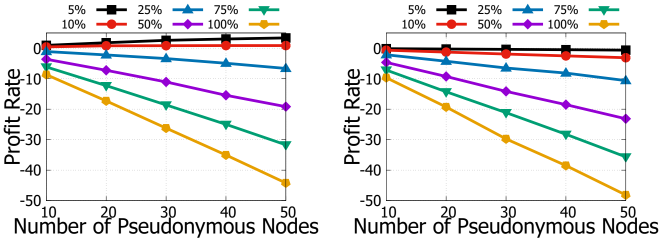

Fig. 3. The simulation result of the Sybil attack. Different lines indicate the adversary pays different percentages of the standard transaction fee to join the activated set. The mean degree of each node is 10 in (a), 50 in (b).

Fig. 3 shows that the profit rate grows linearly with the number of pseudonymous nodes changes. When the slope is positive, more pseudonymous nodes obtain more profits. Otherwise, more pseudonymous nodes suffer more losses. In Fig. 3(a), the adverse node can obtain extra profits if pseudonymous nodes broadcast transactions with no more than 10% of standard transaction fees. The adversary cannot obtain extra profits in Fig. 3(b). The difference between (a) and (b) is the mean degree of each node. We can infer that improving connectivity can improve the safety of our system.

We conclude that our system can resist the Sybil attack. There are two reasons that make the Sybil attack ineffective. First, the adversary only obtains extra profits when their transactions with low transaction fees can be accepted. However, block generators are honest most of the time. They always choose transactions with higher transaction fees for more revenue. It avoids pseudonymous nodes joining the activated set at a low cost. Second, the adversary cannot obtain the revenue for block generators. The adversary can only obtain the revenue for relay nodes. As a result, the adversary cannot obtain extra profits through the Sybil attack, thus our system can resist the Sybil attack.

### C. Activated Set Attack

We design a numerical simulation to demonstrate the performance of the activated set attack we mentioned in Section

VI-A. For the same reason in Section VII-B, we generate the test network by Watts-Strogatz model [38].

We now analyze the cost and profit of the activated set attack for simulation design. To obtain the transaction fees continuously, the adversary has to keep itself in the activated set. Whenever it is not in the activated set, it will immediately broadcast a transaction to refresh the activated time and return to the activated set. The cost of the activated set attack is for the adversary to join the activated set, and the profit is the revenue for relay nodes. The simulations will compare the profit of the adversary when it pays different amounts of transaction fees for joining the activated set with different sizes.

The test network includes $n$ nodes. We randomly choose a node as an adverse node. For simplicity, each node has an index $i \in [1, n]$, and each node broadcasts one transaction in the ascending order of the index to join the activated set. The node that broadcasts the transaction later has the later activated time. The size of the activated set is a variable $x$. The activated set initially includes nodes with indices from $n - x + 1$ to $n$. We assume that all honest nodes broadcast transactions with the standard transaction fee $f_0$. The adversary pays $f$ for each of its transactions. The profit $u$ of the adversary is the total received revenue for relay nodes. We compute the total profit rate $(u - f)/f_0$. For the same reason in Section VII-B, we let block generators obtain 50% of transaction fees and relay nodes obtain another 50%.

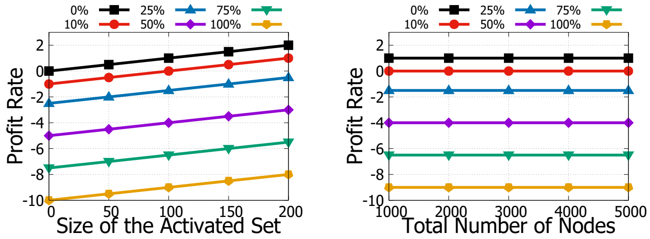

Fig. 4. The simulation result of the activated set attack. Different lines indicate the adversary pays different percentages of the standard transaction fee to join the activated set. (a) shows the results of different sizes of the activated set in the graph with 1000 nodes. (b) shows the results in graphs with a different number of nodes, and the size of the activated set is 10% of the total number of nodes.

Fig. 4 shows that the profit rate decreases linearly with the increase of the transaction fees. In Fig. 4(a), the profit rate grows linearly with the size of the activated set. In Fig. 4(b), the total number of nodes does not affect the profit rate. The adversary obtains extra profits when the profit rate is positive. Otherwise, the adversary cannot obtain extra profits. Therefore, we are interested in zero points of the profit rate. In Fig. 4(a), the zero points of the profit rate appear at 0 activated nodes when the adverse node pays 0% of the standard transaction fee. The zero points of the profit rate appear at 100 activated nodes when the adverse node pays 10% of the standard transaction fee. Based on the simulation result, we infer the relationship between the size of the activated set, the transaction fee paid by the adverse node, and the extra profit. The extra profit is zero when $\frac{Transaction\ fee\ of\ the\ adverse\ node}{Standard\ transaction\ fee} = \frac{Size\ of\ the\ activated\ set}{Total\ number\ of\ nodes}$. For example, if the size of the activated set is 25% of the total number of nodes, the adverse node cannot obtain extra profits if its transaction fee for each transaction is greater than 25%. Fig. 4(b) shows that the total number of nodes does not affect the profit rate, it implies that this conclusion keeps in different sizes of networks.

The above formula shows the transaction fee threshold at which the adversary can obtain extra profits by the activated set attack. Thus, honest nodes can simply resist this attack by rejecting transactions with transaction fees equal to or lower than the threshold. We can conclude our system can resist the activated set attack.

## VIII. Conclusion and Future Work

In this paper, we have proposed the ITF blockchain system focusing on secure incentive allocations over blockchain networks. We have proposed a mechanism to maintain the network topology for the blockchain network. We have proposed an efficient incentive allocation algorithm that rewards relay nodes based on their contributions. We have discussed the security of the ITF system and the incentive allocation algorithm by analyzing several types of attacks. We have conducted simulations to show that our proposed system fairly rewards nodes and resists several types of attacks.

Although we have proposed a mechanism to detect and maintain the topology of the blockchain network, validating the network topology may be difficult for some nodes with limited computing power. Meanwhile, new nodes need to trace all network topology changes to construct the current network topology. We will develop network topology requesting mechanisms in our future work. Moreover, we have discussed that applying the incentive allocation algorithm to our ITF blockchain system. Developing a generalized incentive allocation algorithm will be an important open problem to the blockchain. We will seek the possibility of applying the incentive allocation algorithm to various blockchain systems.

## References

[1] D.-J. Deng, K.-C. Chen, and R.-S. Cheng, "Ieee 802.11 ax: Next generation wireless local area networks," in *10Th international conference on heterogeneous networking for quality, reliability, security and robustness*. IEEE, 2014, pp. 77–82.

[2] S. Nakamoto *et al.*, "Bitcoin: A peer-to-peer electronic cash system," 2008.

[3] M. Babaioff, S. Dobzinski, S. Oren, and A. Zohar, "On bitcoin and red balloons," in *Proceedings of the 13th ACM conference on electronic commerce*, 2012, pp. 56–73.

[4] I. Eyal and E. G. Sirer, "Majority is not enough: Bitcoin mining is vulnerable," in *International Conference on Financial Cryptography and Data Security*. Springer, 2014, pp. 436–454.

[5] R. Pass and E. Shi, "Fruitchains: A fair blockchain," in *Proceedings of the ACM Symposium on Principles of Distributed Computing*. ACM, 2017, pp. 315–324.

[6] Y. Amoussou-Guenou, A. Del Pozzo, M. Potop-Butucaru, and S. Tucci-Piergiovanni, "Correctness and fairness of tendermint-core blockchains," *arXiv preprint arXiv:1805.08429*, 2018.

[7] M. Feldman, K. Lai, I. Stoica, and J. Chuang, "Robust incentive techniques for peer-to-peer networks," in *Proceedings of the 5th ACM conference on Electronic commerce*. ACM, 2004, pp. 102–111.

[8] X. Yan, F. Ye, Y. Yang, and X. Deng, "An autonomous compensation game to facilitate peer data exchange in crowdsensing," in *2017 IEEE/ACM 25th International Symposium on Quality of Service (IWQoS)*. IEEE, 2017, pp. 1–6.

[9] Y. He, H. Li, X. Cheng, Y. Liu, C. Yang, and L. Sun, "A blockchain based truthful incentive mechanism for distributed p2p applications," *IEEE Access*, vol. 6, pp. 27 324–27 335, 2018.

[10] L. Buttyan, L. Dora, M. Felegyhazi, and I. Vajda, "Barter-based cooperation in delay-tolerant personal wireless networks," in *2007 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks*. IEEE, 2007, pp. 1–6.

[11] K. Zhao, S. Tang, B. Zhao, and Y. Wu, "Dynamic and privacy-preserving reputation management for blockchain-based mobile crowdsensing," *IEEE Access*, vol. 7, pp. 74 694–74 710, 2019.

[12] D. Liu, A. Alahmadi, J. Ni, X. Lin *et al.*, "Anonymous reputation system for iiot-enabled retail marketing atop pos blockchain," *IEEE Transactions on Industrial Informatics*, 2019.

[13] M. Feldman, C. Papadimitriou, J. Chuang, and I. Stoica, "Free-riding and whitewashing in peer-to-peer systems," *IEEE Journal on selected areas in communications*, vol. 24, no. 5, pp. 1010–1019, 2006.

[14] Z. Trifa and M. Khemakhem, "Sybil nodes as a mitigation strategy against sybil attack," *Procedia Computer Science*, vol. 32, pp. 1135–1140, 2014.

[15] J. Poon and T. Dryja, "The bitcoin lightning network: Scalable off-chain instant payments," 2016.

[16] V. Sivaraman, S. B. Venkatakrishnan, K. Ruan, P. Negi, L. Yang, R. Mittal, G. Fanti, and M. Alizadeh, "High throughput cryptocurrency routing in payment channel networks," in *17th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 20)*, 2020, pp. 777–796.

[17] Z. Avarikioti, L. Heimbach, Y. Wang, and R. Wattenhofer, "Ride the lightning: The game theory of payment channels," in *International Conference on Financial Cryptography and Data Security*. Springer, 2020, pp. 264–283.

[18] O. Ersoy, S. Roos, and Z. Erkin, "How to profit from payments channels," in *International Conference on Financial Cryptography and Data Security*. Springer, 2020, pp. 284–303.

[19] L. Luu, V. Narayanan, C. Zheng, K. Baweja, S. Gilbert, and P. Saxena, "A secure sharding protocol for open blockchains," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, 2016, pp. 17–30.

[20] M. Zhang, J. Li, Z. Chen, H. Chen, and X. Deng, "Cycledger: A scalable and secure parallel protocol for distributed ledger via sharding," *arXiv preprint arXiv:2001.06778*, 2020.

[21] C. Huang, Z. Wang, H. Chen, Q. Hu, Q. Zhang, W. Wang, and X. Guan, "Repchain: A reputation based secure, fast and high incentive blockchain system via sharding," *IEEE Internet of Things Journal*, 2020.

[22] Q. Wang, X. Zhu, Y. Ni, L. Gu, and H. Zhu, "Blockchain for the iot and industrial iot: A review," *Internet of Things*, vol. 10, p. 100081, 2020.

[23] Z. Xiong, Y. Zhang, D. Niyato, P. Wang, and Z. Han, "When mobile blockchain meets edge computing," *IEEE Communications Magazine*, vol. 56, no. 8, pp. 33–39, 2018.

[24] X. Ding, J. Guo, D. Li, and W. Wu, "An incentive mechanism for building a secure blockchain-based internet of things," *IEEE Transactions on Network Science and Engineering*, 2020.

[25] Y. Huang, Y. Zeng, F. Ye, and Y. Yang, "Incentive assignment in pow and pos hybrid blockchain in pervasive edge environments," in *2020 IEEE/ACM 28th International Symposium on Quality of Service (IWQoS)*. IEEE, 2020, pp. 1–10.

[26] I. Abraham, D. Malkhi, K. Nayak, L. Ren, and A. Spiegelman, "Solidus: An incentive-compatible cryptocurrency based on permissionless byzantine consensus," *CoRR, abs/1612.02916*, 2016.

[27] O. Ersoy, Z. Ren, Z. Erkin, and R. L. Lagendijk, "Transaction propagation on permissionless blockchains: incentive and routing mechanisms," in *2018 Crypto Valley Conference on Blockchain Technology (CVCBT)*. IEEE, 2018, pp. 20–30.

[28] O. Ersoy, E. Zekeriya, and R. L. Lagendijk, "Tulip: A fully incentive compatible blockchain framework amortizing redundant communication," in *2019 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*. IEEE, 2019, pp. 396–405.

[29] L. Li, J. Liu, L. Cheng, S. Qiu, W. Wang, X. Zhang, and Z. Zhang, "Creditcoin: A privacy-preserving blockchain-based incentive announcement network for communications of smart vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 7, pp. 2204–2220, 2018.

[30] X. Wang, Y. Chen, and Q. Zhang, "Incentivizing cooperative relay in utxo-based blockchain network," *Computer Networks*, vol. 185, p. 107631, 2021.

[31] Bitcoin.org, "P2p network guide," "https://developer.bitcoin.org/devguide/p2p_network.html", 2021.

[32] Statista, "Number of blockchain wallet users worldwide from november 2011 to july 11, 2021," "https://www.statista.com/statistics/647374/worldwide-blockchain-wallet-users/", 2021.

[33] Bitnodes, "Global bitcoin nodes distribution," "https://bitnodes.io/", 2021.

[34] T. Chen, Y. Zhu, Z. Li, J. Chen, X. Li, X. Luo, X. Lin, and X. Zhange, "Understanding ethereum via graph analysis," in *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*. IEEE, 2018, pp. 1484–1492.

[35] S. Delgado-Segura, S. Bakshi, C. Pérez-Solà, J. Litton, A. Pachulski, A. Miller, and B. Bhattacharjee, "Txprobe: Discovering bitcoin's network topology using orphan transactions," in *International Conference on Financial Cryptography and Data Security*. Springer, 2019, pp. 550–566.

[36] U. W. Chohan, "The double spending problem and cryptocurrencies," *Available at SSRN 3090174*, 2017.

[37] M. B. Doar, "A better model for generating test networks," in *Proceedings of GLOBECOM'96. 1996 IEEE Global Telecommunications Conference*. IEEE, 1996, pp. 86–93.

[38] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *nature*, vol. 393, no. 6684, p. 440, 1998.