BlockRoam: Blockchain-based Roaming Management System for Future Mobile Networks

Cong T. Nguyen, Diep N. Nguyen, Dinh Thai Hoang, Hoang-Anh Pham,
Nguyen Huynh Tuong, Yong Xiao, and Eryk Dutkiewicz

Abstract—Mobile service providers (MSPs) are particularly vulnerable to roaming frauds, especially ones that exploit the long delay in the data exchange process of the contemporary roaming management systems, causing multi-billion dollars loss each year. In this paper, we introduce BlockRoam, a novel blockchain-based roaming management system that provides an efficient data exchange platform among MSPs and mobile subscribers. Utilizing the Proof-of-Stake (PoS) consensus mechanism and smart contracts, BlockRoam can significantly shorten the information exchanging delay, thereby addressing the roaming fraud problems. Through intensive analysis, we show that the security and performance of such PoS-based blockchain network can be further enhanced by incentivizing more users (e.g., subscribers) to participate in the network. Moreover, users in such networks often join stake pools (e.g., formed by MSPs) to increase their profits. Therefore, we develop an economic model based on Stackelberg game to jointly maximize the profits of the network users and the stake pool, thereby encouraging user participation. We also propose an effective method to guarantee the uniqueness of this game's equilibrium. The performance evaluations show that the proposed economic model helps the MSPs to earn additional profits, attracts more investment to the blockchain network, and enhances the network's security and performance.

Index Terms—Mobile roaming, fraud prevention, proof-of-stake, Stackelberg game, and blockchain.

1 Introduction

1.1 Motivation

With the popularity of IT technologies and smart devices, over 5 billion people have been subscribed to mobile services, generating a $1.03 trillion revenue globally in 2018 [1]. Although the number of subscribers and the revenues will continue to grow, mobile service providers (MSPs) have been facing several obstacles, especially for roaming services. Among them, fraud management is one of the biggest challenges for MSPs with over $32.7 billion annual loss throughout the world [2]. Roaming fraud exploits the inefficiency in managing data exchanges between two MSPs in order to use illegal free-riding services. In particular, when a subscriber moves from its Home Public Mobile Network (HPMN) to a Visited Public Mobile Network (VPMN) and remotely accesses services of the HPMN via the VPMN's facilities, the HPMN has to pay the VPMN for the subscriber's service usage costs incurred according to the roaming agreement. However, the HPMN may not be able to charge the subscriber properly due to the delay in data exchange between the HPMN and VPMN, i.e., the time interval between when the subscriber finished using the service and when the HPMN received the service report from the VPMN [3]. For example, a subscriber can fraudulently obtain subscription from the HPMN, e.g., by SIM cloning or using invalid identities, and uses roaming services in the VPMN. Such roaming frauds can only be detected and responded to after the HPMN receives the service report, which might take more than 4 hours. This long delay, coupled with interoperability issues between different mobile networks, is the main reason why roaming frauds are hard to detect and prevent in current roaming systems, causing significant losses of up to €40,000 in some severe incidents [4].

Recently, the rapid development of blockchain technology has enabled blockchain-based applications in various areas, including Internet-of-Things, healthcare, military, and service providers. In particular, thanks to its advantages of low latency and negligible computational requirement, the PoS consensus mechanism has emerged to be an effective solution to data management in networks consisting of devices with limited computational capacity [5]. Specifically, blockchain technology can be leveraged to significantly reduce the data exchange delay in traditional roaming system, which can help the mobile operators to detect and respond

This work was supported in part by the Joint Technology and Innovation Research Centre—a partnership between the University of Technology Sydney and the VNU Ho Chi Minh City University of Technology (VNU HCMUT) and also partially funded by the European Union's Horizon 2020 programme under the Marie Skłodowska-Curie grant agreement No. 872752. Cong T. Nguyen was partially funded by Vingroup Joint Stock Company and supported by the Domestic Master/ PhD Scholarship Programme of Vingroup Innovation Foundation (VINIF), Vingroup Big Data Institute (VINBIGDATA), code 2020.TS.06. Y. Xiao's work was supported in part by the National Natural Science Foundation of China under Grant 62071193 and the Key R & D Program of Hubei Province of China under Grant 2020BAA002. (Corresponding author: Yong Xiao).

- *Cong T. Nguyen, Hoang-Anh Pham, and Nguyen Huynh Tuong are with the Ho Chi Minh City University of Technology, VNU-HCM, Vietnam. E-mail: {ntcong.sdh19, anhpham, htnguyen}@hcmut.edu.vn.*
- *Diep N. Nguyen, Dinh Thai Hoang, and Eryk Dutkiewicz are with the School of Electrical and Data Engineering, University of Technology Sydney, Australia. E-mail: {diep.nguyen, hoang.dinh, eryk.dutkiewicz}@uts.edu.au.*
- *Y. Xiao is with the School of Electronic Information and Communications at the Huazhong University of Science and Technology, Wuhan, China (e-mail: yongxiao@hust.edu.cn). Y. Xiao is also with the Pazhou Lab, Guangzhou, China.*

to the frauds much sooner, thereby minimizing the financial loss. Moreover, the privacy of the mobile roamers can be enhanced thanks to blockchain's advanced cryptography techniques such as digital signatures and asymmetric keys. This can help to protect roamers' sensitive information such as travel location and travel history. Note that deanonymization attack, e.g., attempting to link a blockchain account to an IP address [6], [7], or guessing patterns of transaction [7] may compromise users' privacy. However, these types of attacks are not effective when targeting mobile users in roaming as mobile users' IPs are usually dynamic. Furthermore, current systems often rely on Data Clearing Houses (DCHs) to process and transmit roaming service records. In addition to the middleman fee which the mobile operators have to pay the DCH, the reliance on such a centralized entity means that if the DCH is down, the current system cannot exchange data. Therefore, in this paper, we propose BlockRoam, a PoS blockchain solution to address the high delay problem in existing roaming systems.

1.2 Related Work

Typically, a roaming fraud protection system consists of preventive and reactive layers as illustrated in Fig. 1 [3]. The preventive layer prevents fraud perpetration by validating subscribers' authentication, auditing subscribers' credit, limiting services duration, and so on. Although these measures can help to mitigate roaming frauds, they have a negative impact on the Quality-of-Service provided to the subscribers, e.g., frequent validation and service limitation will lower customer satisfaction. The reactive layer typically consists of four main stages to detect and react to roaming fraud attacks. The roaming data, e.g., service records, exchanged between MSPs is first collected at the data collection stage and processed at the fraud detection stage to detect potential fraud cases [3]. Each case is then supervised manually in the supervision stage. The service usage is terminated if a fraud attack is confirmed at the response stage. Among these stages, data collection is often the bottleneck in the roaming fraud protection system. Techniques employed at this stage can only support data collection in near real-time with a limited number of subscribers, e.g., Fraud Information Gathering System [8], or shorten the data exchanging delay to 4 hours, e.g., Near Real Time Roaming Data Exchange [9]. Due to the sequential nature of the system, other stages cannot be activated if the data has not been collected. Consequently, although fraud attacks such as SIM cloning can also perpetrate locally in the HPMN, their consequences are much more severe in the roaming scenario due to the delay in data exchange, e.g., it takes up to 18 hours on average before an international roaming fraud attack can be stopped with the current system [4].

With outstanding performance in data integrity, decentralization, and privacy-preserving, blockchain has been emerging to be a secure and effective solution for data management in many decentralized networks. As a result, blockchain-based solutions for mobile roaming have been introduced recently by some organizations, e.g., IBM [10], Deutsche Telekom and SK Telecom [11], and Enterprise Ethereum Alliance [12], focusing on identity management, automating billing processes, and fraud prevention. In particular, these solutions focus on developing blockchain's asymmetric keys and digital signatures to manage subscriber identities and propose smart contracts to set up roaming pacts and automate billing processes. With enhanced identity management and automatic billing, fraud attacks can be significantly reduced. However, most of these solutions are still at the early stage of development and are facing several technical challenges.

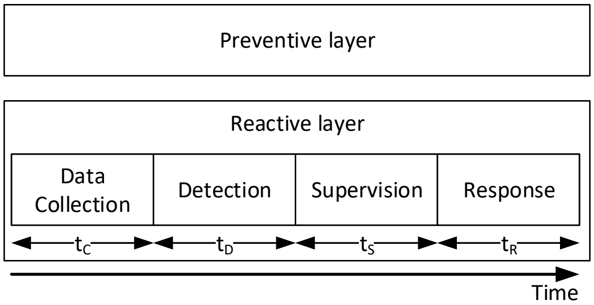


Fig. 1: Illustration of the current fraud protection system.

Specifically, most of current blockchain-based data management systems often employ the Proof-of-Work (PoW) consensus mechanism, e.g., Bitcoin [13]. However, the PoW mechanism consumes massive amounts of energy, e.g., the Bitcoin network's energy consumption is higher than that of many countries [14]. Moreover, PoW-based networks often take a long time to reach consensus, e.g. one hour on average [5]. Thus, a new consensus mechanism, namely Proof-of-Stake (PoS), has been developed with significant advantages over the PoW mechanism, including reduced energy consumption and delay [5]. Recently, a PoS-based blockchain network, namely Bubbletone [15], has been introduced for MSPs to address roaming fraud problems. Using the PoS-based consensus mechanism and smart contracts, the blockchain-based Bubbletone system provides a general platform for various MSP-to-MSP and MSP-to-subscriber interactions in the roaming environment. Nevertheless, the consensus mechanism design is not thoroughly discussed in [15].

In addition, more users (e.g., mobile subscribers) participate in a PoS-based blockchain network means better the performance and security of the network are. Thus, it is important to incentivize more users to participate in the network. In current PoS-based blockchain systems, some stakes, e.g., network tokens, are paid to the users as a reward for consensus participation. However, a user with a few stakes is less likely to receive the reward. Moreover, some blockchain networks such as [15] impose a high stake requirement for consensus participation. Consequently, the stakeholders, i.e., subscribers, are inclined to join a stake pool (formed by MSPs) to earn more rewards. Furthermore, a stake pool can earn profits from the investments of the stakeholders by charging a portion of each stakeholder's reward [5]. As a result, the formation of a stake pool can be beneficial if it can incentivize more subscribers and MSPs to join the network. Therefore, the design of stake pool and network parameters has a significant impact on the performance of a blockchain network, yet studies on this topic are still limited. The stake pool formation in PoS-based blockchain networks was analyzed in our previous work in [5]. However, [5] only considers the investment strategies of the users while the stake pool's pricing policy is assumed to be static. In practice, however, the pool has to design its pricing policy to maximize the profits while attracting more investments from the stakeholders.

1.3 Contributions and Paper Organization

The main contributions of this paper are briefly summarized as follows:

- We propose BlockRoam, an effective blockchain-based roaming service management system to provide a transparent, secure, and automatic platform for data exchanging between the MSPs. In particular, by employing the PoS consensus mechanism, BlockRoam can achieve a delay of less than 3 minutes as will be shown later in Section 3, which is much lower than the 4-hour delay of traditional roaming management systems. In addition to the reduced latency, BlockRoam can automate various roaming processes thanks to smart contracts [16], and thus roaming frauds can be significantly reduced. Moreover, the MSPs often rely on Data Clearing Houses (DCHs) to process and exchange data, which incurs additional costs [3]. In our proposed system, the transactions are stored in the blockchain and processed by smart contracts, and thus the service fees for DCHs can be eliminated. Furthermore, the privacy and security of the subscribers in BlockRoam are significantly enhanced thanks to the blockchain's advanced cryptography techniques [17].
- We analyze existing PoS-based consensus mechanisms [19]–[25] to show that they are not suitable for roaming management due to their limitations in terms of security and performance. Therefore, we develop a consensus mechanism for BlockRoam, which can meet strict security requirements, mitigate a wide variety of blockchain attacks, and achieve a much better performance in terms of transaction confirmation time compared to those of existing mechanisms.
- We introduce an economic model based on the Stackelberg game theory in order to jointly maximize the profits of the stake pool and the stakeholders. By analyzing utility functions of the stake pool and stakeholders, we develop a Mixed Integer Linear Programming model to find the Stackelberg equilibrium of our proposed game. We also propose an effective method that can guarantee to achieve the unique equilibrium for this game. The proposed economic approach can help to maximize the profits of the stake pool and the stakeholders, as well as attracting more investment and improving BlockRoam's security and performance.
- Extensive simulations have been performed to evaluate the performance of our game theoretic model. Particularly, we simulate the game to show that the model can brings additional benefits for the stake pool and the stakeholders. Moreover, we also examine the influence of important parameters on the outcome of the game. Furthermore, adversarial attacks scenarios are also simulated to show that the proposed economic model can help to improve the network's security and performance by attracting more investments to the network. These results are especially crucial in designing appropriate parameters (e.g., total network stakes, pool fees, and rewards) to improve BlockRoam's security and performance.

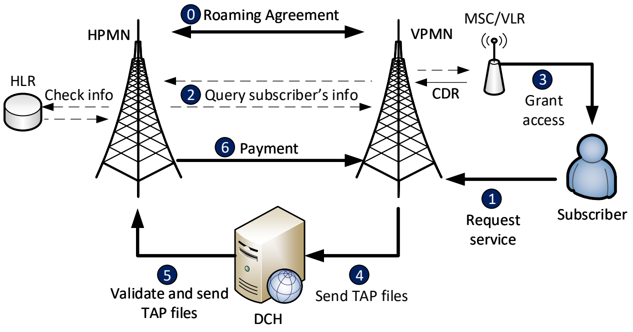


Fig. 2: Illustration of a typical roaming system [3].

The remainder of this paper is organized as follows. We first provide the background about current mobile roaming systems and blockchain technology and introduce BlockRoam in Section 2. We then analyze the security and performance of BlockRoam in Section 3. After that, we formulate and analyze the stake pool and stakeholders game in Section 4. Finally, simulations and numerical results are presented in Section 5, and conclusions are summarized in Section 6.

2 Background and System Model

2.1 Current Roaming Systems

The current roaming system is illustrated in Fig. 2 [3]. In the current system, firstly, a roaming pact is established between two MSPs. Then, when a subscriber wants to use services from its HPMN while being in the service area of the VPMN, the subscriber sends a request to the VPMN. Then, the VPMN queries the HPMN about the services that the subscriber has subscribed to. This information is stored in the Home Location Register (HLR) database of the HPMN. If the subscription information is correct, the VPMN will provide the subscriber access to the corresponding services (e.g., voice or data service) through the Mobile Switching Center/Visited Location Register (MSC/VLR). The Call Detail Records (CDRs) are then sent to both networks where the CDRs are processed for subscription billings and invoices generation. Afterward, the VPMN sends a Transfer Account Procedure (TAP) file which contains the CDR information to the HPMN. Usually, there is a Data Clearing House (DCH) company acting as a middleman, which validates and transmits the TAP files for the VPMN. Once the HPMN receives the TAP files, it will pay the VPMN in accordance with the roaming pact [3].

Fraud attacks in roaming occur when a subscriber gains access to the roaming services, but the HPMN is unable to charge the subscriber for the services provided. In this case, the HPMN still has to pay the VPNM for the facilities provided during the roaming process, which may result in significant financial loss. For example, a fraudulent SIM can use up to 18 hours of service on average, and in some incidents, the loss rate is up to €40,000 per hour [4]. The current roaming system is vulnerable to roaming fraud attacks mainly because of the delay in data exchanging between the HPMN and the VPMN. Even with the Near Real Time Roaming Data Exchange scheme [9], the data exchange can be delayed up to 4 hours, and thus it may take a long time to detect and determine the fraud. Even if the fraud is found, it is still difficult for the HPMN to response as it does not have direct control over the VPMN's facilities [3].

2.2 Blockchain Fundamentals

A blockchain is a sequence (chain) of blocks, where each block consists of data (transactions) shared among users in the network. When a transaction is generated by a user, it will be first verified by miners, i.e., nodes who participate in the consensus process, to verify the transaction. After the transaction is verified and added to a new block, the block will be broadcast to the rest of the nodes in the network. Based on the distributed consensus mechanism, a block will be selected from all the blocks proposed by the miners to append to the chain [17]. Besides the transactions, a block also contains a hash pointer created by the hash functions which map all the block contents and the last block's pointer to the current block's pointer. Therefore, any change in previous blocks will result in a different hash value in the next one, and it can be traced back to the first block of the chain. As a result, the whole blockchain is tamper-evident, i.e., any attempt to alter the previous blocks can be immediately detected. This is one of the most crucial advantages of blockchain technology compared to other security mechanisms. Another advantage is that a blockchain network is decentralized, and thus there is no single point of failure, i.e., the network's operation is ensured even when some nodes are failed. In contrast, for the current roaming system, if the DCH is failed, the CDRs and TAP files cannot be transmitted, and in this case, the whole system will stop working.

A smart contract is a program stored in the blockchain network consisting of a set of rules created by users. If the rules are satisfied, the contract will automatically be enforced by the consensus mechanism. The content of a smart contract is visible to all network users, thus transparency is ensured [16]. For example, an HPMN and a VPMN can negotiate with each other and make a smart contract on the blockchain, which is triggered when a transaction with CDR data is sent to the smart contract address. Then, when the transaction is verified and added into the blockchain, all consensus participants execute the contract code and trigger the events according to the terms of agreement written in the contract, e.g., the HPMN automatically pays the VPMN as per their agreement.

The distributed consensus mechanism is the backbone of a blockchain network, which governs most of the blockchain's operations and ensures that once the data is stored in a block, it is extremely difficult to be altered without the consensus of most of the nodes in the network. Currently, most of the blockchain networks have been employing the PoW consensus mechanisms. In the PoW, the users compete with each other in a solution searching procedure where a user with higher computational power may have higher opportunities to be the block winner who will add a new block to the chain and receive the reward. This competition leads to the waste of energy in PoW-based blockchain networks. Moreover, PoW-based blockchain networks often experience high delays in reaching consensus due to security reasons. This makes PoW consensus mechanisms inappropriate to implement in mobile roaming systems requiring low delay for fraud prevention.

Unlike the PoW, each block in PoS-based blockchain networks is dedicated to an authorized participant (leader) for mining in advance based on stakes of stakeholders in the network. This mechanism has many advantages over the PoW, including lower energy consumption and delay, and thus PoS-based blockchain applications can be employed effectively in networks with thousands of users [5]. Currently, there are several variations of the PoS mechanism, each has some desirable characteristics that are suitable for roaming management as well as some limitations that hinder their applicability in this specific context. In the following, we discuss advantages as well as disadvantages of each mechanism in details.

- *Proof-of-Activity (PoA)* [23] is one of the first PoS mechanisms proposed. This mechanism uses the block header of previous blocks to determine the leader for the current block, which helps to ensure unbiased randomness and prevent grinding attacks as proven in [23]. However, this mechanism is a hybrid PoW-PoS mechanism, and thus it has inherent limitations of PoW mechanism such as high energy consumption and long delay.
- *Casper* [20] is another PoW-PoS hybrid mechanism. Although this mechanism is proven to be secure and able to mitigate many attacks, it still has performance limitations because of the PoW mechanism.
- *Chain-of-Activity (CoA)* [22] is a pure PoS mechanism, and thus it can achieve a relatively low delay (transaction confirmation time) (6 minutes) and requires negligible energy consumption. Nevertheless, the security of this mechanism is not proven rigorously in the paper, and its real-world application network has a relatively low transaction throughput (60 transactions per second).
- *Tendermint* [21], developed based on a Byzantine Fault Tolerance (BFT) protocol, can achieve very low delay and high throughput. However, Tendermint relies on a set of validators to vote for the consensus, but how these validators are chosen is not discussed in the paper. Moreover, this mechanism requires high a communication complexity, i.e., $O(n^3)$, and the security analysis in the paper is not extensive (does not consider several attacks).
- *Ouroboros* [19] is a PoS mechanism with strong theoretical background and rigorous security analysis. The mechanism is proven to be secure, satisfying

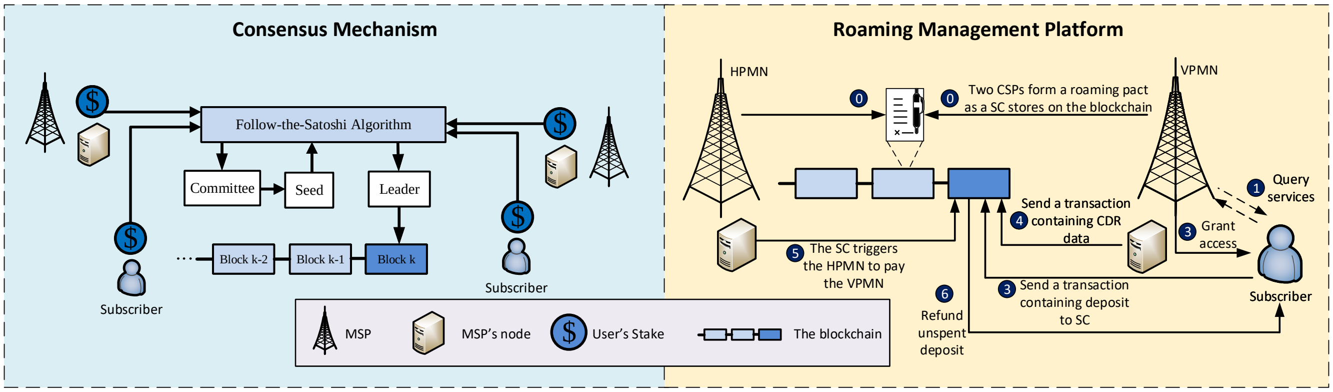

Consensus Mechanism
Follow-the-Satoshi Algorithm
Committee
Seed
Leader
Block k-2
Block k-1
Block k
Subscriber
Subscriber
MSP
MSP's node
User's Stake
The blockchain
Roaming Management Platform
HPMN
VPMN
0
0 Two CSPs form a roaming pact as a SC stores on the blockchain
1 Query services
3 Grant access
4 Send a transaction containing CDR data
5 The SC triggers the HPMN to pay the VPMN
3 Send a transaction containing deposit to SC
6 Refund unspent deposit
Subscriber

Fig. 3: Illustration of the proposed BlockRoam system.

the persistence and liveness properties [26] with overwhelming probability, and able to mitigate many attacks. However, in case of a strong adversary, the delay is significantly increased.

- *Algorand* [24] is proven to be secure and can achieve high performance. However, the mechanism can tolerate only an adversarial ratio of 1/3, and there is no incentive mechanism and attack analysis in the paper
- *Delegated Proof-of-Stake (DPoS)* [25] is a variation of PoS that employs a committee to create blocks. However, this mechanism requires a lot more communications, is more prone to centralization, and can tolerate a smaller adversarial ratio compared to those of the other PoS mechanisms

The main advantages and limitations of the considered consensus mechanisms are summarized in Table 1. As observed in the table, all the consensus mechanisms have security flaws or performance limitations that make these mechanism unsuitable for the roaming management application. Thus, in the next Section, we will propose a consensus mechanism for BlockRoam to address these issues.

2.3 BlockRoam

2.3.1 Network Model

Our proposed blockchain-based system consists of two main components, namely the roaming management platform and the consensus mechanism as illustrated in Fig. 3. The roaming management platform supports complex interactions between the users, automates various roaming processes, and provides a universal currency, i.e., blockchain network tokens, for payments. In addition to the roaming processes, the network can also take part in the consensus mechanism to maintain the network's operations and security, store data (e.g., roaming pacts, subscriber information, and transaction history), and execute roaming processes such as payments and processing CDRs.

2.3.2 Roaming Management Procedure

The roaming process, the main procedure of the roaming management platform, consists of seven main steps as follows:

- *Step 0:* Two MSPs form a roaming pact consisting of tariff plans for services offered to the subscribers and the payment agreement between two MSPs. This roaming pact is made in the form of a smart contract and stored in the blockchain.
- *Step 1:* When a subscriber (roamer) wants to use services from its HPMN, the subscriber queries the VPMN and receives available tariff plans as per the roaming agreement between the VPMN and the HPMN.
- *Step 2:* If the subscriber agrees to use the service, the subscriber sends a transaction containing a sufficient amount of money (in form of digital tokens) to the smart contract's address.
- *Step 3:* When the transaction is verified and sent successfully, the VPMN will grant the subscriber access to roaming facilities.
- *Step 4:* When the subscriber finishes its roaming service, the VPMN sends a transaction to the smart contract's address, which consists of the CDR data of the provided service.
- *Step 5:* The smart contract then automatically calculates the subscriber's service fee and sends it to the HPMN. The smart contract also triggers a transaction from the HPMN to the VPMN for payment of the service.
- *Step 6:* Finally, the smart contract sends the unused tokens to the subscriber.

2.3.3 Benefits

BlockRoam has the following advantages over the traditional roaming system:

- *Roaming fraud prevention:* The main obstacle to prevent and react to fraud attacks is the significant delay in data exchange, i.e., up to 4 hours. Our proposed system employs the PoS mechanism to speed up the data exchanging process, e.g., approximately 3 minutes on average as later shown in Section 3, and thus fraud attacks can be detected much earlier. Moreover, by using smart contracts, the billing process is executed right after the service usage finished. As a result, roaming fraud can be significantly mitigated.

TABLE 1: Advantages and limitations of several PoS consensus mechanisms

Consensus Mechanism	Advantages	Limitations
Proof-of-Activity [23]	Low communication complexity, can mitigate several attacks	Need PoW, high energy consumption, long delay, security analysis is not extensive
Chain-of-Activity [22]	Low delay, low communication complexity, can mitigate several attacks	Low transaction throughput, security analysis is not extensive
Casper [20]	Secure, can mitigate several attacks	Need PoW, high energy consumption, long delay
Tendermint [21]	Low delay	Security analysis is not extensive, high communication complexity
Ouroboros [19]	Secure, can mitigate several attacks, defined incentive mechanism, low communication complexity	Long delay in case of adversarial attacks
Algorand [24]	Secure, low delay, high transaction throughput, low communication complexity	Can tolerate low adversarial ratio, no incentive mechanism, does not analyze attacks
DPoS [25]	Secure, low delay	High communication complexity, more centralized, can tolerate low adversarial ratio.

- *Cost saving:* In our proposed system, the CDRs are stored in the blockchain and processed by smart contracts. Therefore, the DCHs are no longer needed, and thus the middleman fees are eliminated. Moreover, our system automates various processes, such as subscribers billing and HPMN payments, which can further reduce operational costs. Furthermore, our system's energy consumption is negligible compared to that of PoW-based systems, and thus our energy cost is much lower.
- *Security and privacy:* Using cryptographically secure mechanisms, the privacy and security of the subscribers can be significantly improved. Each subscriber in the network uses a pair of public and private keys for identification and verification. The network only needs the subscriber's digital signature which can be easily verified and almost impossible to forge. This also protects the anonymity of the subscribers, as the subscriber's real-life identity is completely unrelated to the network identity.

3 BlockRoam's Consensus Mechanism

We have shown that the existing PoS mechanisms are not suitable for the roaming management application due to either security flaws, or insufficient performance ability. Therefore, in this section, we propose a novel consensus mechanism for BlockRoam. We also conduct analyses to show that the proposed consensus mechanism can satisfy the strict security requirements and achieve more desirable performance compared to existing mechanisms.

3.1 Proposed Consensus Mechanism

3.1.1 Epochs and Time Slots

In our proposed consensus mechanism, time is divided into epochs, each of which is composed of N_e time slots. At the first time slot of epoch e_k, a committee, consists of some users (stakeholders), executes an election protocol to elect one leader for each time slot in epoch e_k. The election protocol also selects the committee members for the epoch e_{k+1}. If a leader fails to broadcast its block during its designated time slot (being offline during its time slot), an empty block will be added to the chain. The leader is also instructed to not change its broadcast blocks at any later time.

3.1.2 Leader and Committee Election Protocol

To elect the leaders and committee members, the committee members of epoch e_k execute the Publicly Verifiable Secret Sharing (PVSS) protocol [18] to create seeds for the Follow-the-Satoshi (FTS) algorithm [5]. The PVSS protocol allows the protocol participants to produce unbiased randomness in the form of strings and any network user to verify these strings. Moreover, the PVSS protocol can tolerate an adversarial ratio of up to 1/2, and this protocol is very efficient in terms of communication complexity, i.e., $O(m)$ where m is the number of committee members [18]. Once the random strings are created, they are used as the seeds for the FTS algorithm (a hash function that takes any string as input and outputs some token indices [5]). The current owners of these tokens are then chosen as leaders of epoch e_k and committee members of epoch e_{k+1}.

3.1.3 Incentive Mechanism

The incentive mechanism plays a crucial role in ensuring that the stakeholders follow the consensus mechanism properly. To this end, the incentive mechanism needs to incentivize participation in the consensus mechanism via a reward scheme and penalize malicious behavior via a penalty scheme. In the reward scheme, a leader will receive a fixed number of tokens when the leader adds a new block to the chain. The probability P_n that user n is selected by the FTS algorithm in a network of N stakeholders is

$$P_i = \frac{s_n}{\sum_{n=1}^{N} s_n}, \tag{1}$$

where s_n is the number of stakes (tokens) of stakeholder n. As observed from 1, the more stakes a stakeholder has, the higher chance it can be selected to be the leader and able to obtain the reward. For the penalty scheme, the leader is required to make a deposit that will be locked during its designated epoch to prevent nothing-at-stake, bribe [5], and transaction denial attacks [19]. The stakes of committee members are also locked during the epoch that they are serving in the committee to prevent long-range attacks [5].

3.2 Security Analysis

3.2.1 Blockchain Properties

To maintain the blockchain's operations and security, a consensus mechanism must satisfy the following properties [26]:

- **Persistence:** Once a transaction is confirmed by an honest user, all other honest users will also confirm that transaction, and its position is the same for all honest users.
- **Liveness:** After a sufficient period of time, a valid transaction will be confirmed by all honest users.

In our proposed system, persistence ensures that once a transaction is confirmed, it cannot be reverted. Without persistence, a fraudster can use the roaming services for free. For example, a fraudster can perform a double-spending attack by firstly sending a transaction Tx_1 to the smart contract. Then, after the VPMN has granted the fraudster access to the roaming service, the fraudster broadcasts a transaction Tx_2 which sends the tokens of Tx_1 to another address (e.g., the fraudster's second account). If Tx_1 has not been confirmed, Tx_2 is still valid and may be confirmed by honest users.

While the persistence property ensures data immutability, the liveness property ensures that every valid transaction will eventually be included in the chain. Without liveness, an attacker might successfully block every transaction coming from the MSP, and consequently, the roaming process cannot commence. It has been proven in [26] that the persistence and liveness properties are ensured if the consensus mechanism satisfies the following properties:

- **Common prefix (CP) with parameter $\kappa \in \mathbb{N}$:** For any pair of honest users, their versions of the chain $\mathcal{C}_1, \mathcal{C}_2$ must share a common prefix. Specifically, assuming that $\mathcal{C}_2$ is longer than $\mathcal{C}_1$, removing κ last blocks of $\mathcal{C}_1$ results in the prefix of $\mathcal{C}_2$.
- **Chain growth (CG) with parameter $\varsigma \in \mathbb{N}$ and $\tau \in (0, 1]$:** A chain possessed by an honest user at time $t + \varsigma$ will be at least $\varsigma\tau$ blocks longer than the chain it possesses at time t.
- **Chain quality (CQ) with parameter $l \in \mathbb{N}$ and $\mu \in (0, 1]$:** Consider any part of the chain that has at least l blocks, the ratio of blocks created by the adversary is at most $1 - \mu$.

We prove that our proposed consensus mechanism can satisfy the common prefix, chain growth, and chain quality properties with overwhelming probabilities in the following Theorem.

Theorem 1. *BlockRoam's consensus mechanism satisfy the common prefix, chain growth, and chain quality properties with overwhelming probabilities.*

Proof: See Appendix A. □

TABLE 2: Transaction confirmation times in minutes

Adversarial ratio	Bitcoin	Cardano	BlockRoam
0.10	50	5	1
0.15	80	8	1.3
0.20	110	12	1.6
0.25	150	18	1.6
0.30	240	31	2
0.35	410	60	2.3
0.40	890	148	2.6
0.45	3400	663	3

3.2.2 Roaming Fraud Protection Ability

To evaluate the roaming fraud protection ability of our system, we focus on the average resolution time t_{total}, i.e., the average time between the occurrence of a roaming fraud attack and the execution of the responses to the attack. As observed in Fig. 1, t_{total} is the sum of every stage's duration at the reactive layer, i.e., $t_{total} = t_C + t_D + t_S + t_R$. Since our proposed system can achieve a much lower t_C compared to the traditional roaming system, i.e., approximately 3 minutes (as later shown in Section 3.3) compared to 4 hours, the t_{total} of our system is nearly 4 hours shorter than that of the traditional roaming system.

3.2.3 Blockchain Attacks Mitigation

In the following Theorem, we prove that our proposed BlockRoam can also be able to mitigate and prevent a variety of emerging blockchain attacks such as double spending, grinding, bribe, nothing-at-stakes, and long-range attacks.

Theorem 2. *BlockRoam can mitigate double-spending, grinding, nothing-at-stakes, bribe, transaction denial, and long-range attacks as long as the adversary does not control more than 50% total network stakes.*

Proof: See Appendix B. □

When the adversary controls more than 50% of the total network stakes, both the persistence and liveness properties are no longer guaranteed [19]. Consequently, attacks such as double-spending, nothing-at-stakes, and transaction denial attacks can no longer be mitigated.

3.3 Performance Analysis

In Table 2, we examine and compare the transaction confirmation times under different adversarial ratio (percentage of stakes in PoS or computational power in PoW that the adversary controls) of a PoW blockchain network (Bitcoin), a PoS network with delayed finality (Cardano), and BlockRoam. The transaction confirmation time is the time it takes to reach a common prefix violation probability $\Pr_{\text{CP}} \leq 0.1\%$. Based on (14), κ can be determined, and then κ is multiplied with the slot time to calculate the transaction confirmation time. Our slot time is set to be 20 seconds (the same as that of Cardano [28]). The transaction confirmation times of Bitcoin and Cardano are presented in [19].

As observed in Table 2, the more stakes the adversary controls, the longer the transaction confirmation time is. Moreover, 51% attack [19] can break most of the PoW-based and PoS-based blockchain networks. Specifically, an adversary controlling more than 51% of total computational power in a PoW-based network or 51% of total stakes in a PoS-based network can successfully perform many attacks, including double-spending, nothing-at-stakes, and transaction denial attacks. Therefore, it is critical to attract more participants to our PoS-based blockchain system in order to increase the network's total stakes, thereby improving the common prefix violation probability and transaction confirmation time. In the next section, we will introduce an effective economic model that can jointly maximize profits for the participants, encouraging them to participate in the network and thus improving the network's performance and security.

4 Economic Model

4.1 Stake Pools and Stakeholders

In a PoS-based blockchain network, the probability that an individual user (stakeholder) with a small number of stakes is selected to be the leader is low as shown in (1). Moreover, when a stakeholder is selected to be the leader, it needs to be online during its designated time slot to (1) collect transactions from other users, (2) validate these transactions, (3) create a block containing valid transaction, (4) broadcast the block to the network. Therefore, if the stakeholder's connection is poor, it fails to create a valid block, and consequently it cannot obtain the block reward. Thus, stakeholders who participate in the consensus process need to maintain a strong connection to the network, which incurs an operational cost, e.g., $40 to $300 per month [29]. Therefore, small stakeholders often pool their stakes together to increase their opportunities to be the leaders and share operational costs, which results in the formation of stake pools, e.g., [30]–[32]. Such formation of a stake pool is also beneficial for the blockchain because no transaction is processed when the leader fails to create a valid block (which reduces transaction throughput). In BlockRoam, the stakeholders, e.g., the subscribers, might be more inclined to join the stake pool (e.g., formed by MSPs) to reduce their operational costs and have more stable incomes. A stake pool often charges a part of the stakeholder's profits for joining the pool, e.g., the Stakecube pool charges 3% of each reward a stakeholder receives [31]. In this section, we introduce an economic model using Stackelberg game in order to jointly maximize the profits of the stake pool and stakeholders, which is beneficial for MSPs and BlockRoam's operation and security.

We consider a PoS-based blockchain network with one stake pool and N stakeholders. The stakeholders have stake budgets $\mathbf{B} = (B_1, \dots, B_N)$ and individual operational costs $\mathbf{C} = (C_1, \dots, C_N)$. The stake pool has its own stake σ, and the pool defines a cost c and a fee α in advance for users who are interested in participating in the pool. The pool's cost is charged for joining the pool and maintaining its operations. The pool's fee is the profit margin of the pool's owner, which usually ranges from 1% to 9% in real-world stake pools, e.g., [30]–[32]. The stakeholders can use their budgets to invest p_i stakes to the pool and m_i stakes for self-mining (individually participate in the consensus process), such that $p_i + m_i \leq B_i$. Let denote $\mathcal{N}_p$ to be the set of stakeholders who invest in the pool, the probability P^w that the pool is selected to be the leader and obtains a block reward R is proportional to the pool's stakes in the total network stakes, i.e.,

$$P^w = \frac{\sigma + \sum_{n \in \mathcal{N}_p} p_n}{\sigma + \sum_{n \in \mathcal{N}_p} p_n + \sum_{j=1}^{N} m_j}. \tag{2}$$

After receiving the reward R, the pool calculates each stakeholder's reward r_i^p based on the proportion P_i^p of stakeholder i's stakes in the total stakes of the pool, which is

$$P_i^p = \frac{p_i}{\sigma + \sum_{n \in \mathcal{N}_p} p_n}. \tag{3}$$

The pool then charges a fee for α percentage from each stakeholder's reward and a cost of $c\mathrm{e}^{-p_i}$ before the reward is finally sent to each stakeholder. Since the cost decreases exponentially as the stakes increase, it encourages the stakeholders to invest more stakes to the pool. Thus, when a stakeholder i invests p_i stakes to the pool, the stakeholder's expected reward r_i^p is given by

$$\begin{aligned} r_i^p &= P^w P_i^p (1-\alpha) R - c\mathrm{e}^{-p_i}, \\ &= \frac{p_i}{\sigma + \sum_{n \in \mathcal{N}_p} p_n + \sum_{j=1}^{N} m_j}(1-\alpha)R - c\mathrm{e}^{-p_i}. \end{aligned} \tag{4}$$

In the case if the stakeholder i uses m_i stakes to self-mine, its expected reward is

$$r_i^m = \Big(\frac{m_i}{\sigma + \sum_{n \in \mathcal{N}_p} p_n + \sum_{j=1}^{N} m_j}\Big) R - C_i, \tag{5}$$

where $\dfrac{m_i}{\sigma + \sum_{n \in \mathcal{N}_p} p_n + \sum_{j=1}^{N} m_j}$ represents the proportion of stakeholder i's stakes in the total network stakes. Then, the profit of the pool can be calculated as follows:

$$\begin{aligned} U_p = &\frac{\sigma}{\sigma + \sum_{n \in \mathcal{N}_p} p_n + \sum_{j=1}^{N} m_j} R \\ &+ \sum_{i \in \mathcal{N}_p} \Big(\frac{p_i \alpha}{\sigma + \sum_{n \in \mathcal{N}_p} p_n + \sum_{j=1}^{N} m_j} R + c\mathrm{e}^{-p_i}\Big). \end{aligned} \tag{6}$$

The total profit of the pool consists of the profits from its own stakes, i.e., the first term in (6), and the costs and fees it charges the stakeholders, i.e., the second term in (6).

4.2 Stackelberg Game Formulation

In practice, a pool usually announces its cost and fee first, e.g., the fee to join the Stakecube pool can be found on its website [31]. Based on that information, the stakeholders will decide how much to invest. As a result, the interaction between the stake pool and stakeholders can be formulated to be a single-leader-multiple-followers Stackelberg game [33]. In this game, the leader is the stake pool who first announces its strategy, i.e., costs and fees to join the pool, and then the stakeholders, i.e., followers, will make their decisions, e.g., to invest to the pool or not.

We denote s_p and s_i to be the strategies of the leader and follower i, respectively. Furthermore, we denote $\mathcal{S}_i$ to be the set of all possible strategies of follower i. Then, the best response s_i^* of a follower i can be defined to be the strategy set which gives the follower the best payoff given a fixed strategy $s_p = (\alpha, c)$ of the leader, i.e.,

$$U_i(s_i^*, s_p) \geq U_i(s_i', s_p), \forall s_i' \in \mathcal{S}_i. \tag{7}$$

Based on the follower's best response, the Stackelberg strategy for the leader is a strategy s_p^* such that

$$s_p^* = \underset{s_p}{\operatorname{argmax}}\, U_p(s_p, s_i^*). \tag{8}$$

Then, the Stackelberg solution can be defined as the tuple (s_p^*, s_i^*), and its corresponding utility tuple (U_p^*, U_i^*) is the Stackelberg equilibrium of the game. To find the Stackelberg equilibrium, the game can be divided into two stages. At the first stage, the leader announces its strategy. Then, at the second stage, the followers determine their strategies based on the leader's strategy. In the following, the backward-induction-based analysis is carried out to examine the Stackelberg equilibrium of this game.

4.2.1 Follower strategy

In this game, a follower's possible strategies can be divided into four cases:

- *Case 1:* Only invest stakes to the pool.
- *Case 2:* Only invest stakes for self-mining.
- *Case 3:* Simultaneously invest stakes to the pool and for self-mining.
- *Case 4:* Do not invest stakes to the PoS-based blockchain network.

We prove in the following Theorem that a follower's best response is use all its stakes either to invest to the pool or for self-mining.

Theorem 3. *A stakeholder's best response is to invest all stakes either to invest to the pool or for self-mining.*

Proof: See Appendix C. □

Since the a stakeholder's best response is to invest all its stakes, the best response can be deduced from either p_i^* or m_i^*. Therefore, from now on, we can denote the best response of follower i by the number of stakes it invest to the pool p_i^*. Then, the best response p_i^* of follower i can be expressed as a function of the pool's cost and fee as follows

$$p_i^*(\alpha, c) = \begin{cases} 0 & \text{if } C_i < \dfrac{B_i \alpha R}{\sigma + \sum_{j=1}^N B_j} + c\mathrm{e}^{-B_i}, \\ B_i & \text{if } C_i \geq \dfrac{B_i \alpha R}{\sigma + \sum_{j=1}^N B_j} + c\mathrm{e}^{-B_i}. \end{cases} \tag{9}$$

Theorem 4. *Given a strategy of the leader, there exists an optimal strategy for every follower and this strategy is unique.*

Proof: From (9), it can be seen that for every fixed strategy of the leader, a unique best response of every follower can be straightforwardly determined. □

4.2.2 Leader strategy

The backward induction mechanism [33] can be used to find the best strategy of the leader, which is the strategy that yields the highest payoff given the best responses of all followers, i.e., we have

$$\begin{aligned} s_p^* = \underset{s_p=(c,\alpha)}{\operatorname{argmax}}\, U_p(s_p, p_i^*) = \frac{\sigma}{\sigma + \sum_{j=1}^N B_j} R + \\ \sum_{i \in \mathcal{N}_p} \left(\frac{p_i^* \alpha}{\sigma + \sum_{j=1}^N B_j} R + c\mathrm{e}^{-B_i} \right). \end{aligned} \tag{10}$$

Since the total network stakes can be considered a constant, the profit from the pool owner's stake is also a constant (the first term in (10)) and does not need to be optimized. Moreover, since $p_i^*(\alpha, c)$ can only take two values, i.e., 0 or B_i, it can be represented by a binary decision variable $x_i \in \mathbf{x} = \{x_1, \ldots, x_N\}$, such that when $x_i = 1$, $p_i^* = B_i$ and when $x_i = 0$, $p_i^* = 0$. This helps to transform the optimization problem (10) into a Mixed-Integer Programming (MIP) optimization as follows:

$$\begin{aligned} \max_{\alpha, c, \mathbf{x}} \quad & \sum_{i=1}^N x_i \left(\frac{B_i R \alpha}{\sigma + \sum_{j=1}^N B_j} + c\mathrm{e}^{-B_i} \right), \\ \text{s.t.} \quad & \frac{B_i R \alpha}{\sigma + \sum_{j=1}^N B_j} + c\mathrm{e}^{-B_i} \leq L(1 - x_i) + C_i \quad \forall i \in \mathcal{N}, \\ & x_i \in \{0, 1\} \qquad \forall i \in \mathcal{N}, \end{aligned} \tag{11}$$

where L is a sufficiently large number. The goal of (11) is to find the optimal values of $(\alpha, c, \mathbf{x})$ to maximize the pool's profit. The objective function represents the profit of the pool, where the stake pool can only charge the stakeholders who have invested in the pool. The first set of constraints ensures that only when the pool charges follower i less than C_i, x_i can take the value of 1, and thus the profit can be added to the total profit of the pool. The second set of constraints ensures that every x_i is a binary number. However, the objective function is nonlinear, i.e., it contains a multiplication of two decision variables x_i and α, which makes it much more complex to solve [35]. Thus, we transform (11) into an equivalent Mixed-Integer Linear Programming (MILP) model as follows:

$$\begin{aligned} \max_{\alpha, c, \mathbf{x}, \mathbf{y}} \quad & \sum_{i=1}^N y_i, \\ \text{s.t.} \quad & \frac{B_i R \alpha}{\sum_{j=1}^N B_j} + c\mathrm{e}^{-B_i} \leq L(1 - x_i) + C_i \quad \forall i \in \mathcal{N}, \\ & y_i - L x_i \leq 0 \qquad \forall i \in \mathcal{N}, \\ & y_i - L(1 - x_i) \leq \frac{B_i R \alpha}{\sum_{j=1}^N B_j} + c\mathrm{e}^{-B_i} \quad \forall i \in \mathcal{N}, \\ & x_i \in \{0, 1\} \qquad \forall i \in \mathcal{N}, \\ & y_i \in \mathbb{R}^+ \qquad \forall i \in \mathcal{N}. \end{aligned} \tag{12}$$

The transformation from (11) to (12) is done by a standard transformation technique which ensures the equivalence of the two models [34]. In particular, we introduce a new set of continuous variables $\mathbf{y} = \{y_1, \ldots, y_N\}$ which represents the profit which the pool can yield from follower i. Two new sets of auxiliary constraints, i.e., the second and third sets of constraints, are added to set the upper bound for y_i. If $x_i = 0$, i.e., follower i does not invest stakes to the pool, y_i will be upper-bounded by 0. If $x_i = 1$, y_i will be upper-bounded by $\frac{B_i R \alpha}{\sum_{j=1}^N B_j} + c\mathrm{e}^{-B_i}$. Thus, the optimal solution of (12) consists of two optimal values of α and c as shown in (10).

4.2.3 Existence of the Stackelberg equilibrium

The existence of the Stackelberg equilibrium is proven via the existence of the optimal solutions of (12) in the following

Theorem.

Theorem 5. *There exists at least one Stackelberg equilibrium in the considered stake pool game.*

Proof: See Appendix D. □

4.2.4 Uniqueness of the Stackelberg equilibrium

Although there always exists at least one Stackelberg equilibrium in this game, the uniqueness of the equilibrium cannot be guaranteed because both α and c are continuous variables. Consequently, there may be multiple pairs of α and c to achieve the same optimal utility as will be shown later in Section 5. In the conventional Stackelberg game model, the leader has only one primary priority, that is, to maximize the profit. Therefore, we propose a secondary priority for the leader, which is to minimize α. This serves two purposes, i.e., to attract followers with high stakes (as the amount the pool charges via the fee is proportional to the stakes) and to determine the unique optimal strategy for the game (i.e., the unique optimal strategy for both the leader and followers). Under the proposed approach, we can always obtain the unique Stackelberg equilibrium as proven in Theorem 6.

Theorem 6. *The considered stake pool game admits a unique Stackelberg equilibrium.*

Proof: See Appendix E. □

Based on this unique Stackelberg equilibrium, the stake pool can design appropriate parameters, i.e., cost and fee, to maximize its profits and attract more stakeholders to invest in the pool, and at the same time, the stakeholders can determine their best investment strategies to maximize their profits.

5 Performance Evaluation

5.1 Parameter Settings

We first study three small game instances, i.e., $\mathcal{G}_1$ to $\mathcal{G}_3$, to clearly show the relation between the leader and the followers in different situations. In these instances, we examine the utility functions of the stake pool and stakeholders. Particularly, we present their corresponding utilities over a range of fees and costs, thereby demonstrating the effects of the stake pool strategy on the profit of the stakeholders and the stake pool. In $\mathcal{G}_1$, we consider a small game consisting one stakeholder and one stake pool with $C_1 = 0.1$, $b_1 = 5$, $R = 10$, and $\sigma = 10$. Then, we extend this game to $\mathcal{G}_2$ by considering five followers with the same configurations as that of the follower in $\mathcal{G}_1$, while other parameters are unchanged. After that, we consider game $\mathcal{G}_3$. Parameters are similar as those of $\mathcal{G}_2$ except that the followers have different budgets $\mathbf{B} = (5, 10, 13, 6, 8)$, operational costs $\mathbf{C} = (0.1, 0.3, 0.2, 0.6, 0.5)$, and $R = 50$.

To evaluate more general cases, we simulate 13 instances $\mathcal{G}_4$ to $\mathcal{G}_{16}$, each with 1,000 followers and different parameters as shown in Table 3. Among them, the first five games $\mathcal{G}_4$ to $\mathcal{G}_8$ are simulated with network parameters, such as R, $\mathbf{C}$, and $\mathbf{B}$, generated based on several real-world PoS-based blockchain networks [36]–[40]. Particularly, the values of R is determined using the number of coins these networks pay out (as block reward) per one block. For the values of $\mathbf{C}$, we first calculate the reference value C_r as follow:

$$C_r = \frac{100}{V_R N_b}, \tag{13}$$

where 100 is the average cost per month (in $) to participate in the consensus process, V_R is the monetary value of each coin, and N_b is the number of blocks produced per month. As a result, C_r represents on average how many coins it costs to participate in the consensus process for one block. Then, the ranges of $\mathbf{C}$ can be determined based on C_r. For $\mathbf{B}$, we estimate the ranges by dividing the total number of coins in circulation and the total number of stakeholders in the network. Then, B_n and C_n of each stakeholder are generated randomly with normal distribution in the ranges listed in Table 3. The eight instances $\mathcal{G}_9$ to $\mathcal{G}_{16}$ are simulated to study the impacts of important parameters, i.e., R, $\mathbf{B}$, $\mathbf{C}$, and σ, on the game outcome. Taking $\mathcal{G}_4$ as a reference, we vary a single parameter at a time to evaluate the impacts of each parameter. For example, to study the impacts of R, we decrease R ten times (compared to $\mathcal{G}_4$) in $\mathcal{G}_9$ and increase R ten times in $\mathcal{G}_{10}$, while all the other parameters are kept the same. The results, including the optimal leader strategy, optimal profit, and percentage of the network stakes invested in the pool, are obtained by solving the MILP optimization (12).

To evaluate the effects of the economic model on the network's security and performance, we simulate six game instances, $\mathcal{G}_{17}$ to $\mathcal{G}_{22}$. In instance $\mathcal{G}_{17}$, we simulate the network with a stake pool, similar to the previous game instances. In contrast, we simulate the network without a stake pool in instance $\mathcal{G}_{18}$. Since there is no stake pool, each stakeholder in this instance only has two choices, i.e., to participate in the consensus process if its operational cost is less than its profit ($C_i < \frac{B_i R}{\sum_{n=1}^{N} B_i}$), or does not participate in the consensus process if its operational cost is higher than its profits. Then, we examine the cases where there is an adversary who tries to attack the network with the same adversarial budget $B_{\mathcal{A}}$ in both instances. Under such adversarial attacks, we compare the security and performance of the network (with and without the stake pool) in terms of common prefix violation probability and transaction confirmation time. The common prefix violation probability is calculated using (14). Based on this, we find the minimum value of κ such that $\Pr_{\text{CP}} < 0.1\%$ and multiply it with a block time of 20 seconds to determine the transaction confirmation time. For $\mathcal{G}_{17}$ and $\mathcal{G}_{18}$, we simulate a weak adversary with $B_{\mathcal{A}} = 20,000$ tokens. Similarly, we simulate a medium adversary with $B_{\mathcal{A}} = 40,000$ tokens for $\mathcal{G}_{20}$ and $\mathcal{G}_{21}$ and a strong adversary with $B_{\mathcal{A}} = 60,000$ tokens for $\mathcal{G}_{21}$ and $\mathcal{G}_{22}$. The other parameters of $\mathcal{G}_{17}$ to $\mathcal{G}_{22}$ are the same as those of $\mathcal{G}_4$.

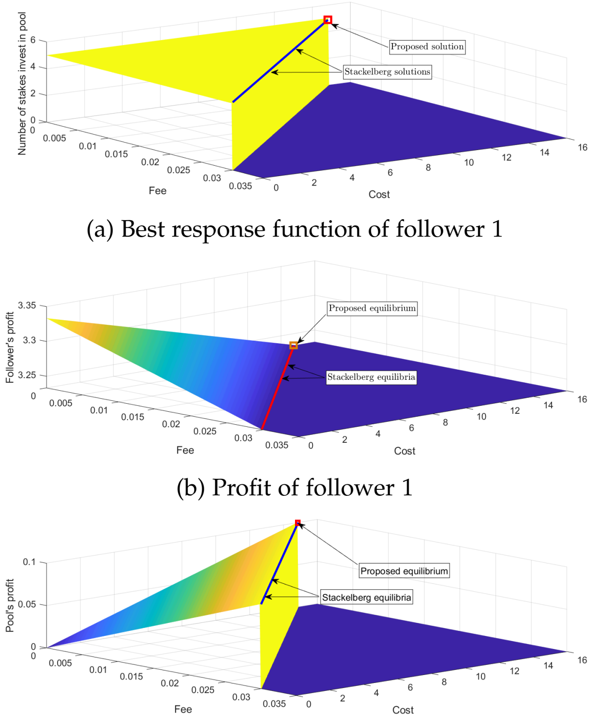
5.2 Numerical Results

5.2.1 Leader and Follower's Utilities

The best response function of follower 1 in $\mathcal{G}_1$ is illustrated in Fig. 4(a). Based on its best response, the profit of follower 1 can be determined. In this game, the profit of the follower decreases as the pool's fee and cost increase as shown in

TABLE 3: Parameters and results of 13 simulation instances.

$\mathcal{G}$	Parameters					Stackelberg equilibrium			
	R	**B** range	**C** range	σ	Based on	c^*	$\alpha^*(\%)$	U_p^*	% stake of the pool
4	1000	[1,250]	[0.05,0.1]	1000	Cardano [36]	3.2	4.0	28.95	69.5
5	200	[1,1000]	[0.0001,0.15]	1000	Algorand [37]	0.06	1.6	1.81	56.6
6	3.81	[1,400]	[0.0001,0.002]	1000	Cosmos [38]	0.1	14.4	0.35	61.2
7	78	[80,160]	[0.0001,0.02]	1000	Tezos [39]	40.1	6.1	2.29	48.9
8	500	[1,5000]	[0.001,0.3]	1000	NEM [40]	0.003	13.01	40.92	62.9
9	**100**	[1,250]	[0.05,0.1]	1000	Cardano	0.003	40.4	28.08	69.5
10	**10000**	[1,250]	[0.05,0.1]	1000	Cardano	0.207	0.4	29.13	69.5
11	1000	[1,250]	**[0.01,0.02]**	1000	Cardano	0.04	0.8	5.82	69.5
12	1000	[1,250]	**[0.25,0.5]**	1000	Cardano	0.04	20.5	140.54	69.5
13	1000	**[1,25]**	[0.05,0.1]	1000	Cardano	0.2	4.7	36.51	72.1
14	1000	**[1,2500]**	[0.05,0.1]	1000	Cardano	356.1	4.0	28.21	70.1
15	1000	[1,250]	[0.05,0.1]	**1**	Cardano	0.04	4.0	28.31	69.5
16	1000	[1,250]	[0.05,0.1]	**100000**	Cardano	0.02	10.9	28.15	69.5



(a) Best response function of follower 1

(b) Profit of follower 1

(c) Pool's profit

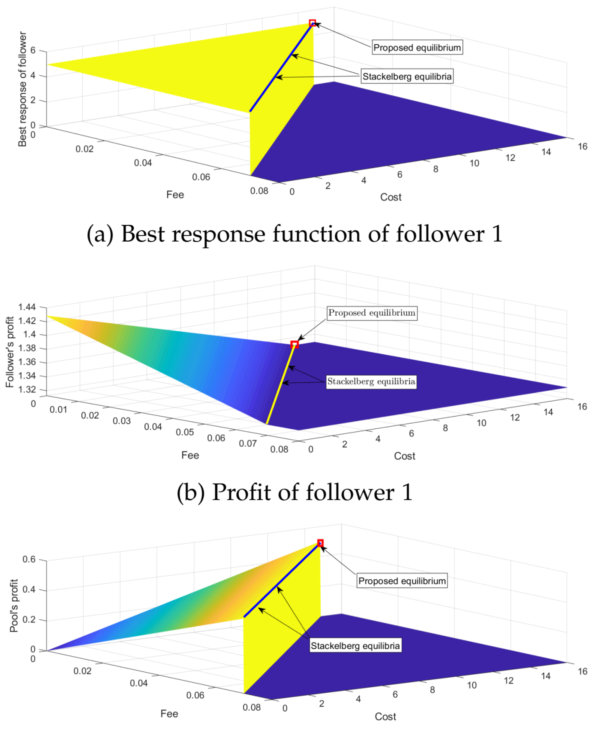
Fig. 4: Profit and best response of the leader and follower in $\mathcal{G}_1$.

Fig. 4(b), but it is still higher than self-mining. The profit of the pool is illustrated in Fig. 4(c). Since there is only one follower in $\mathcal{G}_1$, the profit of the pool only comes from follower 1, and thus it is upper-bounded by C_1. In this game, any pair of (c, α) that satisfies $\frac{\alpha R B_i}{\sigma + C_i} + c e^{-B_i} = C_i = \frac{50}{15}\alpha + 0.007c = 0.1$ is a Stackelberg solution, which leads to multiple Stackelberg equilibria. Nevertheless, under our proposed approach, we can find the unique Stackelberg equilibrium for this game at $(c^*, \alpha^*) = (14.8, 0)$.

In $\mathcal{G}_2$, since the followers have the same budgets and operational costs, their best response and profit functions are the same, which are illustrated in Fig. 5(a) and Fig. 5(b), respectively. These functions are similar to that of $\mathcal{G}_1$, except that the fee threshold is higher (7%). This is because there are more followers in $\mathcal{G}_2$, and thus (c, α) must satisfy $\frac{\alpha R B_i}{\sigma + C_i} + c e^{-B_i} = C_i = \frac{50}{35}\alpha + 0.007c = 0.1$. The pool's profit in $\mathcal{G}_2$ is illustrated in Fig. 5(c), which is upper-bounded by $5C_i$ in this game. The unique proposed equilibrium of this game has a corresponding solution $(c^*, \alpha^*) = (14.8, 0)$ as shown in Fig. 5(c).

In $\mathcal{G}_3$, each follower's best response is illustrated in 6(a). Typically, the higher a follower's budget is, the higher cost and the lower fee that follower is willing to accept, and vice versa. For example, follower 3 with the highest budget only accepts a fee of no more than 1.6%, and follower 1 with the lowest budget only accepts a cost lower than 15. This is because the budget is proportional to the fee the pool charges, while the cost decreases exponentially as the budget increases. The pool's profit in $\mathcal{G}_3$ is illustrated in Fig. 6(b), with the leader's optimal strategy $(c^*, \alpha^*) = (171.3, 3.0\%)$ and optimal profit $U_p^* = 1.19$. Fig. 6(c) illustrates the profit the pool receives from each follower. Interestingly, at the obtained Stackelberg equilibrium of $\mathcal{G}_3$, the follower with the highest stake, i.e, follower 3, does not invest to the pool. The reason is that follower 3 has a relatively low operational cost, and thus the follower is more inclined to mine if the pool's cost and fee are too high. If the pool tries to incentivize all followers to invest by reducing α and c, its profit is only $U_p = 0.68$.

The results of more general cases are shown in Table 3. The five instances $\mathcal{G}_4$ to $\mathcal{G}_8$ are simulated with parameters adopted from several real-world blockchain networks [36]–[40]. The results show that the leader's optimal strategy and profit are significantly influenced by the network's parameters. For example, we obtain the optimal solution of $\mathcal{G}_4$ where $(c^*, \alpha^*) = (3.2, 4.0\%)$, $U_p^* = 28.95$, and approximately 69.5% of the total network's stakes (including σ) are invested to the pool. The profit that the pool earns from each

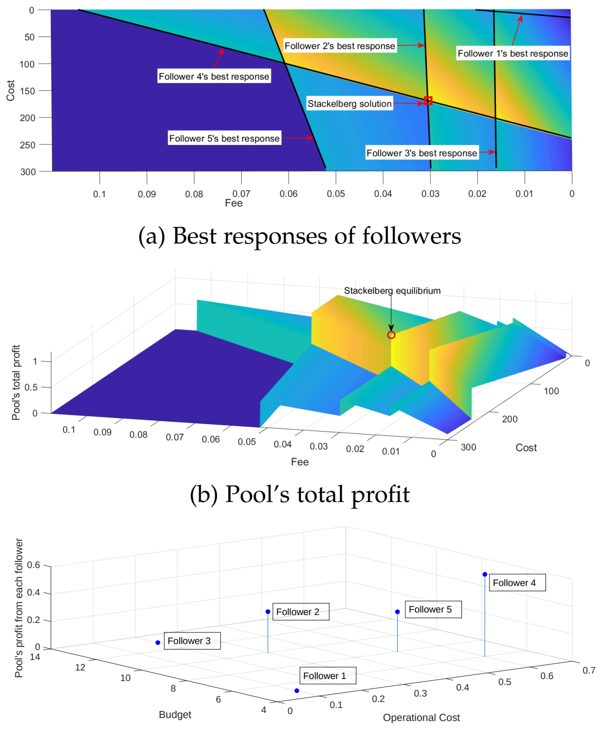


(a) Best response function of follower 1

(b) Profit of follower 1

(c) Pool's profit

Fig. 5: Profit and best response of the leader and follower in $\mathcal{G}_2$.



(a) Best responses of followers

(b) Pool's total profit

(c) Pool's profit from each follower

Fig. 6: Profit and best response of the leader and followers in $\mathcal{G}_3$.

follower depends on each follower's budget and operational cost, as shown in Fig. 7. Typically, a follower with higher cost and budget can give the pool more profit. However, similar to $\mathcal{G}_3$, if the budget is too high, the follower might not want to invest stakes to the pool, e.g., the followers with budget B_i greater than 150 do not join the pool in $\mathcal{G}_4$.

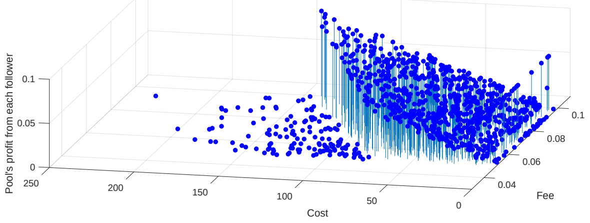


Fig. 7: Pool's profit from each follower in $\mathcal{G}_4$.

5.2.2 Impacts of Parameters

The eight games $\mathcal{G}_9$ to $\mathcal{G}_{16}$ are simulated to study the impacts of important parameters R, $\mathbf{B}$, $\mathbf{c}$, and σ, on the game's outcome. The impacts of those parameters are briefly described as follows:

- *Block reward R:* $\mathcal{G}_9$ and $\mathcal{G}_{10}$ are simulated to show the impact of R. As R increases, the pool's profit increases. However, the followers' operational costs are constant. Therefore, the pool has to decrease α when R increases, otherwise the followers will self-mine.
- *Operational costs $\mathbf{C}$:* $\mathcal{G}_{11}$ and $\mathcal{G}_{12}$ show how the followers' operational cost impacts the game's outcome. As the $\mathbf{C}$ increase, the pool can increase its profit by increasing α. The reason is that the followers' profits from self-mining are inversely proportional to the $\mathbf{C}$, and thus self-mining becomes less profitable if $\mathbf{C}$ are too high.
- *Budgets $\mathbf{B}$:* $\mathcal{G}_{13}$ and $\mathcal{G}_{14}$ show that as the budgets of followers increase, the pool can increase c but it has to reduce α. This is because the profit the pool receives via α is proportional to $\mathbf{B}$, while the profit the pool gets from c decreases exponentially as $\mathbf{B}$ increase. Moreover, as $\mathbf{B}$ increase, the stakeholders invest fewer stakes to the pool and consequently the pool's profit decreases. The reason is that when $\mathbf{B}$ increase, the profit from self-mining also increases, and thus the followers prefer to self-mine.
- *The pool owner's stake σ:* The last two games show that as σ increases, although there are more stakes invested in the pool, its profit slightly decreases. The reason is that σ is inversely proportional to the pool's profit from each follower, and thus increasing σ means that the pool charges less from each follower. Consequently, the pool's profit decreases even though more followers invest to the pool.

5.2.3 Network Security and Performance

Fig. 8 illustrates the common prefix violation probability in instances $\mathcal{G}_{17}$ to $\mathcal{G}_{22}$. As observed from the figure, the instances with a stake pool achieve a lower common prefix

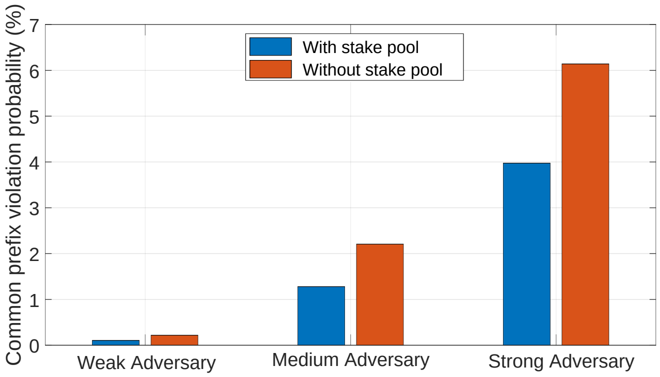


Fig. 8: Common prefix violation probability under different adversarial power.

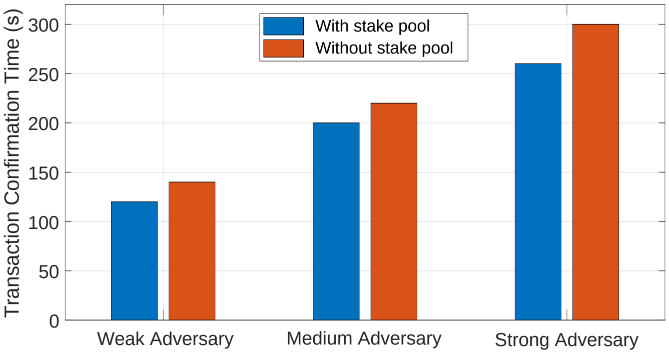


Fig. 9: Transaction confirmation time under different adversarial power.

violation probability compared to the instances without a stake pool. For example, for the medium adversary setting, the network achieves a 1.28 % violation probability, whereas the probability is 2.20 % if there is no stake pool. This is because if there is no stake pool, the stakeholders with small budgets may have negative utility if they participate in the consensus process (if their operational costs are higher than the reward they can obtain). Thus, the stakeholders holding few stakes may not participate in the consensus process, resulting in lower total network stakes. Consequently, the adversarial ratio can be increased, and the adversary may have higher chances to successfully attack the network.

Fig. 9 illustrates the transaction confirmation time in instances $\mathcal{G}_{17}$ to $\mathcal{G}_{22}$. Similar to the common prefix violation probability, the transaction confirmation time of the instances with a stake pool are lower than those of the instances without a stake pool. The reason is that, when the common prefix violation probability is higher than 0.1%, the stakeholders have to wait for more blocks (higher κ) to confirm a transaction. Since the common prefix violation probabilities are higher in the instances without stake pool as discussed above, the transaction confirmation time is also higher in these cases. For example, in $\mathcal{G}_{22}$, the stakeholders have to wait for 15 blocks to confirm a transaction, whereas they have to wait for 13 blocks in $\mathcal{G}_{21}$, and thus the transaction confirmation time is lower in $\mathcal{G}_{21}$

5.3 Summary of Findings

The key findings of the considered stake pool game are summarized as follows:

- We have proved that for a rational stakeholder, its best strategy is to invest all stakes from its budget to the blockchain network.
- We have proved that for each stakeholder, its best strategy is to invest all its stakes either to the pool or for self-mining.
- We have proposed an approach for the leader to decide its optimal strategy. Under this approach, there always exists the optimal and unique best strategies for the stakeholders and the stake pool owner. This approach also helps the stake pool to attract stakeholders with high stakes.
- We have shown that the proposed economic model can enhance the network's security and performance.

6 Conclusion

To address the problem of roaming fraud for mobile service providers, we have proposed BlockRoam, a novel blockchain-based roaming management system which consists of our thoroughly analyzed PoS consensus mechanism and a smart-contract-enabled roaming management platform. Moreover, we have analyzed and showed that BlockRoam's security and performance can be enhanced by incentivizing more users to participate in the network. Therefore, we have developed an economic model based on Stackelberg game to jointly maximize the profits of network users, thereby incentivizing their participation. We have analyzed and determined the best strategies for the stakeholders and the stake pool. We have also proposed an effective solution that results in a unique equilibrium for our economic model. Lastly, we have evaluated the impacts of important parameters on the strategies and the equilibrium of the game. The proposed economic model can help the mobile service providers to earn additional profits, attract more investment to the blockchain network, and enhance the network's security and performance.

References

[1] GSMA Intelligence, "The Mobile Economy 2019," GSM Association, 2019. [Online]. Available: https://www.gsmaintelligence.com/research/?file=b9a6e6202ee1d5f787cfebb95d3639c5&download. [Accessed: 16-Aug-2019]

[2] L. Papachristou, "Report: US$32.7 Billion Lost in Telecom Fraud Annually," *Organized Crime and Corruption Reporting Project*. [Online]. Available: https://www.occrp.org/en/27-ccwatch/cc-watch-briefs/9436-report-us-32-7-billion-lost-in-telecom-fraud-annually. [Accessed: 16-Aug-2019].

[3] G. Macia-Fernandez, P. Garcia-Teodoro, and J. Diaz-Verdejo, "Fraud in roaming scenarios: an overview," in *IEEE Wireless Communications*, vol. 16, no. 6, pp. 88–94, Dec. 2009.

[4] Starhome Mach, "Starhome Mach: Operator's Roaming Fraud Losses Can Reach €40,000 Per Hour," *PR Newswire: press release distribution, targeting, monitoring and marketing*, 29-Jun-2018. [Online]. Available: https://www.prnewswire.com/news-releases/starhome-mach-operators-roaming-fraud-losses-can-reach-40000-per-hour-598836021.html. [Accessed: 16-Sep-2019].

[5] C. T. Nguyen, D. T. Hoang, D. N. Nguyen, D. Niyato, H. T. Nguyen, and E. Dutkiewicz, “Proof-of-Stake Consensus Mechanisms for Future Blockchain Networks: Fundamentals, Applications and Opportunities,” in *IEEE Access*, vol. 7, pp. 85727–85745, Jun. 2019.

[6] R. Henry, A. Herzberg and A. Kate, “Blockchain access privacy: Challenges and directions,” *IEEE Security & Privacy*, vol. 16, no. 4, pp. 38-45, Aug. 2018.

[7] Q. Feng, D. He, S. Zeadally, M. K. Khan and N. Kumar, “A survey on privacy protection in blockchain system,” *Journal of Network and Computer Applications*, vol. 126, pp. 45-58, Jan. 2019.

[8] 3GPP, “3GPP TS 22.031 V15.0.0,” Technical Specification 22.031, Jun-2018.

[9] GSMA, “GSMA Speeds Up The Transfer Of Roaming Call Records,” *Newsroom*, 21-Mar-2012. [Online]. Available: https://www.gsma.com/newsroom/press-release/gsma-speeds-up-the-transfer-of-roaming-call-records/. [Accessed: 13-Nov-2019].

[10] IBM, “Reimagining telecommunications with blockchains,” *IBM Institute for Business Value*. [Online]. Available: https://www.ibm.com/thought-leadership/institute-business-value/report/blockchaintelco. [Accessed: 16-Aug-2019].

[11] Deutsche Telekom AG, “Deutsche Telekom and SK Telecom pave the way for the future,” *Deutsche Telekom*, 26-Feb-2019. [Online]. Available: https://www.telekom.com/en/media/media-information/archive/deutsche-telekom-and-sk-telecom-pave-the-way-for-the-future-564180. [Accessed: 16-Aug-2019].

[12] M. Boddy, “EEA Publishes Blockchain Uses for T-Mobile and Other Major Telecoms,” *Cointelegraph*, 30-Aug-2019. [Online]. Available: https://cointelegraph.com/news/enterprise-ethereum-alliance-publishes-on-blockchain-uses-in-telecoms. [Accessed: 20-Sep-2019].

[13] S. Nakamoto. (May 2008). *Bitcoin: A Peer-to-Peer Electronic Cash System*. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[14] “Bitcoin Energy Consumption Index,” *Digiconomist*. [Online]. Available: https://digiconomist.net/bitcoin-energy-consumption. [Accessed: 13-Nov-2019].

[15] A. Kulichevskiy. (03-Oct-2017). *Bubbletone blockchain white paper*. [Online]. Available: https://icos.icobox.io/uploads/whitepaper/2017/10/59e8dcfa89537.pdf [Accessed: 16-Aug-2019].

[16] L. Luu, D. Chu, H. Olickel, P. Saxena, and A. Hobor, “Making Smart Contracts Smarter,” in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security - CCS16*, Vienna, Austria, Oct. 2016, pp. 254-269.

[17] W. Wang et al., “A Survey on Consensus Mechanisms and Mining Strategy Management in Blockchain Networks,” in *IEEE Access*, vol. 7, pp. 22328–22370, Jan. 2019.

[18] B. Schoenmakers, “A simple publicly verifiable secret sharing scheme and its application to electronic voting,” *Annual International Cryptology Conference*, Santa Barbara, California, USA, Aug. 15-19, 1999, pp. 148-164.

[19] A. Kiayias, A. Russell, B. David, and R. Oliynykov, “Ouroboros: A Provably Secure Proof-of-Stake Blockchain Protocol,” in *Proc. 37th Annu. Int. Cryptolog. Conf. (CRYPTO)*, Santa Barbara, CA, USA, Aug. 2017, pp. 357–388.

[20] V. Buterin and V. Griffith, “Casper the friendly finality gadget,” 2017, *arXiv preprint arXiv:1710.09437*. [Online]. Available: https://arxiv.org/abs/1710.09437

[21] E. Buchman, J. Kwon, and Z. Milosevic (Sep 2018) *The latest gossip on BFT consensus*. [Online]. Available: https://tendermint.com/static/docs/tendermint.pdf

[22] I. Bentov, A. Gabizon, and A. Mizrahi, “Cryptocurrencies without proof of work,” in *International Conference on Financial Cryptography and Data Security*. Barbados, Feb. 2016, pp. 142–157.

[23] I. Bentov, C. Lee, A. Mizrahi, and M. Rosenfeld, “Proof of activity: Extending bitcoin’s proof of work via proof of stake (extended abstract),” *ACM SIGMETRICS Performance Evaluation Review*, vol. 42, no. 3, pp. 34–37, Dec. 2014.

[24] Y. Gilad, R. Hemo, S. Micali, G. Vlachos, and N. Zeldovich, “Algorand: Scaling byzantine agreements for cryptocurrencies,” in *Proceedings of the 26th Symposium on Operating Systems Principles*, Oct. 2017, pp. 51–68.

[25] Y. Xiao, N. Zhang, W. Lou and Y. T. Hou, “A Survey of Distributed Consensus Protocols for Blockchain Networks,” 2020, *arXiv preprint arXiv:1904.04098*. [Online]. Available: https://arxiv.org/abs/1904.04098

[26] J. Garay, A. Kiayias, and N. Leonardos, “The Bitcoin Backbone Protocol: Analysis and Applications,” in *Advances in Cryptology - EUROCRYPT 2015 Lecture Notes in Computer Science*, vol. 9057. E. Oswald, M. Fischlin, Eds. Berlin: Springer, 2015, pp. 281–310.

[27] M. Mitzenmacher and E. Upfal, *Probability and computing: Randomization and probabilistic techniques in algorithms and data analysis*. Cambridge: Cambridge University Press, 2017.

[28] “Cardano Blockchain Explorer,” *Cardano Blockchain Explorer*. [Online]. Available: https://cardanoexplorer.com/. [Accessed: 05-Dec-2019].

[29] Jotunn, “How Many Stake Pools?,” *Cardano Forum*, 21-Sep-2018. [Online]. Available: https://forum.cardano.org/t/how-many-stake-pools/16132/12. [Accessed: 25-Sep-2019].

[30] “Ultrapool,” *Decred Voting Service - Welcome*. [Online]. Available: https://ultrapool.eu/. [Accessed: 16-Aug-2019].

[31] “Stakecube,” *Crypto Shib*. [Online]. Available: https://cryptoshib.com/stakecube/. [Accessed: 16-Aug-2019].

[32] “Earn profits by holdings cryptoassets,” *MyCointainer*. [Online]. Available: https://www.mycointainer.com/. [Accessed: 16-Aug-2019].

[33] Z. Han, D. Niyato, W. Saad, T. Başar, and A. Hjørungnes, *Game theory in wireless and communication networks: theory, models, and applications*. Cambridge Univ. Press, 2012.

[34] F. Glover, “Improved Linear Integer Programming Formulations of Nonlinear Integer Problems,” in *Management Science*, vol. 22, no. 4, pp. 455–460, Dec. 1975.

[35] M. X. Goemans, *Advanced algorithms*. Massachusetts Institute of Technology. Laboratory for Computer Science, 1994.

[36] StakingRewards, “Cardano,” *Digital Asset Research Platform for Staking & Dividends*. [Online]. Available: https://stakingrewards.com/asset/ada. [Accessed: 16-Aug-2019].

[37] StakingRewards, “Algorand,” *Digital Asset Research Platform for Staking & Dividends*. [Online]. Available: https://stakingrewards.com/asset/algo. [Accessed: 16-Aug-2019].

[38] StakingRewards, “Cosmos,” *Digital Asset Research Platform for Staking & Dividends*. [Online]. Available: https://stakingrewards.com/asset/atom. [Accessed: 16-Aug-2019].

[39] StakingRewards, “Tezos,” *Digital Asset Research Platform for Staking & Dividends*. [Online]. Available: https://stakingrewards.com/asset/xtz. [Accessed: 16-Aug-2019].

[40] StakingRewards, “NEM,” *Digital Asset Research Platform for Staking & Dividends*. [Online]. Available: https://stakingrewards.com/asset/xem. [Accessed: 16-Aug-2019].

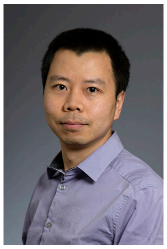
Cong T. Nguyen received his B.E. degree in Electrical Engineering and Information from the Frankfurt University of Applied Sciences in 2014, his M.Sc. degree in Global Production Engineering and Management from the Technical University of Berlin in 2016. Since 2019, he has been a Ph.D. student at the UTS-HCMUT Joint Technology and Innovation Research Centre between Ho Chi Minh University of Technology and the University of Technology Sydney (UTS). His research areas include operations research, blockchain technology, game theory and optimizations.

Diep N. Nguyen (M'13–SM'19) received the M.E. degree in electrical and computer engineering from the University of California at San Diego (UCSD) and the Ph.D. degree in electrical and computer engineering from The University of Arizona (UA). He was a DECRA Research Fellow with Macquarie University and a Member of Technical Staff with Broadcom, CA, USA, ARCON Corporation, Boston, consulting the Federal Administration of Aviation, on turning detection of UAVs and aircraft, and the U.S. Air Force Research Laboratory, on anti-jamming. He is currently a Faculty Member with the Faculty of Engineering and Information Technology, University of Technology Sydney (UTS). His recent research interests include computer networking, wireless communications, and machine learning application, with emphasis on systems' performance and security/privacy. He has received several awards from LG Electronics, UCSD, The University of Arizona, the U.S. National Science Foundation, and the Australian Research Council.

Dinh Thai Hoang (M'16) is currently a faculty member at the School of Electrical and Data Engineering, University of Technology Sydney, Australia. He received his Ph.D. in Computer Science and Engineering from the Nanyang Technological University, Singapore, in 2016. His research interests include emerging topics in wireless communications and networking such as ambient backscatter communications, vehicular communications, cybersecurity, IoT, and 5G networks. He is an Exemplary Reviewer of IEEE Transactions on Communications in 2018 and an Exemplary Reviewer of IEEE Transactions on Wireless Communications in 2017 and 2018. Currently, he is an Editor of IEEE Wireless Communications Letters and IEEE Transactions on Cognitive Communications and Networking.

Hoang-Anh Pham received his BEng in Computer Science and Engineering in 2005 from Ho Chi Minh City University of Technology, (HCMUT in short), VNU-HCM, Vietnam. From 2005 to 2008, he was a faculty member at the Faculty of Computer Science and Engineering, HCMUT. In 2010 and 2014, he received his MSc and PhD in Information and Communications Engineering from MYONGJI University, South Korea, respectively. He is currently a senior lecturer at Faculty of Computer Science and Engineering, HCMUT. He has served as the Director of Internet of Things Lab since 2016, and the Director of HCMUT-Renesas SuperH Lab (specializing in Embedded Systems and Robotics) since 2018. His current research interests include computer networking, data communications, cyber-physical systems, Internet of Things, and Blockchain.

Huynh Tuong Nguyen Dr. Huynh Tuong Nguyen, a faculty member at Ho Chi Minh City University, is an expert in algorithms and resolutions (simulation, modelling & optimization) for real-life problems including: manufacturing scheduling, transportation problems, education management and assessment, digital currency and cryptography. He holds a PhD in Computer Science from Fran¸cois Rabelais University and his work has appeared in Asian Journal of Computer Science and Information Technology, European Journal of Operational Research, Journal of Scheduling and Mathematical Problems in Engineering.



Yong Xiao (S'09-M'13-SM'15) is a professor in the School of Electronic Information and Communications at the Huazhong University of Science and Technology (HUST), Wuhan, China. He is also the associate group leader in the network intelligence group of IMT-2030 (6G promoting group) and the vice director of 5G Verticals Innovation Laboratory at HUST. His research interests include machine learning, game theory, and their applications in cloud/fog/mobile edge computing, green communication systems, wireless networks, and Internet-of-Things (IoT).

Eryk Dutkiewicz (M'05–SM'15) received the B.E. degree in electrical and electronic engineering and the M.Sc. degree in applied mathematics from The University of Adelaide, in 1988 and 1992, respectively, and the Ph.D. degree in telecommunications from the University of Wollongong, in 1996. His industry experience includes management of the Wireless Research Laboratory at Motorola, in 2000. He is currently the Head of the School of Electrical and Data Engineering, University of Technology Sydney, Australia. He holds a professorial appointment at Hokkaido University, Japan. His current research interests include 5G/6G and the Internet-of-Things networks.