This article has been accepted for publication in IEEE Transactions on Services Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSC.2023.3240235

FedChain: Secure Proof-of-Stake-based Framework for Federated-blockchain Systems

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

Abstract—In this paper, we propose FedChain, a novel framework for federated-blockchain systems, to enable effective transferring of tokens between different blockchain networks. Particularly, we first introduce a federated-blockchain system together with a cross-chain transfer protocol to facilitate the secure and decentralized transfer of tokens between chains. We then develop a novel PoS-based consensus mechanism for FedChain, which can satisfy strict security requirements, prevent various blockchain-specific attacks, and achieve a more desirable performance compared to those of other existing consensus mechanisms. Moreover, a Stackelberg game model is developed to examine and address the problem of centralization in the FedChain system. Furthermore, the game model can enhance the security and performance of FedChain. By analyzing interactions between the stakeholders and chain operators, we can prove the uniqueness of the Stackelberg equilibrium and find the exact formula for this equilibrium. These results are especially important for the stakeholders to determine their best investment strategies and for the chain operators to design the optimal policy to maximize their benefits and security protection for FedChain. Simulations results then clearly show that the FedChain framework can help stakeholders to maximize their profits and the chain operators to design appropriate parameters to enhance FedChain's security and performance.

Index Terms—Blockchain, Proof-of-Stake, cross-chain transfer, sidechain, multiple-blockchain, and Stackelberg game.

1 Introduction

1.1 Motivation

Over the last few years, the development of the blockchain technology has attracted massive attention. A blockchain is an append-only ledger of transactions shared among the participants in a peer-to-peer network. With the help of consensus mechanisms, once a transaction enters the blockchain, it cannot be changed without the consensus of the majority of the network. Beside data immutability, the consensus mechanism also plays a key role in ensuring that such a decentralized network can reach the consensus without a central authority, thereby avoiding the single-point-of-failure. Moreover, advanced cryptography techniques such as digital signatures and asymmetric keys [1], [2] enable blockchain users to create easily verifiable but impossible to forge proofs of authentication for assets (i.e., blockchain tokens) while enhancing the privacy of users. As a result, blockchain can enable trusted transactions among network participants even in an open and decentralized environment. With such outstanding benefits, blockchain has been implemented as the backbone of numerous applications in many areas such as finance, healthcare, ,Internet-of-Things (IoT) [1]–[3], and Federated Learning [4], [5], [41].

Despite its popularity and potential, blockchain has been facing various challenges. The rapid development of blockchain and the massive popularity of cryptocurrency have lead to the creation of a plethora of blockchain networks. For example, the number of cryptocurrency networks has increased nearly four times in just one year (from 2000 cryptocurrencies in 2019 to 7400 by the time this article is written, i.e., December 2020 [6]). These blockchain networks are currently employing diverse consensus mechanisms, which results in severe fragmentation since these networks cannot communicate with each other. However, there are many blockchain-based applications where the ability to transfer assets between different blockchains is essential. For example, in [41], a blockchain framework is developed for federated learning, where the consensus nodes need to verify the local gradient updates from the training nodes. Although the proposed framework's effectiveness is proven, the ability to communicate with different blockchains is still desirable. For example, if the number of training nodes is too high, transaction processing and gradient verification may take very long, and thus having two chains to speed up the process would be beneficial. Moreover, transfer learning [42], [43] (where the knowledge can be transferred among different federated learning models) can significantly improve the federated learning speed and accuracy. However, for single blockchain networks, users who want to exchange tokens have to rely on trusted centralized exchange platforms, e.g., Binance [7] and Kraken [8], which is against the decentralized nature of blockchain and poses serious security threats. Particularly, there have been many attacks on these exchanges, resulting in a cumulative loss of more than $1 billion [9]

- *Cong T. Nguyen, 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: cong.nguyen@student.uts.edu.au and {diep.nguyen, hoang.dinh, eryk.dutkiewicz}@uts.edu.au.*
- *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.*
- *Yong Xiao is with the School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan, China. E-mail: yongxiao@hust.edu.cn*

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: Huazhong University of Science and Technology. Downloaded on January 31,2023 at 09:41:52 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Services Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSC.2023.3240235

over the last few years. Moreover, the trade-off between performance and security in consensus mechanism designs usually leads to high delay and low processing throughput. For example, Bitcoin needs approximately 1 hour to confirm a transaction and can only process less than 7 transactions per second [2], which hinders blockchain applicability in many scenarios. This low transaction processing capability can be addressed by using sharding mechanisms [53] to split the blockchain network into multiple sub-network to improve throughput. However, these sub-networks are still controlled by the same network operator, and they also have the same type of tokens. Therefore, sharding cannot enable the transfers of assets among different blockchains. Thus, this necessitates an effective framework that not only allows the interoperability among blockchains networks, but also guarantees the security and performance of each network.

To address these problems, the sidechain technology [10] has been developed to enable the formation of the federated-blockchain system. In a federated-blockchain system, there are multiple blockchains, and users in the system can transfer their assets to any blockchain within. However, the development of the sidechain technology is still in a nascent stage, and it does not fully satisfy the security nor the performance requirements of federated-blockchain systems. Particularly, the ability to transfer assets between multiple chains may lead to centralization to a single chain, e.g., mining power centralization in Proof-of-Work (PoW) and stakes centralization in Proof-of-Stake (PoS). This poses a security threat to the other chains in the same federation [10]. Moreover, most current sidechain applications still employ the PoW mechanism which requires huge energy consumption and has very low transaction processing capabilities [1]–[3]. Therefore, a secure and effective framework, which can address both security and performance issues for cross-chain transfers, is in urgent need for the future development and practical applications of blockchain technology. For example, in coalition loyalty programs, users need to exchange their loyalty points among different programs. However, different companies in the coalition might have different blockchains to store their customers' loyalty points (in the forms of blockchain tokens). For the coalition loyalty program to operate properly, customers need to be able to exchange their points freely among different blockchain networks. Similarly, in retail payment, vendors might only accept a certain type of tokens, and thus the users need to exchange their tokens to another type. In these practical scenarios, an efficient and secure platform is needed that allows users to exchange their assets across multiple blockchains.

1.2 Related Work

Sidechain technology was first introduced in [10] as a novel method to facilitate cross-chain transfers. Particularly, sidechain technology's mechanisms, such as two-way peg and Simplified Payment Verification (SPV) proof [10], enable a set of validators to verify and confirm transactions between different blockchains. Although this work paves the way for many research works and applications, the security and performance issues of sidechain are only briefly mentioned and not well investigated [10]. After the introduction of the sidechain technology, there have been several notable real-world applications such as PoA [12], Liquid [13], and RSK [14]. However, these applications are facing several challenges. In particular, the PoA approach relies on a fixed federation of 23 validators to validate the cross-chain transactions between the Ethereum [15] and several sidechains. This results in a low decentralization level for the consensus process. Moreover, these validators' identities are publicly known, making them easier to be targeted by attackers. Similarly, the Liquid approach [13] also relies on a federation to validate cross-chain transactions. Although these validators are not publicly known, they are chosen only by the network operators, and thus Liquid is not a public blockchain network. Moreover, Liquid is using a version of the PoW consensus mechanism which requires even more computational resources than Bitcoin (Liquid requires the validators to run a Bitcoin node in parallel with a Liquid node). Similar to Liquid, RSK employs a federation to validate transactions via a PoW-based mechanism. Although RSK is more decentralized, i.e., the federation in RSK is determined by public voting, RSK is still limited by the huge energy consumption of the PoW mechanism.

Different from the PoW mechanism, the PoS mechanism enables the blockchain participants to reach the consensus by proving tokens ownership. As a result, the PoS mechanism is much more energy-efficient and can achieve higher transaction processing speed compared to those of the PoW mechanism [1]–[3]. Due to those advantages, recent research works in the area of the sidechain technology have shifted towards the PoS mechanism. In [16], a protocol is developed for cross-chain transfers between a primary blockchain (main chain) and a secondary chain (sidechain). To validate the cross-chain transactions, the protocol relies on a set of certifiers who are chosen by the main chain. A major advantage of the proposed protocol is the independence between the side chain and main chain in terms of security and operations. However, the security of this protocol is not guaranteed. In [17], the authors propose a sidechain system, in which both the sidechain and the main chain employ a PoS mechanism, i.e., Ouroboros. Unlike the previous works, this work focuses more on the security aspects of the sidechain technology, providing formal definitions and robust security analyses. However, the risk of centralization is not addressed. Similar to [16], the authors in [18] also introduce a cross-chain transfer protocol to allow interoperability between a main chain and a side chain. The cross-chain transfer protocol in [18] is proposed with formal definitions, and a consensus mechanism is also presented in a similar way as in [17]. However, the security of the protocol is not guaranteed, and the risk of centralization is also unaddressed. In [51], a PoS-based framework is proposed for a federated-blockchain system. Cross-chain transactions in this framework are processed by a group of validators. These validators are chosen based on their stakes once per day, and they are rewarded for their validation. However, this framework lacks formal security analysis, and it requires more than 66% of the network stakes to be controlled by honest users (Fedchain only requires 51%). In [52], a framework for federated-blockchain is developed based on the Tendermint consensus mechanism [29]. In this framework, cross-chain transactions are processed by a group of fixed validators. Such setting may result in a

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: Huazhong University of Science and Technology. Downloaded on January 31,2023 at 09:41:52 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Services Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSC.2023.3240235

higher risk of centralization as these validators are predetermined and known by the whole network. In [54], a novel cross-chain transfer method is proposed. By requiring the transaction's sender and receiver to vote on a transaction, this method allows the transfers of assets among different parties on different blockchain network. In [55], a cross-chain commitment protocol is developed to enable asset transfers among different blockchains. Different from previous work, this protocol consider the cases where users need to send their transactions on time to a specific smart contract to transfer their assets. A common limitation of both [54] and [55] is that the risk of centralization is not considered. Comparisons between our work and the related works are summarized in Table 1.

To the best of our knowledge, the risk of centralization in federated-blockchain systems has not been addressed in any previous work. Specifically, the ability to transfer tokens between chains may lead to situations where the users centralize to a single chain in the system, e.g., the chain which gives the highest rewards for consensus participation. Such centralization of tokens and users may have negative impacts on the security and performance of the other blockchains in the same system. The reason is that the state of each PoS blockchain is determined by the majority of stakes (tokens), i.e., users who have more stakes will be very likely to be selected to add new blocks. Consequently, it is easier for attackers to target the blockchains that have fewer tokens. This can significantly impact these blockchains' security. Furthermore, since the cross-chain transfer requires the confirmation of transactions in both the originating and destination chains, the centralization of stakes also reduces the overall system performance. More detailed analysis of these negative impacts will be presented in Section 3.

1.3 Contributions and Paper Organization

The main contributions of this paper are as follows:

- Propose FedChain, an effective and secure framework for cross-chain transfers in federated-blockchain systems. Particularly, Fedchain facilitates two-way transfers of assets between any two different chains in the system by utilizing the sidechain technology.
- Develop a novel PoS-based consensus mechanism for the individual blockchains in FedChain that can satisfy the persistence and liveness properties [33], prevent many blockchain-specific attacks, and achieve a more desirable transaction confirmation time compared to other mechanisms such as [26]–[32].
- Develop an incentive mechanism using a Stackelberg game model [39] for FedChain in order to address the problem of centralization in the sidechain technology, provide additional benefits for the users, and enhance FedChain's security and performance. To the best of our knowledge, this is the first paper addressing the risk of centralization in federated-blockchain systems. Furthermore, we can prove the uniqueness of the Stackelberg equilibrium and find the exact formula for this equilibrium. These results help the stakeholders to determine their best investment strategies and the chain operators to design the optimal incentive policy.
- Perform extensive simulations to evaluate the system performance of FedChain. The simulation results then confirm the analytical results and show that FedChain can help the users to maximize their profit and the blockchain operators to determine their optimal blockchain parameters to improve the system's security and performance.

In our previous publication [49], we develop a blockchain-based framework for a different scenario, namely coalition loyalty programs. In this framework, an existing consensus mechanism is adopted, the performance is not compared with other consensus mechanisms, and the security is proven only for only the Common Prefix property under a single static adversary setting. Unlike [49], the consensus mechanism and analyses in this paper are much more generalized and extensive. Particularly, we (i) provide detailed descriptions of the consensus mechanism, (ii) prove that it can satisfy the common prefix, chain growth, and chain quality properties under two different adversary settings, (iii) prove that it can prevent many blockchain-specific attacks, (iii) provide detailed performance analysis. Moreover, the game considered in [49] is a non-cooperative game, whereas in this paper we formulate a two-stage Stackelberg game.

The rest of this paper is organized as follows. We first present the federated-blockchain framework in Section 2. We analyze the proposed consensus mechanism for our framework in Section 3. After that, we introduce and analyze the Stackelberg game in Section 4. Finally, simulations and numerical results are presented in Section 5, and conclusions are drawn in Section 6.

2 Federated-blockchain System

2.1 System Overview

Before elaborating on our proposed consensus mechanism and incentive mechanism, we provide a brief overview of the federated-blockchain system and the cross-chain transfer procedure in this section [10], [11]. As illustrated in Fig. 1, the system is composed of two types of entities as follows:

- **Chains (blockchains):** In FedChain, individual blockchain networks, managed by blockchain operators, can communicate with each other via the cross-chain transfer protocol. Each chain has its own type of token and an individual consensus mechanism. When a new blockchain network wants to join the system, it only needs to negotiate with the existing chains and create smart contracts accordingly.
- **Users:** Users are the participants of the chains in the system. These users can freely exchange different types of tokens by using the smart contracts created by the operators. They can also participate in the consensus mechanism in every chain to earn economic profits through block rewards.

2.2 Cross-chain Transfer Procedure

The SPV mechanism [10] allows tokens from one chain to be securely transferred to another at a predetermined rate.

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: Huazhong University of Science and Technology. Downloaded on January 31,2023 at 09:41:52 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Services Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSC.2023.3240235

TABLE 1
Comparisons with related works

Article	Validators	Consensus	Security Analysis	Centralization Risk
[12]	Fixed, publicly known	PoW	Unproven	Unaddressed
[13]	Chosen by network operators (private)	PoW	Unproven	Unaddressed
[14]	Chosen by public voting	PoW	Unproven	Unaddressed
[16]	Chosen by public voting	PoS	Unproven	Unaddressed
[17]	Chosen by public voting	PoS	Proven	Unaddressed
[18]	Chosen by public voting	PoS	Unproven	Unaddressed
[47]	Chosen by public voting	PoS	Unproven	Unaddressed
[48]	Fixed, publicly known	PoS	Proven	Unaddressed
[50]	Vote by transaction sender and receiver	Undefined	Proven	Unaddressed
[51]	Use smart contract	Undefined	Proven	Unaddressed
Our proposed FedChain framework	Use smart contract	PoS	Proven	Addressed

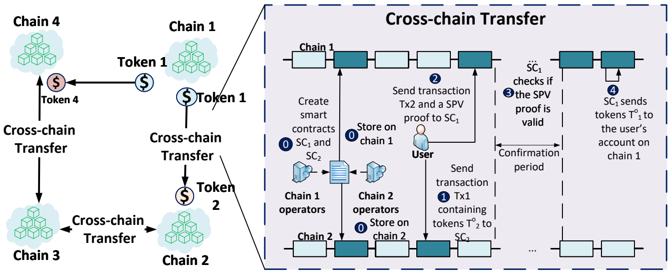


Fig. 1. The federated-blockchain system.

When a user wants to prove that a transfer transaction from an originating chain to a destination chain is valid (not conflicting, digital signature matched the account), an SPV proof is submitted. This proof shows that the transfer transaction belongs to a valid block of the originating chain. Although this process takes a long time for confirmation, it eliminates the risk of centralization and single-point-of-failure [11]. Therefore, the SPV proof is selected as the cross-chain transfer mechanism in our proposed FedChain. As illustrated in Fig. 1, the SPV-based token exchange procedure consists of several steps as follows:

- *Step 0:* Two chains negotiate an agreement which specifies the exchange rate between the two tokens. The chain operators then create in each chain a smart contract according to the agreement.
- *Step 1:* When a user wants to exchange T_2^o tokens into T_1^o tokens, the user sends a transaction Tx1, containing T_2^o tokens, from its account on chain 2 to the smart contract SC_2.
- *Step 2:* The user then sends a transaction Tx2 and an SPV proof from its account on chain 1 to SC_1. Tx2 then triggers SC_1 to validate the SPV proof.
- *Step 3:* During the confirmation period, SC_1 checks (1) the validation of the SPV proof and (2) any conflicts of the submitted SPV proof.
- *Step 4:* After the confirmation period, SC_1 sends a number of T_1^o tokens to the customer's address on chain 1 in accordance with the exchange rate.

The security features of the SPV proof mechanism are proven in [10]. The SPV proof points to the block that contains the cross-chain transfer transaction in the originating chain. Therefore, the validators only have to validate the block that contains the transaction. Thus, the security of the SPV proof only relies on the security of the originating chain, i.e., the SPV proof is secure if the originating chain is secure. However, this leads to a drawback of the SPV proof mechanism, which is the low confirmation speed (the validators have to wait until the transaction is confirmed on the originating chain). Moreover, as the stakes can be transferred between chains, if the security of one chain is violated, the whole system will fail. Therefore, in the next section, we will propose an effective consensus mechanism that can achieve lower transaction confirmation time compared to other conventional mechanisms while satisfying the persistence and liveness properties [33] and being able to prevent various blockchain attacks.

3 FedChain's Consensus Mechanism

In this section, we develop an effective consensus mechanism for FedChain with four new consensus rules based on the consensus mechanism proposed in [26]. Compared with other conventional consensus mechanisms such as [26]–[32], our proposed consensus mechanism can satisfy both the liveness and persistence properties, prevent various blockchain attacks, and achieve an especially low transaction confirmation time as discussed in the following.

3.1 Proposed Consensus Mechanism

3.1.1 *Epochs and time slots*

As illustrated in Fig. 2, time is divided into epochs, and each epoch is divided into time slots in FedChain's consensus mechanism. At the first time slot of epoch e_k, a committee consisting of some users (stakeholders) executes an election protocol to elect the leaders for the epoch e_k, such that for each time slot there is one designated leader who adds one new block to the chain. Similar to [26], we assume that the network is synchronous [50], and a time slot duration of 20 seconds is sufficient for the leader to broadcast a block to every node in the chain. The committee also select the committee members for the epoch e_{k+1}.

3.1.2 *Leaders and committee election protocol*

To elect the leaders and committee, the current epoch's committee members execute the Publicly Verifiable Secret Sharing (PVSS) protocol [35] to create seeds for the Follow-the-Satoshi (FTS) algorithm [3]. The PVSS protocol allows the

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: Huazhong University of Science and Technology. Downloaded on January 31,2023 at 09:41:52 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Services Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSC.2023.3240235

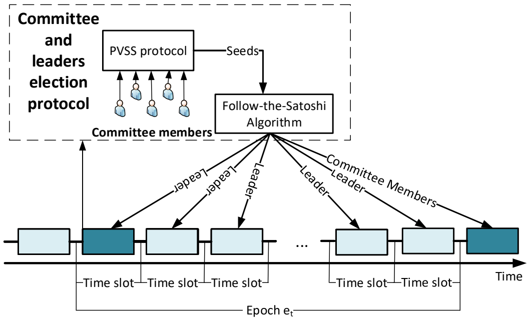


Fig. 2. Epoch-based committee and leader election.

participants to produce unbiased randomness in the form of strings and any network user to verify these strings, as long as the majority (51%) of participants are honest (abiding by the rule of the consensus mechanism), as proven in [35]. Once the random strings are created, they are used as the seeds for the FTS algorithm. The FTS algorithm is a hash function that takes any string as input and outputs token indices [3]. The current owners of these tokens are then chosen as the leaders of this epoch or committee members of the next epoch. The probability P_n that user n is selected to be the leader and committee member by the FTS algorithm in a network of N stakeholders is

$$P_n = \frac{s_n}{\sum_{i=1}^{N} s_i}, \tag{1}$$

where s_n is the number of stakes of stakeholder n. As observed in (1), the more stakes a stakeholder has, the higher chance it can be selected to be the leader. Compared to [26], we design four new consensus rules as follow:

- I_1: After executing the PVSS protocol, the leader list is broadcast to every node in the chain.
- I_2: If a leader fails to broadcast its block during its designated time slot (e.g., being offline during its time slot), an empty block will be added to the chain
- I_3: Once a block is broadcast, the designated leader will not change the block at any later time.
- I_4: Upon receiving two forks (different versions of the chains), honest users adopt the longest valid fork, i.e., the longest fork that has no conflicting blocks and each block is signed by a designated leader.

Rule I_1 can be implemented by instructing the committee members to publish their votes (secret shares) that they used in the PVSS protocol execution, e.g., in the Data field of the block. As long as the adversary does not control more than 50% of the committee, the PVSS protocol can guarantee the unbiased randomness of the result and allow everyone to verify [35]. Rules I_2 and I_3 can be implemented by instructing the leaders to not change their blocks, e.g., change the block's header or transactions. These two rules make sure that a leader cannot change its block once it is broadcast. As a result, every block created by an honest leader will become a checkpoint block. This helps to solidify the whole chain from the genesis block up to the latest honest block. Moreover, Rule I_2 also helps to maintain the chain growth even if the leader cannot broadcast the block in time, e.g., under DDoS attacks. Rule I_4 can be trivially implemented by instructing the stakeholder to check the leader list. Existing consensus mechanisms, e.g., [26]–[30], often adopt the longest chain rule to guarantee chain growth. Alternatively, in our proposed consensus mechanism, we have Rule I_2 to guarantee the chain growth property, and thus we can adopt a more secure version of the longest chain rule, i.e., I_4. These new consensus rules help to considerably reduce the probability that an adversary can successfully create an alternative version of the chain, thereby significantly improving the chain's security and performance. The detailed analysis will be discussed in Theorem 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 consensus participants via a reward scheme and penalize malicious behavior via a penalty scheme. Note that, there are several research works on the design of blockchain's incentive mechanism, such as [19]–[22], but they are only applicable for individual chains with a specific application, e.g., blockchain-based mobile edge computing, consortium blockchains, and vehicular ad-hoc networks. Hence, they cannot be applied for federated-blockchain systems due to strong relations as well as competitions among blockchain service providers and stakeholders.

For the reward scheme, a leader will receive a fixed number of tokens when the leader adds a new block to the chain. This is also to incentivize the leaders to be online during their designated time slots. In single-blockchain settings such as Bitcoin [34] and Cardano [26], the block reward is set at a fixed value for a long period of time, e.g., 4 years in Bitcoin. However, in FedChain, having a fixed block reward scheme may pose security threats. The reason is that the stakes can be transferred between chains in our system, and the total network stakes can also vary in times, e.g., stakes increase from block rewards, and the stakes decrease from cross-chain transfers, etc. Since the probability that a stakeholder is elected to be the leader and able to obtain a block reward depends on the individual chain's stakes, stakeholders may transfer their stakes to a chain with a higher block reward to earn more profits. Consequently, this may attract stakes into a single chain and make it easier for adversaries to control the majority of stakes in the other chains. Therefore, in the following sections, we analyze the stakeholder rational strategy and propose a dynamic reward scheme to protect the decentralization of the whole system. With our proposed dynamic reward scheme, at the end of each epoch, the chains will adjust new block reward values for the next epoch, taking the total network stakes and the final stakes distribution among the chains in the current epoch into account. The dynamic reward scheme will be discussed in more details in Section 4.

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 [3], and transaction denial attacks [26]. The stakes of committee members are also locked during the epoch that they are serving in the committee to prevent long-range attacks [3]. How the proposed penalty scheme can prevent the mentioned attacks will be discussed in the following security analysis.

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: Huazhong University of Science and Technology. Downloaded on January 31,2023 at 09:41:52 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Services Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSC.2023.3240235

3.2 Security Analysis

3.2.1 Adversary and attack models

Since the SPV proof mechanism's security depends on the security of the individual chains, the security of the whole system also relies on the security of each chain. Concerns regarding the privacy of users in sidechains can be addressed using techniques such as the ones proposed in [44]. For example, privacy-enhancing techniques such as Onion routing and Garlic routing [44] can be employed on a sidechain in the form of smart contracts. In this way, transactions from the main chain can be routed through the smart contracts before they are published in the main chain, thereby improving user privacy. . As illustrated in Fig. 3, we consider two types of adversaries as follows:

- **Static Adversary:** This type of adversary uses a stake budget B_A to attack a chain. Let B_n and γ denote the stake budgets of stakeholder n and the honest stake ratio, respectively. Then, the adversarial ratio, i.e., the ratio of adversarial stakes to the total network stakes, is $1-\gamma = \dfrac{B_A}{\sum_{n=1}^{N} B_n + B_A}$.
- **Adaptive Adversary:** In contrast to the static adversary setting, the adaptive adversary does not have a fixed number of stakes. However, this type of adversary can choose to corrupt N_A honest stakeholders and use their stakes to attack. Let $\mathcal{N}_A$ denote the set of corrupted stakeholders, the budget of the adaptive adversary can be defined by $B_A = \sum_{i \in \mathcal{N}_A} B_i$.

The models for the blockchain-specific attacks considered in this paper are as follows:

- **Double-spending attack:** For such kind of attack, the attacker aims to revert a transaction that has been confirmed by the network (to gain back the tokens it has already spent). First, the attacker creates a transaction Tx1 in block $\mathcal{B}_i$ and waits until the block is confirmed. Then, the attacker can either create a conflicting transaction Tx2 or erase the block $\mathcal{B}_i$ from the chain, so that the proof of its spending is gone.
- **Grinding attack:** In grinding attacks, the attacker attempts to influence the leader election protocol to unfairly increase its chance to be selected as a leader. Generally, in protocols where the seeds of the FTS algorithm are derived from the block header, the attacker can check many possible different block contents to determine which one can give the attacker the best chance to be elected as a leader.
- **Nothing-at-stake attacks:** In this attack, the attacker tries to create many forks or conflicting transactions. For example, the attacker can create two transactions to spend the same tokens at two vendors, i.e., Tx1 in fork $\mathcal{C}_1$ and Tx1 in fork $\mathcal{C}_2$. At this point, although both the transactions are not *confirmed*, they are both *valid* (not conflicted within their own fork).
- **Bribe attacks:** For such attacks, the attacker tries to bribe the leaders to create specific blocks, e.g., to support other types of attacks such as double-spending or transaction denial.
- **Transaction denial attack:** In this attack, the attacker tries to prevent transactions of every or some specific users from being included in the chain. To achieve this objective, the attacker has to either block the users' connection to the blockchain or not include the transactions when the attacker is the leader.
- **Long-range attack:** In a long-range attack, a leader immediately transfers its stakes to another account at the beginning of its designated epoch, and thus it can behave maliciously, e.g., performing attacks, for the rest of the epoch without consequences.

3.2.2 Blockchain properties

To maintain the blockchain's security, a consensus mechanism must satisfy the following properties [33]:

- **Persistence:** Once a transaction is more than κ blocks deep in the chain of an honest user, all other honest users will have that transaction in the same position in their chains.
- **Liveness:** After a sufficient period, a valid transaction will be confirmed by all the honest users.

In FedChain, persistence ensures that once a transaction is confirmed, i.e., more than κ blocks deep in the chain, it cannot be reverted. Without the persistence property, the adversary can successfully perform a double-spending attack by firstly sending a transaction to spend some tokens. After that transaction is confirmed, the adversary can create a fork to erase the transaction from the blockchain. If that fork is accepted by the honest users, the adversary can gain back the tokens it already spent. 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 can block every transaction in a blockchain. The persistence and liveness properties are ensured if the consensus mechanism satisfies the following properties [33]:

- **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 $1-\mu$. In the ideal case, $1-\mu$ equals the adversarial ratio $1-\gamma$.

Let $\mathrm{Pr}_{\mathrm{CP}}$, $\mathrm{Pr}_{\mathrm{CG}}$, and $\mathrm{Pr}_{\mathrm{CQ}}$ denote the probabilities that the CP, CG, and CQ properties are violated. We prove that FedChain's consensus mechanism can satisfy the CP, CG, and CQ properties with overwhelming probability, i.e., $\mathrm{Pr}_{\mathrm{CP}}$, $\mathrm{Pr}_{\mathrm{CG}}$, and $\mathrm{Pr}_{\mathrm{CQ}}$ are overwhelmingly low ($< 0.1\%$), in the following Theorem.

Theorem 1. *FedChain's consensus mechanism can satisfy the CP, CG, and CQ properties with* $\mathrm{Pr}_{\mathrm{CP}} = (1-\gamma)^{\kappa}$, $\mathrm{Pr}_{\mathrm{CG}} = 1$, *and* $\mathrm{Pr}_{\mathrm{CQ}} < 1 - \exp\left(\dfrac{l(\gamma-1)\delta^2}{2}\right)$.

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: Huazhong University of Science and Technology. Downloaded on January 31,2023 at 09:41:52 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Services Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSC.2023.3240235

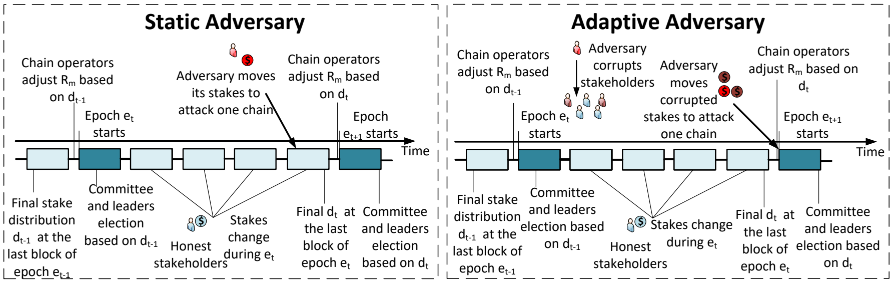

Static Adversary
Chain operators adjust R_m based on d_{t-1}
Adversary moves its stakes to attack one chain
Chain operators adjust R_m based on d_t
Epoch e_t starts
Epoch e_{t+1} starts
Time
Final stake distribution d_{t-1} at the last block of epoch e_{t-1}
Committee and leaders election based on d_{t-1}
Honest stakeholders
Stakes change during e_t
Final d_t at the last block of epoch e_t
Committee and leaders election based on d_t
Adaptive Adversary
Chain operators adjust R_m based on d_{t-1}
Adversary corrupts stakeholders
Adversary moves corrupted stakes to attack one chain
Chain operators adjust R_m based on d_t
Epoch e_t starts
Epoch e_{t+1} starts
Time
Final stake distribution d_{t-1} at the last block of epoch e_{t-1}
Committee and leaders election based on d_{t-1}
Honest stakeholders
Stakes change during e_t
Final d_t at the last block of epoch e_t
Committee and leaders election based on d_t

Fig. 3. Illustrations of the considered adversaries.

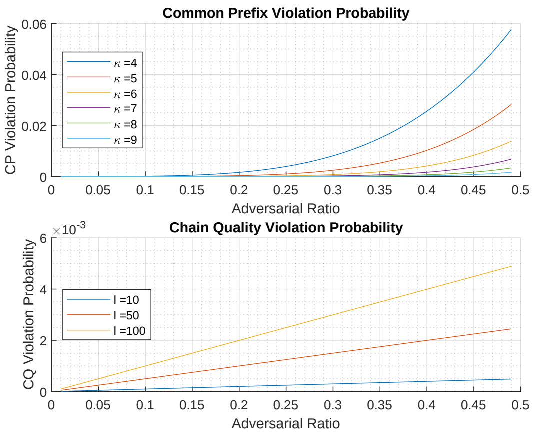


Fig. 4. Blockchain properties violation probabilities.

Proof: We first prove $\Pr_{\mathrm{CP}}$ by showing that the adversary needs to be the leader for κ consecutive blocks to violate CP. We then prove $\Pr_{\mathrm{CG}} = 1$ by using Rule I_2. Finally, we prove $\Pr_{\mathrm{CQ}}$ by using the random walk and Chernoff bound. The detailed proof is provided in Appendix A. □

Fig. 4 illustrates the CP and CQ violation probabilities under different parameter values. As the adversarial ratio increases (i.e., the adversary controls more stakes in the chain), the attacker has more chances to successfully attack. However, the higher κ is, the lower the CP violation probability is. This means that the longer since a transaction is added to the chain, the more stable the transaction becomes. For example, if a transaction is at least seven blocks deep in the chain, the adversary has less than 1% chance to revert it, even if the adversary controls nearly 50% of the total network stakes. In contrast, if the transaction is only four blocks deep, the adversary with 49% stakes has more than 5% chance to revert the transaction. This implies that the more stakes the adversary controls, the longer it takes to confirm a transaction, which is directly related to the performance and security of the chain. For the $\Pr_{\mathrm{CQ}}$, the more blocks we consider, the higher chance the adversary can create more than $(1-\gamma)l$ blocks. For example, an adversary controlling 30% of network stakes has less than 0.1% chance to create more than three in ten blocks, but it has around 0.3% chance to create more than 30 in 100 blocks. This could be harmful to the network if the adversary wants to reduce the network's throughput (i.e., blocks/time slot) by creating only empty blocks every time it is the leader.

3.2.3 Blockchain attacks prevention

In the following Theorem, we prove that our FedChain's consensus mechanism is able to prevent a variety of emerging blockchain attacks such as double spending, grinding, bribe, nothing-at-stakes, and long-range attacks.

Theorem 2. *FedChain's consensus mechanism can prevent double-spending, nothing-at-stakes, bribe, transaction denial attacks, grinding, and long-range attacks according to the considered adversary models.*

Proof: We prove that double-spending and nothing-at-stakes attacks are prevented if CP is not violated. Then, we prove that grinding attacks can be prevented by the PVSS protocol, and bribe attacks are prevented because the adversary does not know the leader in advance. Moreover, transaction denial attacks could be prevented if CG and CQ hold. Furthermore, long-range attacks are prevented because the leader's stakes are locked during the epoch. The detailed proof is provided in Appendix B. □

3.3 Performance Analysis

From the security perspective, we prove that the higher the adversarial ratio is, the higher the probabilities that the adversary can successfully perform attacks on the chain. Similarly, the adversarial ratio also has a negative impact on the performance of the network. In this performance analysis, we aim to analyze and compare the performance of our proposed consensus mechanism when it is employed by individual blockchains in the federated-blockchain system. As shown in Table 2, we examine and compare the transaction confirmation time 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 FedChain's consensus mechanism. The transaction confirmation time of Cardano and Bitcoin, obtained from [26], is under optimal network conditions. This means that the time is theoretically calculated, only taking into account the effects of the adversarial ratio [26]. Specifically, the transaction confirmation time is the time it takes to reach a CP violation probability $\Pr_{\mathrm{CP}} \leq 0.1\%$. For Fedchain's consensus mechanism, κ can be determined based on (6), and then κ is multiplied with the time slot duration to calculate the transaction confirmation time. Our time slot duration is set to be 20 seconds (the same as that of Cardano [37]).

As observed in Table 2, the more stakes the adversary controls, the longer the transaction confirmation time is.

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: Huazhong University of Science and Technology. Downloaded on January 31,2023 at 09:41:52 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Services Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSC.2023.3240235

TABLE 2
Transaction confirmation time in minutes

Adversarial Ratio	Bitcoin	Cardano	FedChain's Consensus Mechanism
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

Moreover, the PVSS protocol no longer ensures unbiased randomness if the adversary controls more than 50% stakes in a chain. Therefore, it is critical to attract more participants to individual chains in order to increase the network's total stakes and prevent the adversary from controlling more than 50% of network stakes. In the next section, we will introduce an effective incentive mechanism developed based on a Stackelberg game model that can jointly maximize profits for the participants and significantly enhance the network's performance and security for chain operators.

4 Stackelberg Game Formulation

In practice, chains usually announce their block rewards first, and then the stakeholders will decide how much to invest accordingly. Therefore, the interaction between the chains and stakeholders in FedChain can be formulated as a multiple-leaders-multiple-followers Stackelberg game model [39]. In this game, the leaders are the chains (managed by the chain operators) who first announce their block rewards, and then the stakeholders, i.e., followers, will make their decisions, e.g., how much to invest in each chain. It is worth noting that there are some approaches that apply the Stackelberg game models to blockchain systems in the literature, such as [23]–[25]. Nevertheless, these models can be applied for individual blockchains only, and thus they cannot be directly adopted for federated blockchain systems in which competitions between multiple blockchain service providers and stakeholders are taken into considerations.

4.1 Stakeholders and Chain Operators

FedChain consists of a set $\mathcal{M}$ of M chains and a set $\mathcal{N}$ of N followers. The leaders offers block rewards $\mathbf{R} = (R_1, \ldots, R_M)$. Stakeholders possess stakes with budgets, denoted as $\mathbf{B} = (B_1, \ldots, B_N)$. The stakeholders can use their stakes to take part in the consensus process of every chain to earn additional profits. Particularly, when stakeholder n invests s_n^m to chain m, its expected payoff U_n^m is:

$$U_n^m = \frac{s_n^m}{s_n^m + \sum_{i \in \mathcal{N}_{-n}} s_i^m} R_m, \tag{2}$$

where $\mathcal{N}_{-n}$ is the set of all stakeholders except stakeholder n. In the considered system, the stakeholders can freely invest within their budgets to any chain, i.e., $\sum_{m=1}^{M} s_n^m \leq B_n$. Thus, the total payoff of stakeholder n is

$$U_n = \sum_{m=1}^{M} U_n^m = \sum_{m=1}^{M} \left(\frac{s_n^m}{s_n^m + T_m} R_m \right), \tag{3}$$

where $T_m = \sum_{i \in \mathcal{N}_{-n}} s_i^m$ expresses the total stakes invested in chain m by all the other stakeholders.

4.2 Game Theoretical Analysis

4.2.1 Followers' strategy

To analyze the game, we first examine the existence of the follower sub-game equilibrium in Theorem 3.

Theorem 3. *There exists at least one Nash equilibrium in the follower sub-game.*

Proof: We prove existence of the equilibrium by proving that the strategy space is convex and U_n is concave $\forall n \in \mathcal{N}$ [39]. The detailed proof can be found in Appendix C. □

Then, we examine the uniqueness of the equilibrium in Theorem 4.

Theorem 4. *The follower sub-game equilibrium is unique, and the convergence to the equilibrium is guaranteed.*

Proof: We prove the uniqueness by showing U_n satisfies Rosen Theorem's conditions [48]. The detailed proof is provided in Appendix D. □

In this game, the stakeholders can invest any number of stakes within their budgets. However, as shown in Theorem 5, a rational stakeholder will always invest all its budget regardless of the other stakeholders' strategies.

Theorem 5. *For every follower n, the strategies that invest less than its total budget, i.e., $\sum_{m=1}^{M} s_n^m < B_n$, always give lower payoffs than the strategy that invests all the budget, i.e., $\sum_{m=1}^{M} s_n^m = B_n$, regardless of other followers' strategies.*

Proof: We compare the utility functions in two cases to prove that investing all stakes always brings more profits. The detailed proof is provided in Appendix E. □

As a result of Theorem 5, the strategies which invest less than the total budget can be removed from the strategy space of every follower. Then, we can reformulate the utility function to reflect the budget constraint as follows:

$$U_n = \sum_{m=1}^{M-1} \left(\frac{s_n^m}{s_n^m + T_m} R_m \right) + \frac{B_n - \sum_{m=1}^{M-1} s_n^m}{B_n - \sum_{m=1}^{M-1} s_n^m + T_M} R_M. \tag{4}$$

With the existence and uniqueness guaranteed, the only question remained is how to find the equilibrium point. Interestingly, for the considered game model, we can prove the exact formula of the equilibrium in Theorem 6.

Theorem 6. *The point where every follower's strategy satisfies $s_n^{*m} = B_n \frac{R_m}{\sum_{i=1}^{M} R_i}, \forall m \in \mathcal{M}, \forall n \in \mathcal{N}$ is the unique equilibrium of the follower sub-game.*

Proof: We prove that at $s_n^{*m} = B_n \frac{R_m}{\sum_{i=1}^{M} R_i}, \forall m \in \mathcal{M}, \forall n \in \mathcal{N}$, all the followers can maximize their profits,, and thus this is the equilibrium. The detailed proof is provided in Appendix F. □

Then, we can conclude that there is a unique sub-game equilibrium for every leader strategy set, and at the equilibrium the stakeholders play their optimal strategies s_n^{*m}.

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: Huazhong University of Science and Technology. Downloaded on January 31,2023 at 09:41:52 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Services Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSC.2023.3240235

This optimal strategy only depends on the stakeholder's total budget and the ratios of block rewards between the chains. Therefore, every stakeholder will always invest according to its unique optimal strategy. In the next stage, we will analyze the leader strategy to determine the optimal block reward for the leaders.

4.2.2 Leader strategy

The proposed incentive mechanism for FedChain has two main aims. The first one is to attract stakes to improve the individual chain's performance and security. The second aim is to ensure the decentralization of the system, i.e., encourage the stakeholders to distribute their stakes evenly across all the chains. For these two aims, we propose a utility function U_m for the leaders as follows:

$$\begin{aligned} U_m &= \sum_{n=1}^{N} \omega_m^n s_n^{*m} - R_m \\ &= \sum_{n=1}^{N} \frac{B_n R_m}{\sum_{i=1}^{M} R_i} \ln\left(\frac{B_n R_m}{\sum_{i=1}^{M} R_i}\right) - R_m, \end{aligned} \tag{5}$$

where ω_m^n is a weight factor which can be defined by $\omega_m^n = \ln(s_n^{*m})$. By using the logarithm of the stakes as the weight factor, we can achieve two main aims. In particular, from this designed utility function, a leader can attract more stakes invested to its pool by increasing its block reward. However, at a certain level, if this leader keeps increasing its block reward to get more stakes, its utility will be decreased. As a result, this utility function encourages the chain operator to set an appropriate level of block reward such that it can attract sufficient stakes to the chain while ensuring that individual stakeholders do not control too much of the network stakes. Moreover, this also discourages the chain operators from setting a too high block reward that will cause the centralization of stakes into a single chain in FedChain. Then, we proceed to find the equilibrium of the upper sub-game and the Stackelberg equilibrium of the considered Stackelberg game in Theorem 7.

Theorem 7. *The point where every leader's strategy is* $R_m^* = \dfrac{M-1}{M^2}\sum_{n=1}^{N} B_n\left(1+\ln\left(\dfrac{B_n}{M}\right)\right)$ *and every follower's strategy satisfies* $s_n^{*m} = B_n \dfrac{R_m}{\sum_{i=1}^{M} R_i}, \forall m \in \mathcal{M}, \forall n \in \mathcal{N}$ *is the unique Stackelberg equilibrium of the considered game. Moreover, the convergence to the Stackelberg equilibrium is guaranteed.*

Proof: We solve $\dfrac{\mathrm{d}U_m}{\mathrm{d}R_m} = 0$ to find R_m^*. Since R_m^* is uniquely defined by constants, the equilibrium is unique. The detailed proof is provided in Appendix G. □

Interestingly, the result from Theorem 7 shows that the optimal strategies are the same for all the chain operators. The reason is that since stakes can be transferred, the security of the whole system is as strong as that of the weakest chain. Therefore, the highest utility can only be achieved when every chain is equally secure.

5 Performance Evaluation

In this section, we conduct experiments and simulations to (i) show that the proposed Stackelberg game can help the stakeholders to maximize their profits, (ii) confirm our analytical results, and (iii) demonstrate that the proposed incentive mechanism can enhance FedChain's security and performance. To this end, we first examine the utility function of a stakeholder to confirm our results from Theorem 6 and show that the Stackelberg game model can help to maximize the stakeholder's profit. After that, to evaluate the security and performance of the FedChain, we implement extensive simulations under various settings. In the simulations, we first show that the rational stakeholders will act according to our proposed Stackelberg game-theoretical analysis. We will then demonstrate that the FedChain's consensus mechanism can satisfy the security properties and attain reasonable performance even under extreme adversarial scenarios. Furthermore, we will show that under the same simulation setting, the proposed dynamic reward scheme achieves better security and performance compared to those of the static reward scheme.

5.1 Simulation Setting

First, we examine the utility function of stakeholder 1 in a small case which consists of two stakeholders and three chains. The stakeholders have budgets $\mathbf{B} = [100, 300]$, and the chains set block rewards to be $\mathbf{R} = [10, 20, 30]$. In this experiment, the strategy of stakeholder 2 is fixed according to Theorem 6. Then, we simulate a system with N stakeholders and M chains under different adversarial models (static and adaptive), reward schemes (static and dynamic), and different adversarial levels (weak, medium, and strong). The simulation parameters are presented in Table 3.

The simulation has several steps as presented in Algorithm 1 (see Appendix H). In particular, at the beginning, each stakeholder has a budget $B_i \in [\mathrm{LB}, \mathrm{UB}]$ generated randomly with uniform distribution. Each chain operator then sets a block reward R_m based on Theorem 7's result in the case of the dynamic reward scheme. In the static reward scheme, R_m are fixed as constants based on several real-world PoS blockchain networks [45]–[47]. After the block rewards are set, the stakeholders make their decisions. To find the best strategies for each stakeholder, we employ the Matlab fmincon function [38], starting from stakeholder 1. Then, the newly found optimal strategy is fixed for the stakeholder, and the algorithm continues to find the best response for stakeholder 2 until stakeholder N. After that, the adversary begins to attack. In the static adversary scenario, the adversarial stakes budget B_A is constant and predetermined. In the adaptive adversary scenario, the adversary chooses a number N_A of stakeholders to corrupt, making their stakes to be adversarial stakes, i.e., the adversarial stakes budget is $\sum_{i\in\mathcal{N}_A^S} B_i$. Then, we measure the impacts of the adversary on $\Pr_{\mathrm{CP}}$, $\Pr_{\mathrm{CQ}}$, transaction confirmation time, and transaction throughput. Finally, we simulate the stake changes by randomly choosing N_Δ stakeholders and changing their budgets by $\pm\Delta_s B_n, \Delta_s \in (0, 1)$. The epoch is then ended, and the simulation moves to the next epoch until the stopping criteria are met, i.e., after n_e epochs.

During the simulation, we measure several important security and performance criteria. First, we measure the stake distribution at the beginning of each epoch to see if the rational stakeholders invest according to our game-theoretical

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: Huazhong University of Science and Technology. Downloaded on January 31,2023 at 09:41:52 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Services Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSC.2023.3240235

TABLE 3
Parameter setting

Parameter	Weak Adversary	Medium Adversary	Strong Adversary
N	100	100	100
M	3	3	3
LB	50	50	50
UB	100	100	100
Δ_s	$(0,1)$	$(0,1)$	$(0,1)$
B_A	500	1000	1500
N_A	10	20	30
n_e	10	10	10

analysis. Then, we examine four different scenarios. In the first two scenarios, we simulate a static adversary who will try to attack the chains under the static and dynamic reward schemes. In the remaining scenarios, an adaptive adversary will try to attack the chains. For each type of adversary, we simulate three different levels of adversary capacity (low, medium, and high) as shown in Table 1.

In terms of security, we measure the CP and CQ violation probabilities. These probabilities can be determined by (6) and (8), respectively. In terms of performance, we measure how much the adversaries can negatively impact the transaction confirmation time and transaction throughput. To calculate the transaction confirmation time, for each chain, we find the value of κ such that $\Pr_{\text{CP}} < 0.1\%$. For the transaction throughput, we want to examine the case where the adversary wants to reduce the transaction processing capability of one of the chains. Specifically, the adversary will move all its stakes to a chain and participate in the leader selection process. For every block the adversary is elected to be the leader, it creates an empty block without any transaction, thereby reducing the network's transaction throughput. In the simulation, we measure a transaction throughput reduction threshold Θ, such that the probability that the adversary can reduce the transaction throughput more than Θ is overwhelmingly low (i.e., $\Pr_{\text{CQ}} < 0.1\%$).

5.2 Performance Results

5.2.1 Economical benefits

Fig. 5 illustrates the utility function of stakeholder 1 in the case where stakeholder 2 invest according to s_n^{*m}. As observed from the figure, stakeholder 1 can achieve maximum utility when it also invests according to s_n^{*m}. Particularly, stakeholder 1 achieves a utility $U_1^* = 15$ with the optimal strategy $\mathbf{s}_1^* = [16.6, 33.3, 50]$. This result shows that our Stackelberg game model can help the stakeholders to achieve maximum profits. Moreover, the ratios between s_1^{*1}, s_1^{*2}, and s_1^{*3} are the same as the ratios between R_1, R_2, and R_3, which confirms our results in Theorem 6.

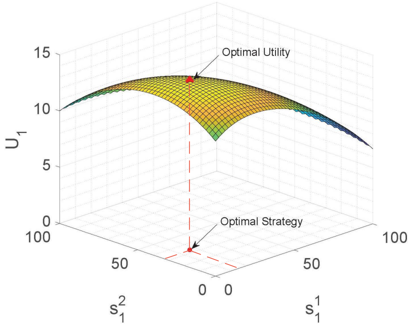


Fig. 5. Stakeholder's utility function.

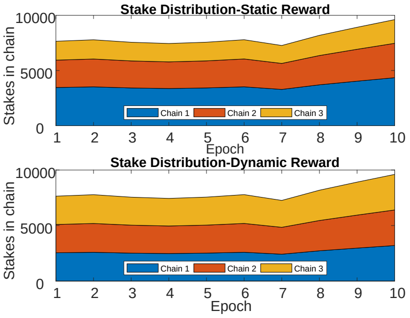


Fig. 6. Stake distribution.

5.2.2 Stake distribution

Fig. 6 illustrates the stake distribution at the end of each epoch. As can be seen from the figure, although the total number of stakes vary across the epochs, the ratio of stakes invested in each chain remains unchanged in both the dynamic and static reward schemes. Moreover, we can observe that the stakes are distributed more evenly in the dynamic reward scheme, which is more beneficial to the chains' security and performance. Furthermore, the stake ratios in both schemes equal the ratio of the block rewards, which confirms our analytical results in Theorem 6.

5.2.3 Security properties

Fig. 7 and Fig. 8 illustrate $\Pr_{\text{CP}}$ of each chain at the end of each epoch under the static and adaptive adversary settings, respectively. From the figures, we can observe that the more stakes the adversary controls, the higher chance it can violate the security of the system. For example, in the static adversary setting, with a low budget (weak adversary), $\Pr_{\text{CP}}$ is at most 0.02%, whereas this probability increases to 1.5% in case of an adversary with a high budget (strong adversary). Secondly, the total system stakes have different effects on the chains' security under the static and adaptive adversary setting. For instance, the system has the highest stakes in the last epoch. At this epoch, $\Pr_{\text{CP}}$ achieve the lowest value under the static adversary because the static adversary has a fixed budget. However, $\Pr_{\text{CP}}$ achieve the highest value under the adaptive adversary setting because the adaptive adversary can corrupt the stakeholders with the most stakes. Therefore, it is crucial to not only attract more stakes to the system but also to incentivize more diversity, i.e., encourage the stakeholders to split their stakes across more chains. We can observe the effect of such diversity between the dynamic and the static reward schemes. Although the total network stakes are the same, the dynamic scheme, which encourages equal stakes distribution, achieves much lower $\Pr_{\text{CP}}$, e.g., at most 14% compared to 24% of the static reward scheme.

Fig. 9 and Fig. 10 illustrate $\Pr_{\text{CQ}}$ of each chain under the static and adaptive adversary settings, respectively. Similar

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: Huazhong University of Science and Technology. Downloaded on January 31,2023 at 09:41:52 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Services Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSC.2023.3240235

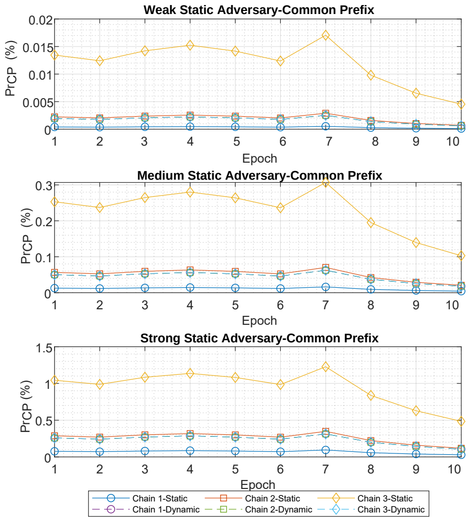


Fig. 7. $\Pr_{\mathrm{CP}}$ under static adversary settings.

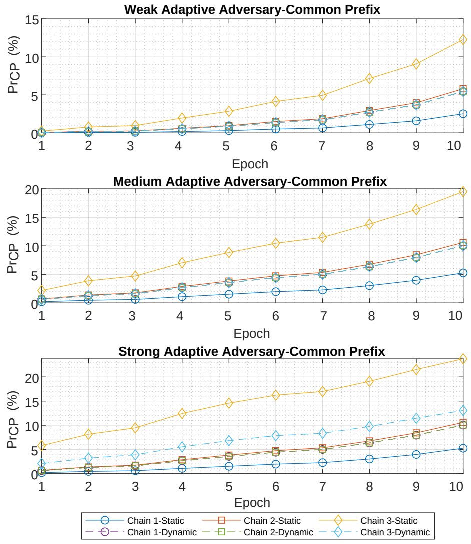


Fig. 8. $\Pr_{\mathrm{CP}}$ under adaptive adversary settings.

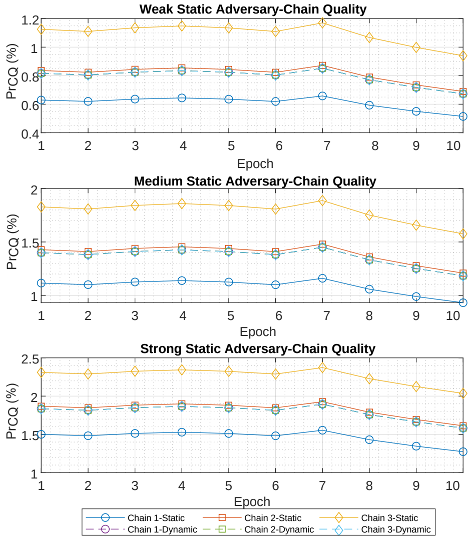


Fig. 9. $\Pr_{\mathrm{CQ}}$ under static adversary settings.

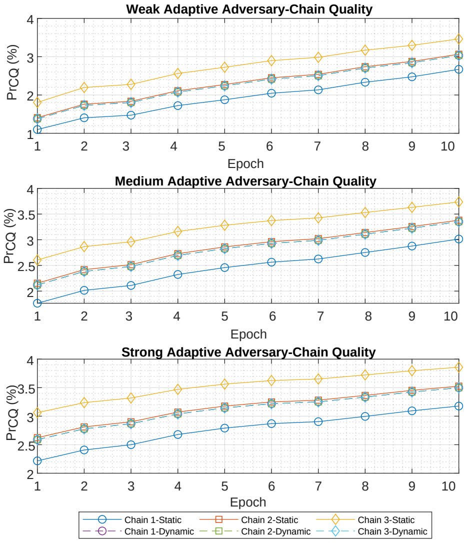


Fig. 10. $\Pr_{\mathrm{CQ}}$ under adaptive adversary settings.

to the $\Pr_{\mathrm{CP}}$, we can draw several conclusions from examining $\Pr_{\mathrm{CQ}}$. Firstly, the stronger the adversary is, the higher chance it violates system security. For example, in the weak adaptive adversary scenario, $\Pr_{\mathrm{CQ}}$ is at most 1.2%, whereas this probability increases to 2.4% in the case of a strong adaptive adversary. Generally, $\Pr_{\mathrm{CQ}}$ gets higher in the case of the adaptive adversary. The reason is that according to the simulation setting, the adversary can corrupt more stakes compared to B_A in the case of the static adversary. Secondly, similar to the results of $\Pr_{\mathrm{CP}}$, $\Pr_{\mathrm{CQ}}$ is inversely proportional to the total system stakes in the case of the static adversary, and it is proportional to the total system stakes in the case of the adaptive adversary. As a result, we can observe that the dynamic scheme achieves lower $\Pr_{\mathrm{CQ}}$, e.g., at most 14% $\Pr_{\mathrm{CQ}}$ compared to 24%. Moreover, since the security of the system is only as good as that of its weakest chain (especially with the SPV proof mechanism), it can be observed that the dynamic reward scheme achieves better security compared to the static reward scheme, i.e., the chains of the dynamic reward scheme always achieve better $\Pr_{\mathrm{CP}}$ and $\Pr_{\mathrm{CQ}}$ compared to those of the weakest chain under the static reward scheme (i.e., Chain 3).

5.2.4 Performance properties

Fig. 11 and Fig. 12 illustrate the transaction confirmation time of each chain under the static and adaptive adversary settings, respectively. From the figures, we can observe that the stronger the adversary is, the more it can negatively affect the system performance. For example, the chains takes at most 120 seconds to confirm a transaction in case of a weak static adversary, but it takes up to 220 seconds in case of a strong static adversary. This is because the transaction confirmation time is directly related to $\Pr_{\mathrm{CP}}$. A stronger adversary has a higher chance to violate the CP property, and thus the users have to wait longer to confirm a transaction. Moreover, we can also observe that the transaction confirmation time is inversely proportional

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: Huazhong University of Science and Technology. Downloaded on January 31,2023 at 09:41:52 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Services Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSC.2023.3240235

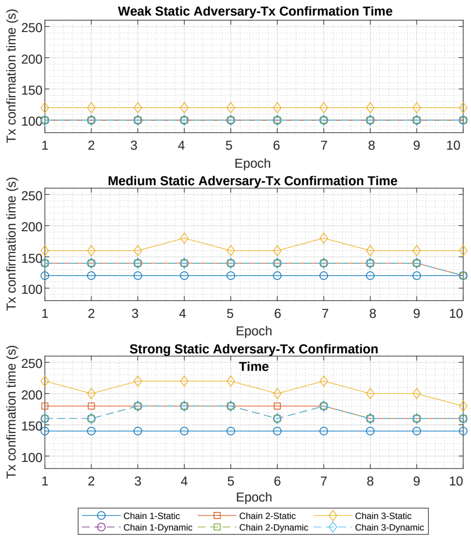


Fig. 11. Transaction confirmation time under static adversary settings.

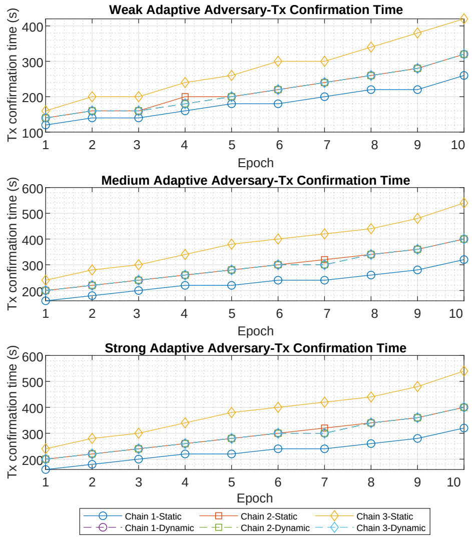


Fig. 12. Transaction confirmation time under adaptive adversary settings.

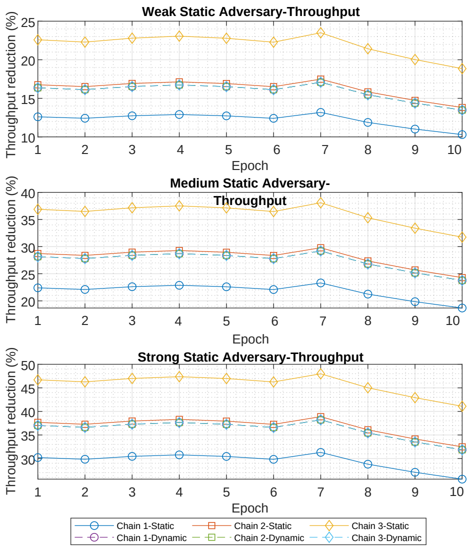


Fig. 13. Transaction throughput under static adversary settings.

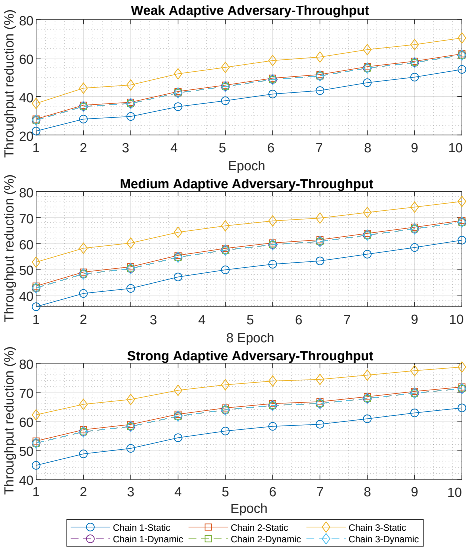


Fig. 14. Transaction throughput under adaptive adversary settings.

to the total stakes of the system in the static adversary settings, whereas the opposite holds true in the adaptive adversary settings. The reason is the same as that of the $\Pr_{\text{CP}}$ scenarios, i.e., the adaptive adversary can corrupt more stakes, whereas B_A of the static adversary is fixed. Furthermore, the transaction confirmation time of the three chains under the dynamic reward schemes is always better than at least two chains under the static reward scheme.

Fig. 13 and Fig. 14 illustrate the transaction throughput reduction percentages of each chain under the static and adaptive adversary setting, respectively. Similar to the previous scenarios, we can observe that a stronger adversary can cause more negative impacts on the system performance, e.g., a weak static adversary can reduce the throughput by at most 24%, whereas the strong static adversary can reduce the throughput by nearly 50%. Moreover, it can be observed that as the system has more stakes, the static adversary becomes weaker, whereas the adaptive adversary becomes stronger, similar to the previous scenario. Finally, one can observe that the performances of the three chains in the dynamic scheme are better than those of at least two chains in the static scheme.

6 Conclusion

In this paper, we have introduced FedChain, an effective framework for federated-blockchain systems together with a cross-chain transfer protocol to facilitate the secure and decentralized transfer of tokens between the blockchains. In this framework, we have proposed a novel consensus mechanism which can satisfy the CP, CG, and CQ properties, prevent various blockchain-specific attacks, and achieve better transaction confirmation time compared to existing consensus mechanisms. Robust theoretical analyses have been then conducted to prove FedChain's consensus mechanism security and performance properties. After that, a

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: Huazhong University of Science and Technology. Downloaded on January 31,2023 at 09:41:52 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Services Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSC.2023.3240235

Stackelberg game model has been developed to examine the interactions between the stakeholders and the blockchains managed by chain operators. This model can provide additional profits for the stakeholders and enhance the security and performance of the blockchains. Through analyses of the Stackelberg game model, we can prove the uniqueness of the Stackelberg equilibrium and find the exact formula for this equilibrium. These results are especially important for the stakeholders to determine their best investment strategies and for the chain operators to design the optimal policy, i.e., block rewards. Finally, extensive experiments and simulations have been conducted to show that our proposed framework can help stakeholders to maximize their profits and the chain operator to design appropriate parameters to enhance FedChain's security and performance.

References

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

[2] Y. Xiao, N. Zhang, W. Lou and Y. T. Hou, "A survey of distributed consensus protocols for blockchain networks," *IEEE Communications Surveys & Tutorials*, vol. 2, no. 22, pp. 1432-1465, Jan, 2020.

[3] 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," *IEEE Access*, vol. 7, pp. 85727–85745, June 2019.

[4] S. R. Pokhrel and J. Choi, "Federated learning with blockchain for autonomous vehicles: Analysis and design challenges," *IEEE Transactions on Communications*, vol. 68, no. 8, pp. 4734-4746, Apr. 2020.

[5] H. Chen, S. A. Asif, J. Park, C. C. Shen and M. Bennis, "Blockchained Federated Learning with Model Validation and Proof-of-Stake Inspired Consensus," Jan. 2021, *arXiv:2101.03300*.

[6] "Global Charts," *CoinMarketCap*. [Online]. Available: https://coinmarketcap.com/charts/. Accessed on: 04-Nov-2020.

[7] "Buy & sell Crypto in minutes," *Binance*. [Online]. Available: https://www.binance.com/en. Accessed on: 04-Nov-2020.

[8] "Bitcoin & Cryptocurrency Exchange," *Kraken*. [Online]. Available: https://www.kraken.com/. Accessed on: 04-Nov-2020.

[9] Selfkey, "A Comprehensive List of Cryptocurrency Exchange Hacks," *Selfkey*. [Online]. Available: https://selfkey.org/list-of-cryptocurrency-exchange-hacks/. Accessed on: 04-Nov-2020.

[10] A. Back *et al.* (Oct. 2008). Enabling blockchain innovations with pegged sidechains. [Online]. Available: http://kevinriggen.com/files/sidechains.pdf

[11] A. Singh *et al.*, "Sidechain technologies in blockchain networks: An examination and state-of-the-art review," *Journal of Network and Computer Applications*, vol. 149, Jan. 2020, doi: https://doi.org/10.1016/j.jnca.2019.102471

[12] V. Arasev. (Sep. 2018). POA network whitepaper. [Online]. Available: https://github.com/poanetwork/wiki/wiki/POA-Network-Whitepaper

[13] J. Dilley *et al.*, "Strong federations: an interoperable blockchain solution to centralized third-party risks," 2016, *arXiv:1612.05491*.

[14] S. Lerner. (Jan. 2016). Drivechains, Sidechains and Hybrid 2-way Peg Designs [Online]. Available: https://docs.rsk.co/Drivechains_Sidechains_and_Hybrid_2-way_peg_Designs_R9.pdf.

[15] G. Wood, "Ethereum: A secure decentralised generalised transaction ledger," Ethereum Project, Zug, Switzerland, Yellow Paper EIP-150 Rev., Aug. 2017, vol. 151.

[16] A. Garoffolo and R. Viglione, "Sidechains: Decoupled consensus between chains," 2018, *arXiv:1812.05441*.

[17] P. Gaži, A. Kiayias and D. Zindros, "Proof-of-Stake Sidechains," in *2019 IEEE Symposium on Security and Privacy (SP)*, San Francisco, CA, USA, May 18-19, 2019, pp. 139-156.

[18] A. Garoffolo, D. Kaidalov and R. Oliynykov, "Zendoo: a zk-SNARK verifiable cross-chain transfer protocol enabling decoupled and decentralized sidechains," 2020, *arXiv:2002.01847*.

[19] W. Sun, J. Liu, Y. Yue and P. Wang, "Joint Resource Allocation and Incentive Design for Blockchain-Based Mobile Edge Computing," *IEEE Transactions on Wireless Communications*, vol. 19, no. 9, pp. 6050-6064, Sept. 2020.

[20] J. Kang *et al.*, "Incentivizing Consensus Propagation in Proof-of-Stake Based Consortium Blockchain Networks," *IEEE Wireless Communications Letters*, vol. 8, no. 1, pp. 157-160, Feb. 2019.

[21] M. S. Iftikhar, N. Javaid, O. Samuel, M. Shoaib and M. Imran, "An Incentive Scheme for VANETs based on Traffic Event Validation using Blockchain," *2020 International Wireless Communications and Mobile Computing (IWCMC)*, Limassol, Cyprus, 2020, pp. 2133-2137.

[22] J. Kang, Z. Xiong, D. Niyato, S. Xie and D. I. Kim, "Securing Data Sharing from the Sky: Integrating Blockchains into Drones in 5G and Beyond,"*IEEE Network*, vol. 35, no. 1, pp. 78-85, Feb. 2021.

[23] Z. Xiong, J. Kang, D. Niyato, P. Wang and H. V. Poor, "Cloud/Edge Computing Service Management in Blockchain Networks: Multi-Leader Multi-Follower Game-Based ADMM for Pricing," *IEEE Transactions on Services Computing*, vol. 13, no. 2, pp. 356-367, Apr. 2020.

[24] S. Guo, Y. Dai, S. Guo, X. Qiu and F. Qi, "Blockchain Meets Edge Computing: Stackelberg Game and Double Auction Based Task Offloading for Mobile Blockchain," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 5, pp. 5549-5561, May 2020.

[25] C. T. Nguyen, D. N. Nguyen, D. T. Hoang, H. Pham, N. H. Tuong and E. Dutkiewicz, "Blockchain and Stackelberg Game Model for Roaming Fraud Prevention and Profit Maximization," *2020 IEEE Wireless Communications and Networking Conference (WCNC)*, Seoul, South Korea, 2020, pp. 1-6.

[26] 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.

[27] P. Daian, R. Pass, E. Shi, "Snow white: Robustly reconfigurable consensus and applications to provably secure proof of stake," in *International Conference on Financial Cryptography and Data Security*, Saint Kitts and Nevis, Feb. 18-22, 2019, pp. 23-41.

[28] V. Buterin and V. Griffith, "Casper the friendly finality gadget," 2017, *arXiv:1710.09437*.

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

[30] 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.

[31] 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.

[32] 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.

[33] J. Garay, A. Kiayias and N. Leonardos "The bitcoin backbone protocol: Analysis and applications," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Sofia, Bulgaria, April 26-30, 2015, pp. 281-310.

[34] S. Nakamoto, (May 2008). "Bitcoin: A peer-to-peer electronic cash system". [Online]. Available: https://bitcoin.org/bitcoin.pdf

[35] 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.

[36] L. Luu, D. Chu, H. Olickel, P. Saxena, and A. Hobor, "Making smart contracts smarter," in *Proc. of the ACM SIGSAC Conference on Computer and Communications Security*, Vienna, Austria, Oct. 2016, pp. 254-269.

[37] Cardano, "Ouroboros Proof of Stake Algorithm," *Cardano*. [Online]. Available: https://cardanodocs.com/cardano/proof-of-stake/.

[38] MathWorks, "Fmincon," *Mathworks*. [Online]. Available: https://www.mathworks.com/help/optim/ug/fmincon.html.

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

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

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: Huazhong University of Science and Technology. Downloaded on January 31,2023 at 09:41:52 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Services Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSC.2023.3240235

[41] Y. Li *et al.*, C. Chen, N. Liu, H. Huang, Z. Zheng and Q. Yan, "A blockchain-Based decentralized Federated Learning framework with committee consensus," *IEEE Network*, vol. 35, no. 1, pp. 234-241, Jan. 2021.

[42] A. Haque, M. S. Ghani and T. Mahmood, "Decentralized Transfer Learning using blockchain & IPFS for Deep Learning," in *International Conference on Information Networking (ICOIN)*, Barcelona, Spain, Jan 7-10, 2020, pp. 170-177.

[43] C. T. Nguyen *et al.*, "Transfer Learning for wireless networks: A comprehensive survey,"*Proceedings of the IEEE*, vol. 110, no. 8, pp. 1073-1115, Aug. 2022.

[44] R. M. Parizi, S. Homayoun, A. Yazdinejad, A. Dehghantanha and K. -K. R. Choo, "Integrating privacy enhancing techniques into blockchains using sidechains," in *2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE)*, Edmonton, AB, Canada, May 5-8, 2019, pp. 1-4.

[45] StakingRewards, "Cardano," *Digital Asset Research Platform for Staking & Dividends*. [Online]. Available: https://stakingrewards.com/asset/ada. [Accessed: 16-Aug-2020].

[46] StakingRewards, "Algorand," *Digital Asset Research Platform for Staking & Dividends*. [Online]. Available: https://stakingrewards.com/asset/algo. [Accessed: 16-Aug-2020].

[47] StakingRewards, "Tezos," *Digital Asset Research Platform for Staking & Dividends*. [Online]. Available: https://stakingrewards.com/asset/xtz. [Accessed: 16-Aug-2020].

[48] J. B. Rosen, "Existence and Uniqueness of Equilibrium Points for Concave N-Person Games," *Econometrica*, vol. 33, no. 3, pp. 520-534, July 1965.

[49] C. T. Nguyen *et al.*, "Blockchain-based Secure Platform for Coalition Loyalty Program Management," *2021 IEEE Wireless Communications and Networking Conference (WCNC)*, Nanjing, China, Mar. 29 - Apr.1 , 2021, pp. 1-6.

[50] S. Gupta, J. Hellings, M. Sadoghi, "Fault-Tolerant distributed transactions on blockchain," *Synthesis Lectures on Data Management*, vol. 16, no.1, pp.1-268, Feb. 2021.

[51] G. Wood. (Nov. 2016). POLKADOT: Vision for a heterogeneous multi-chain framework. [Online]. Available: https://polkadot.network/PolkaDotPaper.pdf

[52] J. Kwon and E. Buchman. (2019). Cosmos Whitepaper. [Online]. Available: https://v1.cosmos.network/resources/whitepaper

[53] M. Zamani, M. Movahedi and M. Raykova "Rapidchain: Scaling blockchain via full sharding," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, Toronto, Canada, Oct. 15-19, 2018, pp. 931-948.

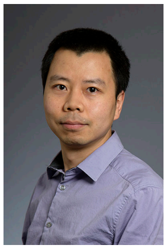
[54] M. Herlihy, B. Liskov and L. Shrira, "Cross-chain deals and adversarial commerce," *VLDB Journal*, Aug. 2021, https://doi.org/10.1007/s00778-021-00686-1.

[55] V. Zakhary, D. Agrawal, A. El Abbadi, "Atomic commitment across blockchains", 2019, *arXiv:1905.02847*

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.

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.

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.



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).

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. 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.

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.

Huynh Tuong Nguyen Dr. Huynh Tuong Nguyen, a faculty member at Ho Chi Minh City University of Technology, is an expert in algorithms and resolutions for real-life problems including: manufacturing scheduling, transportation problems, and cryptography. He holds a PhD in Computer Science from Francois 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.

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: Huazhong University of Science and Technology. Downloaded on January 31,2023 at 09:41:52 UTC from IEEE Xplore. Restrictions apply.