Contract Theory Framework for Cryptoeconomics

by
Jing Li

A dissertation submitted to the Department of Electrical and Computer Engineering,

College of Cullen Engineering

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in Electrical and Computer Engineering

Chair of Committee: Zhu Han

Committee Member: Miao Pan

Committee Member: Weidong Shi

Committee Member: Hien Van Nguyen

Committee Member: Dusit Niyato

Committee Member: Xiao-Ping Zhang

University of Houston
May 2022

ACKNOWLEDGMENTS

I am indebted to my advisor, Dr. Zhu Han, for his continued patience, guidance, and assistance through each stage of my research. I have benefited greatly from his profound knowledge and invaluable insights. I would like express my deep gratitude to Dr. Zhitao Guan, my Master advisor from North China Electric Power University, for providing me the opportunity to start my research journey and encouraging me to go abroad for further study.

Very special thanks to Dr. Miao Pan, Dr. Weidong Shi, Dr. Xiao-Ping Zhang, Dr. Hien Van Nguyen, and Dr. Dusit Niyato, for their kind help and support on this dissertation. Further, I am so grateful to Dr. Dusit Niyato from Nanyang Technological University, for his great inspiration and collaboration throughout all my research work. Also, many thanks to Dr. Tingting Liu, for her patient instructions in the every beginning of my PhD study.

My friends Wen Xie, Kai-Chu Tsai and YingMiao Liu, deserve my thanks who provide me unconditional moral support during the covid-19 epidemic. I would like to thank Dr. Jingyi Wang, Dr. Dawei Chen for their generous help during my lives at the university.

My heartfelt thank must go to my parents and my husband for their hope and love they had given to me. It is so difficult for me to express my love and appreciation to my husband, Dr. Shi Pu. He is my best friend, the love of my life. During my intense Ph.D. study, he has sacrificed a lot and always made my research and career a top priority in our lives. His optimistic spirit accompanied me through all the darkest moments of my Ph.D. research. Thank you for the strength you gave me. I have had the great fortune to meet Mr. YuNing Liu, the finest vocalist in my mind, and I am profoundly grateful to him for taking the edge off my anxiety and stress. His songs and videos have accompanied me through every single sleepless night.

I would like to praise and thank God, who has instructed me in the way of wisdom and lead me along straight paths, so that I am able to overcome all the challenges in my life. Thank God for granting me the steady faith, infinite hope and endless love. May God bless us.

ABSTRACT

Cryptoeconomics is the research on how incentives should construct a decentralized and distributed cryptographic system. Economic incentives are used to motivate the efforts and govern the allocation of resources in the cryptoeconomic ecosystem, ensuring specific types of information security qualities. Compared to the costly and time-consuming cryptography, incentives obtained through game theory are much more cost-efficient and easier to implement. However, there lacks sufficient research on the incentive issue of cryptoeconomics.

We investigate the various incentives of blockchain networks to fill in the gaps in cryptoeconomics research. The first research focuses on the blockchain network with shards and adopts the security-deposit-based consensus protocol, studying the problem of how to balance the security incentive and the economic incentive. The contract theory is utilized to formulate the problem between temporary blockchain leaders and validators. Compared with fixed deposits, flexible deposits can provide sufficient financial incentives for the participants without losing the security incentives. In the second work, we adopt the cyber insurance idea and propose the insurance contract to help determine the withdrawal delay and the insurance claim to relieve the loss of victims. Specifically, instead of requiring the insurance premium from the validators, the cyber insurer first signs the contract with the blockchain representative (e.g., beacon chain). Then the blockchain representative would sign a series of contracts with the validators. Through the simulations, we demonstrate that the proposed model can provide adaptive insurance contracts for the different validators and keep the profits of the blockchain network and the cyber insurer. In the last work, we propose a random-contract-based scheme to maximize the service provider's revenue and assign the service buyers the feasible service price under the framework of a sidechain linked to the public blockchain. We systematically demonstrate random contracts' superiority under the increasing absolute risk-aversion assumption. The simulation results show that random contracts can provide more significant revenue for sidechains by an average of 24.70% compared to deterministic contracts. Efficient service payments can be reduced by an average of 44.65% compared to the main chain's cost.

TABLE OF CONTENTS

LIST OF FIGURES

1 Introduction

In the last decade, the blockchain technology has been considerably researched and developed by both academic communities and industrial circles [1]. A blockchain is a distributed ledger of transaction records, which is jointly maintained and supported by all participating parties [2]. The transactions can be initiated by any party in the system, and then verified by all of the participants after being recorded in the blocks of the public ledger. Due to the cryptographic design, the transaction records of the same block are organized in a Merkle Hash Tree, and all the blocks are sequentially connected before and after with each other, which makes it difficult to tamper with a single transaction record, let alone a single block. As a distributed, decentralized and tamper-resistant public ledger, blockchain has found a wide range of applications in both financial and non-financial world, e.g., the stablecoin project called Libra that is initiated by Facebook [3] and Vehicle Passport of SHIFTMobility [4].

The advent of blockchain technology benefits a wide range of areas, including finance, business, transportation, entertainment, etc. Due to the sophistication and diversity of the blockchain system, the interpretation of blockchain is manifested into two camps [5]: (1) Blockchain is a solution to computer science problems. (2) Blockchain is a solution to incentive mechanism design problems.

In earlier research of blockchain, the first camp had made enormous contributions on the technique components. It is worth noting that blockchain is not a single new technology but a combination of multiple technologies. The springing up and brisk developing of computer and network technologies directly promoted the blockchain evolution [6]. The concept of blockchain was first outlined in 1991 [7]. Stuart Haber and W. Scott Stornetta presented the fundamental notion of blockchain, a chain of hashed records, to address the time-stamping problem, the embryonic form of data structure in the blockchain, while not defining the name "blockchain." Then, in 1997, Adam Back created Hashcash [8], a Proof of Work mechanism for email anti-spam and anti-DoS (denial of service) that has since formed the foundation for most cryptocurrency projects. A brief history of blockchain is shown in Fig. 1.

Until the advent of Bitcoin in January 2009, that blockchain had its first real-world

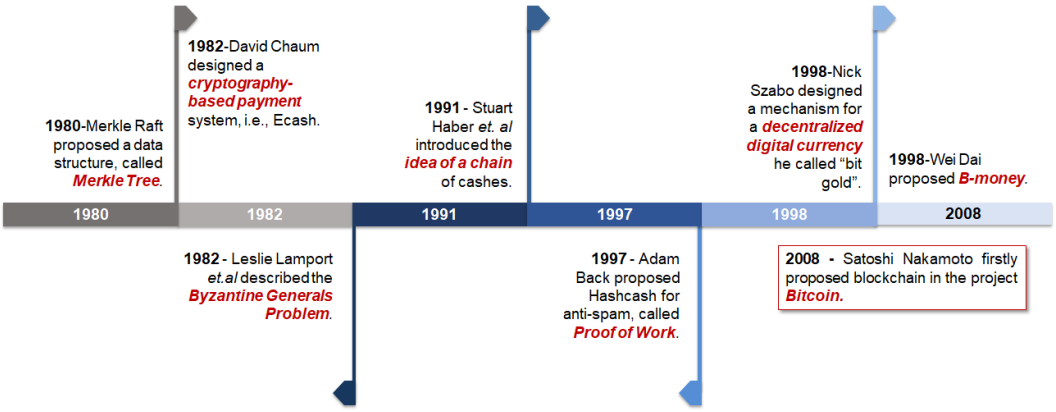


Figure 1: Origins of Blockchain.

application. According to bitcoin's white paper [2], "Bitcoin: A peer to peer Electronic currency system," Satoshi Nakamoto adopted cryptography and Proof-of-Work in the Bitcoin network to ensure data security and consistency and introduced incentive mechanisms to allow transactions to be completed without the involvement of a third party, ushering in a new era of decentralization.

To better understand the second camp, we first give a rough description of blockchain technical components from the first camp perspective. As a collection of technologies, blockchain is composed of a bunch of interdependent and interrelated components. To facilitate the description, we adopt the recognized five-layer architecture (shown in Fig. 2), and explain the components and layers in the following.

- **Data Layer:** The data structure of most conventional blockchains is described as a linked chain of blocks in which transactions are organized in a sequential manner. In addition to Bitcoin, various PoW-based blockchain projects also use chain structure to manage the blocks. Each block of Bitcoin includes a Merkle Hash tree [9], which is a tree-based data structure of transaction hashes. The root hash of tree is obtained by calculating the leaf nodes, which refer to the hashes of transaction data.

  The tree-based data structure uses hash function to ensure the transaction data security and integrity. To verify the authenticity of transactions, a digital signature [10] is required before sending any transaction. Just as all kinds of cryptocurrency

projects' names indicate, 'crypto-' means that these projects must rely on cryptography for security. Bitcoin uses an elliptical curve cryptography to generate the public keys for users.

All the data and information on the Bitcoin are duplicated and scattered in different nodes throughout the network. For the full nodes, they store the metadata in LevelDB [11]. Some other blockchain projects, such as Ethereum [12], also utilize LevelDB to keep all the data.

- **Network Layer:** The network layer refers to the network model, network routing protocols, as well as some other communication protocols. The public blockchain is built upon a peer-to-peer (P2P) network, in which each node joins by connecting to some other nodes. A P2P network consists of a group of computers/client that communicates with each other. The term "peer" means that every node is treated equally in the network. There is no centralized parties with nodes. The peer nodes serve both as providers of resources and as consumers of services. The distributed and decentralized P2P network is the foundation of blockchain.

  The network layer functions are almost carried out by P2P network. In addition to mining and transferring value, some blockchain nodes also have the same functions with P2P nodes. To participate in the network, all nodes must have the routing function [6], which help them to share information with each other. The most often used unstructured P2P Network Protocol is the Gossip Protocol [13]. After a miner generates a block, the others will broadcast the result and block via a gossip protocol. Bitcoin changed the way it distributed gossip messages in 2015 to improve privacy. It currently employs a technique known as "diffusion" [14]. Another well-known communication protocol is called Kademlia (KAD) protocol [15]. The KAD network refers to a P2P network that implements the KAD protocol. As a more efficient protocol, some blockchain projects adopt KAD network as their network layer to enable the blocks and transactions transmission optimization [16].

- **Consensus Layer:** The consensus layer specifies the rules for nodes to reach

an agreement on blockchain's state. The popular consensus protocols include the Proof of Work (PoW) in Bitcoin and the Proof of Stake (PoS) in Ethereum. To illustrate the consensus layer, we discuss the PoW algorithm here as it is the most common algorithm for permissionless blockchains, used by Bitcoin [2]. Take PoW as an example, the protocol determines who is eligible to create a new block, the time slot between two contiguous blocks, and stipulates all nodes to work on the longest chain. Different consensus algorithms employ various principles to determine the rules for nodes based on their actual needs. Ethereum is now experiencing the transition from PoW to PoS since PoW is energy intensive and costly [17]. Unlike PoW's elite hardware requirements, PoS only needs participants to stake some cryptocurrencies to the main chain. The creator of each block is selected randomly, similar to the miners of PoW, responsible for finalizing transactions and working on the longest chain [18]. Some other protocols can be referred to Proof of Authority [19] and Byzantine Agreement [20].

- **Incentive Layer:** The incentives layer establishes an incentive system using the blockchain's cryptocurrency. In the initial design of Bitcoin, the incentives refer to block reward and transaction fee, incentivizing the miners to work on the longest chain and encouraging the other participants to finalize the blockchain's ledger. From the Bitcoin project it is clear that any permissionless blockchain system requires an incentive strategy to keep it running. Miners should be appropriately compensated for their effort, and incentives should push them to act honestly. As the nexus to bind the different technologies to form the blockchain, incentives are apparently less discussed by the first camp and, however, have been emphasized and characterized by the second camp. Ethereum proposed the concept of the security deposit in their Casper protocol [21], which will be one of the most essential incentive components in Ethereum 2.0. According to the latest update of staking deposit requirements on the Ethereum website, everyone needs to stake some ethers to the network before joining the Ethereum [22]. Moreover, anyone who wants to become a full validator must stake 32 ethers. Another crucial incentive is the voting privilege, which is less used in PoW-based blockchain but

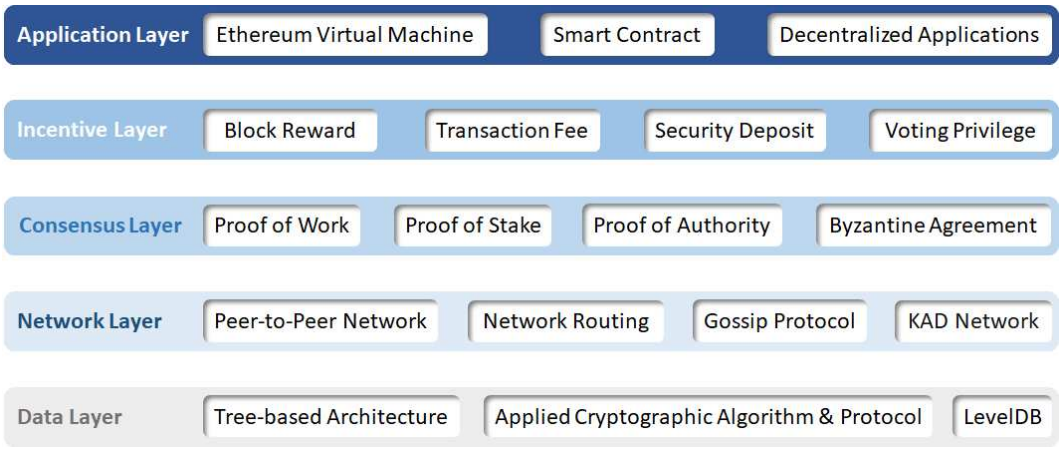


Figure 2: Five Layers of Blockchain Technology.

indispensable to PoS-based blockchain. The reason is that PoS protocols rely on voting mechanisms to reach consensus [23].

- **Application Layer:** The application layer includes Ethereum Virtual Machine, smart contracts, decentralized apps (DApps), and so on. The Ethereum Virtual Machine is a software framework that allows developers to construct Ethereum-based decentralized applications (DApps). All Ethereum accounts' data and smart contracts codes are stored on this virtual machine [24]. Similar to the Ethereum accounts (a.k.a., Externally Owned Account), smart contracts are also a type of account (Contract Account). It is a collection of code with a unique address/account, which can be created by any developers and can operate automatically on Ethereum [25]. Dapps that interface with the blockchain network makes up the most important part of the application layer. These Dapps interoperate with the blockchain network via APIs. Unlike the traditional apps (centralized apps), the DApps run on a decentralized network environment and often require the users to interact with the developer's smart contract to get the download permits. Applications can send instructions to all the underlying layers, which enables all layers to cooperate with each other to perform more advanced functions.

The first camp is much bigger than the second one. One main reason is that many techniques of blockchain have existed for decades. The researchers in computer science have an abundance of reference literature and research experiences due to years of accumulation. There are only a handful of literatures for researchers, engineers, and

graduate/undergraduate students to theoretically understand the dynamics in economic incentives of blockchain networks from a game-theoretic perspective. The computer science content determines the existence of a blockchain framework. Simultaneously, the incentive mechanism improves the system performance by regulating participants' behaviors and coordinating all operations through the costs and benefits. Researchers have gradually realized the role of economic incentives in decentralized and distributed systems.

1.1 Background and Motivation

Ethereum founder Vlad [26] first defined Cryptoeconomics as "A formal discipline that studies protocols that govern the production, distribution, and consumption of goods and services in a decentralized digital economy." The other versions of definition are listed as follows: "Cryptoeconomics is the application of incentive mechanism design to information security problems." Vlad identified that cryptoeconomics has played a crucial role in distributed systems, that is to encourage more entries and to incentivize the desired behaviours [5]. Vitalik [27] expounded on the concept as "Cryptoeconomics is about building systems that have certain desired properties, use cryptography to prove properties about messages that happened in the past, use economic incentives defined inside the system to encourage desired properties to hold into the future." For Josh Stark [28], he proposed that "Cryptoeconomics is the practical science of using economic mechanisms to build distributed systems, where the financial incentives guarantee essential properties of that system and where the economic mechanisms are guaranteed by cryptography."

Bitcoin is the first as well as the most significant instance of cryptoeconomics [2]. The integration of cryptography and mechanism design sparks a revolutionary shift from a traditional P2P network to a blockchain network. Although cryptography is robust when assuring the security and privacy of a system, the cost of development and deployment is increasingly expensive due to the unpredictable attacks and risks. By issuing tokens and offering fees [29], the bitcoin network incentivizes trustless participants to operate as required. It compensates miners' work on the longest chain, coordinating all

parties to defend against the Sybil attack [30] and preserving the system's security and stability. Bitcoin uses cryptoeconomics to solve two problems: 1) the security problem is that how to resist the Sybil attack; 2) the incentive problem is that how to motivate the unknown participants to participate correctly. Specifically, we can learn the uses of incentives according to Vitalik's talk [31]: 1) Rewards: increase actors' token balances if they do something good, e.g., block reward and transaction fee. 2) Penalties: reduce actors' token balances if illegal behavior occurs, e.g., security deposit. 3) Privileges: incentivize participants by giving them decision-making right, e.g., voting weight. Consequently, we conclude the interpretation of cryptoeconomics concept in two folds: 1) It provides the theoretical interpretation, from the perspective of untrusted economic networks, of the consensus protocols assisted by cryptographical functionalities in decentralized blockchain networks regarding the activities of the entities in the network and the dynamics of the network as a whole. 2) It extends the analytical framework based on the economic networking point of view to modeling, designing, and analyzing the participant interactions in any ecosystem that is extended from or build upon blockchain networks.

Therefore, on one hand, the behavioral analysis from the economic perspective of the blockchain networks answers a series of fundamental questions regarding cryptography and distributed system security. Such an analytical approach plays a vital role in designing appropriate protocols in digital ledger networks, especially for those built upon massive peer-to-peer networks without an explicit governance infrastructure. On the other hand, from the engineering perspective, a well-functioning, scalable cryptoeconomic network is able to serve as an efficient platform for decision arbitration and allocation of the resources ranging from physical utilities (e.g., hardware) to financial assets, and more broadly, various conceptual resources including data, trust and social attention (e.g., votes). As a result, the convergence of computer networking, cryptography and economic theory sheds light on better characterization of the decentralized/self-organized systems particularly relying upon the advance of the blockchain technologies. This, in return, requires a comprehensive study of the technical building blocks, such

as consensus protocols, incentive mechanisms, cryptographic and networking functionalities, and all the related primitives from an interdisciplinary perspective.

Compared with the abundant research of camp one, scant attention has been paid to the incentive mechanism design for distributed systems. Due to this reason, this dissertation aims to fill the research gap from a game-theoretic perspective. This dissertation studies the incentive pricing issues in the different blockchain-based ecosystems, including security deposit, online reward and transaction fee. To obtain the optimal incentives, this dissertation exploits the contract theory framework to formulate the problems.

In the next section, this dissertation reviews some existing incentive mechanisms and projects in cryptoeconomics. Specifically, this dissertation provides a comprehensive overview of research efforts from industrial and academic areas and highlights the research gaps that this dissertation intends to address.

1.2 Overview of cryptoeconomics and mechanism design

As for the cryptoeconomics, what should we expect from the outcome of cryptoeconomics? What is the outcome criterion of cryptoeconomics research? According to Ethereum Foundation's talk [32], a good outcome should meet the following requirements: 1) Easier Exchange: The incentive must have good liquidity and can be liquidated very quickly. 2) Trustless Trades: The trades can be processed without a third party. 3) Liquidity for small markets: The liquidity of outcomes should not be limited or impacted by market size.

A key conclusion of the talk is that if the mechanism cannot help the system know your customers (KYC) better and always lead to the rich getting richer, then a bad outcome would always exist. Unfortunately, a considerable number of research using game theory and economics theory cannot achieve the fairness as expected. A better mechanism should be as simple as possible, therefore reducing the dependency on parameter selection. Moreover, the mechanism must be feasible and easy to implement in a distributed and decentralized system.

As blockchain technology evolves, more and more researchers have been making

Model	Parameters	Security Margin
Honest Majority	Honest users are more than or equal to 2/3 of total users.	0.5
Uncoordinated Majority	The coordinated users account for less than ½ of total users.	0.25
Coordinated Majority	The coordinated users account for up to 100% of total users.	0
Bribing Attacker	Budget > (block reward + tx_fees) * number_of_blocks	0

Figure 3: A new interpretation of Bitcoin.

strides in academia research and commercial applications, demonstrating that cryptoeconomics promises to provide sufficient stability, persistence and robustness. The success of cryptoeconomics applications corroborates Vitalik's point and enriches the related conceptual architecture.

As a decentralized system that embedded with an incentive mechanism, Bitcoin is the canonical example of a cryptoeconomics application. The cryptoeconomics resource of Bitcoin refers to the hashing power which is required by Proof of Work. Bitcoin is apparently grounded on the cryptoeconomics assumptions. Bitcoin is acknowledged as the first peer-to-peer digital currency payment system due to its underlying technology, which refers to a decentralized and distributed database comprising a various components. As the first application of cryptoeconomics system: 1) Bitcoin uses Proof of Work consensus to resist Sybil attack. 2) Bitcoin uses block rewards and transaction fees to compensate miners for their effort, and to incentivize them to work on the longest chain. 3) Most of Bitcoin's users are honest and rational. 4) Bitcoin has no access restriction. Anyone can enter or exit the network at any time. 5) Any two of Bitcoin users can communicate with each other relatively quickly. 6) Users in Bitcoin network are anonymous, and there is no way to restore users' real identities. Fig. 3 describes the interpretations of Bitcoin project under the cryptoeconomics analysis framework [27].

Another canonical example of cryptoeconomics is Schelling Coin. Schelling Coins are decentralized oracle construction. Its underlying mechanism relies on a game-theoretic concept known as Schelling points, which is proposed by Thomas Schelling in his book, *The Strategy of Conflict* [33]. The way it works is as follows. Suppose two strangers are in different rooms and have not communicated beforehand. They need to pick up the same number from a set of numbers: **10000 34592 45183 40569 857**. If success, both

of them will get rewards. Otherwise, they will be punished. In theory, each number has the same probability to be selected. However, in practice, the probability of selecting **10000** is far more greater than the others. The reason is that **10000** looks much more special than the others. The uniqueness results in a natural convergence point.

In addition to these two cases, some other well-known cryptoeconomics instances have also been widely explored. Most of the projects focus on the efficiency issue. The main reason for this is that the scalability problem has inhibited the prospects of blockchain development. This chapter introduces two of the most representative ones. Arbitrum technology is one of the potential solutions to Ethereum's recent transaction cost problem, which is proposed by Offchain Labs [34]. Arbitrum is able to lower transaction costs and traffic congestion by transferring the data and transaction from Ethereum's primary network (layer 1) to a secondary network (layer 2). Layer 2 scaling solutions are used to store data outside of Ethereum's blockchain, are believed to be critical solutions for Ethereum's scaling problem.

The other instance is Keepers, which is proposed by ChainLink, is a decentralized network that allows developers and researchers to develop and monitor smart contracts, reducing the latency, increasing the process efficiency and reserving the computation resources. Instead of competing with each other, nodes in Keepers are incentivized to perform all registered jobs [35]. DeFi protocols like bZx [36] and xToken [37] have integrated Keepers to enhance functionality and improve user experience without compromising security and privacy.

However, even the published industrial projects lack ingenious incentives design. Offchain Lab is still working on the token-based incentives [34]. Keeper also exists the costly and unpredictable bounties issues [38]. Lots of research is exploring feasible incentive designs in the context of cryptoeconomics.

As for the academic research, there are some existing research work designing the economic incentive for a blockchain network using various game models. Game theory, as a branch of economics, focuses on mathematical models and studies the interactions among rational participants [39]. It is advantageous in mechanism design, benefiting the

designer, and offering feasible mechanisms for rational game players, such as Evolutionary Game [40], Stackelberg Game [41] and Contract Theory [42]. Liu *et al.* studied the dynamics of mining pool selection in [40]. Considering the hash rate for puzzle-solving and block propagation delay, this work worked out the strategy of mining pool selection. Feng *et al.* employed a cyber-insurance framework in the blockchain-based service market in [41]. They modeled the interaction among the entities as a two- stage Stackelberg game and obtained the optimal pricing strategies. Kang *et al.* designed a comprehensive incentive mechanism for a blockchain-enabled Internet of Vehicles in [43]. Specifically, they used contract theory to model the interactions of miners and finally obtained the optimal rewards for all participants.

In particular, contract theory is a study of economic incentive design, especially in the presence of asymmetric information [44]. Some existing research works on investigating the incentive mechanism design in the engineering area, include wireless communication [45, 46, 47], mobile network [48, 49, 50], and blockchain network [51, 52]. Liu *et al.* proposed a contract model to determine the service prices in a two-layer wireless caching network in [45], ensuring the network service provider's profits and compensating for the participants' cost. In [46], Asheralieva *et al.* combined the contract theory and Lyapunov optimization to design an optimal content sharing scheme in a wireless content delivery network. Based on the combined model, this work enabled to lower down the time cost and determine the optimal policy in the queuing system. Gao *et al.* adopted the contract model to design the time resource assignment and price the corresponding fees in Backscatter-Assisted RF-Powered Netowrks [47]. In [48] and [49], Liu *et al.* developed a contract model in the small-cell caching system and edge caching with 5g wireless network, respectively, analyzing the interactions between the different parties involved in the market and obtaining the optimal service fees and the profits. In [50], Kang *et al.* proposed an effective incentive mechanism by using contract theory to motivate high-reputation mobile devices with high-quality data to participate in model learning. Li *et al.* devised a security deposit pricing scheme using the joint models under the contract theory framework in [51]. This work can provide sufficient economic incentives without reducing the security incentive. In [52], Su *et al.* proposed an energy allocation

mechanism using contract theory in the energy blockchain that facilitated with electric vehicles, maximizing the operator's profit while satisfying the vehicle users' demands.

The rest of this dissertation is organized as follow. Chapter 2 provides a security deposit pricing scheme by using the joint models under the contract theory framework. Chapter 3 considers the potential attacks which may occur in the Proof-of-Stake based blockchain, establishing a cyber-insurance framework using a hierarchical game model. Chapter 4 determines the transaction fee prices on the sidechain using the Geometric Brownian Motion and random contract model.

2 Security Deposits Pricing with Contract Theory

In this chapter, we introduce a flexible pricing scheme for security deposits. Considering the limited budgets of internet of things, we propose two joint models under the contract theory framework to determine the optimal price of different types of security deposits.

2.1 Introduction

As a network of connected devices, Internet of Things (IoT) is capable of collecting, sharing and analyzing data with each other [53]. Specifically, IoT is the collection of technologies, including identification and tracking technologies, wired and wireless sensor networks, machine-to-machine interfaces, enhanced communication protocols and so on. Due to the flexible and powerful data management, there are multiple industrial solutions based on IoT, e.g., IBM Industrial 4.0 [54] and Ericsson Industrial 4.0 [55]. However, the traditional Internet of Things system has limited scalability and single point of failure issues because of its centralized architecture. With the advent of blockchain technology, the combination of IoT and blockchain has a promising application in the future.

2.1.1 Related Work

As the development of industry, the blockchain-based IoT platform has been a research focus. Due to the intensive computation required by Proof of Work, IoT nodes, as the computationally lightweight nodes, are unable to participate in mining. There are multiple related researches regarding the resource management and allocation. In order to reduce the cost and increase the availability of data management, Feng *et al.* [56] construct a decentralized platform for data storage and trading in a wireless powered IoT crowd-sensing system. Asheralieva *et al.* [57] introduce mobile-edge computing and unmanned aerial vehicles to help relay and run the blockchain tasks that include the data collected from IoT. Other related works can refer to [58], [59] and [60]. The other application of blockchain in IoT is to preserve the data security and privacy. Zhao *et al.* [61] design a secure and privacy-preserving system by applying the federal learning the blockchain technology. Gai *et al.* [62] combine the edge computing with the

blockchain-based IoT together, which is called blockchain-based Internet of Edge model and becomes a scalable, controllable privacy-preserving and efficient system. In the industrial area, the first blockchain IoT platforms is IOTA [63]. Other blockchain-based solutions include IoT Chain [64], Atonomi [65], Chain of Things [66], and so on.

Proof of Work (PoW), known as the original consensus protocol, is applied for the Bitcoin project [2]. It requires all the participants to solve a hash puzzle based on SHA-256 by brute-force, wherein the first one that solves the problem (i.e., miner) has the right to generate a block and obtains a reward for its work. Under the similar framework of PoW, a number of alternative protocols have been proposed to improve the performance of original PoW protocols, e.g., Proofs of Useful Work [67] and REM [68]. However, the competition of solving hash puzzle incurs huge energy expenditure for all the participants. To solve this problem, a new idea of the consensus protocol called Proof of Stake (PoS) was first suggested on Bitcoin Forum [69] and improved by Peercoin [70]. PoS shares the same purpose of PoW, but unlike PoW's leader selection depending on the hash puzzle calculation, it relies on the number of coins (i.e., stake), together with the ownership duration (i.e., coin age). Iddo *et al.* [71] point out the rationale behind PoS is that the entities with stake are more suitable to retain the security in order to prevent losses caused by the system erodes. Based on these core ideas of PoS, numerous frameworks with different puzzle designs have been proposed in [72], [73], [74] and [21].

Despite the rapid development, there is still a huge gap between the blockchain technology and real applications. Take the Bitcoin blockchain and Ethereum as examples, Bitcoin blockchain could handle 3-6 transactions per second (tps) [75], and Ethereum 1.0 can only process 7-15 transactions per second [76]. Compared with Visa [77], which is capable of handling more than 65,000 transaction messages per second, it is clear that neither of them can be considered as an alternative for daily transactions. One straightforward method to increase the throughput and reduce the latency is to partition the entities into parallel sub-groups (i.e., committees), which are responsible for generating and sustaining the sub-blocks (i.e., sharding) synchronously. Sharding is a term derived from distributed database and can be applied to the blockchain network for realizing

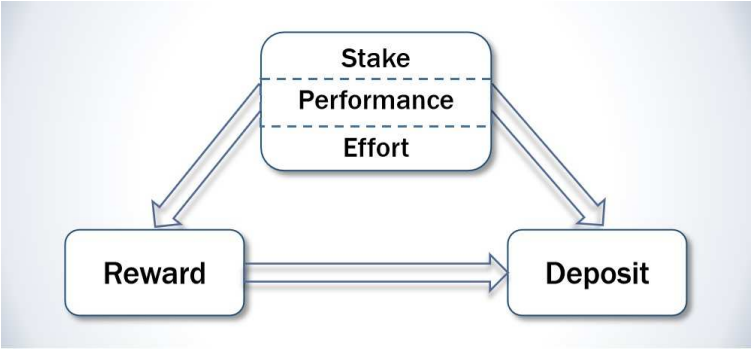


Figure 4: Relationship among the features (i.e., stake, performance and effort), reward and deposit of a participant.

the scalability [78].

To further satisfy the requirements in the blockchain network with shards, protocols that support high-throughput such as those in [79] and [80] were proposed. Take the ongoing Ethereum 2.0 project as the example, it publishes two possible PoS consensus protocols, i.e., Casper CBC that was proposed by Vlad Zamfir [81] and Casper FFG that was proposed by Vitalik and Virgil [21]. Since the project is always up to update and well-maintained, we adopt the latter in this work according to the latest document of github [82], which is a security-deposit based economic consensus protocol that can be deployed atop any *proposal mechanism*. PoS's security is deemed to derive from the size of **security deposits** rather than the number of participants, which can be set to greatly exceed the gains from the reward, and thus, proof of stake provides strictly stronger security incentives than proof of work [21]. Here we clarify two novel concepts: **security incentive** and **economic incentive**. For security incentive, it means participants have to behave in a legitimate way in the case that their deposits are slashed due to violating the rules. In this case, the penalty is regarded as the cost to motivate the majority of participants to sustain the security of the whole system. For economic incentive, the participants are motivated to follow the rules by the rewards issued from system.

Ethereum 2.0 adopts the combination of sharding and Casper to handle the scalability issue. It requires all participants to initiate a one-way transaction of 32 ethers to a deposit contract on Ethereum 1.0 [86]. However, 32-ether blocks out the participant whose stake is lower than that. For the participants whose stake values are far more than

Table 1: Related works comparison

Paper	Scenario	Model	Hidden information	Hidden action
[45]	Wireless communication	AS[1]	✓	×
[43]	Blockchain & Internet of vehicles		✓	×
[51]	Blockchain		✓	×
[83]	Fog computing	MH[2]	×	✓
[84]	Cognitive radio networks	JAM[3]	✓	✓
[85]	Mobile networks	TM[4]	×	✓
Our work	Blockchain & IoT	JAM JAT[5]	✓	✓

[1] AS represents Adverse Selection.
[2] MH represents Moral Hazard.
[3] JAM represents Joint Adverse Selection and Moral Hazard.
[4] TM represents Tournament Model.
[5] JAT represents Joint Adverse Selection and Tournament Model.

32 ethers, the security incentive constraint is weakened. Moreover, it is not fair to issue the same reward to the participants who have different stakes, performances and efforts. As shown in Fig. 4, we list the relationship among the features (i.e., stake, performance and effort), reward and deposit of a participant. Rationally speaking, the reward should be issued according to the participant's stake, performance or effort, which leads to an effective economic incentive for the participant; the deposit that is submitted to system should be greater than the reward and less than the stake. This means that it is affordable to the participant.

2.1.2 Motivation and Contribution

A sharded blockchain with PoS is a promising framework that allows the participants to run the transaction-related functions at a lower cost than the PoW-based blockchain. However, the steep security deposit price is an obstacle to the lower stake participants, such as the IoT device owners. IoT refers to a wide range of kinds of stuff and technique, benefits not only the industrial fields but also the individual life experiences, including smart wearables, smart homes, and smart communities. In this work, any individual or institute who possesses the IoT devices can be viewed as the 'validator.' The diversity of heterogeneous devices owned by the validators indicates various distribution of stake and performance. According to [21] and [86], staking the deposit is only a method to

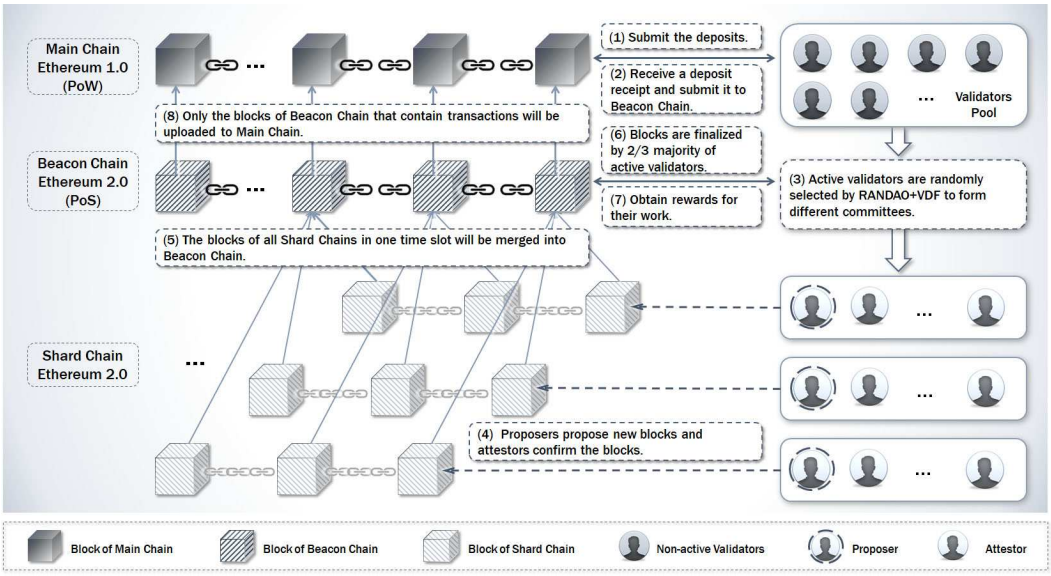


Figure 5: An overview of the blockchain network with shards.

prohibit illegal behaviors in participating blockchain rather than an extra cost. Moreover, any honest participant will be rewarded accordingly for their efforts in processing the transaction-related tasks. After exiting the blockchain network, the honest participants will get the security deposit returned. Considering both the disadvantage and advantage of participating in such a blockchain network, we dedicate to studying the mathematical models of security deposit pricing for these owners with different stake and performance.

Before designing an incentive scheme in the similar scenario (e.g., Ethereum 2.0), we analyze the problems that lead to the difficulty in designing an economic incentive mechanism, from the perspective of the intrinsic characteristics in blockchain. For a blockchain network with shards, even though there is a beacon chain in charge of the administrative transactions, the problem of *information asymmetry* still exists. Owing to the anonymity of a blockchain network and weak leadership of the beacon chain [82], the problem mainly manifested in the following aspects: (a) the beacon chain cannot determine the exact stake value of a single participant. (b) The real performance and effort exerted by the participant are unknown to others except themselves. Note that even though the timestamps contained in the work (i.e., block proposal, transaction or block confirmation) can be used as a metric of performance, they are not accurate

[87], e.g., the block proposal times are accurate only to within an hour or two in general. (c) The stake, performances and efforts are not uniformly distributed among these heterogeneous nodes.

The blockchain network should set the deposit greatly exceed the reward to regulate a validator's behavior [21]. But the validators cannot afford it if the deposit exceeds the stake in their accounts. In other words, the reward for each participant is the lower bound of their deposit, while the personal stake is the upper bond. Consequently, the reward is the key to determine the deposit. One easy way is to set a uniform reward and deposit for all validators with the same weights [88]. However, designing an adaptive deposit pricing scheme is challenging in terms of asymmetric information. As the study of how incentivizing the different parties in the presence of asymmetric information [44], contract theory is advantageous in mechanism design, benefiting the designer, and accordingly offering feasible mechanisms for rational game player. However, in practice, there must be a contract designer able to obtain the prior distribution of participants and design the contract items to the potential participants, which is against the conventional blockchain network with a single chain. That is also why contract theory cannot be applied in such a decentralized and autonomous system. In contrast, a sharded blockchain network consists of one main chain, one beacon chain, and various committees. With a weak leadership due to the decentralized architecture, the beacon chain can be the contract designer and can deploy the contract items among the validators during each epoch.

In this work, with the consideration of all the above aspects, we utilize the contract theory [44] to formulate the problems. In particular, contract theory can help overcome the information asymmetry when designing the mechanisms. As shown in Table 1, contract theory has been widely applied to variety of areas, e.g., Wireless Edge Caching Networks [45], Fog computing [83], Cognitive Radio Network [84] and Mobile Crowdsourcing [85]. Compared with the other research studies regarding the blockchain and our previous work [51], we consider more sophisticated cases in the blockchain network with shards. Generally, there is a positive correlation between the performance and effort of a participant, but the stake has nothing to do with the performance or effort, which

can be purchased online. When taking stake and effort or all of them into consideration at the same time, we present two different cases in the following: (a) **Stake-oriented** There is a significant difference among the validators' stakes, but they vary slightly in performances (or efforts), or in other words, their performances are far better than the task needs. (b) **Effort-oriented** There is a significant difference among the validators' performances (efforts), or in other words, the system values performances (efforts) more than stakes.

Here we summarize the main contributions of this work in the following.

1. We proposed two joint models based on contract theory, aiming to determine the participants' rewards according to their stakes, performances (efforts). Then the deposits can be specified based on the different rewards. That means, the participants obtain the different rewards and submit the different deposits, in which the deposits exceed the rewards. Compared with the mainly emphasized security incentive in Ethereum 2.0, the two proposed schemes are able to balance the security incentive and economic incentive for the beacon chain and the participants.

2. The first joint model is the combination of *adverse selection* and *moral hazard*. The *adverse selection* is used for distinguishing participants' stake type, and the *moral hazard* is used for determining the performance (or effort) exerted by a participant of a certain stake type.

3. The second joint model is the combination of *tournament* and *adverse selection*, which is applied to the second case. We utilize *tournament*, a competition of performances (or efforts), to distinguish the participants' capability. Then in a certain performance (or effort) type, *adverse selection* can help to determine the minimum stake threshold for the participants.

4. The simulation results and the related analysis are provided for the both joint models. We compare the impacts from the results of different key parameters, and present optimal rewards and security deposits to prove the feasibility and efficacy of the two models.

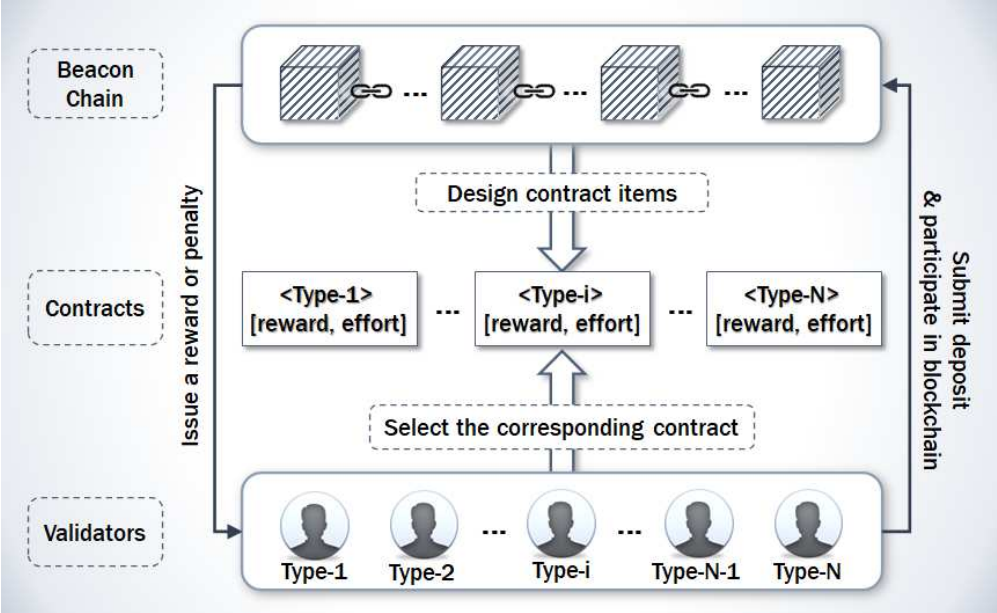


Figure 6: An overview of system model.

2.2 System Model

As shown in Fig. 5, a complete blockchain network with shards consists of a PoW-based blockchain (main chain), a PoS-based blockchain (beacon chain), a validator pool and a number of committees. In Proof-of-Stake, the consensus nodes (i.e., IoT devices or nodes in this work) are known as validators. The main chain is constituted by verified blocks with all the finalized transactions, the beacon chain contains the administrative transactions that are associated with all the validators and shards. More details of work process are included in Fig. 5. The major terminologies are listed in Table 2.

Let the beacon chain with weak leadership be responsible for designing the contracts. At the beginning of each epoch, even before the first step described in Fig. 5, the validators will sign the contracts according to their own stake or effort. The optimal reward and deposit are determined by the contracts based on the previous statistics. The details of designing and issuing contracts are listed in Fig. 6. Throughout the work, we set the temporary manager of the beacon chain as the contract designer. We classify the validators into different types regarding their stake in the blockchain account to design an adaptive deposit pricing mechanism. We adopt the validator's completing task time to evaluate its effort (e.g., voting time on chain [21], [86]).

Table 2: Terminology [82]

Term	Description
Beacon chain	The central proof-of-stake chain that is the base of the sharding system.
Shard chain	One of the chains on which user transactions take place and account data is stored.
Validator	A registered participant in the beacon chain. Anyone can become one by sending ether into Ethereum 1.0 deposit contract.
Proposer	A validator who proposes a new block.
Attestor	A validator who verifies and confirms the blocks.
Slot	A period during which one proposer creates a beacon chain block.
Epoch	An aligned span of slots during which all validators get exactly one chance to make an attestation.

2.2.1 Validator Model

Definition 1 *We classify the validator nodes into different types: type-1, type-2, ..., type-*$\mathbb{N}$*. The classification criterion is based on their stake or performance (effort). Let* θ_i *represent the type, and the type* θ_i *follows the inequation, which is expressed as*

$$\theta_1 < \theta_2 < ... < \theta_{\mathbb{N}}. \tag{1}$$

In this work, a validator who has more stake or higher performance will be identified as a higher type, which is set to be an index in Section 2.3 without loss of generality.

The utility function of a validator node is given by

$$U_V = \omega(\mathbb{R}) - \psi(e) - \xi(s) - \varphi(\theta, \mathbb{R}, e), \tag{2}$$

where $\mathbb{R}$ represents the reward that should be paid for the validator nodes, e denotes the efforts made by the validators, $\omega(\cdot)$ is the evaluation function of the reward $\mathbb{R}$, $\psi(\cdot)$ is the cost function of the effort e, $\xi(\cdot)$ is defined to evaluate the lock-up cost. According to [86] and [21], the penalty for any violation is a validator's entire deposit, we set $\varphi(\cdot)$ as the function of expected penalty (i.e., deposit).

2.2.2 Beacon Chain Model

The utility function of beacon chain is expressed as

$$U_{BC} = \pi(q) - \mu(\mathbb{R}), \tag{3}$$

where $\pi(\cdot)$ is the metric functions that evaluate the quality of effort received from a validator and increasing in the observed quality q, and $\mu(\cdot)$ is defined to evaluate the reward issued to the validator and increasing in the issued reward $\mathbb{R}$. The gap between the two terms denotes the beacon chain's utility obtained from the validator. From the beacon chain's perspective, it expects to extract the maximized utility by requiring a higher quality of validator's effort and offering a lower reward.

2.2.3 Definitions of Contract Theory

Definition 2 *Spence-Mirrlees Single-crossing Property For any contract* $(\mathbb{R}_i, e_i)$ *with different type* θ_i*, if the utility function of validators satisfies the inequation, which is given by*

$$\frac{\partial}{\partial \theta}\Big[- \frac{\partial U_V / \partial \mathbb{R}}{\partial U_V / \partial e}\Big] > 0. \tag{4}$$

The number of constraints can be effectively reduced [89].

Definition 3 *(Individual Rationality) IR means that only when a positive utility assigned by the beacon chain, the validator can accept the contract, which is given by*

$$U_{V(i)} \geq 0, \quad \forall i \in \{1, \ldots, i, \ldots, \mathbb{N}\}. \tag{5}$$

The basic idea of IR constraint reduction is that the constraints of all types will hold if the IR constraint of type-1 can be satisfied.

Definition 4 *(Incentive Compatibility) Here are four definitions regarding the IC constraints between type-i and type-j:*

(a) If $\forall j \in \big\{1, \ldots, i-1\big\}$*, the constraints are called Downward Incentive Constraints (**DICs**).*

*(b) If $j = i-1$, the constraint is called Local Downward Incentive Constraint (**LDIC**).*

*(c) If $\forall j \in \{i+1, \ldots, N\}$, the constraints are called Upward Incentive Constraints (**UIC**s).*

*(d) If $j = i+1$, the constraint is called Local Upward Incentive Constraint (**LUIC**).*

IC refers to that a validator of type-i can only obtain the maximum profit by choosing their own contract instead of all the others' contracts. The basic idea of IC constraint reduction is that if for type-i the LDIC holds, then its all the DICs hold, and the same as its LUIC and UICs.

2.3 The joint Design of Adverse Selection and Moral Hazard

In this section, we consider the scenario that there is a significant difference among the validators' stake values, but they vary relatively little in performance, or in other words, their performances are far better than that the task needs. So the stake value will be considered to be the main criterion of rewarding level. However, the stake value information of a single validator node is unknown to the beacon chain. Assuming that it can only obtain part of the information about the validator nodes, such as their stake value distribution and the possible value range.

In a PoS based blockchain network, all the participants can lock their coins (i.e., stake) in the network and will be randomly selected as the future block manager. According to [21] and [90], any holder of ether (i.e., the cryptocurrency issued by Ethereum) in Ethereum 1.0 can participate in Eth2 by submitting the security deposit. For ease of description, we call the account saving of an ether holder in Ethereum 1.0 as 'stake'. The main idea of pricing the security deposit is that the price must greatly exceed the reward issued by network [21]. Our goal is to design an affordable security deposit without impacting the security incentive. Consequently, an affordable security deposit should be greater than the validator's reward, but less than its stake. Moreover, the reward, security deposit and the voting weight of a validator should be positively associated with its stake. Thus, we classify the validators into different types based on their stake, e.g., $\theta \in \{\theta_1, ..., \theta_i, ..., \theta_{\mathbb{N}}\}$. A higher-type participant needs to stake a much higher

deposit to obtain more reward and more significant weight in blockchain, such as voting. Nevertheless, it will be fined much higher if any violation occurs. What opposite to a higher-type participant is, the lower one only needs to stake a minor deposit in exchange for a meager income and far less weight, which hardly impacts the voting results. We list the main notations of this model in Table 3.

For the different type θ_i, we set the different fixed salary f_i (a reward for one's participation) and the different bonus b_i (a reward for one's effort). Intuitively, for $\forall \theta_i, i \in \{1, \ldots, \mathbb{N}\}$, the salaries and bonuses satisfy the inequalities, which is given by

$$f_1 < \ldots < f_i < \ldots < f_{\mathbb{N}} \quad \text{and} \quad b_1 < \ldots < b_i < \ldots < b_{\mathbb{N}}. \tag{6}$$

Thus, the reward $\mathbb{R}$ distributed by the beacon chain now is described as a tuple (f_i, b_i). The base salary depends on a validator's type, but the bonus varies in the validator's effort. It is difficult to directly measure validators' efforts. As we known, an extensively applied metric of the whole blockchain network performance is *throughput*(e.g., [88] and [91]), which is the total number of transactions processable within a second. Thus, we consider that the completion time of a single task (i.e., transaction verification or voting) can be the metric of a validator's effort. For a specific validator belonging to type i, the time T_i of completing the task observed by the system consists of two parts: actual completing time t_i and the network delay ε, we have

$$T_i = \begin{cases} t_i + \varepsilon, & \varepsilon > 0, \\ t_i, & \varepsilon = 0. \end{cases} \tag{7}$$

The whole network delay are assumed to follow a normal distribution [92] with mean μ and variance σ. A direct connection between effort and time is that making more effort in the same task will take less time. That is, the exerted effort e_i is decreasing in the actual completing time t_i and the observed quality of effort q_i is decreasing in the observed time T_i. Similar to [43], we define a time-related metric function to measure the effort of a validator and the observed quality of a response, which is expressed as

$$e_i = \alpha_1 (T_{max} - t_i)^{a_1} \quad \text{and} \quad q_i = \alpha_2 (T_{max} - T_i)^{a_2}, \tag{8}$$

where T_{max} is the maximum acceptable response time of the beacon chain, $e_{i,k} \geq q_{i,k}$, α_1, α_2, a_1 and a_2 are the pre-defined coefficients and $e'(t) < 0, q'(t) < 0$. That means, by virtue of the network delay, the observed task quality is always less than or equals to the actual effort that is made by the validator node, and the effort level and the response quality decrease with the increment of time.

With the assumption of risk neutrality, we define an evaluation function $\omega_{i,k}$ regarding the base salary f_i, bonus b_i and effort e_i, which derives from the linear incentive schemes in [93] and [44]. If a validator k of type i signs the corresponding contract (f_i, b_i), its evaluation function of total reward is expressed by

$$\omega_{i,k} = \theta_i \beta_1 f_i + \theta_i \beta_2 b_i e_{i,k}. \tag{9}$$

According to the cost model in [93], the cost function is associated with the effort level and the observation error. In the proposed model, the total cost consists of three parts: the cost of exerting effort, the cost of staking deposit into blockchain, and the potential penalty due to the inefficient performance. For given unit effort cost c, and the maximum acceptable response time T_{max}, we define cost functions of the effort, staking and penalty in the following equations:

$$\psi(e_{i,k}) = c^2 (e_{i,k})^2, \tag{10}$$

$$\xi(\theta_i, b_i) = \theta_i^2 b_i^2 + b_i, \tag{11}$$

$$\text{and} \quad \varphi(\theta_i, \mathbb{R}_i, e_{i,k}) = \rho(\theta_i, \mathbb{R}_i)\gamma(1 - e_{i,k}/T_{max})^2. \tag{12}$$

Similar to [43] and [44], we set a quadratic cost function with respect to effort level. The equation in 11 is not the actual lock-up cost (e.g., the lock-up cost function is generally denoted by an increasing natural exponential function), but an individual evaluation function with respect to the deposit. Considering the deposit is positively correlated with the reward, we adopt the bonus instead of deposit to simplify the calculation. This quadratic function is convex, where $\sum(\xi(\theta_i, b_i)) \leq \xi(\theta_i, \sum b_i)$. It incurs more cost for those who intend to collude together. Consequently, the function in 11 is increasing

in the validator's type and deposit, which is monotonic just as the actual lock-up cost function. $\rho(\theta_i, \mathbb{R}_i)$ denotes the evaluation function of the deposit that the validator node should submit to the beacon chain, which is related to the validator nodes' type and reward. We also use $\gamma(1 - e_{i,k}/T_{max})^2 \in (0, 1]$ to express the failure probability of inefficient performance. Obviously, if validator exerts no effort at all (i.e., $e_{i,k} = 0$), then $U_{i,k} = -b_{i,k}\varepsilon - \gamma\rho(\theta_i, \mathbb{R}_i) < 0$.

Therefore, according to the redefined functions (9), (11), (10) and (12), we can rewrite the utility function (2) of the validator node k, which is given by

$$U_{V(i,k)} = \theta_i\beta_1 f_i + \theta_i\beta_2 b_i e_{i,k} - c^2(e_{i,k})^2 - \theta_i^2 b_i^2 - b_i - \rho(\theta_i, \mathbb{R}_i)\gamma\big(1 - \frac{e_{i,k}}{T_{max}}\big)^2. \tag{13}$$

Owing to the features of blockchain, the beacon chain can only obtain limited information about the validators. Set λ_i as the probability of the validator node k belonging to the certain type-i, where $\sum_{i=1}^{\mathbb{N}} \lambda_i = 1$. Assume that there are $\mathbb{K}$ validator nodes in each type, the utility function of beacon chain is expressed as

$$U_{BC} = \sum_{i=1}^{\mathbb{N}} \lambda_i \bigg(\sum_{k=1}^{\mathbb{K}} \pi(q_{i,k}) - \mu(\mathbb{R}_{i,k}) \bigg). \tag{14}$$

For given fixed salary f_i and bonus b_i, we have

$$\mu(\mathbb{R}_{i,k}) = \mathbb{R}_{i,k} = f_i + b_i q_{i,k}, \tag{15}$$

where $q_{i,k}$ is the observed performance of type-i validator k, and $b_i q_{i,k}$ denotes the reward for its effort.

2.3.1 Problem Formulation

According to the constraint **IR**, for any rational validator node with any effort greater than zero, only when its utility given by the contract is greater than zero can it accept the contract. So we have

$$U_{V(i,k)} \geq 0. \tag{16}$$

The constraint **IC** can guarantee that the rational validator nodes will sign the contracts that are designed specifically for their own types. Since the validator nodes can obtain the greatest utilities only from their own contracts instead of others. We set $U_{V(i,k)}(\mathbb{R}_i, e_i)$ as the type-i validator node k's utility that is obtained from its own type contract, and set $U_{V(i,k)}(\mathbb{R}_j, e_j)$ as the utility that is obtained from the type j contract. Thus, according to the IC constraint, we have

$$U_{V(i,k)}(\mathbb{R}_i) \geq U_{V(i,k)}(\mathbb{R}_j). \tag{17}$$

Therefore, with the constraints (16) and (17), the optimization problem is expressed as

$$\max_{(f_i,b_i)} \quad U_{BC} = \sum_{i=1}^{\mathbb{N}} \lambda_i \Big(\sum_{k=1}^{\mathbb{K}} \pi(q_{i,k}) - f_i - b_i q_{i,k} \Big), \tag{18}$$

s.t.

$$(a) \quad U_{V(i,k)}(\mathbb{R}_i) \geq 0,$$

$$(b) \quad U_{V(i,k)}(\mathbb{R}_i) \geq U_{V(i,k)}(\mathbb{R}_j).$$

However, the problem (18) described above is not a convex problem, and thus cannot be directly solved by the convex optimization tools. We solve this problem by the proposed method in the subsections.

2.3.2 Optimal Solution of Contract

Since the problem is formulated with the joint moral hazard and adverse selection, from the perspective of contract theory [44], its optimal solutions will vary under the different cases. However, in our scenario, the stake value of a validator node is always unknown to the system. That means there always exists the adverse selection problem, and there is no probability that only the moral hazard case exists. Thus, we present the optimal solutions for the joint model (i.e., the joint model of adverse selection and moral hazard) and adverse selection model, respectively. Moreover, we explain the physical meaning behind these mathematical models.

For a certain type i, we can easily determine that the utility of a validator is a

concave function with respect to its effort. Thus, in order to obtain the optimal effort choice for validator k, we first differentiate the utility function of validators with respect to e and set it to zero, which is given by

$$\frac{\partial U_{V(i,k)}}{\partial e} = 0. \tag{19}$$

For ease of analysis and without loss of generality, we conduct the following steps to simplify the calculation. For the effort $e_{i,k} = \alpha_1(T_{max} - t_{i,k})^{a_1}$ and $q_{i,k} = \alpha_2(T_{max} - T_{i,k})^{a_2}$, we set $\alpha_1 = \alpha_2 = a_1 = a_2 = 1$, since the time is one of the general performance metric as in [92]. Given the historical performance data and $\tau(e_i) = \gamma\big(1 - e_{i,k}/T_{max}\big)^2$, we can obtain that $\tau(e_i) \in (0, \tau(\underline{e})]$, where $\underline{e}$ represents the lowest level of effort accepted by system. Then we reduce $\tau(e_i)$ to a constant $\tau(\underline{e})$ without impacting the monotonicity and incentive compatibility of all types of validators. According to our previous analysis of *reward, deposit and stake* in Fig. 4, deposit (also known as penalty if any illegal behaviour occurs) should be greater than all the rewards issued by system, where $\rho(\theta_i, \mathbb{R}_i) \geq \mu(\mathbb{R}_{i,k})$. Therefore, we define $\rho(\theta_i, \mathbb{R}_i) = \kappa\theta_i(f_i + b_i e_{i,k})$, where $1 \leq \kappa$. Finally, the optimal choice of effort for validator k belonging to type i can be obtained according to the first derivative, which is given by

$$e_i^* = \frac{\theta_i b_i}{2c^2}(\beta_2 - \kappa\tau). \tag{20}$$

Since β_2 is a pre-defined parameter, then if $\beta_2 = 2c + \kappa\tau$ holds, regardless of the value of right side. e_i^* can be rewritten as $e_i^* = \theta_i b_i / c$. For a certain type-i, $e_{i,k}^*$ can represent all the validator nodes' optimal choice of efforts, so we use e_i^* instead. Similarly, if the type-i validator node signs the contract that is designed for type-j, its optimal effort choice is expressed as

$$e_{i,j}^* = \frac{\theta_i b_j}{c}. \tag{21}$$

From the equation (20), we can reach the conclusion that, the optimal effort choice e_i^* for all types θ_i is greater than zero, and it will increase with its type θ_i and bonus b_i and decrease with the cost c. From the validator node's perspective, it will have more incentives to respond to the beacon chain to the best of its abilities, if it is identified

as a higher type and rewarded with a higher bonus [44]. By substituting the effort $e_{i,k}$ with e_i^*, we first rewrite the type-i validator node's utility function (13) as

$$U_{V(i)}(\mathbb{R}_i) = \theta_i(\beta_1 - \kappa\tau)f_i - b_i. \tag{22}$$

In the pure adverse selection model, the effort e_i for its type will be a fixed value, i.e, $\hat{e}_i$, which means that a type-i validator is required to exert e_j^* if he signs the type-j contract rather than the optimal effort $e*_{i,j}$. From the equation (22), we can conclude that this validator can obtain the maximum utility from the joint model which is $\theta_i(\beta_1 - \kappa\tau)f_j - b_j$. However, the validator's maximum utility in the pure model is only $\theta_i(\beta_1 - \kappa\tau)f_j - b_j - (\theta_j - \theta_i)^2 b_j^2$, which is apparently less than that of the joint model. Therefore, even if signing the wrong contract, the rational validators are more willing to maximize their utility by adjusting the efforts. Therefore, we only give the solution of joint model below.

After solving the moral hazard problem and obtaining the optimal choice of efforts, we assume that the validators' performance can be obtained (the performance can be expressed as a function of f_i and b_i). However the validator nodes' stake values are still unknown to the beacon chain. Here we set the contract as (f_i, b_i), i.e., the fixed salary f_i and bonus b_i. However, from (18) we can see that there are $\mathbb{N}(\mathbb{N}-1)$ IC constraints and $\mathbb{N}$ IR constraints in total. Consequently, it is still difficult to solve the problem. Thus, we will first reduce the constraints by utilizing the definitions that we described in Section (2.2.3). The monotonicity of f_i and b_i is predefined, and the monotonicity of e_i^* is explained in the previous analysis (20). So we omit the proof of monotonicity here.

- **Reduction of IR Constraints**

According to the IC constraints in (18) and the monotonic condition(1), for type $\forall i \in \{1, ..., \mathbb{N}\}$, we can have

$$U_{V(i)}(f_i, b_i) \geq U_{V(i)}(f_1, b_1) \tag{23}$$

$$\text{and} \quad U_{V(i)}(f_1, b_1) \geq U_{V(1)}(f_1, b_1). \tag{24}$$

Table 3: Main Notations (1)

Symbol	Definition
θ_i	Validators' type
f_i	Fixed salary
b_i	Bonus for the type-i validator's effort
T_i	Observed time of type-i validator completes the task
t_i	Actual completing time
ε	Network delay
e_i	The effort exerted by a validator
q_i	The observable quality of effort
$\rho(\theta_i, \mathbb{R}_i)$	The deposit submitted by type-i validators
ω_i	The evaluation function of reward
ψ	The cost function of effort
ξ	The cost function of stake
φ	The penalty function
π	The evaluation function of response' quality
μ	Total salary

Obviously, for given (23) and (24), we can come to the conclusion that $U_{V(i)}(f_i, b_i) \geq U_{V(1)}(f_1, b_1)$.

- **Reduction of IC Constraints**

According to the IC constraints, for given three adjacent types, i.e., type $i-1$, type i and type $i+1$, which follow $\forall i \in \{2, \ldots, \mathbb{N}-1\}$, we have the following two inequations, which are expressed as

$$U_{V(i+1)}(f_{i+1}, b_{i+1}) \geq U_{V(i+1)}(f_i, b_i) \tag{25}$$

$$\text{and} \quad U_{V(i)}(f_i, b_i) \geq U_{V(i)}(f_{i-1}, b_{i-1}). \tag{26}$$

To proceed the reduction of IC constraints, we set two parameters l and r, which are given by

$$l = (\theta_{i+1} - \theta_i)(\beta_1 - \kappa\tau) f_i \tag{27}$$

$$\text{and} \quad r = (\theta_{i+1} - \theta_i)(\beta_1 - \kappa\tau) f_{i-1}. \tag{28}$$

Based on the monotonic conditions (1), (6) and the optimal result analysis (20), we easily have $l \geq r$. So we add (27) and (28) to (26), and obtain a new inequation

$U_{V(i)}(f_i, b_i) + l \geq U_{V(i)}(f_{i-1}, b_{i-1}) + r$, we have

$$U_{V(i+1)}(f_i, b_i) \geq U_{V(i+1)}(f_{i-1}, b_{i-1}). \tag{29}$$

Based on (25) and (29), we have

$$U_{V(i+1)}(f_{i+1}, b_{i+1}) \geq U_{V(i+1)}(f_{i-1}, b_{i-1}), \tag{30}$$

then repeat the steps (25), (26), (29) and (30), we can obtain the following constraints, which is given by

$$\begin{aligned} U_{V(i+1)}(f_{i+1}, b_{i+1}) &\geq U_{V(i+1)}(f_{i-1}, b_{i-1}) \\ &\geq U_{V(i+1)}(f_{i-3}, b_{i-3}) \\ &\geq ... \\ &\geq U_{V(i+1)}(f_2, b_2) \\ &\geq U_{V(1)}(f_1, b_1). \end{aligned} \tag{31}$$

Similarly, for the type θ_{i-1} and all the contracts which follow $\forall i \in \{2, ..., \mathbb{N}\}$, we can easily obtain the following inequalities by the same steps above, which is expressed as

$$\begin{aligned} U_{V(i-1)}(f_{i-1}, b_{i-1}) &\geq U_{V(i-1)}(f_{i+1}, b_{i+1}) \\ &\geq ... \\ &\geq U_{V(i-1)}(f_{\mathbb{N}}, b_{\mathbb{N}}). \end{aligned} \tag{32}$$

Therefore, the proof of (31) indicates that if the LDICs are satisfied, all the DICs also hold, as well as the LUICs and UICs proved in (32). With the reduced constraints, we set the profit metric function as $\pi(\cdot) = d\theta_i q_i^*$. If there is no network delay, then $q_i = e_i$ holds. The optimization problem can be expressed as

$$\max_{(f_i, b_i)} \quad U_{BC} = \sum_{i=1}^{\mathbb{N}} \lambda_i \left[\frac{d\theta_i^2 b_i}{c} - f_i - \frac{\theta_i b_i^2}{c} \right], \tag{33}$$

s.t.

$$(a) \quad U_{V(i)}(f_i, b_i) = U_{V(i)}(f_{i-1}, b_{i-1}), \quad \forall i \in \{1, ..., \mathbb{N}\},$$

$$(b) \quad U_{V(1)}(f_1, b_1) = 0.$$

To solve this problem, we first set $\Delta_j = (\theta_{j+1} - \theta_j) f_j(\beta_1 - \kappa\tau)$. Based on the $\mathbb{N}$ IR constraints of (33), we add all the equations that is numbered from 1 to i together and get

$$U_{V(i)} = \sum_{j=1}^{i-1} \Delta_j. \tag{34}$$

According to the equation (34), we derive an expression for f_i, which is given by

$$f_i = \frac{b_i + \sum_{j=1}^{i-1} \Delta_j}{\theta_i(\beta_1 - \kappa\tau)}, \quad \forall i \in \{2, ..., \mathbb{N}\}. \tag{35}$$

Note that $f_1 = b_1/\theta_1(\beta_1 - \kappa\tau)$. After substituting (35) into (33), (33) will be converted into a new problem, we differentiate U_{BC} with respect to b_i and then get

$$\frac{\partial U_{BC}}{\partial b_i} = \frac{d\theta_i^2}{c} - \frac{1}{\theta_i(\beta_1 - \kappa\tau)} - \frac{2\theta_i b_i}{c}. \tag{36}$$

Next, by differentiating $\frac{\partial U_{BC}}{\partial f_i}$ with respect to b_i, we have

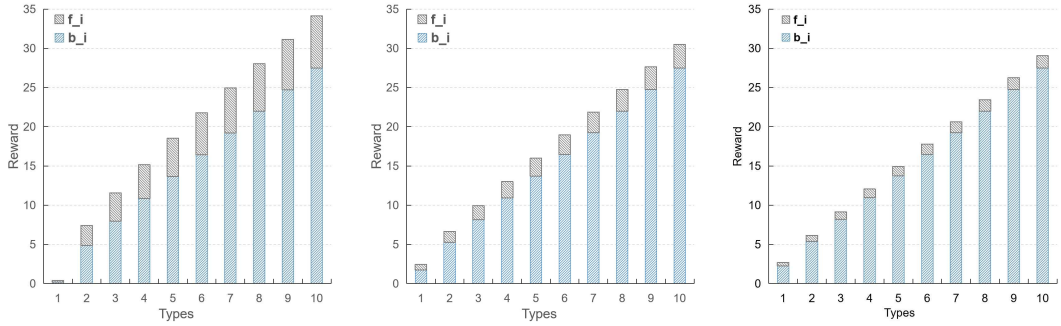
$$\frac{\partial^2 U_{BC}}{\partial (b_i)^2} = -\frac{2\theta_i}{c} < 0. \tag{37}$$

Clearly, we can reach the conclusion that the problem is a convex problem with the result (37). Therefore, we can get the optimal solution (f_i^*, b_i^*) by setting the first derivative as zero, which is expressed as

$$b_i = \frac{d\theta_i}{2} - \frac{c}{2\theta_i^2(\beta_1 - \kappa\tau)}. \tag{38}$$

The network failure occurs mostly within the information propagation among nodes and can cause an extra time cost, which is viewed as the network delay in this work for simplicity.

The network delay of the whole network is assumed to follow a normal distribution with mean μ and variance σ^2 [92]. When we take the network delay into consideration,



(a) $\beta_1 = 1$ (b) $\beta_1 = 2.5$ (c) $\beta_1 = 5$

Figure 7: The impact of evaluate parameter β_1 on the fixed salary f_i and bonus b_i.

Table 4: Parameter Setting

Parameters	Values
Total number of types	$N = 10$
Pre-defined penalty factors	$\kappa = 2$, $\tau = 0.01$
Pre-defined coefficients	$d = 5.5$, $\beta_1 = 5$
Unit cost	$c = 5$
Mean of the network delay	$\mu = 1500$ms
Variance of the network delay	$\sigma = 500$ms

i.e., $\varepsilon > 0$. With $q_i = e_i - \varepsilon$, we can rewrite the optimization problem, which is expressed as

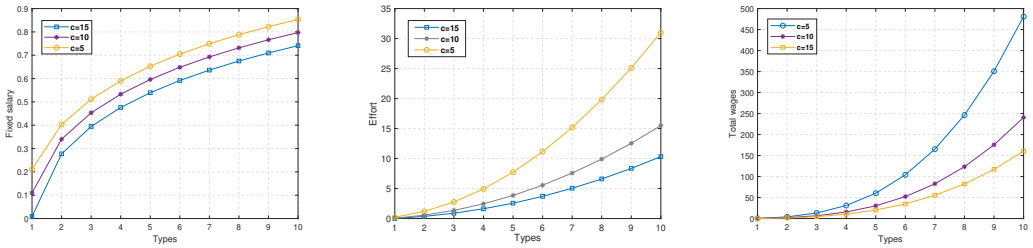
$$\max_{(f_i, b_i)} \quad U_{BC} = \sum_{i=1}^{\mathbb{N}} \lambda_i [d\theta_i \int_0^{\Delta T} (e_i^* - t) \mathrm{d}F(t;\mu) - f_i - b_i \int_0^{\Delta T} (e_i^* - t) \mathrm{d}F(t;\mu)], \qquad (39)$$

where $\varepsilon \sim f(t;\mu,\sigma)$ is a normal distribution, and ΔT is the acceptable tolerant delay for blockchain network. The optimal solutions to such problem is similar as above. We present the optimal results of e_i^*, f_i, and b_i in the following subsection.

2.3.3 Simulation Results and Analysis

In this subsection, we present the simulation results and give the analysis for each illustration with the parameter setting. Then we present the numerical results in the following steps.

According to the utility function in Section 2.3, we set the different parameters β_1 and β_2 to evaluate the fixed salary f_i and bonus b_i. That means, the parameters setting will affect the fixed salary and bonus. Since $\beta_2 = 2c + \kappa\tau$, we first keep the parameters κ, τ fixed and set $d = 5.5$ and $c = 5$, and then increase the β_1.



(a) Fixed salary f_i.

(b) Optimal effort choices e_i^*.

(c) Total reward $f_i + b_i e_i^*$ when adopting the optimal effort choice e_i^*.

Figure 8: The impact of cost c on the fixed salary f_i, optimal effort choice e_i^* and the total reward $f_i + b_i e_i^*$.

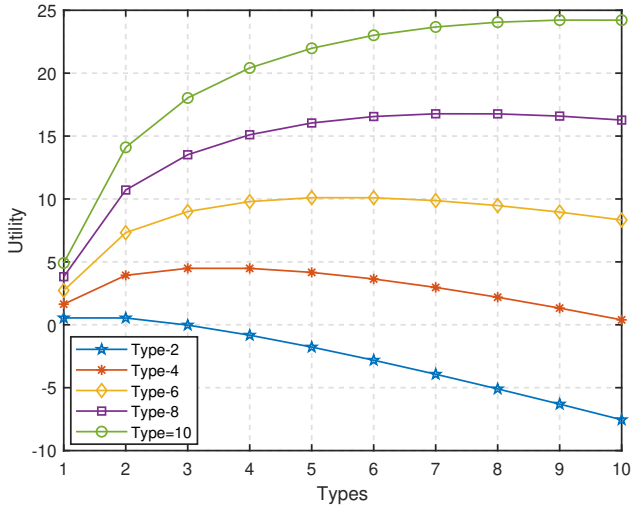


Figure 9: Utilities of validators when sign different contracts.

With x-axis representing the different types of validator nodes, from Fig. 7, we can see that β_1 significantly affects the fixed salary f_i, but has slight effect on the bonus b_i. Moreover, the monotonicity shown in Fig. 7 is consistent with our premise (6).

According to the optimal solutions f_i^*, b_i^* and the optimal effort choice e_i^*, we observe that another important parameter that may affect them is the unit cost c. We increase the cost and try to determine the different impacts on the results. So we first keep the parameters κ and τ fixed and set $d = 3.1$, $\beta_1 = 5$, and then change the value of c. Finally, from Fig. 8(a), Fig. 8(b) and Fig. 8(c) we can see that the fixed salary, an optimal effort choice and total reward decrease along with the increasing of c. We do not list the bonus b_i here, since it varies little as the the cost changes, which is similar to Fig. 7. In fact, even for the fixed salary, the variation is tiny as well.

As shown in Fig. 9, the validators of type-2, type-4, type-6, type-8 and type-10

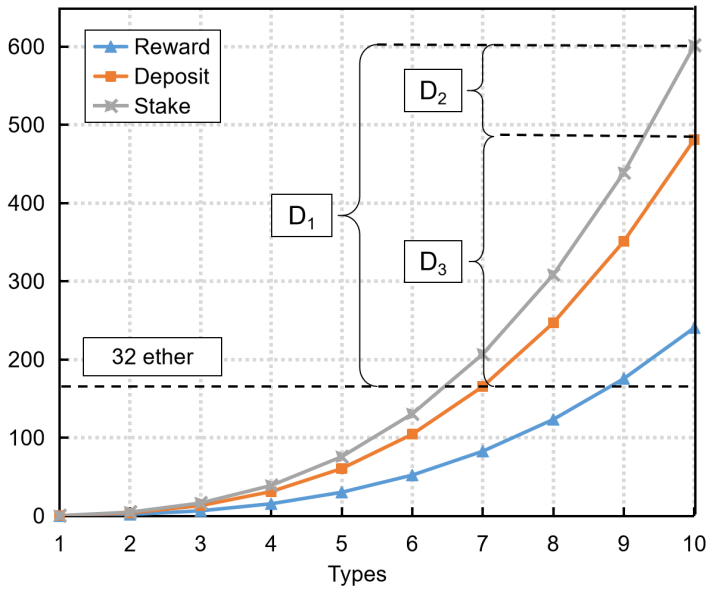


Figure 10: Comparison of reward, deposit, stake and the fixed deposit (e.g., 32 ether).

Algorithm 1 Security deposit pricing solution

Require: Number of types: $\mathbb{N}$, the probability of the different type validator: λ_i, total number of validators: n, average stake of each type: s_i, reward issued to validators: $\mathbb{R}_i$, threshold coefficient: κ, total security deposit in Casper: $\mathbb{D}$;

Ensure: Deposit d_i^*;

1: **if** $\sum_1^{\mathbb{N}} n\lambda_i s_i \geq \mathbb{D}$ **then**
2: **for** $i = 1$; $i \leq \mathbb{N}$; $i++$ **do**
3: Set $\delta = \frac{\lambda_i s_i}{\sum_{j=1}^{\mathbb{N}} \lambda_j s_j}$;
4: $d_i^* = \min\left\{s_i, \max(\frac{\delta\mathbb{D}}{n\lambda_i}, \kappa \cdot \mathbb{R}_i)\right\}$;
5: **end for**
6: **else**
7: **end if**

signing the different contracts have various utilities. It shows apparently that the validators have the maximum utilities only when choosing the contract designed for their own, which proves the IC constraint. Besides, all these maximum incentives are positive, which explains the IR constraint.

At last, Fig. 10 represents the comparison of the proposed scheme and the fixed-deposit scheme (e.g., Ethereum 2.0). The reward curve is the same as the one marked with "$c = 10$" of Fig. 8(c). The stake curve is not obtained from the real data, just from a simulation in this model. However, it is easy to adopt the real data to a curve of this model by adjusting the parameters and setting an "exchange rate". In Ethereum 2.0, only the validators whose stakes are greater than 32 ether can afford it and are eligible

for operation and maintenance within a blockchain. We assume that these validators belong to the types that are greater than 7 in our model. Obviously, 32-ether blocks out these validators whose types are less than 7 and lead to their disqualification from the participation of a blockchain. Not only that, 32-ether cannot ensure an adequate security incentive for the system against the higher type validators. For a better comparison, take the type-10 validator as an example, let D_1 denote the difference between the stake and 32 ethers, D_2 denote the difference between the proposed deposit and stake, and D_3 denote the difference between the proposed deposit and 32 ether. Thus, it is clear that due to $D_2 < D_1$ and $D_3 > 0$, the proposed model can provide a better security incentive than that of the fixed-deposit model for the system, which motivates the validators who possess more stake to behave in a legal way. Additionally, there is no restriction for the lower type validators. All of them are required to submit an affordable deposit and obtain reward for their work by signing the contracts. In this model, reward, deposit and stake of all types always follow the rule that we analyze before, i.e., reward<deposit<stake. According to [21], security derives from the size of the penalty, which equals the amount of deposit submitted by validators, not the number of validators. Thus, we can easily have the relation of deposit's amount between [21] and our work as follows: $\sum_{i=1}^{10} \lambda_i \mathbb{D}_i \geq \sum_{i=7}^{10} \lambda_i \mathbb{D}_7$, where the left term denotes the total amount of deposits in our work and the right term represents the size of deposits in [21] with $\mathbb{D}_i$ is viewed as the required deposit. Together with the same distribution of validators, the deposits required in this model exceed Ethereum 2.0, which means the security is retained rather than being reduced.

An extreme case is that all the potential validators do not have as many as 32-ether in their accounts. However, if the total number of these potential validators n is far greater than that of validators in the original committee (e.g., 128 in [94]), the following inequality may hold, which is given by

$$\sum_{i=1}^{\mathbb{N}} n\lambda_i s_i \geq \mathbb{D}, \tag{40}$$

where $\mathbb{D}$ is the total amount of security deposit required by one committee. We adopt a proportional assignment to determine the security deposit d_j for each validator, which

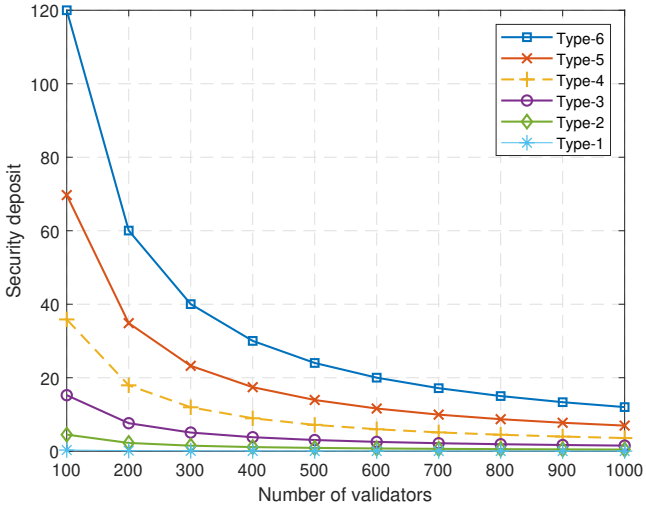


Figure 11: Security deposit pricing when total number of validators varies.

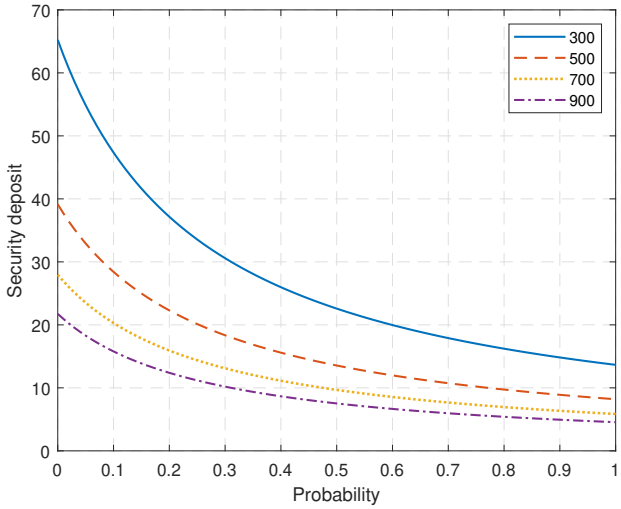


Figure 12: Security deposit pricing when probability of type-6 validators varies.

is expressed as

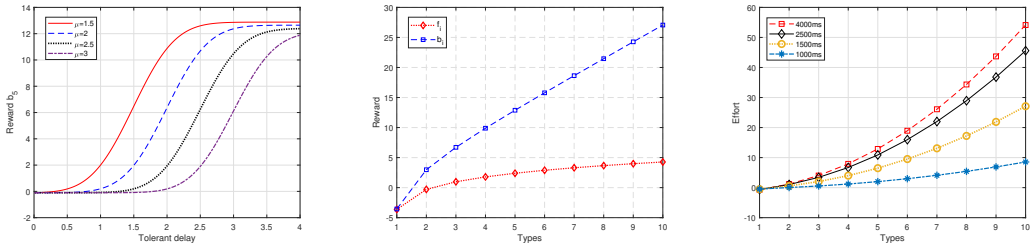
$$d_j = \frac{\lambda_j s_j}{\sum_{i=1}^{\mathbb{N}} \lambda_i s_i} \cdot \frac{\mathbb{D}}{n\lambda_j}. \tag{41}$$

With the inequality (40), we can prove that

$$d_i \leq s_i. \tag{42}$$

Obviously, any assigned deposit is less than or equal to the stake in the validator's account, which indicates such an assignment is feasible.

Additionally, an algorithm (1) is proposed to obtain the optimal deposit. We assume



(a) Bonus b_5. (b) Fixed salary f_i and bonus b_i. (c) Optimal effort choice e_i^*.

Figure 13: The impact of network delay on validators' rewards and efforts.

the validators are uniformly distributed for ease of analysis, and type-7 validators have an average stake of 32-ether, so all the lower types cannot afford 32-ether. We further investigate the security deposit pricing when the total number of validators varies in Fig. (11). We can see that the security deposit is decreasing in the number of validators. However, some of the deposit pricing points are not valid. Take type-6 as an example. It needs to submit up to 120-ether security deposit if the total number is 100. With the previous assumptions, inequality in (40) and equation in (41), we can easily obtain the minimum number of validators: 476. Together with the total number, we also have the security deposit for each validator satisfying the equation: $d_i = s_i$. Moreover, we also can obtain the upper bound and lower bound of the penalty coefficient: $\overline{\kappa} = \frac{s_i}{\mathbb{R}_i}$ and $\underline{\kappa} = \frac{d_i}{\mathbb{R}_i}$. According to our contract design, κ is pre-determined. Through the calculation of boundaries, we can conclude that $\kappa \cdot \mathbb{R}_i$ will be adopt as the security deposit if $\kappa \in [\underline{\kappa}, \overline{\kappa}]$. Consequently, the security deposits charged on the validators exceed $\mathbb{D}$: $\sum_{i=1}^{\mathbb{N}} n\lambda_i \kappa \mathbb{R}_i > \mathbb{D}$. As mentioned in [21], security derives from the security deposit. Thus, the proposed security deposit pricing can even provide more security.

Apart from the total number of validators, another factor may affect the security deposit pricing is the probability of validators' distribution. Take type-6 as an example. We show the variation of security deposit when the distribution probability of validators varies in Fig. 12. Note that all the security deposits are zeros if probability is zero. We can see the security deposit prices (except for the points with probability=0) are decreasing in the probability of validators' distribution. For the same probability, more validators will lead to a lower security deposit prices. Similarly, there exists some invalid points. Combining with Fig. 11, it assist with determining the feasible probability

distribution of different type validators.

The following figures show how the network delay impacts incentives and validators' behaviors. Similar to [92], the system set up a tolerant delay for all the responses. Take the type-5 reward as an example. We will examine how the reward changes over the tolerant delay with the different mean values. With the same parameter setting in table 4, Fig. 13(a) shows the bonus for type-5 validators over the tolerant delay. We can observe that all the bonus values seem to plateau at around 4000ms. More bonuses will be issued to validators if the blockchain network can accept the longer delay. Moreover, the solid curve demonstrates that a lower network delay can result in a higher bonus. However, the maximum values are still lower than that of the no-delay case in Fig. 7(a). With $\mu = 1500$ms and $\Delta T = 3000$ms, Fig. 13(b) plots the base salary and bonus over all the validator types. As the type increases, the rewards for each type of validator are gradually increasing. Similarly, all the rewards are less than that of the no-delay case in Fig. 7(a). Note that the rewards for type-1 and type-2 are lower than zero. Thus, some lowest types of validators can be filtered out due to the network delay. Finally, Fig. 13(c) compares all the effort levels with the different tolerant delay. As is observed, the effort level increases in types and tolerant delay, which indicates that more efforts are required to compensate for the loss caused by network delay. Combining with Fig. 13(a), we can figure out that network delay can decrease the bonus, which means the network delay also acts as a disincentive to effort level.

2.4 The Joint Design of Adverse Selection and Tournament

To take this research one step further, we additionally consider that the network places more weight on the performance (e.g., a higher quality of task completion contributes more market value for the blockchain network) than on the stake. The performance also varies significantly throughout all the validators. In this case, we consider performance as the main criterion of classification. However, a validator node's effort is a hidden action that can be scheduled by itself and is unknown to the beacon chain. To supplement this, "stake" should be considered as a minor criterion of rewarding. We further utilize *adverse selection* to formulate the optimization problem and determine

Table 5: Monotonicity and Mapping Relation

Term	Parameters
Type	$\theta_{\mathbb{K}} > \ldots > \theta_j > \ldots > \theta_1$
Effort	$e_{\mathbb{K}} > \ldots > e_j > \ldots > e_1$
Reciprocal of effort	$e_{\mathbb{K}}^{-1} < \ldots < e_j^{-1} < \ldots < e_1^{-1}$
Performance(Time)	$\Delta t_1 < \ldots < \Delta t_{\mathbb{K}+1-j} < \ldots < \Delta t_{\mathbb{K}}$

the minimum threshold of stake s required from different types of validators.

We assume that the devices' ability distribution can be acquired through the pre-screening in the Proof-of-Work context. To reduce the monitoring cost [44] and maintain the ex-ante effort level, we develop a joint design of adverse selection and tournament to deal with this intricate problem. The beacon chain, as the contract designer, divides the time into $\mathbb{K}$ intervals, and each interval Δt is represented by a type $\theta \in \{\theta_1, ..., \theta_i, ..., \theta_{\mathbb{K}}\}$. Note that the first interval Δt_1 belongs to the $\mathbb{K}$-th type $\theta_{\mathbb{K}}$, and the last interval $\Delta_{\mathbb{K}}$ belongs to the first type θ_1. Assuming that time t is a random variable whose distribution depends on the reciprocal of effort, i.e., e^{-1}. We can intuitively determine that the more effort results in the shorter time. That is, for different effort levels i and j, if $e_i > e_j$ and $e_i^{-1} < e_j^{-1}$, then there is a high probability that $t_i < t_j$.

For better understanding, we clarify the monotonicity and mapping relation of *type*, *effort*, *reciprocal of effort* and *performance* as shown in Table (5).

From Table (5), we can conclude that a validator node wins the j-th place if and only if its effort is the j-th order and it belongs to type-j of all the orders $\{1, ...j, ..., \mathbb{K}\}$. An important point is that the first place of type order denotes the lowest performance level and the $\mathbb{K}$-th place denotes the highest one. Take θ_j as an example, the validator's effort level is e_j and the corresponding time interval is $\Delta t_{(\mathbb{K}+1-j)}$ rather than Δt_j. We list the main notations of this model in Table 6.

Given the analysis regarding the performance, effort, stake and reward, it is clear that the joint model in Section 2.3 does not apply to the scenario that we consider in this section. The basic rationale of tournament is first to check the performance rank order, and then to reward the validator according to the orders. Through pre-screening [95], the beacon chain obtains an ex-ante task completion time distribution. Assume that the computation power distribution is relatively constant during a certain period,

Table 6: Main Notations (2)

Symbol	Definition
$\mathbb{K}$	The total number of types
Δt_i	The predefined threshold time interval of completing task
$\mathbb{R}_1$	The reward determined in tournament model
$P(\text{rank} = j)$	The probability of certain effort level is in the $j-th$ order of all levels
A	The coefficient of absolute risk aversion
$\mathbb{E}$	The optimal effort that maximizes the validator's utility
p, d, γ	The coefficients evaluating the observed performance

and the winning probability of a validator is positively associated with its optimal effort on computing. In other words, a validator can only achieve its expected ranking by maintaining the ex-ante optimal effort level. This assumption can assist with simplifying the model design and calculation. In the following sections, we present the tournament model design, the optimal solution of tournament, the adverse selection model design and the optimal solution of adverse selection model, respectively.

2.4.1 Tournament Model Design

As we analyze in Table (5), the observed performance order and the type order are opposite. That is, a validator with longer completion time of task will obtain a lower reward. Besides, by introducing the adverse selection concept, we first classify the observable time into different ranks, which simplifies the design of tournament. According to the utility function (2) described in the system model, we redefine this function for the validator, which is given by

$$\omega(\mathbb{R}_1) = \frac{(\mathbb{R}_1)^p}{p}, \tag{43}$$

where p is the power coefficient and $0 < p < 1$, $\mathbb{R}_1$ is the reward determined in the tournament model. In this model, we assume that the reward is a convex function of effort e. For simplicity, we replace e with the observed performance (time) t, then we

define $\mathbb{R}_1 = (d/t)^\gamma$, where d is a predefined parameter and $\gamma > 1$. So we have

$$\omega(\frac{1}{t}) = \frac{\gamma d^{\gamma p}(\frac{1}{t})^{\gamma p}}{\gamma p}. \tag{44}$$

Then we have the coefficient of absolute risk aversion for the validators, which is expressed as

$$A(\frac{1}{t}) = -\frac{\omega''}{\omega'} = \frac{1-\gamma p}{1/t}. \tag{45}$$

When we set $\gamma p < 1$, as the increase of performance, the risk aversion coefficient decreases. In this pure model, the virtual cost of stake and the penalty factor is omitted, i.e., $\xi(s) = 0$ and $\varphi(\theta, \mathbb{R}, e) = 0$. Finally, we derive the expected utility function U'_V in the tournament model, which is expressed as

$$U'_V = \sum_{j=1}^{\mathbb{K}} \big\{\omega(\mathbb{R}_{1,j}) P(rank = j) - \psi(e_j)\big\}, \tag{46}$$

where $P(rank = j)$ represents the probability that a certain effort level is in the j-th order of all the levels. It also represents the $(\mathbb{K}+1-j)$-th order of all the time intervals. Thus, we have the probability distribution function, which is given by

$$\begin{aligned} P(rank = j) &= P\left\{(\mathbb{K}-j)\Delta t \le t \le (\mathbb{K}+1-j)\Delta t\right\} \\ &= \int_{(\mathbb{K}-j)\Delta t}^{(\mathbb{K}+1-j)\Delta t} f(t; e^{-1}) dt, \end{aligned} \tag{47}$$

where $f(t; e^{-1})$ is the probability density function of time, and e^{-1} is the expected value of the distribution.

According to the system model, we set $\pi(\cdot) = ge = g/\int t f(t; e^{-1}) dt$ and $\mu(\mathbb{R}) = \mathbb{R}_1$, where g is a pre-defined evaluation coefficient. Then the optimization problem is expressed as

$$\max_{(\mathbb{R}_1, \mathbb{E})} \quad U'_{BC} = \frac{g}{\int t f(t; e^{-1}) dt} - \mathbb{R}_1, \tag{48}$$

s.t.

$$(a) \quad \mathbb{E} = arg\max_{e} \int v(\mathbb{B}) f(t; e^{-1}) dt - \psi(e),$$

$$(b) \quad \omega(\mathbb{R}_1) \int_{\Delta t} f(t; \mathbb{E}^{-1}) dt - \psi(\mathbb{E}) \geq \bar{u}.$$

where $\bar{u}$ is the minimal and acceptable utility of a validator node.

2.4.2 Optimal Solution of Tournament

From the rational standpoint, all the validator nodes try to achieve a certain rank order in their power, in order to obtain the maximum utilities. In general, we assume the distribution of the number of task completions per unit of time by validators is in conformity with the Exponential Distribution, i.e., $F(t; e^{-1}) \sim E(e^{-1})$, and set $\psi(e) = \frac{c}{2}(e^2)$. Therefore, we can get the optimal effort choice for different rank order by taking the first derivative of the validators' utility function, which is given by

$$\sum_{j=1}^{\mathbb{K}} \left\{ \frac{\partial \omega(\mathbb{R}_{1,j}) P(rank = j)}{\partial e_j} - \psi'(e_j) \right\} = 0. \tag{49}$$

Due to the design of the joint model, the validators belong to the same type will share the same reward, so we set a step function $W(\mathbb{R}_{1,j}) = \frac{h_1(\frac{1}{t_{\mathbb{K}-j+1}})^{h_2}}{h_2}$ to replace the evaluation reward function, where $h_1 = \gamma d^{\gamma p}$, $h_2 = \gamma p$ and $t_{\mathbb{K}-j}$ denotes the starting point value of time interval $\Delta t_{\mathbb{K}-j+1}$. The equation (49) can be given by

$$\frac{h_1(\frac{1}{t_{\mathbb{K}-j+1}})^{h_2}}{h_2} \frac{\partial P(rank = j)}{\partial e_j} - ce_j = 0. \tag{50}$$

Then we will get all the optimal effort choice e_j^* for the different performance of validator nodes by utilizing mathematical tool, such as Matlab.

2.4.3 Problem of Adverse Selection

Since the different time intervals correspond to the different optimal efforts, suppose there are $\mathbb{K}$ time intervals. The type θ is expressed as

$$\theta_i = ae_i, \quad \forall i \in \{1, ..., \mathbb{K}\}, \tag{51}$$

where a is a pre-defined parameter, and e_i is the same as e_i^*.

With the system model under consideration, we present the type-i validator node's utility function under the adverse selection model in the following equation,

$$U_{V(i)} = \theta_i \omega(\mathbb{R}_i) - \eta s_i - \rho(\mathbb{R}_i) P_i(t_i \geq T_{max}), \tag{52}$$

where $\mathbb{R}_i = \mathbb{R}_{1,i} + \mathbb{R}_{2,i}$, $\mathbb{R}_{1,i}$ is determined by the tournament model, $\mathbb{R}_{2,i}$ is the reward determined by the adverse model, η is the virtual unit cost of stake value, s_i denotes the average stake value threshold of type-i validator nodes, $\rho(\cdot)$ denotes the penalty according to the reward, and $P_i(t \geq T_{max}) = 1 - F(T_{max}; e_i^{-1})$, which is the probability of submission delay. Based on the contract theory, for $\forall i, j \in \{1, ..., \mathbb{K}\}$ and $i \neq j$, the **IR** constraint is: $U_{V(i)}(\mathbb{R}_i, s_i) \geq 0$, and the **IC** constraint is: $U_{V(i)}(\mathbb{R}_i, s_i) \geq U_{V(i)}(\mathbb{R}_j, s_j)$.

2.4.4 Utility Function of Beacon Chain

Let λ_i denote the prior distribution probability of type i. According to all types of validator nodes, $\forall i \in \{1, ..., \mathbb{K}\}$, the objective of the Beacon Chain is to maximize the expected utility function. This means higher ranked validator nodes with higher stake value are more desired. According to the system model, we set $\pi(\cdot) = g_1(\theta_i)^{z_1} + g_2(s_i)^{z_2}$. With the **IR**, **IC** constraints and the monotonicity condition, the optimization problem is expressed as

$$\max_{(\mathbb{R}_j, s_i)} \quad U_{BC} = \sum_{i=1}^{\mathbb{N}} \lambda_i \Big(g_1(\theta_i)^{z_1} + g_2(s_i)^{z_2} - \mu \mathbb{R}_i \Big), \tag{53}$$

s.t.

$$(a) \quad U_{V(i)}(\mathbb{R}_i, s_i) \geq 0,$$

$$(b) \quad U_{V(i)}(\mathbb{R}_i, s_i) \geq U_{V(i)}(\mathbb{R}_j, s_j),$$

$$(c) \quad \theta_1 < \ldots < \theta_i < \ldots < \theta_{\mathbb{K}},$$

where g_1, g_2, z_1, z_2 and μ are pre-defined parameters.

2.4.5 Optimal Solution of Adverse Selection

In order to solve the problem (53) formulated in Section 2.4.4, we first prove the monotonicity condition, and then reduce the number of the constraints by showing the proofs of IC and IR constraints. The details are listed in the following steps.

Lemma 1 *(Monotonicity) For any contract* $(\mathbb{R}_i, s_i)$*, given* $\theta_i \geq \theta_j$ *and* $\mathbb{R}_i \geq \mathbb{R}_j$*,* s_i *and* s_j *will also satisfy the monotonic relation, that is* $s_i \geq s_j$*.*

Proof 1 *According to the IC constraints of different types of validator nodes, we can obtain*

$$U_{V(j)}(\mathbb{R}_j, s_j) \geq U_{V(j)}(\mathbb{R}_i, s_i). \tag{54}$$

After transforming the inequation (54), we have

$$\eta(s_i - s_j) \geq (\theta_j \omega(\mathbb{R}_i) - \rho(\mathbb{R}_i)P_i) - (\theta_j \omega(\mathbb{R}_j) - \rho(\mathbb{R}_j)P_j, \tag{55}$$

then we set a new function $h(\mathbb{R}_i) = \theta_j \omega(\mathbb{R}_i) - \rho(\mathbb{R}_i)P_i)$*, and take the first derivative of* $h(\mathbb{B}_i)$*, which is given by*

$$h'(\mathbb{R}_i) = \theta_j \omega'(\mathbb{R}_i) - \rho'(\mathbb{R}_i)P_i. \tag{56}$$

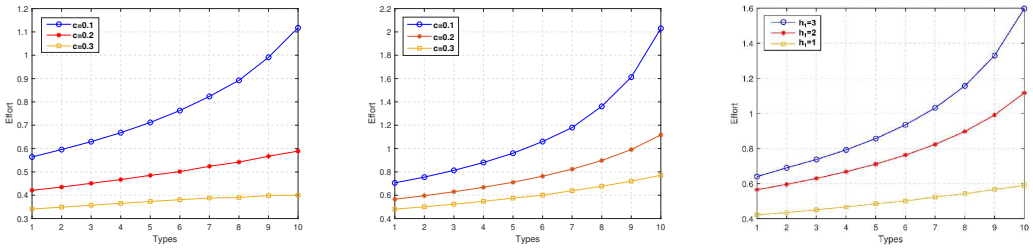
If $h'(\mathbb{R}_i) > 0$*, we have* $\theta_j v'(\mathbb{R}_i) > \rho'(\mathbb{R}_i)P_i$*. Note that the minimum value of* θ_j *is* θ_1*, i.e.,* $j = 1$*, and the maximum value of* P_i *is 1. So we set the slope of function* $\omega(\cdot)$ *and* $\rho(\cdot)$ *as* $\theta_1 \omega' > \rho'$*. Now the equation (55) is rewritten as*

$$\eta(s_i - s_j) \geq h(\mathbb{R}_i) - h(\mathbb{R}_j). \tag{57}$$

It is clear that when $\theta_1 \omega' > \rho'$*, we have* $h'(\mathbb{R}_i) > 0$*. For given* $\mathbb{R}_i > \mathbb{R}_j$*, we have* $h(\mathbb{R}_i) - h(\mathbb{R}_j) > 0$*, that means* $s_i > s_j$ *exists when* $w_2 \neq 0$*. Therefore, we complete the proof of monotonic relation.* ■

- **Reduction of IR Constraints**

The Monotonicity Condition already holds due to the solutions of tournament. Here the proof is omitted. Since we have proved the reduction of IR and IC constraints in



(a) $h_1 = 1$. (b) $h_2 = 2$. (c) c=0.2.

Figure 14: The impact of cost c and evaluation parameter h_1 on the optimal effort choice e_i^*.

the former section, we only give the brief description toward the proofs in the following description. The IR constraints of all types will hold if the IR constraint of type 1 can be satisfied. According to the IC constraints in (53), for type $\forall i \in \{1, ..., \mathbb{K}\}$, we first have

$$U_{V(i)}(\mathbb{R}_i, s_i) \geq U_{V(i)}(\mathbb{R}_1, s_1). \tag{58}$$

Then based on the monotonicity conditions, we have the inequation, which are expressed as

$$U_{V(i)}(\mathbb{R}_1, s_1) \geq U_{V(1)}(\mathbb{R}_1, s_1) \tag{59}$$

$$\text{and} \quad U_{V(i+1)}(\mathbb{R}_i, s_i) \geq U_{V(i+1)}(\mathbb{R}_{i-1}, s_{i-1}). \tag{60}$$

- **Reduction of IC Constraints**

Consider three adjacent types, i.e., type $i-1$, type i and type $i+1$, which follow $\forall i \in \{2, ..., \mathbb{K}-1\}$. According to the IC constraints, then we have two inequalities, which are expressed as

$$U_{V(i+1)}(\mathbb{R}_{i+1}, s_{i+1}) \geq U_{V(i+1)}(\mathbb{R}_i, s_i) \tag{61}$$

$$\text{and} \quad U_{V(i)}(\mathbb{R}_i, s_i) \geq U_{V(i)}(\mathbb{R}_{i-1}, s_{i-1}). \tag{62}$$

We can easily get $(\theta_{i+1} - \theta_i)\omega(\mathbb{R}_i) \geq (\theta_{i+1} - \theta_i)v(\mathbb{R}_{i-1})$ by virtue of the monotonic conditions and add it to (62), and then we have a new inequation. Based on (61) and (60), we have

$$U_{V(i+1)}(\mathbb{R}_{i+1}, s_{i+1}) \geq U_{V(i+1)}(\mathbb{R}_{i-1}, s_{i-1}). \tag{63}$$

The rest of the details are all the same as that of previous Section 2.3.2. With the reduced constraints, the optimization problem can be expressed as

$$\max_{(\mathbb{R}_j, s_i)} \quad U_{BC} = \sum_{i=1}^{\mathbb{N}} \lambda_i \bigg(g_1(\theta_i)^{z_1} + g_2(S_i)^{z_2} - \mu \mathbb{R}_i \bigg), \tag{64}$$

s.t.

$$(a) \quad U_{V(1)}(\mathbb{R}_1, s_1) = 0,$$

$$(b) \quad U_{V(i)}(\mathbb{R}_i, s_i) = U_{V(i)}(\mathbb{R}_{i-1}, s_{i-1}),$$

$$(c) \quad \theta_1 < ... < \theta_i < ... < \theta_{\mathbb{K}}.$$

Based on the $\mathbb{K}$ IC constraints of (64), we add all the equations that are numbered from 1 to i together and get

$$U_{V(i)} = \sum_{j=1}^{i-1} \Delta_j + \rho(\mathbb{R}_1) P_1, \tag{65}$$

where $\Delta_j = \omega(\mathbb{R}_j)(\theta_{j+1} - \theta_j)$. For simplicity, we set $\omega(\mathbb{R}_j) = \mathbb{R}_j = \mathbb{R}_{1,j} + \mathbb{R}_{2,j}$. According to the equation (52), we derive an expression for s_i, which is given by

$$\mathbb{R}_{2,i} = \frac{\eta s_i + \sum_{j=1}^{i-1}}{\theta_i} - \mathbb{R}_{1,j} \tag{66}$$

$$\text{and} \quad \mathbb{R}_{2,1} = \frac{\eta s_1 + 2\mathbb{R}_{1,1} P_1 - \theta_1 \mathbb{R}_{1,1}}{\theta_1 - 2P_1}. \tag{67}$$

After substituting (66) into (64), (64) will be converted into a new problem, we differentiate U_{BC} with respect to S_i and then get $\frac{\partial U_{BC}}{\partial s_i} = g_2 z_2 (s_i)^{z_2 - 1} - \frac{\eta}{\theta_i}$. Next, by differentiating $\frac{\partial U_{BC}}{\partial \mathbb{B}_i}$ with respect to $\mathbb{B}_i$, we have

$$\frac{\partial^2 U_{BC}}{\partial (s_i)^2} = g_2 z_2 (z_2 - 1)(s_i)^{z_2 - 2}. \tag{68}$$

When we set $0 < z_2 < 1$, we have $\frac{\partial^2 U_{BC}}{\partial (s_i)^2} < 0$. Obviously, we can conclude that the problem is a convex problem. Therefore, we can get the optimal solution $(\mathbb{R}_i^*, s_i^*)$ by

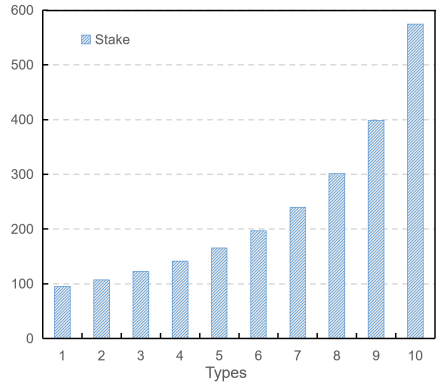


Figure 15: The minimum thresholds of stake for all the types.

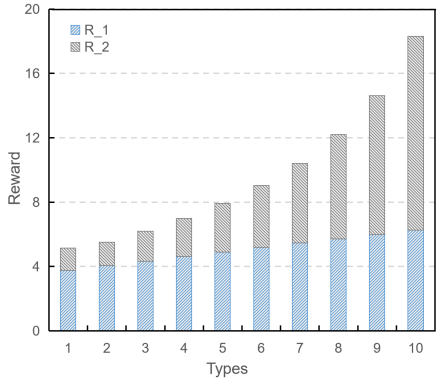


Figure 16: Two kinds of reward for all the types.

setting the first derivative to zero, we have

$$s_i^* = \left(\frac{\eta}{g_2 z_2 \theta_1}\right)^{\frac{1}{z_2 - 1}}. \tag{69}$$

Then the following optimal results can be obtained by standard optimization solvers.

2.4.6 Simulation Results and Numerical Analysis

In this section, we present the simulation results to prove our optimal solutions, and describe the effect of different parameters on these solutions. According to the optimal solution of Section 2.4.2, two important parameters h_1 and c will affect the reward $\mathbb{R}_1$ and effort. So we first set h_1 as a fixed value and set $h_2 = 0.8$, and then from Fig. 14(a) and Fig. 14(b), we see the optimal effort choice e_i^* decreases as the cost c increases. Next, set c as a fixed value (e.g., $c = 0.2$), from Fig. 14(c), we conclude that the optimal effort choice e_i^* increases with the parameter h_1. For the three figures, it is clear that the optimal effort choices for different types meet the monotonicity condition.

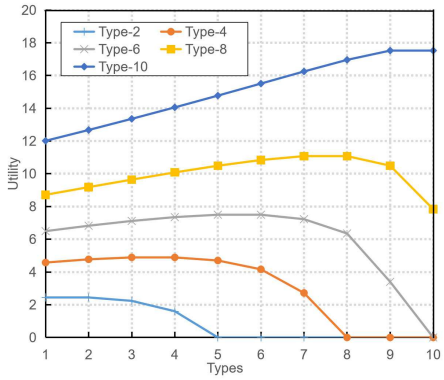


Figure 17: Utilities of validators when signing the different contracts.

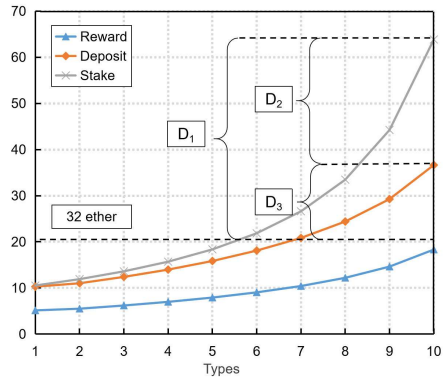


Figure 18: Comparison of reward, deposit, stake and the fixed deposit (e.g., 32 ether)

For the following simulations, we focus on the minimum threshold of stake for different types, the total rewards and the utility of different contracts. The optimal results in the adverse model are obtained based on the tournament model. We first set $a = 30$, and compare the result with the optimal effort curve (i.e., $h_1 = 2$) as shown in Fig. 14(c). So we have the optimal stake results for different types, from Fig. 15, the greater type, the more stake is required. Fig. 16 shows the variation of the two kinds of rewards that are determined by the proposed joint model. Both of them, as well as their sum, are consistent with the monotonicity condition. In order to simplify Fig. 17, we set all the utility values that are lower than zero to be zero. We can observe that the validators have the maximum utilities only when choosing the contract designed for their own, which proves the IC constraint. Besides, all these maximum incentives are positive, which explains the IR constraint.

At last, the Fig. 18 represents the comparison of the proposed scheme and the fixed-deposit scheme (e.g., Ethereum 2.0). We adopt the stake curve presented in Fig.

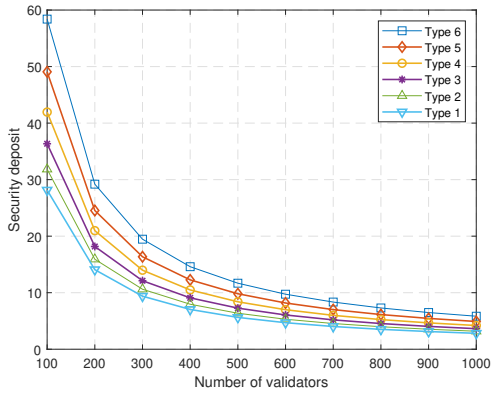


Figure 19: Security deposit pricing when the total number of validators varies.

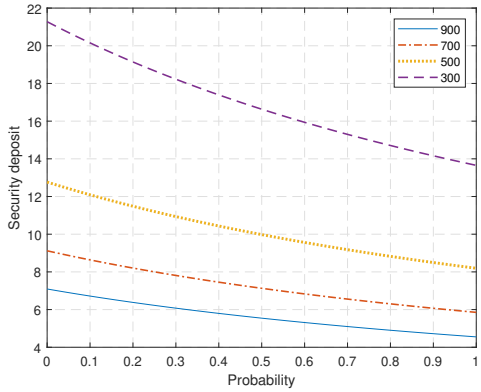


Figure 20: Security deposit pricing when the probability of type-6 validators varies.

15 and the reward curve presented in Fig. 16. Without loss of generality, we scale down the stake to make it accommodate the reward in the joint model. In Ethereum 2.0, only the validators whose stakes are greater than 32 ether can afford it and are eligible for operation and maintenance within a blockchain. Just like the assumption that we make in the Section (2.3), 32-ether blocks out those validators whose types are less than 7 and leads to their disqualification from the participation of a blockchain. Not only that, 32-ether policy cannot ensure an adequate security incentive for the system against the higher type validators. Take the type-10 validator as an example, let D_1 denote the difference between the stake and 32 ether, D_2 denote the difference between the proposed deposit and stake, and D_3 denote the difference between the proposed deposit and 32 ether. Thus, it is clear to see that, due to $D_2 < D_1$ and $D_3 > 0$, this model can also provide the better security incentive and economic incentive. The reward, deposit and stake of all types also follow the rule we analyze before, i.e.,

reward<deposit<stake. Compared with the joint model in Section (2.3), the stake has become a criterion of access that is defined by the contracts rather than a constant identity. That means, the deposit determined in this model may block some cash-strapped participants. However, this model mainly focuses on an effort-oriented scenario, unlike the stake-oriented scenario described in Section (2.3), it provides a novel incentive design to balance the security incentive and economic incentive, and makes a trade-off between the number and the incentive of participants. Similarly, we set type-7 as the threshold type of validators, which means only the validators marked with the types higher than 7 can afford the fixed deposit in Ethereum 2.0. However, the proposed model allows a wide range of validators to participate in the PoS blockchain. For the higher-type ($i \geq 7$) validators, they need to submit a higher deposit than Ethereum 2.0. The lower-type ($i < 7$) validators can contribute a minor amount of deposits while they are prohibited from participating in Ethereum 2.0. Finally, we can reach the same conclusion as in section 2.3.3, i.e., the security of this model is not decreased compared to Ethereum 2.0.

Analogously, there also exists an extreme case of this joint scheme. Through the calculation, if none of the required stakes is greater than 32-ether but $\sum_{i=1}^{\mathbb{N}} n\lambda_i s_i \geq \mathbb{D}$ exists, we can price the security deposit according to the assignment algorithm 1. All the validators' locked stake is less than 32-ether, which can be mapped into type-i ($i \in \{1, \cdots, 6\}$) of the proposed scheme. We set an exchange rate to simplify the calculation, which is type-7 validators' stake to 32-ether. With the assumption of uniform distribution of validators, we will first examine the total number of validators' impact on security deposit pricing. According to the security incentive in inequality (40) and the assignment in (41), we have the security deposit for all types of validators in Fig.19. However, some pricing points are invalid due to the constraint: $d_i \leq s_i$. Given the prior probability and stake of validators, we can obtain the minimum number of validators through the equations in (40) and (42). From Fig. 19, we can intuitively conclude that all validators' deposits are decreasing in the total number of validators. More validators participating can lead to a smaller share for an individual. Fig. 19 also shows that the security deposit pricing is increasing in the validators' types, which indicates that

the assignment is consistent with the monotonicity condition. We also explore how the probability of a specific type will impact security deposit pricing. Take type-6 validators as an example. From Fig. 20, we can observe that the security deposits are decreasing in the probability of validators (except for $\lambda_6 = 0$). The idea behind these two figures is similar: the more validators there are, the smaller share there is. Nevertheless, sometimes there may exist some unfeasible pricing points. We can easily filter out these points by using the inequality in (40). Consequently, given any stake value, we can determine the minimum number and feasible probability of validators based on the above two figures.

2.5 Application and Implementation

The proposed work can be applied in any blockchain network with an administrative party, where the 'deposit' can represent any proof or token in the circulation of such a scenario. To implement the proposed models in a practical blockchain system, we can consider the following steps. First, as the administrator in the blockchain network, the beacon chain can obtain the distribution of potential validators' stake and performance without any private information and classify them into different types. Based on the practical scenario, the administrator can determine which joint model is more satisfied. Second, the administrator starts to design the contract with the historical data and apply the empirical data into the proposed model, adjusting the system parameters and coefficients to test the optimal results until they are consistent with practice. Then he can obtain the feasible system parameters. Next, the beacon chain will broadcast the contract set to all the potential validators. They will evaluate the utilities provided by all the contracts by using their specific private information and decide to accept a particular contract or not. Last, the validators will exert efforts to fulfill their contractual obligations. By observing the performances, the beacon chain will evaluate the efforts and pay the validators with stipulated rewards. If any violation occurs, the corresponding validator's deposit will be slashed entirely.

2.6 Summary

In this work, we first gave the explanation of security incentive and economic incentive, and then analyzed the relationship of the features (i.e., stake, performance and effort), reward and deposit of a participant. With two complicated scenarios we discussed before, we proposed two joint models under the contract theory framework to balance the security incentive and economic incentive. As a result, both the economic incentive and the security incentive are kept as the same time. Compared with the mainly emphasized security incentive in Ethereum 2.0, the simulation results exactly show that the proposed optimal contract-based approach can achieve a better balance between the security incentive and economic incentive.

3 Resist Attack using Cyber-insurance Framework

Sharding is a promising solution to achieving scalability within the blockchain network. A sharded blockchain network consists of a beacon chain and several committees powered by the participants (i.e. validators) through the Proof-of-Stake (PoS) consensus protocol. Efficient and scalable as it can be, the sharded blockchain based on PoS is vulnerable to discouragement attack. A discouragement attack occurs when malicious validators censor messages to discourage validators from participating in the network. Furthermore, no rate-limiting validator rotation (enter/exit quickly) makes it more challenging to detect such an attack. In this work, considering the undetermined rotation and the discouragement attack, we render the beacon chain an intermediary, allowing the beacon chain to interact with validators and the cyber-insurer, aiming to encourage the validators' stable rotation through insurance compensation. Specifically, we utilize a two-stage hierarchical game-based model to formulate the complicated interactions under the cyber insurance framework. In the first stage, the beacon chain develops compensatory strategies according to the insurer's profile. In the second stage, the beacon chain designs a series of contracts for validators, including insurance items, compensatory strategies, and rotation requirements. Consequently, the proposed scheme incentivizes validators to remain online by transferring risk to the cyber insurer and enables the sharded blockchain network to weaken the attack's impact through validators' stable rotation. This work presents closed-form solutions for the proposed model, in which the beacon chain and the cyber insurer can gain maximized profits. The simulations demonstrate the feasibility and superiority of the proposed model.

3.1 Introduction

Blockchain is an ingenious invention of Nakamoto and is described in the remarkable project [2]. It is a permissionless platform with the characteristics of decentralization, tamper-resistance and transparency [1]. With the advent of Bitcoin (BTC) [96], blockchain technology has acquired significant attention. Ethereum is another world's leading programmable project [97] based on the blockchain technology framework [12].

Unlike Bitcoin, mostly focusing on financial issues, Ethereum aims to be a "World Computer", which allows everyone to be a developer to write its own code and create new kinds of applications [12]. A blockchain is constructed by a series of undeniable blocks, where each block is generated and confirmed by the different parties. These blocks are connected before and after within one chain. The core technology of coordinating all the participants across the distributed network is called consensus protocol. The first practicable consensus protocol in the blockchain framework is known as the Proof of Work (PoW) [2], in which miners can only win the opportunity of mining block by competing in hash rate against others. In the early development stage of the blockchain, PoW indeed provides the benefits, such as Denial-of-Service (DoS) attack defense and Sybil attack defense. The success of Bitcoin [97] has proved this point.

Considering the aggravation of hash rate competition causes a massive waste of resources, numerous researchers seek new alternatives that serve the same function. Proof of Stake (PoS) is first proposed in the Bitcoin Forum [69], i.e. the leader selection relies on the number of stakes rather than the computational resources. That means the leader selection of PoS follows a relative deterministic way compared with that of PoW, such as by turns. Moreover, there is no need to issue more rewards to compensate for energy consumption. As described above, the significant advantages of PoS can be summarized in three aspects: comparable decentralization, economic saving, and security [98].

With the extensive research on consensus protocols, more potential attacks are being studied. Cyber-attacks can take more implicit forms. Vitalik explores a new type of attack that may disrupt the whole blockchain network, which is called *Discouragement Attack* [99]. A discouragement attack means the attackers would act maliciously inside a consensus mechanism to gradually reduce other validators' revenue, even at a certain cost to themselves. The final purpose is to encourage the validators to drop out of the mechanism. It is worth noting that a discouragement attack is the cheapest way to corrupt the incentive scheme of a PoS based blockchain. In the best case, attackers can do this without losing a cent only by censoring the honest nodes and driving their reward to zero. Consequently, they will encourage rational validators to drop out of the

system [100].

Recently, Vitalik published a new idea about the validator sharding set update [101], which prevented the validators from withdrawing. In the previous design, every validator is able to exit/enter the sharding committees at the end of each round [102]. That means the malicious validators can perform a large-scale attack without being detected because the system cannot perform the detection in a short time. A better way is to set a withdrawal delay, which means the validators must wait in a queue for withdrawing before being free to exit the committee. In a nutshell, this idea is to make time for the system to detect the attack and malicious nodes, allowing the system to prevent the discouragement attack. However, there are still some open questions, i.e. determining the withdrawal delay for all the validators and selecting the validators in a queue. Considering the risk introduced by the discouragement attack, the blockchain network requires an appropriate incentive scheme to encourage the validators to stay online and neutralize the whole network's risk.

3.1.1 Related Work

As the core of blockchain technology, the consensus protocol has been the research priority for blockchain experts. With the advantage of energy-efficient, PoS is increasingly the focus of research attention. Here are some PoS schemes but are not completely different from the framework of PoW, e.g. Proof of Activity [103], Proof of Burn [104] and Proof of Luck [105].

The joint efforts from both industry and academia areas make great strides in moving blockchain technology forward, which brings opportunities and benefits for a wide range of areas, including finance, business, industry. IBM developed a blockchain platform based on Hyperledger Fabric [106], providing solutions and services for multiple institutions [107], including financial services, supply chain, manufacturing, media, and entertainment, retail. The applications of blockchain technology to some hot industrial areas, e.g. the Internet of Things, also attracted extensive attention, e.g. Atonomi [65], Chain of Things [66] and IoTeX [108].

With the decentralized, tamper-resistance nature, the blockchain-mediated application presents a vast potential. Choo *et al.* [109] discuss the ongoing challenges when applying blockchain in industry, governments, and academia. Guan *et al.* [110] adopt the blockchain technology to securely aggregate and store the near-real-time data and preserve the users' privacy. Pal *et al.* [111] design a decentralized and asynchronous delegation model by using the blockchain technology and demonstrate the feasibility by using Ethereum private blockchain platform. Liang *et al.* [112] utilize the blockchain technology to establish a decentralized storage system, which realizes the dynamic storage and update and fast repair. Yang *et al.* [113] introduce the blockchain technology to help protect the topology privacy in Mobile Edge Computing (MEC). Marwa *et al.* [114] apply the blockchain to build up a privacy-preserving framework in the smart power networks.

Although blockchain technology presents various advantages, there still exist scalability and efficiency issues inside. Ethereum 2.0 [115] combines sharding technology and PoS, focusing on further improving network efficiency in a scalable way. A creative structure proposed by Ethereum Foundation is the beacon chain, introducing PoS into Ethereum, where there was only PoW before. All the data structures inside the beacon chain are akin to the public chain powered by Proof of Work. However, the different aspect is that the beacon chain is operated by a committee composed of many validators using Proof-of-Stake consensus. Such a committee is pseudo-randomly assigned to verify a shard of the current block. As the intermediary between the public chain and the numerous committees, the beacon chain is responsible for managing administrative transactions throughout the whole blockchain network with shardings, which include a registry of validator addresses, the state of each validator, attestations, and links to shards. The beacon chain can coordinate with the whole network and ensure a smooth transition between a pure PoW blockchain system to a pure PoS one.

However, the blockchain network deployed with PoS is vulnerable to discouragement attack, a critical threat to most of the PoS-based blockchains but is less studied. Saito [100] proposed a solution that can ensure the cost of producing a block fluctuates depending on the degree of support a node receives from the rest of the network. They

adopted a new workload measurement that is a derivative of the volume of transaction fees gathered from other nodes. Apart from developing a new consensus protocol, applying cryptography is an alternative. But it is virtually impossible to discriminate between attackers and honest validators inside a PoS mechanism, especially the attacker sacrificing short-term profit just looks like a victim. Moreover, relying on consensus development and cryptography application to deal with a specific cyber risk is exceptionally costly to an existing blockchain network.

One of the most effective ways to transfer cyber risk is cyber insurance. The market for cyber insurance has vitality spurred in recent years, which is expected to reach $5 billion by 2018 and exceed $7.5 billion by 2020 [116]. The market with colossal potential motivates more and more researchers to investigate it in various network scenarios. Khalili *et al.* [95] have explored the interdependent nature of cybersecurity and the latest Internet measurement for evaluating the security posture. They focus on the theoretical details more, the other promising works regarding cyber insurance, see, e.g. [117], [118], and [119]. Their "interdependent nature" idea does an excellent job explaining the relation between the entities and the networks, which can also apply to the participants and the blockchain networks. Feng *et al.* [41], adopt the cyber insurance tool to neutralize the cyber risk caused by double-spending in the blockchain network and model the problem as a two-stage Stackelberg game. Lu *et al.* [120] introduce cyber insurance to heterogeneous wireless networks (HWNs) and reviews the cyber risks of the enabling technologies for HWNs. As we discussed above, none of the works consider the discouragement attack.

3.1.2 Motivation and Contribution

Inspired by the above work, we explore the discouragement attack within the PoS mechanism in the blockchain networks with shards. We adopt cyber insurance as a risk-management to mitigate the risk and motivate the validators' online time (i.e. withdraw delay [101]). Owing to the anonymity nature and the weak leadership of the beacon chain, the problems of *hidden information* and *hidden action* coexist. Hidden information and hidden action are two terms specific to economics. In this work, the validators'

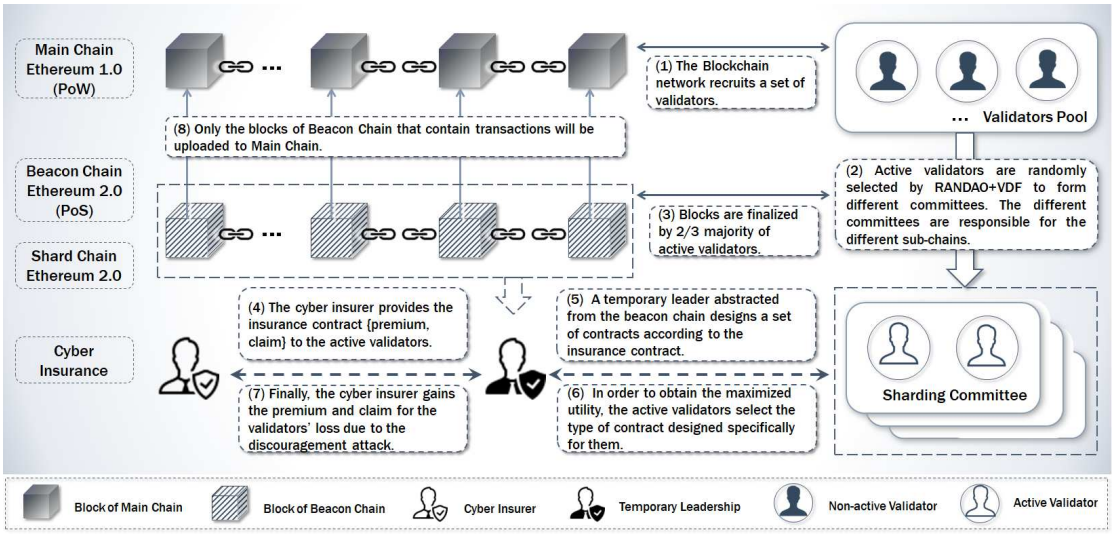


Figure 21: An overview of the sharded blockchain network with cyber insurance.

type information is the hidden information and their efforts after singing the contracts are the hidden action. Besides, considering the cyber insurer, the interactions of the three parties, i.e. cyber insurer, blockchain infrastructure provider, and validators, are complex and difficult to analyze. There is not an existing model that can be applied directly to formulate the problem. One practical model to set up a cyber insurance framework is the contract theory [44], which has been extensively studied, e.g. [51], [50], [121] and [122]. Specifically, we designed a cyber-insurance framework, modeling the interactions between the beacon chain and its underlying validators in [122] but without the role and effect of cyber-insurer. We did not take the interactions between cyber-insurer and blockchain into account and just assumed that cyber-insurer could gain nothing from the insurance, which is less applicable in reality. Therefore, we establish a two-stage hierarchical game model by combining the Stackelberg game and contract theory, formulating the complicated interactions among cyber-insurer, blockchain, and validators, and allowing the cyber-insurer and blockchain to obtain the maximized profits. First of all, we analyze the discouragement attack model and the expected loss for all kinds of validators (i.e. malicious, censored, and uncensored). Then, the cyber-insurer will take the lead in the upper sub-game of Stage I (more details regarding this model are given in Section 3.3), determining the premium and claim factor. Consider that the blockchain infrastructure provider follows the leader's rules and determines a discount factor for premium.In Stage II, the blockchain infrastructure provider designs a series

of contracts for participants, determining the contract components based on the attack model and cyber-insurer's premium and claim factor. By such a design, the information asymmetry can be overcome by contract theory [44]. As a result, the validator needs to pay only a discounted premium but is entitled to cyber insurance compensation. The cyber insurer needs to pay only a discounted claim but will obtain a full premium. It is the blockchain to compensate for the premium and claim for the cyber insurer and validators, respectively.

We summarize the main contributions of this work in the following.

- We utilize cyber insurance as risk-management (i.e. it works as an economic mechanism) that motivates the validators to be online and reduces their losses from the discouragement attack. The proposed scheme allows the blockchain to keep more validators with high online active participation for a required time, which contributes to the market value and makes time for the developers to detect discouragement attacks and malicious nodes.

- We propose a hierarchical game model by combing the Stackelberg game and contract theory together, analyzing the interactions among the cyber-insurer, blockchain infrastructure provider, and validators, and then formulating the problems in two stages.

- In the hierarchical game model, the cyber-insurer is the leader in upper Stage I, determining the premium and claim factor for the validators. Based on the cyber-insurer's strategy in Stage I, the blockchain infrastructure provider, as the follower in lower Stage II, determines the contract items, including discount and compensation factors. The validators would be given a set of insurance contracts in Stage II. Thus, we can maximize the cyber insurer and blockchain infrastructure provider's profit and determine the online duration for the validators.

- Detailed illustrations are given to show the solving process. Accordingly, we present the closed-form of the optimal premium, claim, and compensation parameters obtained through backward induction. With the extensive simulations, we compare the impacts of different key parameters on the optimal results and

show the profit differences between the proposed model and the benchmark model, demonstrating the feasibility and efficacy of the proposed model.

3.2 System Model

This section will introduce the discouragement attack model, the reward distribution function, the expected loss, and then present the utility models for the validators, blockchain, and the cyber insurer. The overview of the sharded blockchain with cyber insurance model can be referred to Fig. 21. We consider the market with one cyber insurer and one blockchain network. From the incentive perspective, the beacon chain tries to maximize the participation (i.e. delay and online deposits) of validators to hold its market value. The malicious ones would like to choose the contract with a small 'delay', but they cannot escape from the security review when waiting in the queue. The honest ones prefer contracts with a longer online time, which means a higher claim for the potential risk.

The core idea of discouragement attack is long-term censorship on honest participants, which the attackers conduct through the power of their collusion [99]. As long as some messages are published for audit, verification, or packaging, the attackers act illegally inside the PoS consensus mechanism to reduce other validators' revenue [123]. Fig. 22 depicts a comparison regarding the discouragement attack effects with/without cyber-insurance. According to the rewarding function in [99], the malicious users initiate censorship on these honest users to decrease the rewards in phase (1). Then in phase (2), the honest users realize that the rewards are slashed round after round, even though they vote honestly. Without enough incentives, these honest users will intend to drop out of the system in phase (3). Finally, the honest users' withdrawal enables the malicious users to launch much more severe attacks, such as the double-spending attack. However, cyber insurance can guarantee honest users' rewards by compensating for their loss in phases (a) and (b). With an unaffected reward, the honest users will tend to stay in the system for a longer time in phase (c), which forces the malicious one to drop out of the system due to the increasing cost.

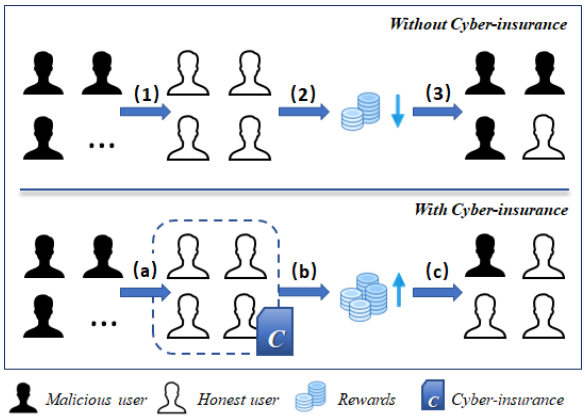


Figure 22: How does the cyber-insurance work when discouragement attack occurs.

3.2.1 Discouragement Attack Model

Discouragement attack means that the malicious validators are controlled by the attacker acting illegally inside the consensus mechanism in order to reduce other validators' revenue. Although it is virtually impossible to censor any transaction on a blockchain network, censorship can occur when signing the signatures [123]. Take the sharded blockchain as an example. In [99], there are N validators in a single committee, sharing a maximum total reward of R in each round. If the total number of whom signing the messages is M ($M \leq N$), then they can earn a reward of $\frac{R}{N} \cdot \frac{M}{N}$ because of signing a message. When the attackers collude to stop including messages from some other validators (not all validators), the censored validators will gain nothing because their signatures are not included in the finalized results [124]. Following that M is virtually reduced, both the reward of attackers and non-censored validators decreases. Even though the attackers are forced to inflict economic damage to themselves due to censorship, they still have the incentive to expect long-term profit in the future. Consequently, the victims realize that their incentive is decreasing and then exit the blockchain network. A possible, as well as the worst case, is that most honest validators drop out of the network with all attackers remaining, allowing the attackers to easily launch the double-spending attack or any other severe attacks on the chain manipulating the finalized blocks.

To better analyze the attack model, [99] introduces a useful concept called *griefing*

factor and a reward distribution function with the bounded griefing factor. The discouragement attack can be classified into two types: majority attack and minority attack. A majority attack means the attacker has a greater power to control a majority of the validators, while the minority attack is the opposite case. In a majority attack, the validators can be identified as three types: the malicious validators, the censored validators, and the uncensored validators. In a minority attack, there are only two types of validators: the malicious validators and the remaining validators. We will explain the reward distribution and loss function under the majority attack in the following. Suppose there are N validators in committee $\mathcal{I}$, $\hat{n}$ malicious validators, $\hat{k}$ censored validators, $N-\hat{k}-\hat{n}$ uncensored validators, and the total reward for each round is R. Each validator contributing to a PoS blockchain's work earns R/N if no one is malicious. But if there exist malicious validators, according to Vitalik's idea [125], we have the reward distribution function: $\frac{R(N-\hat{k})}{N^2}$. Thus, we have the loss function l_1 for each censored validator and the loss function l_2 for each uncensored validator, which are expressed as

$$l_1 = \frac{R}{N} \tag{70}$$

$$\text{and} \quad l_2 = \frac{R}{N} - \frac{R(N-\hat{k})}{N^2} = \frac{R\hat{k}}{N^2}. \tag{71}$$

Thus, we have the expected loss function, which is given by

$$\mathcal{L} = \left\{ \frac{\hat{k}}{(N-\hat{n})} l_1 + \frac{(N-\hat{k}-\hat{n})}{(N-\hat{n})} l_2 \right\}, \tag{72}$$

where $\hat{k}/(N-\hat{n})$ denotes the probability of being censored, and $(N-\hat{k}-\hat{n})/(N-\hat{n})$ denotes the probability of not being censored.

3.2.2 Cyber Insurer Utility Model

This work will first classify all the validators into the different types according to their activeness. For the validators of type-i, the cyber insurer determines the premium p_i and coverage factor β_i. For the cyber insurer, it can get profit from the gap between the premium and the claim. The premium p_i is given by the type-i validators, as well as the blockchain infrastructure provider. The claim is determined by the loss $\mathcal{L}$ and

the coverage factor β_i. Thus, the cyber insurer's utility obtained from a validator can be expressed by

$$U_{c,i}(p_i, \beta_i) = p_i - \beta_i \mathcal{L}. \tag{73}$$

3.2.3 Blockchain Utility Model

The blockchain network is decentralized, which means no central trusted party is dealing with the administrative transaction. However, it is not the same case in a blockchain network with shards. According to [12], the beacon chain consists of the dedicated blocks for recording the administrative transactions. The block managers can be considered to be temporary leaders. Moreover, the contract items and the related details on execution can be encoded into a smart contract.

The blockchain infrastructure provider acts as a medium between the cyber insurer and the validators. It follows the cyber insurer's strategies and determines a series of compensation strategies (*discount factor*,*compensation factor*) for all types validators to further incentivize their online time d_i. By such design, the whole blockchain network's value can be increased by motivating the validators with higher activeness to stay online for a longer time. The utility function is given by

$$U_{b,i} = \Pi(\theta_i, d_i) - \alpha_i p_i - \Theta(u\beta_i \mathcal{L}), \tag{74}$$

where $\Pi(\cdot)$ is an increasing function that is used for calculating the revenue from validators' delay d_i and activeness θ_i, the second term is the compensatory premium for the cyber insurer with the discount factor α_i, and the last term is the compensatory claim for the validators with the compensation factor u.

3.2.4 Validator Utility Model

The validators are randomly selected from the validator pool. Thus, there exists a variety of validators with different activeness. We first classify the validators into different types: type-1, type-2, ..., type-i, ..., type-$\mathbb{N}$. The classification criterion is based on their history activeness in the blockchain. In order to analyze the activeness of the different validators without loss of generality, we model the activeness of all the

validators as the Normal Distribution $\hat{\mathcal{A}}$ with mean μ and variance σ^2. For a certain range of activeness $[a_i, a_{i+1})$, we model it as a truncated normal distribution $\mathcal{A}_i$, which is expressed as

$$\mathcal{A}_i(x; \mu, \sigma, a_i, a_{i+1}) = \frac{\frac{1}{\sigma}\phi(\frac{x-\mu}{\sigma})}{\Phi(\frac{a_{i+1}-\mu}{\sigma}) - \Phi(\frac{a_i-\mu}{\sigma})}, \tag{75}$$

where the validators whose activeness lies in interval $[a_i, a_{i+1})$ belong to type i. The activeness $\hat{a}_i$ observed and analyzed by the blockchain infrastructure provider contains a random noise, which is given by

$$\pi(\hat{a}_i, \mathcal{N}) = \hat{a}_i - \mathcal{N}, \tag{76}$$

where $\mathcal{N}$ is the network delay, a zero mean Gaussian noise with variance σ'.

All validators have options of not buying a contract, directly buying an insurance contract from the cyber-insurer, or buying a mixed contract from the blockchain. Intuitively, an honest and rational validator tends to buy a contract (either from the cyber-insurer or blockchain) to prevent monetary loss caused by discouragement attacks. The single contract from the cyber-insurer allows the buyers free to enter and exit the blockchain network. By contrast, a mixed contract provided by blockchain can additionally offer a discount factor and a compensation factor in exchange within a fixed length for the validator being online. For incentive reasons, validators buying mixed contracts from blockchain is profitable. Even having to stay online for a fixed duration, the validators' benefits will be protected by the mixed contract with a lower cost and a higher claim. We will present and explain the validator's utility of choosing contracts of different categories in the following.

A. No Contract

If the validator does not sign any contract, it will bear the total cost of its effort as well as the potential loss caused by the discouragement attack. Therefore, the type-i validator's utility per unit time is

$$u_i^- = -c[\pi(\hat{a}_i)]^2 - \mathcal{L}. \tag{77}$$

Thus, the expected utility without contract is given by $U_i^- = E(-c[\pi(\hat{a}_i)]^2 - \mathcal{L}) = -c\hat{a}_i^2 + c\sigma'^2 - \mathcal{L}$. Obviously, due to the potential loss, the validator would choose to do nothing to lower the total cost. Note that "doing nothing" means the validator will not do any other activities that the consensus protocol does not require.

B. With Single Contract

If the validator signs the contract with the cyber insurer directly, it can only obtain the insurance contract (p_i, β_i), where p_i is the premium and β_i is the claim factor with $0 \leq \beta \leq 1$. Without the compensation factors, the validator receives the profit is given by

$$\mathbb{P}_i^- = -p_i - \mathcal{L} + \beta_i \mathcal{L}. \tag{78}$$

C. With Mixed Contract

This work considers the blockchain infrastructure provider to be an intermediary to interact with the cyber insurer and the validator, further motivating the validators by providing the mixed contract with a discount factor and a compensation factor. Thus, the type-i validators will be provided a series of mixed contracts $\{[p_i, \beta_i], [\alpha_i, u\beta_i, d_i]\}$, where α_i is the discount factor with $0 \leq \alpha \leq 1$, u is the compensation factor with $0 \leq u \leq 1$ and d_i denotes the delay (i.e. the validator's online time), it means the validator will receive a discount of the insurance contract through keeping active online. For a base premium p and a loss $\mathcal{L}$, the validator receives the profit from the mixed contract $(p_i, \alpha_i, \beta_i, d_i)$ is expressed as

$$\mathbb{P}_i^+ = -p_i + \alpha_i p_i - \mathcal{L} + \beta_i \mathcal{L} + u\beta_i \mathcal{L}. \tag{79}$$

Thus, we have the profit given by the mixed contract is given by $f(p_i, \alpha_i, \beta_i) = \mathbb{P}_i^+ - \mathbb{P}_i^- = \alpha_i p_i + u\beta_i \mathcal{L}$, which means the type-$i$ validator needs to pay a premium $p_i - \alpha_i p_i$ instead of p_i, and obtains $(1+u)\beta_i \mathcal{L}$ as the claim for the loss.

Intuitively, the validator with a high activeness is desired by the blockchain infrastructure provider, since a high activeness contributes to a more value of the network. To motivate the validators, a high discount factor should come with a high coverage factor,

which both are offered by the contract. Based on the cyber insure strategy (p_i, β_i), the blockchain infrastructure provider determines its own policy. We set $\alpha_i = g(\beta_i)$ for the ease of analysis, with $g'(\beta_i) > 0$ and $g''(\beta_i) \leq 0$. Then $f(p_i, \alpha_i, \beta_i)$ can be rewritten as $G(p_i, \alpha_i) := f(p_i, \beta_i, \alpha_i)$. With the pre-defined compensation factor u, let $(p_i, \beta_i, \alpha_i, d_i)$ denote the mixed contract $\{[p_i, \beta_i], [\alpha_i, u\beta_i, d_i]\}$. Therefore we first have the utility function per round for the validators, which is given by

$$u_i^+ = \mathcal{A}_i G(p_i, \alpha_i) + \gamma \mathcal{A}_i \pi(\hat{a}_i) - c[\pi(\hat{a}_i)]^2 - h(d_i), \tag{80}$$

where γ is the pre-defined coefficient and the second term $\gamma \mathcal{A}_i \pi(\hat{a}_i)$ denotes the benefit from the observed activeness that is evaluated by the contract, and $h(d_i)$ represents capital lockup cost with the delay d_i per round, which means the validator would suffer a loss because its deposit is locked up in the blockchain network and cannot be redeemed for the period d_i, which is expressed as

$$h(d_i) = \mathrm{d}\left\{e_0 \mathcal{D} \exp\left\{\epsilon_0 d_i\right\}\right\}, \tag{81}$$

where ϵ_0 is the annual interest rate with $0 < \epsilon_0 < 1$, e_0 is the pre-defined coefficient and $\mathcal{D}$ is the lockup deposit. Thus, we have $h(\cdot)$ is an increasing function of d_i. The time for each round is so tiny compared with the online time d_i, we use the differentiation to present the discounted value approximately.

3.3 Problem Formulation of The Hierarchical Game

This section will model the interactions of the cyber insurer, blockchain infrastructure provider, and validators as a hierarchical game, where the interaction between the cyber insurer and blockchain infrastructure provider is a Stackelberg and the interaction between the blockchain infrastructure provider and validators is a contract game. As shown in Fig. 23, Stage I denotes the Stackelberg game, while Stage II represents the contract game.

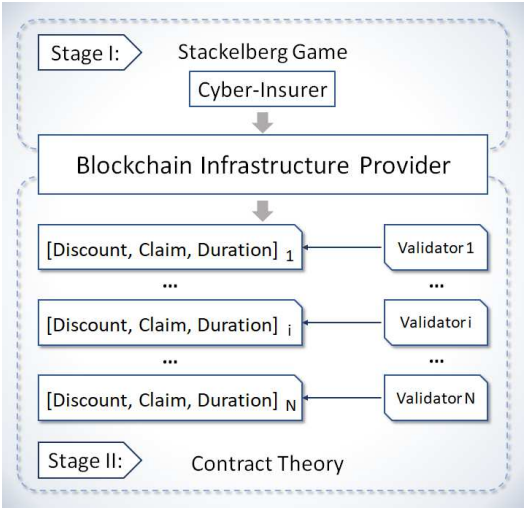


Figure 23: A two-stage hierarchical game model.

3.3.1 Stage I: Stackelberg game formulation

The Stackelberg game includes two sub-games of strategies made by the participants in the model. Cyber-insurer will determine the premium and claim strategies in the upper game after considering blockchain's action. The blockchain infrastructure provider follows the cyber insurer's rules and determines the contracts for validators accordingly in the lower game. The anonymity of the blockchain network makes it more difficult for the cyber insurer to have accurate knowledge of the validators. The cyber insurer can only obtain the probability distribution of the validators based on their activeness history. Let λ_i denote the probability of the type$-i$ validators with $\sum_{i=1}^{\mathbb{N}} \lambda_i = 1$. According to Section 2, the revenue of cyber insurer obtained from all the types of validators is $U_c = \sum_{i=1}^{\mathbb{N}} \{u_c(p_i, \beta_i)\}$. Thus, the utility function of the cyber insurer in the leader's sub-game is expressed as

$$U_c = \sum_{i=1}^{\mathbb{N}} \lambda_i \Big\{ p_i - \beta_i \mathcal{L} \Big\}. \tag{82}$$

Once acquiring the knowledge of the insurance contract from the cyber insurer, the blockchain infrastructure provider determines an incentive policy for each type of the validators in order to gain more market value and resist the discouragement attack through validators' rotation. Based on the premium and claim factor, the blockchain infrastructure provider determines a discount factor for the validators' premium and a

compensation factor for the validators' claim. Thus, we have the utility function for the blockchain infrastructure provider, which is given by

$$U_b = \sum_{i \in \mathbb{N}} \lambda_i \Big\{ \Pi(\theta_i, d_i) - p_i \alpha_i - \Theta(u \beta_i \mathcal{L}) \Big\}, \tag{83}$$

where $\Pi(\cdot)$ is the evaluation function of the validators' delay, the second term $p_i \alpha_i$ is the discount premium for the validators and $\Theta(\cdot)$ is the cost of compensation claim.

3.3.2 Stage II: Contract Model Design

In Stage II, we elaborate the contract theory framework to model the interactions between the blockchain infrastructure provider and the validators. The blockchain infrastructure provider acts as a medium between the cyber insurer and the validators. Given an insurance contract with the premium and claim factor determined by the cyber insurer, the validators obtain a series of contracts from the blockchain. To incentivize the validators and determine the validators' delay, the blockchain infrastructure provider designs a set of contracts, including the discount factor, compensation factor, and online requirement. Recall the utility function of validators in Section 80, and we have the expectation function, which is given by

$$U_i^+ = \theta_i G(p_i, \alpha_i) + \gamma \theta_i \hat{a}_i - c \hat{a}_i^2 + c \sigma'^2 - h(d_i), \tag{84}$$

in which θ_i represents the expected value of the truncated normal distribution $\mathcal{A}_i$, i.e. $\theta_i = E(\mathcal{A}_i)$. According to the probability density function of the truncated normal distribution in Section 3.2.4, the type information is expressed as

$$\theta_i = \int_{a_i}^{a_{i+1}} \frac{x \frac{1}{\sigma} \phi(\frac{x-\mu}{\sigma})}{\Phi(\frac{a_{i+1}-\mu}{\sigma}) - \Phi(\frac{a_i-\mu}{\sigma})} \mathrm{d}x. \tag{85}$$

In the utility function in (84), the optimal activeness for a validator of type-i within the contract can be obtained by $a_i^* := \arg\max \big\{ \theta_i G(p_i, \alpha_i) + \gamma \theta_i \hat{a}_i - c \hat{a}_i^2 + c \sigma'^2 - h(d_i) \big\}$. Then we solve this problem by setting its first derivative condition as zero, which is expressed as $\frac{\partial U_i^+}{\partial \hat{a}_i} = 0$. Given $\hat{a}_i^* = \frac{\gamma \theta_i}{2c}$, it is easy to enable optimal activeness to approach the expected value or a higher one through setting γ. Thus, we rewrite the utility

function in (84) by substituting $\hat{a}_i^* = \frac{\gamma\theta_i}{2c}$, which is expressed as

$$U_i^+ = \theta_i G(p_i, \alpha_i) + \frac{\gamma^2\theta_i^2}{4c} + c\sigma'^2 - h(d_i). \tag{86}$$

We will prove the feasibility of the contract by introducing the constraints of Individual Rationality (**IR**) and Incentive Compatibility (**IC**). Individual Rationality (**IR**) and Incentive Compatibility (**IC**) are two significant constraints of the contract theory [44], which enable the rational validators to select the specific contract that is designed for their own types rather than others.

Definition 5 *Individual Rationality (IR). IR means that a rational type-i ($\forall i \in \{1, \ldots, \mathbb{N}\}$) validator will accept a contract only when the utility provided by the contract is not less than that of no-contract case, i.e.,*

$$U_i^+(p_i, \beta_i, \alpha_i, d_i) = \theta_i G(p_i, \alpha_i) + \frac{\gamma^2\theta_i^2}{4c} + c\sigma'^2 - h(d_i) \geq U_i^-(0, 0, 0, 0), \tag{87}$$

where $U_i^-(0, 0, 0, 0)$ denotes the utility function without the contract.

Definition 6 *Incentive Compatibility (IC). IC means that a type-i validator can only obtain the maximum profit by choosing the contract $(p_i, \beta_i, \alpha_i, d_i)$ rather than all the other contracts $(p_j, \beta_j, \alpha_j, d_j)$ $(\forall i, j, i \neq j)$, i.e.,*

$$U_i^+(p_i, \beta_i, \alpha_i, d_i) \geq U_i^+(p_j, \beta_j, \alpha_j, d_j). \tag{88}$$

A contract is considered to be feasible only when these two constraints are satisfied. The contract designer can only maximize its profit with IR and IC constraints. Therefore, the problem of the contract model can be expressed as

$$\max_{(p_i, \beta_i, \alpha_i, d_i)} \quad U_b = \sum_{i \in \mathbb{N}} \lambda_i \Big\{ \Pi(\theta_i, d_i) - p_i\alpha_i - \Theta(u\beta_i\mathcal{L}) \Big\}, \tag{89}$$

s.t.

$$(22a) \quad U_i^+(p_i, \beta_i, \alpha_i, d_i) \geq U_i^-(0, 0, 0, 0),$$

$$(22b) \quad U_i^+(p_i, \beta_i, \alpha_i, d_i) \geq U_i^+(p_j, \beta_j, \alpha_j, d_j),$$

$$(22c) \quad \theta_1 < \theta_2 < \cdots < \theta_{\mathbb{N}},$$

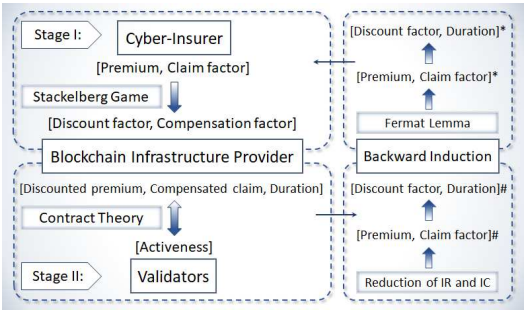


Figure 24: An overview of problem formulation and solution.

wherein (22a) and (22b) are the IR and IC constraints, respectively, and (22c) is the monotonicity condition. Obviously, the problem (89) is not a convex problem and cannot be solved directly. Thus, we transform this problem by reducing the constraints in the following.

3.4 Optimal Solution for the Hierarchical Game

In this section, we apply backward induction to solve the problems. That means we will first solve Stage II's problem by assuming that the solution of Stage I is given. Then we finally obtain all the solutions by substituting them to the problems in Stage I. As shown in Fig. 24, in Stage II, given the premium$^{\#}$ and claim factor$^{\#}$ ('#' means that the parameters are assumed to be known in Stage II), we can obtain the optimal discount factor and online time by reducing the IR and IC constraints under the contract framework. Then in Stage I, after substituting the parameters obtained in Stage II into the objective function, we finally are able to get the optimal premium* and claim factor* ('*' means that the parameters are the final optimal solutions). The algorithms for solving the problems are presented in Algorithm 1 and Algorithm 2.

3.4.1 Stage II: Contract Theory Model

In this sub-section, we reduce the numbers of IC and IR constraints according to the contract theory framework [44] and then obtain a new optimal problem with the reduced constraints.

Lemma 2 *(Reduction of IR) For all the types, $\forall i \in \{1, \ldots, \mathbb{N}\}$, if we have $\theta_1 < \theta_2 < \cdots < \theta_{\mathbb{N}}$, then the IR constraint of type-i holds only when the constraint of the lowest*

type validator is satisfied.

Proof 2 *According to the IR constraint* $U_i^+(p_i, \beta_i, \alpha_i, d_i) \geq U_i^-(0,0,0,0)$*, for type-*1 *validators, we have*

$$U_1^+(p_1, \beta_1, \alpha_1, d_1) \geq U_1^-(0,0,0,0). \tag{90}$$

According to the IC constraint $U_i^+(p_i, \beta_i, \alpha_i, d_i) \geq U_i^+(p_j, \beta_j, \alpha_j, d_j)$ *and* $\theta_i > \theta_1$*, we have*

$$U_i^+(p_i, \beta_i, \alpha_i, d_i) \geq U_i^+(p_1, \beta_1, \alpha_1, d_1) \quad \textit{and} \quad U_i^+(p_1, \beta_1, \alpha_1, d_1) \geq U_1^+(p_1, \beta_1, \alpha_1, d_1). \tag{91}$$

Obviously, for given (90) and (91), we can come to the conclusion that $U_i^+(p_i, \beta_i, \alpha_i, d_i) \geq U_1^-(0,0,0,0)$*. Thus, we complete this proof.* ■

Lemma 3 *(Monotonicity) For any contract* $(p_i, \beta_i, \alpha_i, d_i)$*,*$p_i \geq p_j$*,* $\beta_i \geq \beta_j$*,* $\alpha_i \geq \alpha_j$ *and* $d_i \geq d_j$ *if and only if* $\theta_i \geq \theta_j$*.*

Proof 3 *For ease of expression, we use* $G(\beta_i)$ *instead in the following proofs. According to the IC constraint* $U_i^+(p_i, \beta_i, \alpha_i, d_i) \geq U_i^+(p_j, \beta_j, \alpha_j, d_j)$*, we can obtain the following inequalities,*

$$\theta_i G(\beta_i) + \frac{\gamma^2\theta_i^2}{4c} + c\sigma'^2 - h(d_i) \geq \theta_i G(\beta_j) + \frac{\gamma^2\theta_i^2}{4c} + c\sigma'^2 - h(d_j) \tag{92}$$

$$\textit{and} \quad \theta_j G(\beta_j) + \frac{\gamma^2\theta_j^2}{4c} + c\sigma'^2 - h(d_j) \geq \theta_j G(\beta_i) + \frac{\gamma^2\theta_j^2}{4c} + c\sigma'^2 - h(d_i). \tag{93}$$

Then, we can obtain a new inequation by adding (92) and (93) together, which is given by

$$(\theta_i - \theta_j)[G(\beta_i) - G(\beta_j)] \geq 0. \tag{94}$$

(a) Sufficiency *If* $\theta_i > \theta_j$*, we can get* $G(\beta_i) - G(\beta_j) \geq 0$ *by deriving from (94). As* $G'(\beta_i) > 0$*, we can conclude that* $\beta_i > \beta_j$*. So the sufficiency condition is proved.*

(b) Necessity *The inequation in (94) can be transformed and rewritten as*

$$\theta_i[G(\beta_i) - G(\beta_j)] \geq \theta_j[G(\beta_i) - G(\beta_j)], \tag{95}$$

where $G'(\beta_i) > 0$ and $\beta_i > \beta_j$. We have $G(\beta_i) - G(\beta_j) > 0$ and conclude that $\theta_i > \theta_j$ easily. ■

Proposition 1 *For all the contracts $(p_i, \beta_i, \alpha_i, d_i)$ $(\forall i \in \{1, \ldots, \mathbb{N}\})$, we have $d_i \geq d_j$, if and only if $\beta_i \geq \beta_j$.*

Subproof 1 *According to the IC constraint expressed in (93), we have*

$$h(d_i) - h(d_j) \geq \theta_j[G(\beta_i) - G(\beta_j)]. \tag{96}$$

(a) Sufficiency *If $\beta_i \geq \beta_j$, we can conclude that $G(\beta_i) - G(\beta_j) > 0$ due to $G'(\beta_i) > 0$. Then, we $h(d_i) - h(d_j) > 0$ and $h(d_i) > h(d_j)$. Since $h'(d_i) > 0$, then $d_i > d_j$.*

(b) Necessity *We have the following inequation according to the IC constraint expressed in (92), which is given by*

$$\theta_i[G(\beta_i) - G(\beta_j)] \geq h(d_i) - h(d_j). \tag{97}$$

If $d_i \geq d_j$, we have $h(d_i) - h(d_j) > 0$ due to $h'(d_i) > 0$, which implies $\theta_i[G(\beta_i) - G(\beta_j)] > 0$. Since $G'(\beta_i) > 0$, then we easily have $\beta_i > \beta_j$. The proof is completed. ■

Lemma 3 indicates that in such a feasible contract, the validators who keep active online for a longer time will gain a higher discount factor and coverage factor.

Lemma 4 *(Reduction of IC) There are four definitions regarding the IC constraints between type-i and type-j $(\forall i \neq j)$:*

*(a) If $\forall j \in \{1, 2, \ldots, i-1\}$, the constraints are called Downward Incentive Constraints (**DICs**).*

*(b) If $j = i-1$, the constraint is called Local Downward Incentive Constraint (**LDIC**).*

*(c) If $\forall j \in \{i+1, \ldots, N\}$, the constraints are called Upward Incentive Constraints (**UICs**).*

*(d) If $j = i+1$, the constraint is called Local Upward Incentive Constraint (**LUIC**).*

With the monotonicity conditions $\theta_1 < \theta_2 < \cdots < \theta_{\mathbb{N}}$, the DICs can be reduced as LDICs, i.e. $U_i^+(p_i, \beta_i, \alpha_i, d_i) \geq U_i^+(p_{i-1}, \beta_{i-1}, \alpha_{i-1}, d_{i-1})$ and the UICs can be reduced as the LUICs, i.e. $U_i^+(p_i, \beta_i, \alpha_i, d_i) \geq U_i^+(p_{i+1}, \beta_{i+1}, \alpha_{i+1}, d_{i+1})$.

Proof 4 *All of the validators are classified into different types, and there exists the IC constraint between any two types. As a result, there are too many IC constraints in total, which will increase the difficulty of computation. Here we will prove that all of the IC constraints can be reduced as LDICs. Consider three adjacent types, i.e. type $i-1$, type i and type $i+1$, which follows $\forall i \in \{1, \ldots, N-1\}$. According to the IC constraints, we revise this inequation $U_i^+(p_i, \beta_i, \alpha_i, d_i) \geq U_i^+(p_j, \beta_j, \alpha_j, d_j)$ and then have the following two inequations,*

$$U_i^+(p_i, \beta_i, \alpha_i, d_i) \geq U_{i+1}^+(p_i, \beta_i, \alpha_i, d_i) \tag{98}$$

$$\text{and} \quad U_i^+(p_i, \beta_i, \alpha_i, d_i) \geq U_i^+(p_{i-1}, \beta_{i-1}, \alpha_{i-1}, d_{i-1}). \tag{99}$$

According to the monotonicity condition $\theta_{i+1} > \theta_i$ and $\beta_i \geq \beta_{i-1}$, we have the inequation, which is expressed as

$$(\theta_{i+1} - \theta_i)G(\beta_i) \geq (\theta_{i+1} - \theta_i)G(\beta_{i-1}). \tag{100}$$

Then we transform the inequation (100) to the following one, which is expressed as

$$\theta_{i+1}G(\beta_i) + \frac{\gamma\theta_{i+1}^2}{4c} - \theta_i G(\beta_i) - \frac{\gamma\theta_i^2}{4c} \geq \theta_{i+1}G(\beta_{i-1}) + \frac{\gamma\theta_{i+1}^2}{4c} \quad -\theta_i G(\beta_{i-1}) - \frac{\gamma\theta_i^2}{4c}. \tag{101}$$

To proceed the reduction of IC constraints, we add (100) to the inequation (101), and obtain a new inequation i.e.,

$$U_{i+1}^+(p_i, \beta_i, \alpha_i, d_i) \geq U_{i+1}^+(p_{i-1}, \beta_{i-1}, \alpha_{i-1}, d_{i-1}). \tag{102}$$

Combine the inequation (98) and (102), we can easily get the following inequalities, which are expressed as

$$U_i^+(p_i, \beta_i, \alpha_i, d_i) \geq U_{i+1}^+(p_i, \beta_i, \alpha_i, d_i) \tag{103}$$

$$\textit{and} \quad U_{i+1}^{+}(p_i, \beta_i, \alpha_i, d_i) \geq U_{i+1}^{+}(p_{i-1}, \beta_{i-1}, \alpha_{i-1}, d_{i-1}). \tag{104}$$

Repeat the steps described above, we can obtain the following constraints, which are given by

$$\begin{aligned} U_{i+1}^{+}(p_{i+1}, \beta_{i+1}, \alpha_{i+1}, d_{i+1}) &\geq U_{i+1}^{+}(\beta_{i-1}, d_{i-1}) \geq U_{i+1}^{+}(p_{i-3}, \beta_{i-3}, \alpha_{i-3}, d_{i-3}) \\ &\geq \cdots \\ &\geq U_{i+1}^{+}(p_1, \beta_1, \alpha_1, d_1) \geq U_1^{+}(p_1, \beta_1, \alpha_1, d_1). \end{aligned} \tag{105}$$

Similarly, for the type θ_{i-1} and all the contracts which follow $\forall i \in \{2, \ldots, \mathbb{N}\}$, we can easily obtain the following inequation by the same steps above, which is expressed as

$$\begin{aligned} U_{i-1}^{+}(p_{i-1}, \beta_{i-1}, \alpha_{i-1}, d_{i-1}) &\geq U_{i-1}^{+}(p_{i+1}, \beta_{i+1}, \alpha_{i+1}, d_{i+1}) \\ &\geq \cdots \\ &\geq U_{i-1}^{+}(p_{\mathbb{N}}, \beta_{\mathbb{N}}, \alpha_{\mathbb{N}}, d_{\mathbb{N}}). \end{aligned} \tag{106}$$

Therefore, we present the proof that if the LDICs are satisfied, all the DICs also hold, as well as the LUICs and UICs proved in (106). ■

From lemma 4, we can conclude that Nash Equilibrium always exists. If an optimal contract portfolio $\Omega_i = \{p_i, \alpha_i, \beta_i, d_i\}$ forms an Nash Equilibrium, for type-i validator and the other alternative contract portfolios Ω_{-i}, we must have

$$U_i^{+}(\Omega_i) \geq U_i^{+}(\Omega_{-i}), \tag{107}$$

which can be derived from lemma 4. We will also demonstrate this conclusion through the simulation in section 3.5.

To solve the new optimization problem defined in (89), we first reduce the IC constraints for all the types of contracts ($\forall i, \in \{2, \ldots, \mathbb{N}\}$) by setting $U_i^{+}(p_i, \beta_i, \alpha_i, d_i) = U_i^{+}(p_{i-1}, \beta_{i-1}, \alpha_{i-1}, d_{i-1})$, which is given by

$$\theta_i G(p_i, \alpha_i) + \frac{\gamma^2 \theta_i^2}{4c} + c\sigma'^2 - h(d_i) = \theta_i G(p_{i-1}, \alpha_{i-1}) + \frac{\gamma^2 \theta_{i-1}^2}{4c} + c\sigma'^2 - h(d_{i-1}). \tag{108}$$

Algorithm 2 Optimal Solution to the problem in Stage II

Require: The premium from Stage I: $\hat{p}$, the claim factor from the Stage I: $\hat{\beta}$, the historical statistical distribution of the validators' activeness: $\mathcal{A}$, the number of types: $\mathbb{N}$, the probability of the different type validator: λ, and the loss: $\mathcal{L}$;

Ensure: The discount factor $\hat{\alpha}$ and the online time $\hat{d}$ for all the validators;

1: //Reduction of IC constraints;
2: **for** $i = \mathbb{N}; i \geq 2; i--$ **do**
3: Set $U_i^+(\hat{p}_i, \hat{\beta}_i, \alpha_i, d_i) = U_i^+(\hat{p}_{i-1}, \hat{\beta}_{i-1}, \alpha_{i-1}, d_{i-1})$;
4: Set $\psi_i = U_i^+(\hat{p}_i, \hat{\beta}_i, \alpha_i, d_i) - U_i^+(\hat{p}_{i-1}, \hat{\beta}_{i-1}, \alpha_{i-1}, d_{i-1})$;
5: **end for**
6: //Reduction of IR constraints;
7: Set $U_1^+(\hat{p}_1, \hat{\beta}_1, \alpha_1, d_1) = U_1^-(0,0,0,0)$;
8: Set $\psi_1 = U_1^+(\hat{p}_1, \hat{\beta}_1, \alpha_1, d_1) - U_1^-(0,0,0,0)$;
9: //Add all reduced IC constraints and IR constraint together;
10: **for** $i = \mathbb{N}; i \geq 1; i--$ **do**
11: Set $\Psi_i = \psi_i$;
12: **for** $j = i-1; j \geq 1; j--$ **do**
13: $\Psi_i = \Psi_i + \psi_j$;
14: **end for**
15: **end for**
16: **for** $i = \mathbb{N}; i \geq 1; i--$ **do**
17: Obtain d_i from $\Psi_i = 0$;
18: Substitute d_i into the problem: $\max\limits_{(\hat{p}_i, \hat{\beta}_i, \alpha_i, d_i)} U_{b,i} = \Pi(\theta_i, d_i) - p_i\alpha_i - \Theta(\mu\beta_i\mathcal{L})$;
19: Compute $\hat{\alpha}_i = \arg\max U_{b,i}(\alpha_i) = e_1 \left\{ \frac{\hat{p}_i[e_2 e_1 \theta_i(1-\eta) - e_1(1-\eta)]}{\mathcal{L}(e_3\theta_i^{\epsilon_3}\mu - e_2\theta_i)} \right\}^{\frac{1-\eta}{\eta}}$;
20: Compute $\hat{d}_i = \frac{1}{\epsilon_0} \log \left\{ \frac{\theta_i G(\hat{\alpha}_i) - \sum_{j=1}^{i-1}\Delta_j + \frac{\gamma^2\theta_1^2}{4c} + \mathcal{L}}{e_0\epsilon_0\mathcal{D}} \right\}$;
21: **end for**

For the type-1 utility function, we reduce the IR constraint by setting $U_i^+(p_i, \beta_i, \alpha_i, d_i) = U_i^-(0,0,0,0)$and have

$$\theta_1 G(p_1, \alpha_1) + \frac{\gamma^2\theta_1^2}{4c} + c\sigma'^2 - h(d_1) = -c\hat{a}_1^2 + c\sigma'^2 - \mathcal{L}. \tag{109}$$

Then we add all the IC constraints and obtain

$$\theta_i G_i = (\theta_i - \theta_{i-1})G_{i-1} + \cdots + \theta_2 G_1 + c\sigma'^2 - h(d_1), \tag{110}$$

where G_i denotes $G(p_i, \alpha_i)$. Then, we add the reduced IR constraint of type-1 in (109) to (110) and have

$$\theta_i G(p_i, \alpha_i) - h(d_i) = \sum_{j=1}^{i-1}\Delta_j - \frac{\gamma^2\theta_1^2}{4c} - \mathcal{L}, \tag{111}$$

where $\Delta_j = (\theta_{j+1} - \theta_j)G(p_j, \alpha_j)$. Without loss of generality, we set $\alpha_i = g(\beta_i) = e_1\beta_i^{(1-\eta)}$, $\Pi(\theta_i, d_i) = e_2 h(d_i) + e_2\theta_i^{\epsilon_2}$ and $\Theta(u\beta_i\mathcal{L}) = e_3\theta_i^{\epsilon_3}u\beta_i\mathcal{L}$, where e_1, e_2 and e_3 are the evaluation factors, and $\epsilon_1 > 0, \epsilon_2 > 0, \epsilon_3 > 0$ and $0 < \eta < 1$ are the pre-defined coefficients. Therefore, we rewrite the objective function of the blockchain, which is expressed as

$$\max_{(p_i,\beta_i,\alpha_i,d_i)} U_b = \sum_{i\in\mathbb{N}} \lambda_i \Big\{ e_2 h(d_i) + e_2\theta_i^{\epsilon_2} - p_i\alpha_i - e_3\theta_i^{\epsilon_3}u\beta_i\mathcal{L}) \Big\}. \tag{112}$$

Based on (111), we have

$$h(d_i) = \theta_i G(p_i, \alpha_i) - \sum_{j=1}^{i-1}\Delta_j + \frac{\gamma^2\theta_1^2}{4c} + \mathcal{L}, \tag{113}$$

and rewrite the objective function (112) by substituting (113), i.e.,

$$\begin{aligned}\max_{(p_i,\beta_i,\alpha_i,d_i)} \quad U_b = \sum_{i\in\mathbb{N}} \lambda_i \Big\{ & e_2[\theta_i G(p_i, \alpha_i) - \sum_{j=1}^{i-1}\Delta_j + \frac{\gamma^2\theta_1^2}{4c} + \mathcal{L}] \\ & + e_2\theta_i^{\epsilon_2} - p_i\alpha_i - \frac{e_3}{e_1^{\frac{1}{(1-\eta)}}}\theta_i^{\epsilon_3}u\alpha_i^{\frac{1}{(1-\eta)}}\mathcal{L}) \Big\}.\end{aligned} \tag{114}$$

Therefore, we have the first derivative of α_i, and then obtain the result by differentiating $\frac{\partial U_b(i)}{\partial \alpha_i}$ with respect to α_i, which are given by

$$\frac{\partial U_b(i)}{\partial \alpha_i} = \frac{e_2\theta_i u\mathcal{L}}{(1-\eta)e_i^{\frac{1}{1-\eta}}}\alpha_i^{\frac{\eta}{1-\eta}} + p_i e_2\theta_i - p_i - \frac{e_3\theta_i^{\epsilon_3}u\mathcal{L}}{(1-\eta)e_1^{\frac{1}{1-\eta}}}\alpha_i^{\frac{\eta}{1-\eta}} \tag{115}$$

$$\text{and} \quad \frac{\partial^2 U_b(i)}{\partial \alpha_i^2} = -\frac{\eta u\mathcal{L}(e_3\theta^{\epsilon_3} - e_2\theta_i)}{(1-\eta)^2 e_1^{(\frac{1}{1-\eta})}}\alpha_i^{\frac{2\eta-1}{1-\eta}}. \tag{116}$$

Obviously, we have $\frac{\partial^2 U_b(i)}{\partial \beta_i^2} < 0$ by setting $e_3\theta_i^{\epsilon_3} > e_2\theta_i$ and come to the conclusion that (112) is a concave function. Finally, given the premium p_i determined by Stage I, we derive the optimal solution α_i^* of the contract model by setting $\frac{\partial U_b(i)}{\partial \alpha_i} = 0$. We have

$$\hat{\alpha_i} := e_1\left\{\frac{p_i[e_2e_1\theta_i(1-\eta) - e_1(1-\eta)]}{u\mathcal{L}(e_3\theta_i^{\epsilon_3} - e_2\theta_i)}\right\}^{\frac{1-\eta}{\eta}}. \tag{117}$$

Algorithm 3 Optimal Solution to the problem in Stage I

Require: The discount factor $\hat{\alpha}$, online time $\hat{d}$ that are obtained from Stage II, the historical statistical distribution of the validators' activeness: $\mathcal{A}$, the number of types: $\mathbb{N}$, the probability of the different type validator: λ, and the loss: $\mathcal{L}$;

Ensure: The Optimal premium: p^*, claim factor: β^*, discount factor: α^*, and online time: d^* for all the validators;

1: **for** $i = \mathbb{N}; i \geq 1; i - -$ **do**
2: Transfer β_i from $\hat{\alpha_i}$: $\beta_i = \left\{ \frac{p_i[e_2 e_1 \theta_i (1-\eta) - e_1(1-\eta)]}{\mathcal{L}(e_3 \theta_i^{\epsilon_3} u - e_2 \theta_i)} \right\}^{\frac{1}{\eta}}$;
3: Substitute β_i into the problem: $U_{c,i} = p_i - \beta_i \mathcal{L}$;
4: Compute $p_i^* = \arg\max U_{c,i}(p_i)$;
5: Compute $\beta_i^* = \left\{ \frac{p_i^*[e_2 e_1 \theta_i (1-\eta) - e_1(1-\eta)]}{\mathcal{L}(e_3 \theta_i^{\epsilon_3} u - e_2 \theta_i)} \right\}^{\frac{1}{\eta}}$;
6: Compute $\alpha_i^* = e_1 \beta_i^{*(1-\eta)}$;
7: Compute $d_i^* = \frac{1}{\epsilon_0} \log \left\{ \frac{\theta_i G(\alpha_i^*) - \sum_{j=1}^{i-1} \Delta_j + \frac{\gamma^2 \theta_1^2}{4c} + \mathcal{L}}{e_0 \epsilon_0 \mathcal{D}} \right\}$;
8: **end for**

According to equation (81) and (111), then we obtain the optimal online time, which is given by $\hat{d}_i := \frac{1}{\epsilon_0} \log \left\{ \frac{\theta_i G(p_i, \beta_i) - \sum_{j=1}^{i-1} \Delta_j + \frac{\gamma^2 \theta_1^2}{4c} + \mathcal{L}}{e_0 \epsilon_0 \mathcal{D}} \right\}$.

3.4.2 Stage I: Stackelberg Game Model

Recall $\alpha_i = g(\beta_i)$, then we obtain the optimal claim factor and p_i, which are given by

$$\hat{\beta}_i := \left\{ \frac{p_i[e_2 e_1 \theta_i (1-\eta) - e_1(1-\eta)]}{u\mathcal{L}(e_3 \theta_i^{\epsilon_3} - e_2 \theta_i)} \right\}^{\frac{1}{\eta}} \tag{118}$$

$$\text{and} \quad p_i = \frac{u\mathcal{L}(e_3 \theta_i^{\epsilon_3} - e_2 \theta_i)}{[e_2 e_1 \theta_i (1-\eta) - e_1(1-\eta)]} \beta_i^{\eta}. \tag{119}$$

Substituting (119) into (82), we obtain a new utility function of the cyber insurer and take the first derivative of β_i, which are expressed as

$$U_c = \sum_{i \in \mathbb{N}} \lambda_i \left\{ \frac{\mathcal{L}(e_3 \theta_i^{\epsilon_3} u - e_2 \theta_i)}{[e_2 e_1 \theta_i (1-\eta) - e_1(1-\eta)]} \beta_i^{\eta} - (1-u)\beta_i \mathcal{L} \right\}. \tag{120}$$

$$\text{and} \quad \frac{\partial U_c(i)}{\partial \beta_i} = \frac{\mathcal{L}\eta(e_3 \theta_i^{\epsilon_3} u - e_2 \theta_i)}{[e_2 e_1 \theta_i (1-\eta) - e_1(1-\eta)]} \beta_i^{\eta-1} - (1-u)\mathcal{L}. \tag{121}$$

By differentiating $\frac{\partial U_c(i)}{\partial \beta_i}$ with respect to β_i, we have $\frac{\partial^2 U_c(i)}{\partial \beta_i^2} = \frac{\mathcal{L}\eta(\eta-1)(e_3 \theta_i^{\epsilon_3} u - e_2 \theta_i)}{[e_2 e_1 \theta_i (1-\eta) - e_1(1-\eta)]} \beta_i^{\eta-2}$.

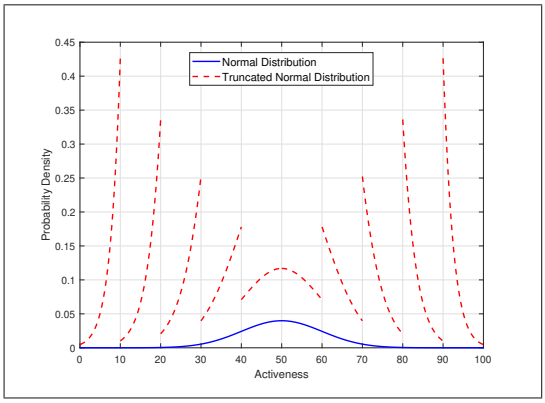


Figure 25: Probability density of the truncated normal distribution and the normal distribution.

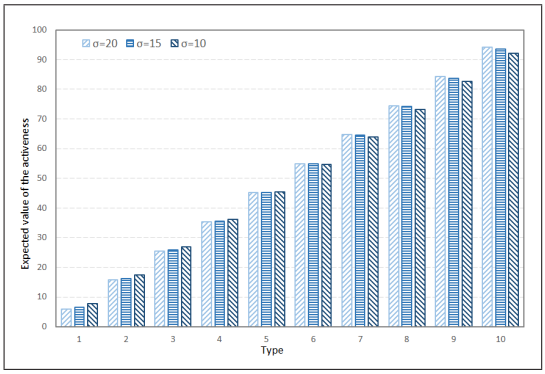


Figure 26: The expected activeness for each type validators.

Obviously, we have $\frac{\partial^2 U_c(i)}{\partial \beta_i^2} <$ with $0 < \eta < 1$. Thus, we can obtain the optimal price of the claim factor by setting $\frac{\partial U_c(i)}{\partial \beta_i} = 0$, i.e.,

$$\beta_i^* = \left\{ \frac{\eta(e_3 \theta_i^{\epsilon_3} u - e_2 \theta_i u)}{[e_2 e_1 \theta_i (1-\eta) - e_1(1-\eta)]} \right\}^{\frac{1}{1-\eta}}. \tag{122}$$

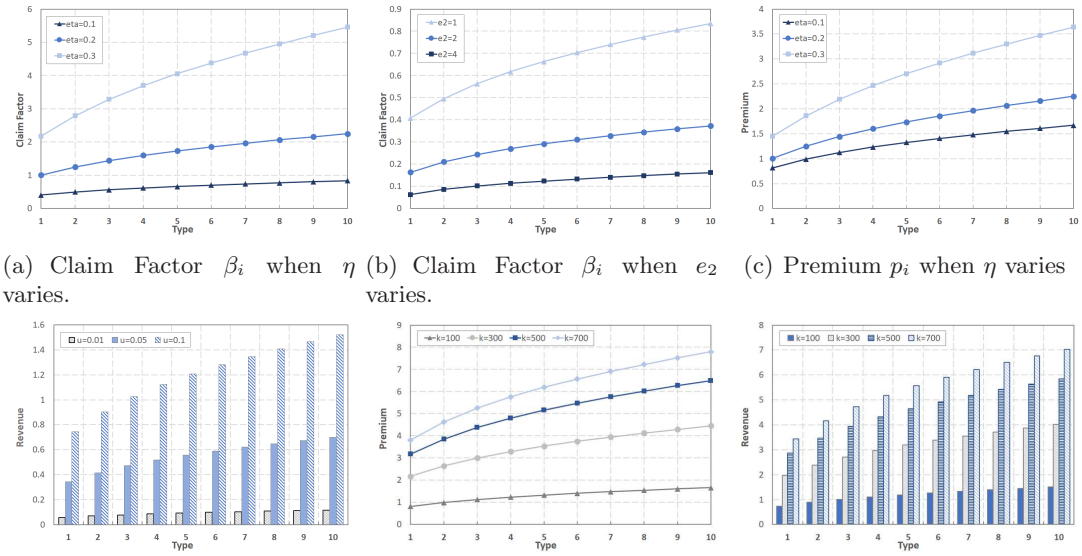
Finally, we have the optimal price of premium, which is expressed as

$$p_i^* = \mathcal{L} \mathcal{F}^{\frac{1}{1-\eta}} (\eta)^{\frac{\eta}{1-\eta}}, \tag{123}$$

where $\mathcal{F} = \frac{(e_3 \theta_i^{\epsilon_3} u - e_2 \theta_i u)}{[e_2 e_1 \theta_i (1-\eta) - e_1(1-\eta)]}$.

3.5 Simulation Results and Numerical Analysis

In this section, we first illustrate the distribution of activeness, present the expected value for each type of validator, and then evaluate the impact of the blockchain infrastructure provider decision on the premium and claim factor. Finally, we show the



(a) Claim Factor β_i when η varies.

(b) Claim Factor β_i when e_2 varies.

(c) Premium p_i when η varies

(d) The revenue of cyber insurer when compensation factor varies.

(e) The premium of cyber insurance when the loss varies.

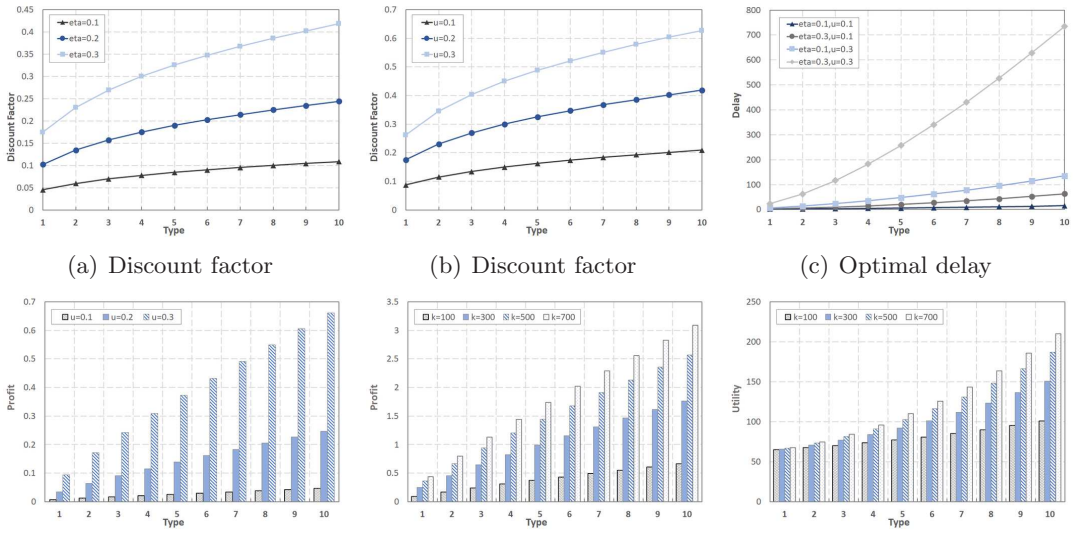
(f) The revenue of cyber insurer when the loss varies.

Figure 27: The impacts of different parameters on the cyber insurer's decision and revenue.

utilities of the validators, the profit of the blockchain infrastructure provider, and the cyber insurer's revenue.

In the initial phase, we consider the activeness of the whole validators to be a *Normal Distribution* with $\mu = 50$ and $\sigma = 10$. In our design, the metric of activeness is uniformly classified into ten intervals, i.e. each interval denotes a single type of validators. When considering the types separately, the probability density function for each type of activeness is modeled as the *Truncated Normal Distribution* with the same parameters $\mu = 50, \sigma = 10$. As shown in Fig. 25, we plot the probability density function (pdf) of truncated normal distribution (i.e. red dot line) according to the pdf of normal distribution (i.e. solid blue line). When only considering a certain type i and its corresponding activeness scope $(a_i, a_{i+1}]$, our plotting result shows that the probability density has been shifted to a higher value. The reason is that the definite integral on the interval $(a_i, a_{i+1}]$ is one while the other elements outside this interval are set to be zero. The probability density change also increases the expected value to a value higher than that of the original normal distribution.

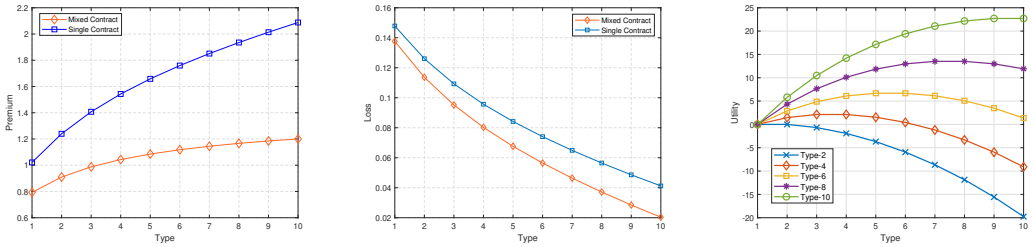
For type i that fits the truncated normal distribution $\mathcal{A}_i(\mu, \sigma, a_i, a_{i+1})$, we present the expected value θ_i with different σ in Fig. 26. We can see that the expected value increases along with the σ decreases in the first five types, but in the last five types,



(a) Discount factor (b) Discount factor (c) Optimal delay

(d) Profit gained by validators when the compensation factor varies. (e) Profit gained by validators when the loss varies. (f) Blockchain's utility when the loss varies.

Figure 28: The impacts of different parameters on blockchain's strategies and profits.



(a) The premium that the different-type validators need to validators pay (b) The loss of the different-type (c) Utilities of type-2, type-4, type-6, type-8 and type-10 validators when signing different contracts.

Figure 29: The comparison with the single contract, and how the utility varies when signing different contracts.

the case is completely opposite. For further analysis, we set $\mu = 50$, $\sigma = 10$, $e_1 = 0.5$, $e_3 = 10$ and $\epsilon_3 = 1.3$. For the discouragement attack assumption, we set $N = 1000$, $\hat{n} = 100$, $k = 100$ and $R = 1000$ for the analysis of the parameters. We list the main parameters in Table 7.

According to the Stackelberg game principle, the cyber insurer's optimal claim and premium strategies depend on the decisions made by the lower sub-game. We next study the impact of blockchain infrastructure provider's decisions. According to the utility function in (114), the blockchain infrastructure provider identifies the weight for each item by using the evaluation parameters, i.e. e_1, e_2, e_3, η and u. According to the closed-form of the optimal solution in (122), it is easy to conclude that the optimal

Table 7: Parameter Setting

Parameters	Values
Mean of validators distribution	$\mu = 50$
Variance of validators distribution	$\sigma = 10$
Number of types	$n = 10$
Number of committee members	$N = 1000$
Number of malicious validators	$\hat{n} = 100$
Total reward for each round	$R = 1000$
Evaluation parameter (1)	$e_0 = 100$
Evaluation parameter (2)	$e_1 = 0.5$
Evaluation parameter (3)	$e_2 = 1$
Evaluation parameter (4)	$e_3 = 10$
Annual interest rate	$\epsilon_0 = 0.135$
Pre-defined coefficients	$\eta = 0.1, 0.2, 0.3$
Number of censored validators	$k = 100, 300, 500, 700$
Compensation factor	$u = 0.1, 0.2, 0.3$
Deposit	$\mathcal{D} = 8500$

solution β_i^* increases along with e_3 and u increase but decreases as e_2 and e_1 grow.

From Fig. 27(a) and Fig. 27(c), we can observe that the claim factor increases when η grows but decreases along with the e_2 grows, where η is applied to define the discount factor α_i. However, the claim factor is a parameter that denotes the claim ratio, which should be less than one. Similar to the previous two sub-figures, the revenue of cyber-insurer also increases as the η grows. Fig. 27(a) only illustrates the growing trend of β_i, wherein $\beta_i \geq 1$ cannot be adopted in a real scenario. Thus, we set $\eta = 0.1$ in the following simulations. From Fig. 27(d), we observe that the compensation factor that the blockchain infrastructure provider determines can also impact the cyber insurer's revenue. The claim should also be given consideration since $\beta_i \geq 1$ when $u > 0.1$, which is not valid. We examine the impact by limiting $u \leq 0.1$. Fig. 27(d) illustrates that a higher compensation factor u would lead to a higher cyber insurer's revenue.

The discouragement attack can be classified into two types: majority attack and minority attack. A majority attack means the attacker has a greater power to control a majority of the validators, while the minority attack is the opposite case. Thus, we set $N = 1000$ validators in total and $\hat{n} = 100$ malicious validators, and let $k = 100$, $k = 300$ denote the minority attack and $k = 500$, $k = 700$ denote the majority attack. Recall the expected loss function in (72), and we can conclude that the expected loss for each

validator will be increased when the censored validators grow in number, which finally approaches its reward per round R/N. Given $e_2 = 1$ and $\eta = 0.1$, we have the growing premium and revenue when increasing the loss, as shown in Fig. 27(e). Besides, Fig. 27(f) also shows that a higher loss brings a higher revenue for the cyber insurer. A positive correlation of the loss, premium, and revenue satisfy the cyber insurer's benefit requirement.

Based on the optimal solutions of cyber insurer, we next investigate the impact of different parameters and loss on the contract (α_i, d_i) individually. In Fig. 28(a) and Fig. 28(b), we set $e_3 = 1.5$ and figure out that, the discount factors α_i of all the contracts increase in terms of η and the compensation factor u. If the metric unit of delay is 'one day', then we set $T = 365$. According to [115], the deposit is 32 ETH (i.e. around 8,500\$). Thus, we set $e_0\epsilon_0 = 13.5$ and $\mathcal{D} = 8,500$ to observe the optimal delay. From Fig. 28(c), we can conclude that η and u have the same impact on the optimal delay, i.e. a higher η and a higher u result in a higher optimal delay. Vitalik suggests "1-3 months to rotate the entire validator set" in [101], and the setting in Fig. 28(c) exactly satisfies the statement.

We can also examine the impact of compensation factor u on the profit $G(p_i, \alpha_i)$. Due to the higher discount factor and the claim factor, validators can obtain a higher profit from the contract (α_i, β_i), as shown in Fig. 28(d). Moreover, a higher compensation factor will bring more benefits for the cyber insurer as well as the validators. Still, it does not mean that the blockchain infrastructure provider compensates them by sacrificing its own profit. Fig. 28(c) shows that a validator must stay online for a longer time in order to obtain a larger compensation factor, wherein a longer delay contributes to a higher profit for the blockchain network. Similarly, when the attacker censors more validators, we observe that the validators gain the increasing profit from their own contracts, as shown in Fig. 28(e), which is consistent with our previous assumption in Section 3.2. In Fig. 28(f), it shows that the blockchain network's utility increases in terms of loss growth.

Finally, we compare the premium and the loss of the mixed contract case with the benchmark scheme 'single contract' case, as illustrated in Fig. 29. In Fig. 29(a), we

observe that the premium in both cases increases with the validators' type. However, for type-i validators, the premium they need to pay in the 'mixed contract' case is less than that of the 'single contract' case. In Fig. 29(b), it shows that the loss in both cases decreases with the validators' type, which means the claim they get from the insurance increases with the validators' type. Apparently, the 'mixed contract' case can provide more insurance claim than a 'single contract' case. We also prove the feasibility of the proposed contract by illustrating the IR and IC constraints in Fig. 29(c). Due to the huge gap of the different utilities, we set a new utility function $U_{i,j} = U_i^+(p_j, \beta_j, \alpha_j, d_j) - U_i^+(p_1, \beta_1, \alpha_1, d_1)$ to denote the original utility without loss of generality. We plot the corresponding utilities of the selected type validators when signing the different contracts in Fig. 29(c), indicating that the type$-i$ validator can only achieve maximum utility when selecting the contract that is exactly designed for his own type. This also confirms the inequality (107). For $\forall i \in \{1, \ldots, \mathbb{N}\}$, we have $U_{i,i} \geq 0$, which means $U_i^+(p_i, \beta_i, \alpha_i, d_i) \geq U_i^+(p_1, \beta_1, \alpha_1, d_1)$. Thus, IC and IR constraints are both satisfied.

3.6 Summary

This work first analyzes the discouragement attack model, the expected loss of the validators in the blockchain networks with shards, designing cyber insurance under a hierarchical model with the contract theory and Stackelberg game. The founders of Ethereum point out that the withdrawal delay mechanism can resist the discouragement attack. However, how to determine an appropriate delay is still an open question. Therefore, we propose an incentive scheme by integrating the cyber insurance idea to neutralize the cyber risks, determining the different validators' withdrawal delays, and providing the insurance claim for their loss. That means the blockchain system can keep more online deposits to resist the discouragement attack via the 'delay' determined in the contract. Simultaneously, the validators stay online to get insured for the loss caused by the discouragement attack. Moreover, the cyber insurer can also benefit from the insurance premium. With few research works on the discouragement attack, we analyze the attack model first and then explore the cyber insurance idea under the contract

theory framework. Based on the simulation and analysis, we can conclude that the proposed contract scheme is able to keep the revenue of cyber-insurer and encourage the validators to be online by providing cyber insurance as an incentive.

4 Transaction Fee Pricing on Sidechain using Random Contract Model

With the evolution of industrial technologies, the internet of things and blockchain show enormous potential. Different individuals and institutes who possess heterogeneous devices indicate diversity in their private information, including the service demand and risk-bearing capability. For those who expect the transactions to be recorded on the blockchain, low efficiency and growing transaction costs are significant obstacles to the application of public blockchain. Furthermore, there is no economic incentive scheme ensuring a timely transaction recording due to the uncertainty of the network's performance and the miner's effort. Considering the asymmetric information (i.e., anonymous private information) and uncertainty, we propose a random-contract-based scheme to maximize the service provider's revenue and assign the service buyers the feasible service price under the framework of a sidechain linked to the public blockchain. Furthermore, we systematically demonstrate random contracts' superiority under the increasing absolute risk-aversion assumption. By applying Geometric Brownian Motion, we study service pricing in different cases. Through the simulation, the numerical results show that random contracts can provide more significant revenue for sidechain by an average of 24.70% compared to the deterministic contracts. Efficient service payments can be reduced by an average of 44.65% compared to the main chain's cost.

4.1 Introduction

Internet of Things (IoT) is a massive network with numerous connected devices, which are embedded with various sensors, allowing them to interact with the environment and communicate with each other via the Internet to achieve specific goals [53]. The explosive growth of industrial requirements brings the number of heterogeneous objects connected to the Internet. With the Fourth Industrial Revolution's evolution, IoT has developed to a new stage, e.g., [54] and [126]. According to [127], nearly 127 new devices are connected to the internet every second. The statistical diagram in [128] predicts that by 2030 around 50 billion of these IoT devices will be installed around the world. Forecasts suggest that by 2025, the overall data volume of installed devices

worldwide will be 79.4 zettabytes [129].

The Internet of Things technology can take advantage in terms of effectively enhancing smart life experience. We can divide the application sets into three folds: smart home, smart city, and smart community. The smart equipment embedded with sensors, including media, air conditioning, and lighting system, can collect and share data via the home network to make an individual's life more convenient, gradually achieving home automation [130]. As mentioned in [131], the smart community is a paradigmatic class of cyber-physical systems with cooperating networked smart homes, gathering data, and sharing information between homes and cross regions to enhance healthcare quality, community security, and emergency response abilities. With the popularization of smart infrastructure, the smart city built upon the Internet of Things will become a reality [132]. Take the city of Houston as an example [133], it embraces the idea of a smart city, aiming to provide a higher quality public service and life experience for its residents, including transportation, public safety, resiliency, and sustainability. Some enterprises have done some research regarding the challenges and opportunities to build smart cities [134]. As for the establishment of a smart city, the connected devices will expand as the service demand grows. The data is valuable, but the centralized data storage of the conventional Internet of Things makes it vulnerable to attacks [135]. The data stored in a distributed ledger can be audited and referred to by any party on blockchain networks. At its core, a blockchain is a set of data blocks secured and connected with cryptography in a time-sequential way [2]. Unlike the centralized system, any global party can validate and audit the transactions organized in a Merkle Hash Tree structure (or other characteristic data structure). Together with the cryptographic mechanisms, blockchain can attain data tamper-resistant and traceable goals.

However, the public chain's slow-growth performance can hardly deal with users' increasing demands. Additionally, the blockchain network will increase transaction fees to incentivize the miners' effort since the mining rewards will be cut in half approximately every four years [2]. One of the prospective methods to tackle such a scalability issue is the application of sidechains. Blockstream firstly proposed the idea of sidechain through their white-paper [136], which describes a functionally similar blockchain linked to the

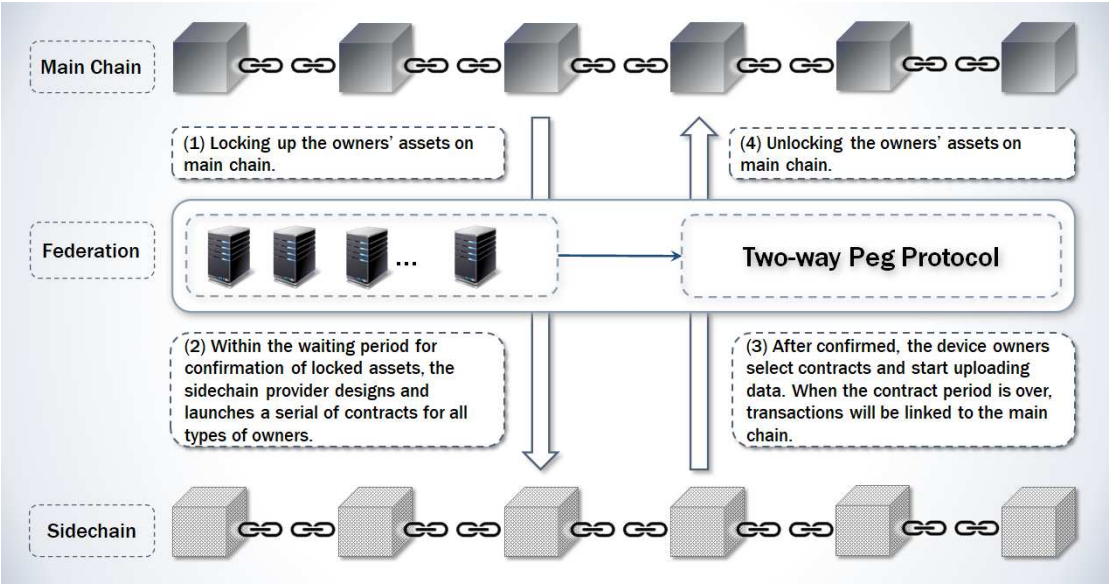


Figure 30: An overview of a sidechain linked to the main chain.

main chain by using the two-way peg protocol. A federation is organized by a sidechain provider (SP), which acts as an intermediate between the sidechain and main chain. The federation launches a two-way peg protocol to link the two chains together, determining when DOs' assets (e.g., cryptocurrency) should be locked up and when the assets can be circulating in the sidechain network. The main chain will qualify and audit the assets and offer approval if the assets are valid. As an extension of functionality, the sidechain shares the same structure with the main chain, including the data structure, consensus mechanism, and block generation procedure. It was initially designed for trading among the different cryptocurrency holders [137].

4.1.1 Related Work

Due to the merits, blockchain technology has found wide applications in various areas, e.g., [138], [139], and [140]. Specifically, blockchain can benefit IoT with its decentralization, security, and privacy. Dai *et al.* [141] discuss and analyze the potentials of incorporating blockchains into IoT plus presenting the technical challenges of the integration. Blockchain has the potential to assist with IoT but also introduces latency and cost problems. With the increasing number of pending transactions, device owners (DOs) have to pay much higher than expected, even absurd amounts [142].

Nevertheless, the functionality of the sidechain has significant potential to be developed. Harishankar *et al.* [143] proposed a scalable and costless sidechain mechanism that supports marketplace-style payment transactions, ensuring the security of all participants' sidechain funds. Li *et al.* [144] proposed an optimized two-way peg protocol to prevent the worthless information injection attack and ensure the quality of data during the authentication information sharing procedure between the main chain and side blockchains. Most research on sidechain is related to security issues. However, the incentive issue is equally important while lacking corresponding research.

In order to establish a smart community, efficient and lower-cost transaction recording services for DOs are necessary for timely data sharing. Little research on economic incentives has been explored specifically for the sidechain network. Game theory, as a branch of economics, focuses on mathematical models and studies the interactions among rational participants [39]. It is advantageous in mechanism design, benefiting the designer, and offering feasible mechanisms for rational game players. In particular, contract theory is a study of economic incentive design, especially in the presence of asymmetric information [44]. Some existing research works on investigating the incentive mechanism design in the engineering area, include wireless communication [45, 46, 47], mobile network [48, 49, 50], and blockchain network [51, 42, 52]. Kang *et al.* combined the reputation system and contract theory to ensure secure miner selection and data sharing in the blockchain-enabled Internet of Vehicles. Su *et al.* [52] proposed a permissioned energy blockchain system with a contract-based energy source allocation scheme, which satisfies the individual needs and maximizes the operator's utility. Wei *et al.* [145] considered the data plan design problem in the mobile network. The mathematical model is very close to a random contract design, but they did not consider the users' risk-bearing capabilities and refined the uncertainty in a deterministic model instead. We compare the related work concerning the scenario and contract type in Table 8. Most of the previous research literature adopts the deterministic contract without considering the risk-aversion situation, meaning that most participants are considered risk-neutral rather than risk-averse, even though the latter case is more practical than the former one.

Table 8: Comparison of Related Work

Paper	Scenario	Risk considered?	Model*
[45],[46], [48]	Content catching	No	D
[47]	RF-powered network	No	D
[50],[145]	Mobile network	No	D
[51]	Shared blockchain	No	D
[42]	Blockchain-enabled Internet of vehicle	No	D
[52]	Energy blockchain	No	D
Our work	Sidechain	Yes	Both

D represents Deterministic Contract.
R represents Random Contract.

4.1.2 Motivation and Contribution

Unlike well-prepared participants in the public blockchain with abundant computation resources or stakes, the DOs' resources and stakes are unevenly distributed, a significant difference in terms of different individuals and institutes. As a result, there is a difference in the DOs' demands and risk-bearing capabilities subject to resources and stakes. Furthermore, the sidechain network has been rarely studied compared to the public blockchain network. Although the sidechain is generally considered the extensive form of a public blockchain, its pricing strategy may differ considerably from its public blockchain. Take the Bitcoin's sidechain project Liquid as an example [146]. The system allows the service provider to determine the fee value and the corresponding transaction confirmation time, e.g., the lowest possible transaction fee on Liquid is 0.1 satoshi (i.e., the smallest unit of a bitcoin). The pricing strategy is therefore much more personalized than that of the public blockchain. Each of the sidechain service providers enjoys considerable autonomy in the running of its own sidechain network. However, the pricing rationale behind it is unclear and makes it challenging for service providers to determine the service fee and obtain maximized revenue. Moreover, the actual transaction confirmation time significantly relies on the miner's effort and network delay, which cannot fully be controlled by the service provider. We can conclude that the transaction fee on a sidechain is much cheaper than the price on its main chain, and the actual transaction confirmation time is uncertain. Therefore, pricing the service fee with such an uncertain service time needs to be studied extensively. When designing a feasible

incentive scheme in such a sidechain network for the DOs, we explore three significant but often neglected influence factors: uncertain service level, risk-bearing capability, and the transaction fees on the main chain. Firstly, we should consider whether the SP has sufficient computation power to deal with all the external environment's uncertainty, offering deterministic services to all DOs. Assigning a deterministic allocation means that the SP makes a massive effort in rescheduling all transactions in the sidechain. However, SP is responsible only for administrative transactions between the main chain and sidechain, instead of any block generation procedure or mining regulation. It is difficult to ensure a deterministic and stable allocation for all the DOs with such weak leadership. Secondly, an eager DO who pays less imposes a greater risk than the reluctant DOs [147]. In other words, DOs have different risk-bearing capabilities, wherein the reluctant DOs are more risk-averse than the eager ones. In practice, the more risk-averse DOs prefer to pay more to avoid risk, obtaining a more stable transaction service. Lastly, the transaction-related services mainly depend on the transaction fee, which is difficult to determine. Since the sidechain is an extension of the main chain, we set the sidechain's transaction fee as a discounted price on the main chain.

In conclusion, the uncertainty of network latency and miner performance can hardly be eliminated due to the external environment and individuals' nature. The individual with smart personal devices and institutes with the smart infrastructure are collectively referred to DOs, which have different risk-bearing capabilities entirely unknown to all the others. That means uncertainty and information asymmetry will coexist. Considering all the mentioned issues above, we propose a random-contract based scheme to maximize the service provider's revenue, assigning a serial of feasible service allocations and corresponding prices to DOs. We also present the necessary assumption and lemmas to investigate deterministic contracts' limitations and random contracts' superiority for further research. Then we apply Geometric Brownian Motion to model the main chain's transaction fees, exploring how to design a random allocation for the DOs without detriment to SP's profit. Regarding the different risk-bearing capabilities, we assign a service with the more considerable variance to lower risk-aversion DO while offering a service with the minor variance to higher risk-aversion DO. The proposed scheme can

compensate for the loss on lower risk-aversion DOs by charging the higher risk-aversion more. The main contributions of this work are summarized as follows:

- We propose a pricing scheme for SP under the framework of a sidechain linked to the main chain. In particular, we design a random contract model to maximize SP's revenue and offer a random transaction recording service allocation to the device owners with uncertainty and asymmetric information.
- Considering the transaction fee fluctuation, we apply the Geometric Brownian Motion model to forecast the transaction fee of the sidechain network. To explore the superiority of random contracts, we also provide the corresponding deterministic forms and demonstrate that random contract dominates the deterministic contract only when the increasing risk aversion assumption holds.
- In terms of the variance of transaction service, we derive the optimal and sub-optimal solutions for both random and deterministic contracts with the unknown case and known case. Through the simulation, we can find that the DO's utility is affected by introducing randomness. However, using the random contract to mitigate the uncertain risk, we can adequately preserve or even improve SP's revenue.

The rest of this work is organized as follows. In Section 4.2, we present the system model, including the deterministic and random models. In Section 4.3, we provide the specific design of the contract, including the problem formulation and optimal solutions for both random and deterministic contracts, and demonstrate the superiority of random contracts. In Section 4.4, we illustrate the simulation results and present the analysis. In Section 4.5, we give a conclusion for the proposed work.

4.2 System Model

In this section, we first present the DO's utility model by modeling other DOs' signings as the social externality, which is expressed in both deterministic and random forms. We then introduce the SP's utility model by applying the Geometric Brownian Motion (GBM) to model the transaction fees on the main chain. Finally, we give the

specific problem formulation under the increasing risk-aversion assumption. An overview of a sidechain linked to the main chain is shown in Fig. 30. We can observe that the two-way peg protocol is the key protocol to link a sidechain and the main chain, transferring the assets and data periodically. A federation is a group of functionaries [136], which possesses the limited authority to deploy such protocol without impacting the decentralization. In general, the sidechain interacts with the main chain periodically. At the beginning of the interaction period, the sidechain will communicate with the main chain via the two-way peg protocol. The main chain locks the assets by creating a transaction, and the sidechain also creates a transaction to confirm the lockup with cryptographic proof. The interaction in the initial phase typically takes one or two days, which allows the federation (i.e., service provider) to design and launch a series of contracts for DOs. Only after locked on the main chain, the cryptocurrency and protocol of the sidechain can be activated and used without further interaction. After the service period is over, the transactions and newly-generated profit in the sidechain will be packaged and transferred back to the same main chain. It follows that initialization and synchronization are very similar within one period [136].

We consider the market with one risk-neutral SP who sells contracts and multiple risk-averse DOs who will decide to buy contracts or not. Since SP can predetermine the rate at which locked assets are exchanged between the main chain and sidechain, the SP also can design a set of contracts for DOs. After signing the contracts, SP can reschedule all the DO's unconfirmed transactions sequence in the transaction pool, allowing the miners to verify and package them in a predetermined order. Then SP issues transaction fees to reward the work of finalizing the transactions. We present utility models for DOs and SPs in Section 4.2.1 and Section 4.2.2, respectively, and the problem formulation in Section 4.2.3. The main notations involved in the following models are listed in Table 9.

4.2.1 Device Owner Model

In the proposed design, SP offers a series of contracts with different allocations, i.e., finalized transactions per unit time T_i and payments P_i. DOs also have a different

Table 9: Main Notations

Symbol	Definition
θ_i	Type-i DO's service demand;
T_i	Finalized transaction per unit time;
P_i	Payment for services provided by SP;
$\mathcal{P}(x)$	Probability of event x;
γ, λ	predefined coefficients;
$\rho(\theta_i)$	Hazard rate with respect to demand θ_i;
$h(\theta_i)$	Measurement of externality with respect to θ_i;
α_i	Discount factor of transaction fee;
R_i	Revenue of a SP obtaining from the contracts;
C	Transaction fee in main chain.

demand θ_i. DOs that share the same demand will be identified as the same type. We view the DO who has the demand θ_i as the type-i DO for analysis convenience.

Without loss of generality, assume the DO with a higher demand is more risk-averse. We elaborate on this assumption in Section 4.2.3. Intuitively, the more risk-aversion DO cannot afford higher risks, such that they prefer to pay substantially more for a stable service with less risk. We use $u(\cdot)$ to denote the utility provided by the contract, which is expressed as

$$u(T_i, P_i; \theta_i) = \int_0^{T_i} \nu(x; \theta_i) \mathrm{d}x - h(\theta_i) - P_i. \tag{124}$$

The first term on the right side of (124) denotes the valuation of T_i for the DO whose demand is θ_i, is decreasing in x and increasing in θ_i, where

$$\int \nu(x; \theta_i) \mathrm{d}x = V(x; \theta_i) \tag{125}$$

$$\text{and} \quad V_{11}{}^1 < 0. \tag{126}$$

Assume there are N types of demand in total, where $\theta_1 < \cdots < \theta_N$, and the contracts also include a redundant contract, where $V(0, 0) = 0$, meaning that DOs always have the option of buying nothing. According to (124), we have the second derivative of u_i with respect to T_i, which is given by

$$\frac{\partial^2 u_i}{\partial (T_i)^2} = V_{11} = \nu_1 < 0. \tag{127}$$

[1]In the following description, we will let $\phi_1(x, y)$ denote $\frac{\partial \phi}{\partial x}$, $\phi_2(x, y)$ denote $\frac{\partial \phi}{\partial y}$, $\phi_{11}(x, y)$ denote $\frac{\partial^2 \phi}{\partial x^2}$, $\phi_{12}(x, y)$ denote $\frac{\partial}{\partial y}\frac{\partial \phi}{\partial x}$ $\phi_{22}(x, y)$ denote $\frac{\partial^2 \phi}{\partial y^2}$,$\phi_{21}(x, y)$ denote $\frac{\partial}{\partial x}\frac{\partial \phi}{\partial y}$.

Thus, we can verify that the utility function is concave in T_i, and such that DOs are risk-averse rather than risk-neutral. The second term on the right side of (124) indicates the negative social externalities experienced by the lower type DOs. Intuitively, more higher-type DOs signing the contracts means the current type DOs probably receive a lower quality of services because of the limited throughput. The transaction verification and block generation depend on the corresponding transaction fee. Higher-type DOs will contribute more transaction fees and likely obtain a shorter verification and packaging time. Thus, lower-type DOs have to suffer the risk that their transactions cannot be verified and packaged in time. We set a metric to evaluate the externality, which is expressed as

$$h(\theta_i) = \lambda \lim_{\Delta\theta \to 0} \left\{ \frac{\mathcal{P}\{\theta_i \leq X\}}{\mathcal{P}\{\theta_i \leq X \leq \theta_i + \Delta\theta\}} \right\}, \tag{128}$$

where the numerator denotes the probability of DOs who have the demand higher than θ_i, and the denominator represents the probability of DOs who have the demand close to θ_i. Thus, for type-i DOs, the externality decreases as the current DOs' proportion grows and increases as the higher-type DOs' proportion grows.

For the continuous case, DOs' types follow a continuous distribution with the probability density function $f(\theta)$, we have

$$h(\theta_i) = \frac{\lambda \int_{\theta_i}^{\infty} f(x)\mathrm{d}x}{\int_{\theta_i}^{\theta_i+\Delta\theta} f(x)\mathrm{d}x} \approx \frac{\lambda(1 - F(\theta_i))}{\Delta\theta f(\theta_i)} = \frac{\gamma}{\rho(\theta_i)}, \tag{129}$$

where $\Delta\theta$ is a constant interval, λ is a parameter that measures the external impact, $\gamma = \lambda/\Delta\theta$ and $\rho(\theta) \equiv \frac{F'(\theta)}{1-F(\theta)}$ is the hazard rate [148] for function $F(\theta)$.

For the discrete case, DOs' types follow a discrete distribution with the probability $p(\theta)$, we have

$$h(\theta_i) = \gamma \frac{\mathcal{P}\{\theta_i < X\}}{\mathcal{P}\{\theta_i = X\}} = \gamma \frac{\sum_{j=i+1}^{N} p(\theta_j)}{p(\theta_i)} = \frac{\gamma}{\rho(\theta_i)}, \tag{130}$$

where $\rho(\theta) \equiv p(\theta_i)/\sum p(\theta_j)$ when θ is a discrete variable. Without loss of generality, we set $h(\theta_j) \geq h(\theta_i)$ for all $\theta_j < \theta_i$.

A. Deterministic contract

According to [149], a rational decision−maker will behave to maximize the expected

value of utility function defined over the probabilistic outcomes. This function is known as the von Neumann−Morgenstern (VNM) utility function. Given the deterministic contract items (T_i, P_i), the VNM utility is expressed as

$$U(T_i, P_i; \theta_i) = E[u(T_i, P_i; \theta_i)] = u(T_i, P_i; \theta_i). \tag{131}$$

B. Random contract

Given the random contract item, the confirmed transaction $\widetilde{T}_i$ is a random variable with distribution $F(s)$ and mean T_i. Therefore, the VNM utility is expressed as

$$U(\widetilde{T}_i, \widetilde{P}_i; \theta_i) = E[u(\widetilde{T}_i, \widetilde{P}_i; \theta_i)] = \int_{-\infty}^{+\infty} \int_{0}^{\widetilde{T}_i} \nu(x; \theta_i) \mathrm{d}x \mathrm{d}F(\widetilde{T}_i; T_i) - \frac{\gamma}{\rho(\theta_i)} - \widetilde{P}_i. \tag{132}$$

4.2.2 Blockchain Service Provider Model

Without loss of generality, we consider the SP to be risk neutral. Thus, we have

$$R_i = \widetilde{P}_i - T_i \cdot \alpha_i C, \tag{133}$$

where the first term is the payment given by DO. The second term denotes the total cost for T_i transactions. Specifically, C represents the type-i transaction fee for verification and packaging into a block and is characterized as a GBM. α_i is the discount factor [146]. We adopt a discounted price of transaction fee in the main chain as the sidechain's transaction fee.

A. Pricing of Transaction Fees

Since the contracts may be effective at some future date and time, not immediately, we adopt a prediction model to obtain a more realistic price of the transaction fee. Considering that some bitcoin pricing models derive from the various Brownian Motions [150], with loss of generality, we apply GBM to model the transaction fee of the main chain [151], which is previously applied for predicting the stock price [152]. We also fit

this model on historical price of Ethereum transaction fee in Section 4.4. Thus, we have

$$\mathrm{d}C = \underbrace{\mu C}_{\widetilde{\mu}}\,\mathrm{d}t + \underbrace{\sigma C}_{\widetilde{\sigma}}\,\mathrm{d}W_t, \tag{134}$$

where W_t is a standard Brownian Motion and μ and σ are constants. The first term, also known as the drift term, is used to model the deterministic trends, and the second term is used to model the unpredictable events in the future.

To solve C, we set $g = \ln C$. According to Itô's lemma [153], we have the stochastic differential equation (SDE), which is expressed as

$$\mathrm{d}g = \left(\frac{\partial g}{\partial t} + \widetilde{\mu}\frac{\partial g}{\partial C} + \frac{1}{2}\widetilde{\sigma}^2\frac{\partial^2 g}{\partial C^2}\right)\mathrm{d}t + \widetilde{\sigma}\frac{\partial g}{\partial C}\mathrm{d}W_t = (\mu - \frac{1}{2}\sigma^2)\mathrm{d}t + \sigma\mathrm{d}W_t. \tag{135}$$

Then we take the integral of both sides for (135) and rewrite this logarithm as an exponential equation, which is given by

$$C = C_0 \exp\left\{C(\bar{T})\right\}, \tag{136}$$

where C_0 is the initial price of transaction fee for type-i. Assume that $\bar{T}$ denotes the maximum length of one unit time, let the price at time $\bar{T}$ approximately represents the fee for each transaction, where $C(\bar{T})$ is the Itô drift-diffusion Process, which is given by

$$g = C(\bar{T}) = \int_0^{\bar{T}} (\mu - \frac{1}{2}\sigma^2)\mathrm{d}t + \int_0^{\bar{T}} \sigma\mathrm{d}W_t = (\mu - \frac{1}{2}\sigma^2)\bar{T} + \sigma W_{\bar{T}}. \tag{137}$$

For any deterministic contract, we have $E(R_i) = P_i - T_i \cdot \alpha_i C_0 \exp\left\{\mu\bar{T}\right\}$. For any random contract, due to $\widetilde{T}_i$ and C are two independent variables, we have $E'(R_i) = \widetilde{P}_i - E(\widetilde{T}_i \cdot \alpha_i C) = \widetilde{P}_i - T_i \cdot \alpha_i C_0 \exp\left\{\mu\bar{T}\right\}$.

4.2.3 Problem Formulation

In this work, our goal is to design a set of contracts, which can extract a maximized profit from DO and provide the services that adapt to DOs' requirements. In general, a deterministic allocation can always benefit the contract provider. However, if the contract buyers have different risk-bearing capabilities, the contract provider can be

more profitable by offering random contracts. To analyze the random contract precisely, we impose the following assumption.

Assumption 1 ***Increasing Absolute Risk Aversion.*** *The absolute risk aversion with respect to DOs' consumption T is non-decreasing in DOs' demand θ. That is,*

$$\frac{\partial(-\frac{\partial^2 U}{\partial T^2}/\frac{\partial U}{\partial T})}{\partial \theta} = \frac{\partial \mathcal{R}_T}{\partial \theta} > 0, \tag{138}$$

where $\mathcal{R}_T = -\frac{\partial^2 U}{\partial T^2}/\frac{\partial U}{\partial T}$, and $-\frac{\partial^2 U}{\partial T^2} < 0$. If $\mathcal{R}_T^i > \mathcal{R}_T^j$, then type$-i$ DO is more risk-averse than type$-j$ DO.

Through this assumption, the higher-type DOs are more risk-averse than the lower-type ones. Otherwise, if DOs are indifferent in risk or the higher-type DOs are less risk-averse than the lower-type ones. As a result, the random contract cannot provide a higher profit over the deterministic contract [44],[154]. According to Assumption 1, we set the valuation function of x, which is expressed as

$$\begin{aligned} \nu(x;\theta_i) = \beta\theta_i e^{-\beta\theta_i(x-\theta_i)}, \quad V(T_i;\theta_i) \quad &= \int_0^{T_i} \beta\theta_i e^{-\beta\theta_i(x-\theta_i)} \mathrm{d}x \\ &= e^{\beta\theta_i^2} - e^{-\beta\theta_i(T_i-\theta_i)}. \end{aligned} \tag{139}$$

For any random variable $\widetilde{T}_i$ follows a normal distribution with mean T_i and variance σ_i, we have the expected function of V, which is expressed as

$$E[V(\widetilde{T}_i;\theta_i)] = \int_{-\infty}^{+\infty} \int_0^{\widetilde{T}_i} \nu(x;\theta_i)\mathrm{d}x\mathrm{d}F(\widetilde{T}_i;T_i) = e^{\theta_i^2} - e^{\theta_i^2 + \frac{\theta_i^2\sigma_i^2}{2} - T_i\theta_i}.^2 \tag{140}$$

To formulate the problem, we next define two vital constraints under the contract theory framework [44]: 1) Individual Rationality (**IR**); 2) Incentive Compatibility (**IC**), which enables the rational DOs to select the specific contract that is designed for their own types rather than others.

Definition 7 *Individual Rationality (IR). IR means that a rational type-i ($\forall i \in \{1,\ldots,N\}$) DO accepts a contract only when the utility is non-negative, which is expressed as*

[2] In fact, $\int_{-\infty}^{+\infty} e^{-\beta\theta_i\widetilde{T}_i} \frac{1}{\sqrt{2\pi}\sigma_i} e^{-\frac{(\widetilde{T}_i - T_i)^2}{2\sigma_i^2}} d\widetilde{T}_i = e^{\frac{\beta^2\theta_i^2\sigma_i^2}{2} - \beta T_i\theta_i}$, where $\int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi}\sigma_i} e^{-\frac{[\widetilde{T}_i - (T_i - \sigma_i^2\theta_i)]^2}{2\sigma_i^2}} d\widetilde{T}_i = 1$.

$$U(\widetilde{T}_i, \widetilde{P}_i; \theta_i) \geq U(0,0) = 0, \tag{141}$$

where $U(0,0) = 0$ denotes the utility function without the contract.

Definition 8 *Incentive Compatibility (IC). IC means that a type-i DO can only obtain the maximum profit by choosing the contract that is designed for itself rather than all the other contracts, which is expressed as*

$$U(\widetilde{T}_i, \widetilde{P}_i; \theta_i) \geq U(\widetilde{T}_j, \widetilde{P}_j; \theta_i), \quad \forall i \neq j. \tag{142}$$

A contract that satisfies the IR and IC constraints can be considered to be feasible. SP obtains the profit from all types of DOs, thus, the optimal problem can be expressed as

$$\max_{(\widetilde{T}_i, \widetilde{P}_i)} \quad R = \sum_{i \in \mathbb{N}} G(\theta_i) \Big\{ \widetilde{P}_i - T_i \cdot \alpha_i C_0 \exp\left\{\mu \bar{T}\right\} \Big\}, \tag{143}$$

s.t.

$$(a) \quad U(\widetilde{T}_i, \widetilde{P}_i; \theta_i) \geq 0,$$

$$(b) \quad U(\widetilde{T}_i, \widetilde{P}_i; \theta_i) \geq U(\widetilde{T}_j, \widetilde{P}_j; \theta_i),$$

$$(c) \quad \theta_1 < \theta_2 < \cdots < \theta_N,$$

where $G(\theta_i)$ denotes the probability of type-i DO with $\sum_{i=1}^{N} G(\theta_i) = 1$. Throughout the random contract model can be easily deduced to the deterministic one.

4.3 Optimal Solution and Contract Feasibility

Since there are N IR constraints and $N(N-1)$ IC constraints in total, problem (143) is non-convex and so that it is difficult to be solved directly. In order to solve the optimization problem, we first reduce the constraints by several lemmas and obtain the closed forms of optimal solutions in Section 4.3.1. Then we demonstrate the feasible and superiority of random contracts through two propositions in Section 4.3.2.

4.3.1 Optimal Solution

In this subsection, we first present and prove the sufficient conditions of IC.

Lemma 5 *For any feasible deterministic contract (T_i, P_i), if $\theta_i > \theta_j$, then $T_i > T_j$.*

This lemma shows that a DO with a higher demand should select a greater transaction service amount.

Proof 5 *According to IC conditions, we have*

$$V(T_i;\theta_i) - h(\theta_i) - P_i \geq V(T_j;\theta_i) - h(\theta_i) - P_j, \tag{144}$$

$$V(T_j;\theta_j) - h(\theta_j) - P_j \geq V(T_i;\theta_j) - h(\theta_j) - P_i. \tag{145}$$

Adding the two inequalities (144) and (145) together, and transposing all the left items to the right side we obtain a new inequality, we have

$$V(T_i;\theta_i) - V(T_i;\theta_j) \geq V(T_j;\theta_i) - V(T_j;\theta_j) \tag{146}$$

$$\text{and} \quad \int_{\theta_j}^{\theta_i} V_2(T_i;x)\mathrm{d}x - \int_{\theta_j}^{\theta_i} V_2(T_j;x)\mathrm{d}x = \int_{\theta_j}^{\theta_i} \left\{ \int_{T_j}^{T_i} V_{21}(y;x)\mathrm{d}y \right\} \mathrm{d}x \geq 0. \tag{147}$$

Since $V_{21} = V_{12} = \nu_2 > 0$, thus we have $T_i \geq T_j$. Similarly, we can have $\theta_i \geq \theta_j$ if $T_i > T_j$. Throughout this work focuses on the random contract, the optimal solution T_i of deterministic contract is exactly the mean for random variable $\widetilde{T}_i$.

Lemma 6 *For any feasible random contract $(\widetilde{T}_i, \widetilde{P}_i)$, if $\theta_i > \theta_j$ and $T_i > T_j$ (where $\widetilde{T}_i$ follows a normal distribution with mean T_i and variance σ_i, and $\widetilde{P}_i$ is fixed), then $\widetilde{P}_i > \widetilde{P}_j$. This lemma shows that a higher-demand DO needs to pay a higher price for service.*

Proof 6 *According to IC condition, we have*

$$E[V(\widetilde{T}_j;\theta_j)] - \widetilde{P}_j \geq E[V(\widetilde{T}_i;\theta_j)] - \widetilde{P}_i. \tag{148}$$

$$\Downarrow$$

$$\begin{aligned} \widetilde{P}_i - \widetilde{P}_j &\geq E[V(\widetilde{T}_i;\theta_j)] - E[V(\widetilde{T}_j;\theta_j)] \\ &= e^{\frac{\theta_j^2\sigma_j^2}{2}}(e^{\theta_j^2 - T_j\theta_j} - e^{\theta_j^2 - T_i\theta_j}) \\ &\geq 0. \end{aligned} \tag{149}$$

Similarly, we can deduce the proof and come to the conclusion $P_i > P_j$ within the deterministic contract.

We will compare the prices and show the proof under the different conditions in the following lemmas for any feasible deterministic contract and its corresponding random contract.

Lemma 7 *For any random contract $(\widetilde{T}_i, \widetilde{P}_i)$ (where $\widetilde{T}_i$ follows a normal distribution with mean T_i and variance σ_i, and $\widetilde{P}_i$ is fixed) and its corresponding deterministic contract (T_i, P_i), if type-i DO is indifferent in choosing between them, then the optimal random contract provides a lower price than that of any deterministic one, which is expressed as*

$$\widetilde{P}_i < P_i. \tag{150}$$

Proof 7 *When a random contract $(\widetilde{T}_i, \widetilde{P}_i)$ and its corresponding deterministic contract (T_i, P_i) are provided at the same time, where $\widetilde{T}_i^*$ and T_i^* are the optimal strategies of these two contracts individually, then the DO's corresponding utility $\widetilde{U}_i$ and U_i are given by*

$$\widetilde{U}_i = E\left\{V(\widetilde{T}_i^*; \theta_i)\right\} - h(\theta_i) - \widetilde{P}_i \tag{151}$$

$$\text{and} \quad U_i = V(T_i^*; \theta_i) - h(\theta_i) - P_i. \tag{152}$$

If the type-θ_i DO is indifferent in the above two contracts, which means $\widetilde{U}_i = U_i$. Then, we can obtain the gap between the prices, which is expressed as

$$P_i - \widetilde{P}_i = V(T_i^*; \theta_i) - E\left\{V(\widetilde{T}_i^*; \theta_i)\right\} > 0, \tag{153}$$

where $V(\cdot)$ is concave in T_i. Therefore, we come to the conclusion that $\widetilde{P}_i < P_i$.

We will show the comparison between the deterministic and random contracts' payment prices if given the relation of their local downward prices.

Lemma 8 *For any feasible random contract $(\widetilde{T}_i, \widetilde{P}_i)$, if we have $\widetilde{P}_i \geq P_i$ for type-i DO,*

then the following inequality always holds for $i+1 \leq N$, which is expressed as

$$\widetilde{P}_{i+1} > P_{i+1}. \tag{154}$$

Proof 8 *According to IC condition, we have*

$$V(T_{i+1};\theta_{i+1}) - P_{i+1} \geq V(T_i;\theta_{i+1}) - P_i \tag{155}$$

$$\textit{and} \quad V(T_i;\theta_i) - P_i \geq V(T_{i+1};\theta_i) - P_{i+1}. \tag{156}$$

We can get the scope of P_{i+1} by transposing (155) and (156), which is expressed as

$$P_{i+1} \in [\pi_1(T_{i+1},T_i;\theta_i) + P_i, \pi_1(T_{i+1},T_i;\theta_{i+1}) + P_i], \tag{157}$$

where $\pi_1(T_{i+1},T_i;\theta_i) = V(T_{i+1};\theta_i) - V(T_i;\theta_i)$. Similarly, we can obtain the scope of $\widetilde{P}_{i+1}$, which is expressed as

$$\widetilde{P}_{i+1} \in [\pi_2(T_{i+1},T_i;\theta_i) + \widetilde{P}_i, \pi_2(T_{i+1},T_i;\theta_{i+1}) + \widetilde{P}_i], \tag{158}$$

where $\pi_2(T_{i+1},T_i;\theta_i) = E[V(T_{i+1};\theta_i)] - E[V(T_i;\theta_i)]$.

We can observe that $\overline{P}_{i+1}$ is the optimal solution when local upward incentive constraints are binding, and $\underline{P}_{i+1}$ is the optimal solution when local downward incentive constraints are binding. Thus, we have

$$\begin{aligned}
\overline{P}_{i+1} &= \pi_1(T_{i+1},T_i;\theta_i) + P_i = \int_{T_i}^{T_{i+1}} \nu(x;\theta_i)\mathrm{d}x + P_i, \\
\underline{P}_{i+1} &= \pi_1(T_{i+1},T_i;\theta_{i+1}) + P_i = \int_{T_i}^{T_{i+1}} \nu(x;\theta_{i+1})\mathrm{d}x + P_i, \\
\overline{\widetilde{P}}_{i+1} &= \pi_2(T_{i+1},T_i;\theta_i) + \widetilde{P}_i = \int_{T_i}^{T_{i+1}} E_1[V(x;\theta_i)]\mathrm{d}x + \widetilde{P}_i, \\
\textit{and} \quad \underline{\widetilde{P}}_{i+1} &= \pi_2(T_{i+1},T_i;\theta_{i+1}) + \widetilde{P}_i = \int_{T_i}^{T_{i+1}} E_1[V(x;\theta_{i+1})]\mathrm{d}x + \widetilde{P}_i.
\end{aligned} \tag{159}$$

Since we have $\nu_2 > 0$ according to (139) and Lemma 5, then $\nu(x;\theta_{i+1}) > \nu(x;\theta_i)$. It

is true that $\overline{P}_{i+1} < \underline{P_{i+1}}$. *Besides, with* $\widetilde{P}_i > P_i$, *we finally obtain*

$$\overline{\widetilde{P}}_{i+1} > \overline{P}_{i+1} \quad \textit{and} \quad \underline{\widetilde{P}}_{i+1} > \underline{P}_{i+1}, \tag{160}$$

where $\nu(x;\theta_i) < E_1[V(x;\theta_i)]$. *Therefore, no matter which IC property is adopt to deal with this problem, we can always have the conclusion that* $\widetilde{P}_{i+1} > P_{i+1}$.

Lemma 9 *For any feasible random contract* $(\widetilde{T}_i, \widetilde{P}_i)$, *if type-*$(i+1)$ *DO who is more risk-averse than type-i with* $\sigma_i \neq 0$ *and* $\sigma_{i+1} = 0$, *and the contracts from type-*θ_1 *to type-*θ_i *are loss-making, then type-*$(i+1)$ *DO can compensate for the loss on all the types lower than* $i+1$ *only when the Local Downward Incentive Constraints (LDIC) if type-*$\{i+1\}$ *is not binding, which is expressed as*

$$U(T_{i+1}, \widetilde{P}_{i+1}; \theta_{i+1}) > U(\widetilde{T}_i, \widetilde{P}_i; \theta_{i+1}). \tag{161}$$

Proof 9 *According to Lemma 8, if the following inequality holds* $\widetilde{P}_i < P_i$. *Then we can easily have*

$$U(T_i, P_i; \theta_{i+1}) > U(\widetilde{T}_i, \widetilde{P}_i; \theta_{i+1}). \tag{162}$$

From Lemma 8, the maximized price $\widetilde{P}_{i+1}$ *is achieved only when the local downward incentive constraints are binding. We have* $U(T_{i+1}, P_{i+1}; \theta_{i+1}) = U(T_i, P_i; \theta_{i+1})$ *with* $\widetilde{P}_{i+1} = P_{i+1}$, *and thus,* $U(T_{i+1}, P_{i+1}; \theta_{i+1}) > U(\widetilde{T}_i, \widetilde{P}_i; \theta_{i+1})$ *holds. If* $\widetilde{P}_i$ *and* P_i *are uncertain, then we have to set* $\widetilde{P}_{i+1} > P_{i+1}$. *Then, we can have*

$$U(T_{i+1}, \widetilde{P}_{i+1}; \theta_{i+1}) < U(T_i, P_i; \theta_{i+1}), \tag{163}$$

which violates the IC condition of the deterministic set of contracts. This conclusion is also explained in [44]. As Lemma 8 implies, since the upper bound of solution is obtained from the binding LDIC, which means all type-$(i+1)$ *DOs have already stayed at their own reservation utility level* (T_i, P_i), *i.e.,* $\underline{P}_{i+1}$ *is not able to be increased to extract more revenue any more. Therefore, the local downward incentive constraints between*

these two types of the deterministic set cannot be binding, which is expressed as

$$U(T_{i+1}, \widetilde{P}_{i+1}; \theta_{i+1}) > U(T_i, P_i; \theta_{i+1}). \tag{164}$$

Assume $\widetilde{P}^*_{i+1} > \overline{P}^*_{i+1}$, SP will have the possibility to adapt the price to compensate for the loss, maximizing the price until $U(T_{i+1}, \widetilde{P}_{i+1}; \theta_{i+1}) > U(\widetilde{T}_i, \widetilde{P}_i; \theta_{i+1})$ holds. If $U(T_{i+1}, \widetilde{P}_{i+1}; \theta_{i+1}) = U(\widetilde{T}_i, \widetilde{P}_i; \theta_{i+1})$, then type-$\{i+1\}$ DO can still select $(\widetilde{T}_i, \widetilde{P}_i)$, which introduces loss again.

Lemma 10 *For any feasible random contract $(\widetilde{T}_i, \widetilde{P}_i)$ with $\sigma_i \neq 0$, the greatest revenue $R^*(\widetilde{T}_i, \widetilde{P}_i; \theta_i)$ can be achieved only when the LDICs are binding, which is expressed as*

$$U^*(\widetilde{T}_i, P_i; \theta_i) = U^*(\widetilde{T}_{i-1}, P_{i-1}; \theta_i). \tag{165}$$

Proof 10 *SP aims to establish a sets of contracts to extract the greatest revenue from the contract receivers as much as it can. According to the utility function (133), a type-i DO pays a total of P_i to SP if he signs the type-i contract. Combining (133) and (132), we have the expected revenue from type-$i+1$ DO, which is expressed as*

$$R^*(T_{i+1}, P_{i+1}; \theta_{i+1}) = S(T_{i+1}; \theta_{i+1}) - \frac{\gamma}{\rho(\theta_{i+1})} - U^*_{i+1}, \tag{166}$$

where $S(T_{i+1}; \theta_{i+1}) = E[V(\widetilde{T}_{i+1}; \theta_{i+1})] - T_{i+1} \cdot \alpha_i C_{i+1}$ is the social surplus generated by purchasing the service. If the designer have the complete information about all types of DOs, it can obtain the maximized revenue $\hat{R}$ by extracting all the surplus and leading to a zero utility for DOs.

However, SP can hardly get to know all DOs in practice. For type-$i+1$ DO, the optimal services level T^*_{i+1} is able to be determined in (166), and proved to be unique. With the determined T^*_{i+1}, the revenue function is vertically shifting along $x = T^*_{i+1}$. Assume $(T^*_{i+1}, R^{**}(T^*_{i+1}, \theta_{i+1}))$ is a possible optimal solution that $R^{**}(T^*_{i+1}, \theta_{i+1}) > R^*(T^*_{i+1}, \theta_{i+1})$. We can observe that $U^{**}(T^*_{i+1}, R^{**}_{i+1}; \theta_{i+1}) < U^*(T^*_{i+1}, R^*_{i+1}; \theta_{i+1}) = U^*(T^*_i, R^*_i; \theta_{i+1})$, where violates IC condition. Therefore, the LDIC is binding in this situation.

A. Optimal Solutions to Deterministic Contract

The IR constraint of the lowest type is binding, so that $U(T_1, P_1; \theta_1) = 0$. Together with Lemma 10, we have

$$U(T_{i+1}, P_{i+1}; \theta_{i+1}) = U(T_i, P_i; \theta_{i+1}) = V(T_i; \theta_{i+1}) - h(\theta_{i+1}) - P_i, \tag{167}$$

where $P_i = R_i + T_i \cdot \alpha_i C$. Therefore, we can further have

$$U(T_{i+1}, P_{i+1}; \theta_{i+1}) - U(T_i, P_i; \theta_i) = V(T_i; \theta_{i+1}) - V(T_i; \theta_i) - h(\theta_{i+1}) + h(\theta_i). \tag{168}$$

Combining all the binding constraints together, we can obtain the utility function and expected revenue from the specific type-i DO, which are given by

$$U(T_i, P_i; \theta_i) = \sum_{j=1}^{i-1} \Big\{ V(T_j; \theta_{j+1}) - V(T_j; \theta_j) \Big\} - h(\theta_i) + h(\theta_1) \tag{169}$$

$$\text{and} \quad R_i^* = V(T_i; \theta_i) - T_i \cdot \alpha_i C - h(\theta_1) - \sum_{j=1}^{i-1} \Big\{ V(T_j; \theta_{j+1}) - V(T_j; \theta_j) \Big\}. \tag{170}$$

If the SP only considers the optimum in the single type, then the optimization problem in (143) equals to $\max\limits_{(T_i)} \quad R_i$. With the transformations above, the revenue function R is only related to T_i. Obviously, R is concave with respect of T_i due to $\frac{\partial^2 R}{\partial T_i^2} \leq 0$. The closed form of optimal solution for all types $i \geq 2$, which is expressed as

$$T_i^* = \theta_i - \frac{\log(\alpha_i C / \beta \theta_i)}{\beta \theta_i}. \tag{171}$$

According to the binding IC constraint in (167), we have the optimal price for the random contracts ($i > 1$), which is expressed as

$$P_i^* = \sum_{j=1}^{i-1} \{ V(T_{j+1}; \theta_{j+1}) - V(T_j; \theta_{j+1}) \} + P_1^*, \tag{172}$$

where $P_1^* = V(T_1^*; \theta_1) - h(\theta_1)$.

B. Optimal Solutions to Random Contract

The risk cannot be measured accurately sometimes, which is denoted by the variance σ in this work. However, it is possible to estimate the upper bound $\overline{\sigma}$ and lower bound 0 of the variance, which can be assigned to some of the lower types and higher types, respectively. Our goal is to provide DOs with random contracts while maximizing the SP's profit to make it equal to or even higher than the deterministic contracts. A lazy way to determine the price is by assigning $\overline{\sigma}$ to all the types except for the highest one. However, this will act as a disincentive to accepting the contracts. A wise way is to provide the maximum incentives without loss of profit. We will first introduce how to obtain feasible allocations only with the upper boundary and lower boundary in the following.

1) Unknown variance

According to Lemma 8, we know that if there exists $\widetilde{P}_i \geq P_i$, then $\widetilde{P}_{i+1} \geq P_{i+1}$ must hold. However, together with Lemma 7, it is obvious that $\widetilde{P}_1 < P_1$. Thus, we will first examine the result of $\widetilde{P}_2 - P_2$ by the given $\overline{\sigma} = \sigma_1 = \sigma_2$ and $T_2^* > T_1^*$, which is given by

$$\widetilde{P}_2 - P_2 = E[V(\widetilde{T}_2;\theta_2)] - E[V(\widetilde{T}_1;\theta_2)] + \widetilde{P}_1 - V(\widetilde{T}_2;\theta_2) + V(T_1,\theta_2) - P_1. \tag{173}$$

If $\widetilde{P}_2 - P_2 \geq 0$, we have $e^{\theta_2^2-\theta_1\theta_2}\beta_1^{\frac{\theta_2}{\theta_1}} - \frac{e^{\frac{1}{2}\overline{\sigma}^2\theta_1^2}-1}{e^{\frac{1}{2}\overline{\sigma}^2\theta_2^2}-1}\beta_1 \geq \beta_2$, where $\beta_1 = \frac{\alpha_i C_1}{\theta_1}$ and $\beta_2 = \frac{\alpha_i C_2}{\theta_2}$. In order to compensate for the loss on the lowest-type DOs, the following inequality must hold, which is given by

$$\widetilde{P}_2 - P_2 \geq P_1 - \widetilde{P}_1. \tag{174}$$

To be more precise, the inequality (174) is given by

$$G(\theta_2)(\widetilde{P}_2 - P_2) \geq G(\theta_1)(P_1 - \widetilde{P}_1), \tag{175}$$

where $G(\theta_i)$ is the probability of type-i DOs. If the optimal price $\widetilde{P}_2^*$ and P_2^* that are obtained from binding LDICs do not satisfy the inequality (175), we can proceed to examine $\widetilde{P}_3$ with $\sigma_3 = \overline{\sigma}_3$, until the highest type.

Due to the limited ability in some cases, the system can hardly detect the specific

Algorithm 4 Algorithm for the Pricing of Random Contract with Variance Boundary

Require: The number of types: N, the probability of the different type DOs: $G(\theta_i)$, and the optimal services level T_i^*,the variance boundary $\overline{\sigma}$;

Ensure: The feasible prices $\widetilde{P}_i^*$ of all the random contracts;

Compute $P_1^* = V(T_1^*; \theta_1) - h(\theta_1)$;
for $i = 2$; $i \leq N$; $i++$ **do**
    $P_i^* = \left\{V(T_i^*; \theta_i) - V(T_{i-1}^*; \theta_i)\right\} + P_{i-1}^*$;
end for
Compute $\widetilde{P}_1^* = EV(T_1^*; \theta_1, \overline{\sigma}) - h(\theta_1)$;
Set $sig = 0, \phi = \widetilde{P}_1^*, S = G(\theta_1)(\phi - P_1)$;
for $i = 2$; $i \leq k$; $i++$ **do**
    if $sig == 0$ **then**
        $\phi = EV(T_i^*; \theta_i) - EV(T_{i-1}^*; \theta_i, \overline{\sigma}) + \phi$;
    else
        $\phi = \left\{V(T_i^*; \theta_i) - V(T_{i-1}^*; \theta_i)\right\} + \phi$;
    end if
    if $G(\theta_i)(\phi - P_i) + S \geq 0$ **then**
        $sig = 1$;
    else
        $sig = 0$;
    end if
    $S = G(\theta_i)(\phi - P_i) + S$;
    $\widetilde{P}_i^* = \phi$;
end for
if $S \geq 0$ **then**
    $\widetilde{P}_{k+1}^* = V(T_{k+1}^*; \theta_{k+1}) - EV(T_k^*; \theta_{k+1}, \overline{\sigma}) + \widetilde{P}_k$;
    for $j = k+2; j \leq N; j++$ **do**
        $\widetilde{P}_j^* = V(T_j^*; \theta_j) - V(T_{j-1}^*; \theta_j) + \widetilde{P}_{j-1}^*$;
    end for
else
    Set $\eta = V(T_{k+1}^*; \theta_k) - V(T_k^*; \theta_k) + P_k$;
    Compute $\varepsilon = \frac{-S}{\sum_{j=k+1}^{N} G(\theta_j)}$;
    Compute $\widetilde{P}_{k+1}^* = \eta + \varepsilon$;
    for $j = k+2; j \leq N; j++$ **do**
        $\widetilde{P}_j^* = V(T_j^*; \theta_{j-1}) - V(T_{j-1}^*; \theta_{j-1}) + \widetilde{P}_{j-1}^*$;
    end for
end if

variance accurately. If we can find a price from the random contract that can compensate for all the loss on the lower types with the upper bound $\bar{\sigma}$, then the other prices can be revised accordingly based on any deterministic one. Otherwise, we have to charge more on the higher types with the lower bound of variance $\sigma = 0$. We can turn to reducing the upward constraints so that their type of contract can provide the reservation utility. We have proved this statement in Lemma 161, which is also explained by Bolton in [44] and by Maskin in [154].

Algorithm 5 Algorithm for the Pricing of Random Contract with All Known Variances.

Require: The number of types: N, the probability of the different type DOs: $G(\theta_i)$, the optimal services level T_i^*, and the optimal prices of deterministic contract P_i^*, all variance σ_i;

Ensure: The feasible prices $\widetilde{P}_i^*$ of all the random contracts;

Compute $\widetilde{P}_1^* = EV(T_1^*; \theta_1, \sigma_1) - h(\theta_1)$;
Set $S = G(\theta_1)(\widetilde{P}_1^* - P_1)$;
for $i = 2$; $i \leq k$; $i++$ **do**
    $\widetilde{P}_i^* = EV(T_i^*; \theta_i, \sigma_i) - EV(T_{i-1}^*; \theta_i, \sigma_i) + \widetilde{P}_{i-1}^*$;
    $S = G(\theta_i)(\widetilde{P}_i^* - P_i) + S$;
end for
if $S \geq 0$ **then**
    $\widetilde{P}_{k+1}^* = V(T_{k+1}^*; \theta_{k+1}) - EV(T_k^*; \theta_{k+1}, \sigma_i) + \widetilde{P}_k$;
    for $j = k+2; j \leq N; j++$ **do**
        $\widetilde{P}_j^* = P_j^* - P_{k+1}^* + \widetilde{P}_{j-1}^*$;
    end for
else
    Set $\eta = V(T_{k+1}^*; \theta_k) - V(T_k^*; \theta_k) + P_k$;
    Compute $\varepsilon = \frac{-S}{\sum_{j=k+1}^{N} G(\theta_j)}$;
    Compute $\widetilde{P}_{k+1}^* = \eta + \varepsilon$;
    for $j = k+2; j \leq N; j++$ **do**
        $\widetilde{P}_j^* = V(T_j^*; \theta_{j-1}) - V(T_{j-1}^*; \theta_{j-1}) + \widetilde{P}_{j-1}^*$;
    end for
end if

In this way, random contracts' profit can be guaranteed higher than or equal to the deterministic contracts. However, it is not easy to extract maximized revenue without known variances. The algorithm is presented in Algorithm 4.

2) Known variance

Since the advanced development of machine learning in dealing with risk detection and estimation, even if costly, some internet service providers prefer such a measurement approach to maximize revenue while providing better services. Thus, SP can obtain more accurate distributions regarding the services, such as $\sigma_1 \geq \cdots \geq \sigma_k \geq \sigma_{k+1} =, \ldots, = \sigma_N = 0$.

Together with Lemma 10, the maximized revenue can be achieved from binding LDICs. Thus, with a particular risk (i.e., known variance), DOs will surely obtain a lower utility, while SP will be beneficial from the random contracts. This statement is proved and explained in Section 4.3.2 by two propositions. We present the details of the algorithm in Algorithm 5.

We can easily obtain the service price of the type-1 random contract through the

binding IR constraint. Assume that all the variances are unknown except variance boundary $\overline{\sigma}$. We first calculate the type-2 price with the boundary. According to Fig. 31, we set up an indicator "$sig = 0$" to denote that the total random contract prices are lower than the deterministic contract prices. Otherwise, "$sig = 1$" means that the random contract prices have already exceeded the deterministic prices. If the sum of random contract prices is lower than that of deterministic prices, we will check the next type's price with the variance boundary. It is possible to improve the price with a random service level compared to the deterministic price. Until the sum is greater than the deterministic prices, the algorithm will turn to calculate the price with zero variance. Assume that $\widetilde{P}_m$ is the first assigned price that $S(\widetilde{P}_m) \geq 0$ holds, we can conclude that $\widetilde{P}_m > P_m$. According to Lemma 8, all the subsequent prices of random contracts satisfy that $\widetilde{P}_{m+1} > P_{m+1}$. Therefore, there is no revenue loss in the random contracts starting from the current type m. When pricing the types i ($\forall i \in \{i, \ldots, k\}$) participants, the contracts will offer a random service level with a variance to type-j ($\forall j \in \{1, \ldots, m\}$) participants, while offering a deterministic service level to the other $(k - m)$ types of participants.

We present the other flowchart in Fig. 3 to illustrate the calculation of prices $\widetilde{P}_j$ ($\forall j \in \{k + 1, \ldots, N\}$). Suppose that the random contract offers a price that can compensate for all the lower types' loss with the upper bound $\overline{\sigma}$. In that case, the other types' prices can be revised accordingly based on any deterministic price, which is shown on the left side of Fig. 32 (denoted with the dotted rectangle box). Otherwise, we have to charge more on the higher types with the lower bound of variance $\underline{\sigma} = 0$. As shown on the right side of Fig. 32 (denoted with the solid rectangle box), we can turn to reduce the upward constraints so that their type of contract can provide the reservation utility. We prove this statement in Lemma 161, which is also explained by Bolton [44].

4.3.2 Feasibility of Random Contract

In this subsection, we will analyze the impact on DO's utility and SP's revenue when introducing the randomness through the following propositions.

Proposition 2 *For any contract satisfying Assumption 1, the optimal deterministic*

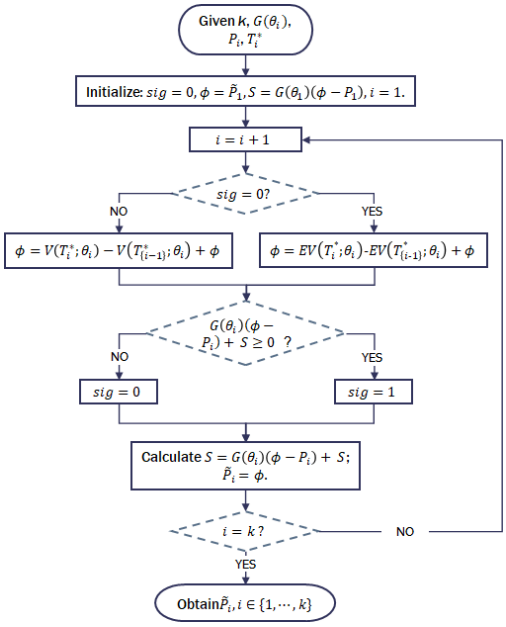


Figure 31: Flowchart of Algorithm 1 (a).

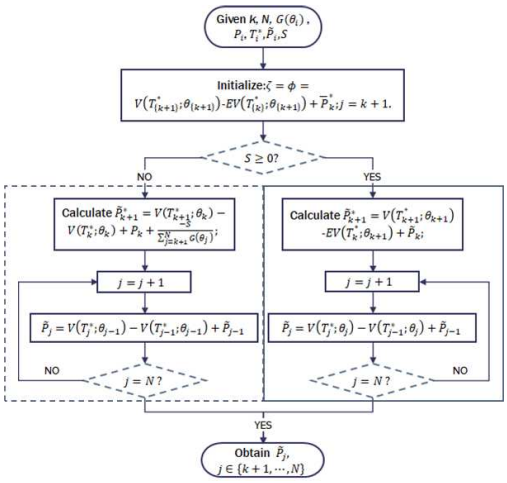


Figure 32: Flowchart of Algorithm 1 (b).

contract provides no lower utility than that of any random one, which is expressed as

$$U^*(\widetilde{T}_i, P_i; \theta_i) \leq U^*(T_i, P_i; \theta_i). \tag{176}$$

Proof 11 *According to the solution (169), we have*

$$U^*(\widetilde{T}_i, P_i; \theta_i) = \sum_{j=1}^{i-1} \left\{ E[V(\widetilde{T}_j; \theta_{j+1})] - E[V(\widetilde{T}_j; \theta_j)] \right\} + \Delta(\theta_j), \tag{177}$$

where $\Delta(\theta_j) = h(\theta_1) - h(\theta_i)$.

We transform the the first term on the right side of equation (177), which is expressed

as

$$\begin{aligned}\varpi &= \sum_{j=1}^{i-1}\Big\{\int_{-\infty}^{+\infty}\Big[V(\widetilde{T}_j;\theta_{j+1}) - V(\widetilde{T}_j;\theta_j)\Big]\mathrm{d}F(\widetilde{T}_j)\Big\} \\ &= \sum_{j=1}^{i-1}\Big\{\int_{\theta_j}^{\theta_{j+1}} E[V_2(\widetilde{T}_j;k)]\mathrm{d}k\Big\}.\end{aligned} \tag{178}$$

For the random variable $\widetilde{T}_i$, we can obtain that $\widetilde{T}_i = V^{-1}(\tau,\theta_i)$ by setting $\tau = V(\widetilde{T}_i,\theta_i)$. Thus, we have $V_2(\widetilde{T}_i,k) = V_2(V^{-1}(\tau,\theta_i),k)$. Then we obtain the second derivative of $V_2(\cdot)$ with respect to $\widetilde{T}_i$, which is given by

$$\frac{\partial V_2}{\partial \widetilde{T}_i} = \frac{\mathrm{d}V_2}{\mathrm{d}\tau}\cdot\frac{\mathrm{d}\tau}{\mathrm{d}\widetilde{T}_i} = \frac{V_{21}}{V_1} = \frac{\nu_2}{\nu}. \tag{179}$$

where $V_{21} = V_{12} = \nu_2$ and $V_1 = \nu$. [2] *Then we obtain the result by rearranging the derivative, which is expressed as*

$$\frac{\partial^2 V_2}{\partial \widetilde{T}_i^{\,2}} = \frac{\partial}{\partial \widetilde{T}_i}\left(\frac{\nu_2}{\nu}\right) = \frac{\nu_{21}\nu - \nu_2\nu_1}{\nu^2} = \frac{\nu_{12}\nu - \nu_1\nu_2}{\nu^2} = \frac{\partial}{\partial\theta_i}\left(\frac{\nu_1}{\nu}\right). \tag{180}$$

According to Assumption 1 and the utility function proposed in this work, we have $\frac{\partial^2 U}{\partial T^2} = V_{11} = \nu_1$ and $\frac{\partial U}{\partial T} = V_1 = \nu$. Thus, $\frac{\partial^2 V_2}{\partial \widetilde{T}^2} < 0$, which means V_2 is concave. Then according to Jensen's inequality, we have

$$\varpi = \sum_{j=1}^{i-1}\int E[V_2(\widetilde{T},k)]\mathrm{d}k \le \sum_{j=1}^{i-1}\int V_2(T,k)\mathrm{d}k. \tag{181}$$

Then we come to the conclusion that $U^*(\widetilde{T}_i,P_i;\theta_i) \le U^*(T_i,P_i;\theta_i)$. We complete this proof in a general form, which means any set of contracts satisfying Assumption 1 have the same conclusion.

From proposition 2, we can get to know that random contract will reduce the utilities of DOs, which means the slashed part may be converted to the profit of SP. We will prove that it is possible for random contracts to catch up to deterministic contracts in profit.

Proposition 3 *For the feasible sets of random contract $(\widetilde{T}_i,\widetilde{P}_i)$ with $\theta_1 < \cdots < \theta_k$*

[2] *$V(T_i,\theta_i)$ is continuous in T_i and θ_i, thus, we have $V_{21} = V_{12}$.*

and $\sigma_i \neq 0$, there exists at least one optimal random contract provides at least as much expected revenue as the deterministic one if the total number of types k is big enough, which is expressed as

$$R^*(\widetilde{T}_i, \widetilde{P}_i; \theta_i) \geq R^*(T_i, P_i; \theta_i). \tag{182}$$

Proof 12 *We define $K(T_i, \theta_i) = E[V(\widetilde{T}_i; \theta_i)]$ and set $N(T_i, \theta_i) = V(T_i, \theta_i) - K(T_i, \theta_i)$. Together with equation (172), we have the gap between prices, which is expressed as*

$$P_i^* - \widetilde{P}_i^* = \sum_{j=1}^{i-1} \left\{ \int_{T_j}^{T_{j+1}} N_1(x, \theta_{j+1}) \mathrm{d}k \right\} + P_1^* - \widetilde{P}_1^*. \tag{183}$$

According to equations (139) and (140), we have

$$\begin{cases} N_1 = V_1 - K_1 < 0, \\ N_{12} = \dfrac{\partial(V_2(\cdot) - K_2(\cdot))}{\partial T} = V_{21} - K_{21} < 0. \end{cases} \tag{184}$$

The gap is decreasing in T_i and θ_i. Therefore, with the variance $\sigma_i \neq 0$, we can finally find out a feasible $\widetilde{P}_i$ that $P_i^* \leq \widetilde{P}_i^*$ and $R^*(\widetilde{T}_i, \widetilde{P}_i; \theta_i) \geq R^*(T_i, P_i; \theta_i)$.

4.4 Numerical Results and Analysis

This section will first calculate the future transaction fee price and other practical parameters in a risk-neutral market. Together with the parameter setting, we present the performance and expenditure comparison between the random and deterministic contracts, as well as the sidechain and main chain. Finally, we also verify the feasibility of the proposed random contract.

We collect the real data of Ethereum in the past two years, including 732 days average transaction fees [155]. Without loss of generality, we fit the price fluctuation with Brownian motion with the differences in logarithmic prices. We first obtain the logarithmic difference through the following equation, which is expressed as

$$d_i = \log(C_i) - \log(C_i - 1), \tag{185}$$

where C_i denotes the price of day i. Then we have the mean and variance of the

Table 10: Parameter Setting

Parameters	Values
Latest Average Transaction fees	$C = 2.927$
The percentage drift	$\mu = -0.0035$
The percentage volatility	$\sigma = 0.3142$
Confirmation period	$\bar{T} = 2$
The number of types	$N = 10$
Pre-defined coefficient	$\beta = 0.1, \gamma = 0.5$
Discount factor	$\alpha_i = \{0.1, 0.2\}$
The number of lower-types	$k = 6$

logarithmic difference sample data, which is expressed as

$$\mu = \sum_{i=1}^{731} d_i/731 \tag{186}$$

$$\text{and} \quad \sigma = \sqrt{\frac{1}{731}\sum_{i=1}^{731}(d_i - \mu)^2}. \tag{187}$$

With historical data, we can calculate sample mean $\mu = -0.0035$ and sample variance $\sigma = 0.3142$. According to the sidechain white-paper in [136], a typical confirmation period $\bar{T}$ is on the order of a day or two. We define a different discount factor of the transaction fee for the lower-type and higher-type DOs, respectively, with $\alpha_i = 0.1, \forall i \in \{1, \ldots, k\}$ and $\alpha_i = 0.2, \forall i \in \{k+1, \ldots, N\}$. We assume the number of DO types is $N = 10$ with $\theta = \{5, 6, 7, 8, 9, 12, 14, 16, 17, 20\}$, and the types follow a uniform distribution with $G(\theta_i) = 0.1$. In the proposed design, the number of higher-type contracts providing deterministic items with $\sigma_i = 0$ is assumed to be 4, so the number of lower-type contracts is $k = N - 4 = 6$. The main parameters are shown in Table 10.

We present the optimal service level T_i^* for each DO's type in Fig. 4. With the different coefficients β, we can observe that all the optimal solutions are practical since the offered services are greater than the demands θ. The trend of T_i^* is increasing in DO's type, which is consistent with Lemma 5 in Section 4.3.1. Moreover, the optimal service level decreases as the coefficient β is growing. When $\beta = 1$, T_i^* almost equals its corresponding demand. In this work, we present the deterministic and random forms of the contract design. The optimal service level is the fixed allocation offered by the

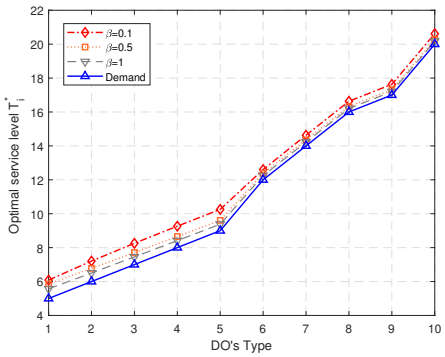


Figure 33: Optimal service level T_i^*.

deterministic contract. Compared to the deterministic contract, the proposed random contract provides a random allocation, a random variable that follows the normal distribution with a particular mean T_i^* and variance σ_i. Considering the uncertainty of network and miners, SP offering a random allocation to the DOs with less risk-aversion is practical. Especially in the decentralized sidechain network, SP cannot control and schedule all the transaction procedures. Thus, SP will only assure a few types of DOs that they can obtain deterministic allocations.

To further analyze the parameters' impacts on the optimal results, we take type-1 DO's service level as an example, showing the changes in optimal results when the coefficient β and discount factor α vary. As shown in Fig. 34, we can observe that the optimal service level can achieve a high value if reducing the discount factor. With a smaller discount factor, SP can interact with the main chain at a lower transaction fee cost, incentivizing SP to improve its service level for a more significant profit. Since we adopt the exponential functions to formulate the utilities, the trend of optimal results would be drastic. Thus, we use the coefficient β to relax the trend. In Fig. 34, only the values exceeding the demand are valid. The levels achieve a peak at around $\beta = 0.1$. All the peak levels are greater than the demand. After the peak values, the optimal service level decreases as β is growing and seems to plateau, starting from $\beta = 0.4$. Therefore, Fig. 34 allows us to determine the feasible sets of these two parameters. For the following simulations and analysis, we set the parameters to $\beta = 0.1$ and $\alpha = \{0.1, 0.2\}$.

Fig. 35 shows the price assignments comparison between the random contract and deterministic contract. We can observe that all the prices increase in DO's types, which

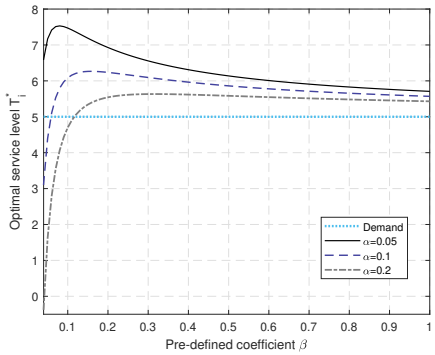


Figure 34: Optimal service level T_i^*.

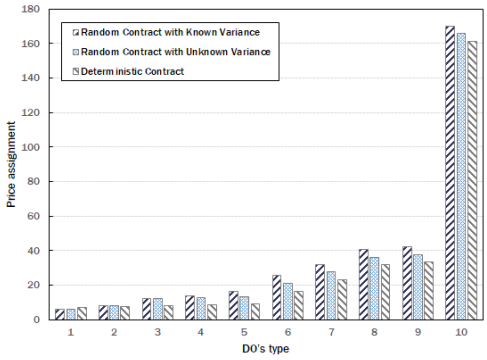


Figure 35: The price assignment for deterministic and random contracts.

substantiates the Lemma 6. For type-1 contracts, the price of deterministic is higher than the random contract due to Lemma 7. Through the algorithm 4, we have $P_1 - \widetilde{P}_1 > \widetilde{P}_2 - P_2$, but $P_1 - \widetilde{P}_1 < \widetilde{P}_2 - P_2 + \widetilde{P}_3 - P_3$. That means the prices of type-3 and type-2 can compensate for the loss on type-1. Through the algorithm 5, we come to the same conclusion. Besides, the gap between deterministic and random contracts is decreasing, as the claim in proposition 3. Starting at the third type, the prices of random contracts are higher than the deterministic ones, confirming the Lemma 8. With unknown variance, we cannot determine a specific price from the binding LDICs of random contracts. We can only obtain the price with a variance boundary and binding LDICs. Once there exists a price $\widetilde{P}_i$ can compensate for all the loss, we have to stop using it and turn to revising the price obtained in deterministic contracts by adding the difference $\widetilde{P}_i - P_i$. If we assume all the lower-type contracts share the same variance, we find out that the IC condition cannot hold within some contracts. With the parameters in Table 10, we can obtain type-3 price, which can compensate for the loss on type-2 and type-1 DOs, so that the following prices can be modified by adding $\widetilde{P}_3 - P_3$. We

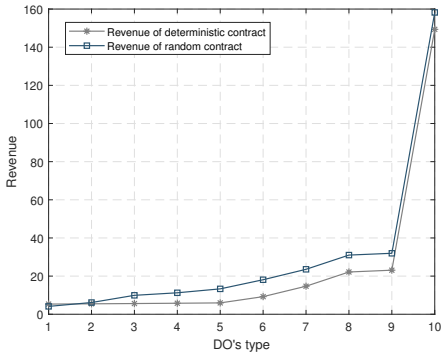


Figure 36: Revenue provided by random and deterministic contracts.

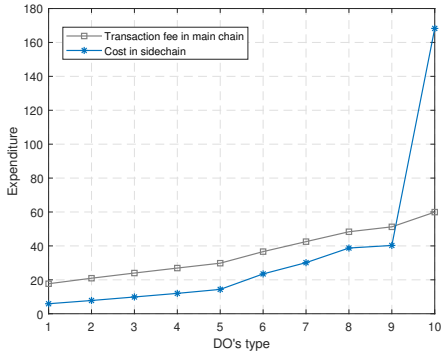


Figure 37: The expenditure comparison between main chain and sidechain.

can work out an accurate price with the known variance that makes the LDIC binds at optimum. Thus, the prices of random contracts with known variance are higher than the other cases.

With the parameter setting and price assignments, we can obtain the revenue comparison between the random and deterministic contracts in Fig. 36. Given $\widetilde{P}_1 < P_1$ in the last figure, we have $\widetilde{R}_1 < R_1$. The following prices are increasing and remain greater than that of deterministic contracts. Fig. 36 depicts that the proposed random contract's revenue dominates any deterministic one except for the first one. By such design, even though the highest-type DOs select type-$N-1$ instead of type-N contract, SP will not suffer losses. Based on the numerical results, the random contracts can improve the revenue by an average of 24.70% compared to the deterministic contracts.

Together with the average predicted transaction fee C, we have the average cost $\mathcal{C}_i$ for each type-i in the main chain, which is expressed as

$$\mathcal{C}_i = T_i^* \cdot C. \tag{188}$$

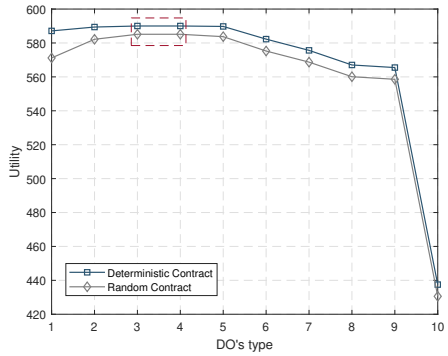


Figure 38: Utilities of type-4 DOs when signing all types of contracts.

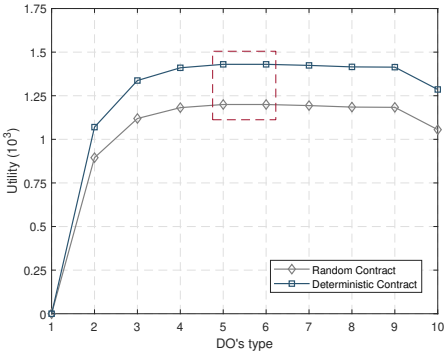


Figure 39: Utilities of type-6 DOs when signing all types of contracts.

We can observe that the prices of the proposed contracts applied in the sidechain are lower than the average transaction fees in the main chain except for the highest type DOs. Due to binding LDICs, type-$(N-1)$ contracts can provide the highest-type DOs with reservation utilities, which means the type-N DOs have the option of choosing type-$(N-1)$ contracts. If so, SP remains profitable, according to Fig. 37. For all the efficient price assignments, the service payments can be reduced by an average of 44.65% compared to the main chain's cost.

To verify the IR and IC conditions of proposed contracts, we select two representative types of DOs, plotting the utilities with all types of random and deterministic contracts. No matter under the random or deterministic cases, Fig. 38 shows that type-4 DOs can obtain the maximized utility only when signing its type or downward neighboring type contract. Moreover, the deterministic contract curve locates upper than the random one because the random contracts make a substantial profit at the expense of DOs' utility. We can come to the same conclusion in Fig. 39. Both of the two figures can prove the IR condition holds due to the non-negative utilities provided by their type contracts.

4.5 Summary

In the conventional public blockchain network, the growing transaction fee leads to more and more pending transactions waiting in the pool. Such a scalability issue has become a severe handicap for all DOs. By applying a sidechain linked to the main chain, the proposed scheme can provide proper pricing and transaction management for heterogeneous device owners in the smart community. We propose a random contract model under asymmetric information for the sidechain network. Concerning the uncertainty of network and miners' effort, we develop a pricing scheme for SP and distribute a random allocation to DOs according to their risk-bearing capability. Specifically, with the increasing risk-aversion assumption, we design a series of contracts with random items, which maximize SP's revenue and bring an equal or more considerable revenue than the deterministic contract. We present several lemmas to prove the superiority of random contracts. Together with the GBM model, we adopt real data of the average Ethereum transaction fee and determine the sidechain price assignment in the simulation Section. We demonstrate that the proposed random contract is dominant by comparing the deterministic contract and fees on the main chain.

This work only considers the pricing strategy for transaction recording. Since the transaction fees are determined solely based on the transaction validation and independent of the mining process, the transaction service pricing includes no mining reward. The service payments collected by the sidechain service provider will be finally issued to the miners in the form of the transaction fee. Generally, the price is affected by supply and demand. The same is true for transaction fee pricing. A proper price can incentivize both the miners and participants, contributing to the blockchain network's performance improvement. Using the game-theoretical approach to defend the node collusion or other malicious behaviors is a promising and significant topic, whereas it is not the focus of this work. Nevertheless, the proposed incentive mechanism can be built atop any consensus layer or security protocol. As an extensive form of the public blockchain, the sidechain is advantageous in reducing the pending transactions number and saving transaction fee costs [15]. A significant challenge of this random contract approach is to estimate and forecast the probability distribution of sidechain performance

on recording the transactions. The other difficulty is verifying the ex-post service levels offered by a sidechain. Fortunately, with the advances in optimization and machine learning techniques, we can obtain an accurate probability distribution regarding the performance of recording transactions. Furthermore, such techniques combined with cryptography allow us to measure and verify the ex-post service level.

5 Future Work and Conclusion

Cryptoeconomics is a new interpretation of blockchain, but the relevant research is scant. For this reason, we study three categories of economic incentives in cryptoeconomics: penalty, reward, and transaction fee. In this dissertation, we first focus on security deposit pricing. To deal with the issue that a fixed security deposit has violated the no entry-limitation principles of the public blockchain, we design a set of flexible security deposits, enabling the participants with various budgets to be able to join the network. We can ensure the economic incentives through flexible security deposits without reducing the security incentives. Then the second work devises a cyber-insurance framework to resist the potential cyber-attacks in PoS-based networks. A representative example of the "potential cyber-attacks" is a discouragement attack. We calculate the loss function of discouragement attacks and use a hierarchical game model to design the cyber-insurance contract for the victims and blockchain infrastructure. In the last work, we propose a random-contract-based scheme to maximize the sidechain service provider's revenue and assign the service buyers the feasible service price under the framework of a sidechain linked to the public blockchain. We systematically demonstrate random contracts' superiority under the increasing absolute risk-aversion assumption. By applying Geometric Brownian Motion, we study service pricing in different cases. This dissertation only covered a minor part of incentive issues. The relevant research of cryptoeconomics is still in its infancy. Below listed are some topics of our future work:

5.1 Voting Weight

Ethereum has made great strides in the attempts to transit from Proof-of-Work (PoW) to Proof-of-Stake (PoS). The security of Ethereum 2.0 is dependent on the quantity of security deposits and the assumption of a deposit-weighted majority rather than a tremendous computation. By voting, all participants will be able to implement various blockchain functions, with the voting weight based on the number of deposits they have made. Furthermore, abundant resources of nodes (e.g., network bandwidth and storage) are crucial to the smooth operation of a blockchain network. In conclusion, the security and functionality of a PoS-based blockchain are remarkably correlated with the

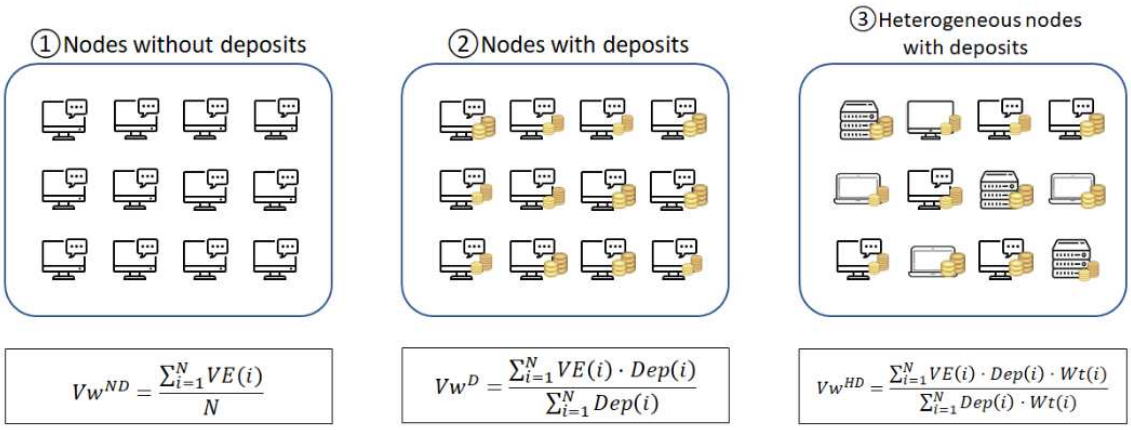


Figure 40: Three types of weight assignment.

security deposit, voting weight, and node diversity. However, there is scant research on these aspects. This work aims to use a Tullock contest to present a weight assignment scheme and a grade-based signaling model to investigate deposit strategies. Regarding the performance as a grade and the security deposit as a costly signal, the combined model enables the blockchain to motivate participants' implemented performance level and the participants' strategies on deposit submission. To obtain the heterogeneous participants' Nash equilibrium of the Tullock contest and the equilibrium sets of the grade-based signaling model is the difficult point of this work. We have reviewed the three major type weight assignment, which is shown in Fig. 40.

5.2 Crypto-asset value stabilization

Crypto-assets are a type of digital assets, which are designed using the decentralized ledger technology. The practical examples includes Bitcoin and stablecoin. BitUSD, the first stablecoin initiative, was launched in 2014 and was collateralized by other cryptocurrencies rather than fiat assets. The groundbreaking digital asset was created by two future cryptocurrency industry leaders, Dan Larimer (EOS) and Charles Hoskinson (Cardano). Tether, which was created in November 2014, has played a critical part in the emergence of cryptocurrencies and is the most well-known stable asset now on the market.

Considering the significant fluctuation of crypto-assets, we aim to offer insurance contracts to the crypto-assets owners, ensure the value will not drop below the reserved

price within a required time span. We establish an insurance framework using contract theory, the items include time span, reserved value, risk type. Insurer needs to evaluate the risk type of participants in advance. The contract items are determined by using Option Pricing Model and risk parameters. If the risk level (according to risk parameters) is high, the insured crypto-assets will be liquidated through automated auctions by using the smart contract. The goals of this work include: 1) A Risk-sharing Decentralized insurance, 2) Transfer the financial risk to option market, and 3) Protect insurant against both technical and financial risks.

5.3 Conclusion

As we discussed before, Bitcoin is the first as well as the most significant instance of cryptoeconomics. The integration of cryptography and mechanism design sparks a revolutionary shift from a traditional P2P network to a blockchain network.

Although cryptography is robust when assuring the security and privacy of a system, the cost of development and deployment is increasingly expensive due to the unpredictable attacks and risks. According to Vlad's talk in [5], cryptoeconomics can benefit the following issues: 1) The dis-incentivization of Byzantine faults: Bitcoin uses the Proof-of-Work consensus and incentive mechanism to solve the Byzantine general problem. 2) The "individual rationality" of deciding whether to run a node on a blockchain protocol: how does the incentive mechanism maximize the users' utilities when running a node on blockchain. 3) The economic barriers to Sybil attacks: A proper economic incentive mechanism is able to resist Sybil attacks without Proof-of-Work.

Based on the explanations and analysis of cryptoeconomics research, we can conclude the scenarios in which cryptoeconomics can be applied as follows: 1) Security of lower layer interactions: any channel between a pair of participants, e.g. state channel and payment channel. 2) Light clients: how should we design a fast and feasible incentive mechanism for a group of participants and deploy it in a decentralize system? 3) Decentralized Applications: encourage the users' participation and activity. 4) DoS resistance of off-chain protocols: guarantee the security of on-chain assets and motivate the efforts of off-chain users. 5) Blockchain-based peer to peer markets: incentivize the

participants from the different chains to interact in a desired way.

This dissertation has attempted to deal with the security deposit pricing for the light clients and enabled the incentive mechanisms to serve for a group of participants. Also, we have analyzed the interactions among various parties involved in a blockchain-based market, designing the compensation scheme for the victims so that we can regulate the participants' behaviours. Finally, we have considered the off-chain transaction fee pricing for a sidechain service provider, and motivating the efforts of off-chain users. There are still lots of research gaps within the cryptoeconomics area, we hope this dissertation can serve as a reference for the later research.

References

[1] W. Wang, D. T. Hoang, P. Hu, Z. Xiong, D. Niyato, P. Wang, Y. Wen, and D. I. Kim, "A survey on consensus mechanisms and mining strategy management in blockchain networks," *IEEE Access*, vol. 7, pp. 22 328–22 370, 2019.

[2] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf, [Accessed: Nov. 14, 2021].

[3] L. A. Members, "Libra white paper," 2020. [Online]. Available: https://libra.org/en-US/white-paper/, [Accessed: Dec. 21, 2021].

[4] S. Inc., "Shiftmobility liberates vehicle data with world's first secure automotive blockchain platform," 2018. [Online]. Available: http://shiftmobility.com/press-releases/shiftmobility-liberates-vehicle-data-worlds-first-secure-automotive-blockchain-platform/, [Accessed: Aug. 3, 2019].

[5] V. Zamfir, "Bip001: Unlimited edition," 2017. [Online]. Available: https://www.youtube.com/watch?v=u6VSPD5TrP4&t=371s, [Accessed: Sept. 19, 2020].

[6] A. M. Antonopoulos, *Mastering Bitcoin: unlocking digital cryptocurrencies*. O'Reilly Media, Inc., 2014.

[7] S. Haber and W. S. Stornetta, "How to time-stamp a digital document," in *Conference on the Theory and Application of Cryptography*. Springer, 1990, pp. 437–455.

[8] A. Back, "Hashcash - a denial of service counter-measure," 2002. [Online]. Available: https://www.cs.miami.edu/home/burt/learning/csc686.211/docs/hashcash.pdf, [Accessed: Jan 15 2022].

[9] R. C. Merkle, "Protocols for public key cryptosystems," in *IEEE Symposium on Security and Privacy*. IEEE Computer Society, 1980, pp. 122–122.

[10] D. Johnson, A. Menezes, and S. Vanstone, "The elliptic curve digital signature algorithm (ecdsa)," *International journal of information security*, vol. 1, no. 1, pp. 36–63, 2001.

[11] Google, "Leveldb," 2011. [Online]. Available: https://opensource.googleblog.com/2011/07/leveldb-fast-persistent-key-value-store.html, [Accessed: Oct. 12, 2021].

[12] E. Foundation, "Ethereum," 2013. [Online]. Available: https://www.ethereum.org/, [Accessed: Dec. 12, 2020].

[13] A. Demers, D. Greene, C. Hauser, W. Irish, J. Larson, S. Shenker, H. Sturgis, D. Swinehart, and D. Terry, "Epidemic algorithms for replicated database maintenance," in *Proceedings of the sixth annual ACM Symposium on Principles of distributed computing*. ACM, 1987, pp. 1–12.

[14] H. QURESHI, "Bitcoin's p2p network," 2019. [Online]. Available: https://nakamoto.com/bitcoins-p2p-network/, [Accessed: Jan. 15, 2021].

[15] P. Maymounkov and D. Mazieres, "Kademlia: A peer-to-peer information system based on the xor metric," in *International Workshop on Peer-to-Peer Systems*. Springer, 2002, pp. 53–65.

[16] E. Rohrer and F. Tschorsch, "Kadcast: A structured approach to broadcast in blockchain networks," in *Proceedings of the 1st ACM Conference on Advances in Financial Technologies*. ACM, 2019, pp. 199–213.

[17] W. Wang, D. T. Hoang, P. Hu, Z. Xiong, D. Niyato, P. Wang, Y. Wen, and D. I. Kim, "A survey on consensus mechanisms and mining strategy management in blockchain networks," *IEEE Access*, vol. 7, pp. 22 328–22 370, 2019.

[18] wackerow, "Proof-of-stake (pos)," 2022. [Online]. Available: https://ethereum.org/en/developers/docs/consensus-mechanisms/pos/, [Accessed: Feb. 22, 2022].

[19] V. Foundation, "Vechain white paper 2.0," 2019. [Online]. Available: https://www.vechain.org/whitepaper/#bit_v48i3, [Accessed: Feb. 15, 2022].

[20] B. Chase and E. MacBrough, "Analysis of the xrp ledger consensus protocol," 2018. [Online]. Available: https:\\arXivpreprintarXiv:1802.07242, [Accessed: Feb. 11, 2019].

[21] V. Buterin and V. Griffith, "Casper the friendly finality gadget," 2017. [Online]. Available: https://arxiv.org/abs/1710.09437, [Accessed: Aug. 10, 2019].

[22] E. Foundation, "How to stake your eth," 2021. [Online]. Available: https://ethereum.org/en/staking/#stake, [Accessed: Feb. 12, 2022].

[23] S. Leonardos, D. Reijsbergen, and G. Piliouras, "Weighted voting on the blockchain: Improving consensus in proof of stake protocols," *International Journal of Network Management*, vol. 30, no. 5, p. e2093, 2020.

[24] E. Community, "Ethereum virtual machine," 2022. [Online]. Available: https://ethereum.org/en/developers/docs/evm/, [Accessed: April 3, 2022].

[25] E. Foundation, "Introduction to smart contracts," 2022. [Online]. Available: https://ethereum.org/en/developers/docs/smart-contracts/, [Accessed: March 3, 2022].

[26] V. Zamfir, "What is cryptoeconomics?" 2015. [Online]. Available: https://www.youtube.com/watch?v=9lw3s7iGUXQ, [Accessed: Jan. 15, 2021].

[27] V. Buterin, "Introduction to cryptoeconomics," 2017. [Online]. Available: https://www.youtube.com/watch?v=pKqdjaH1dRo&t=413s, [Accessed: Aug. 19, 2021].

[28] J. Stark, "Making sense of cryptoeconomics," 2017. [Online]. Available: https://medium.com/l4-media/making-sense-of-cryptoeconomics-5edea77e4e8d, [Accessed: Jan. 4, 2022].

[29] L. Schilling and H. Uhlig, "Some simple bitcoin economics," *Journal of Monetary Economics*, vol. 106, pp. 16–26, 2019.

[30] B. N. Levine, C. Shields, and N. B. Margolin, "A survey of solutions to the sybil attack," *University of Massachusetts Amherst, Amherst, MA*, vol. 7, p. 224, 2006.

[31] V. Buterin, "Cryptoeconomics in 30 minutes by vitalik buterin (devcon5)," 2019. [Online]. Available: https://www.youtube.com/watch?v=GQR1xjQn5Pg, [Accessed: Dec. 19, 2021].

[32] E. Foundation, "Programmable incentives: Intro to cryptoeconomics," 2017. [Online]. Available: https://www.youtube.com/watch?v=9lw3s7iGUXQ, [Accessed: Dec. 3, 2020].

[33] T. C. Schelling, "The strategy of conflict. prospectus for a reorientation of game theory," *Journal of Conflict Resolution*, vol. 2, no. 3, pp. 203–264, 1958.

[34] O. Lab, "Inside arbitrum," 2021. [Online]. Available: https://developer.offchainlabs.com/docs/inside_arbitrum, [Accessed: Jan. 4, 2021].

[35] C. Lab, "Introduction to chainlink keepers," 2021. [Online]. Available: https://docs.chain.link/docs/chainlink-keepers/introduction/, [Accessed: Jan. 5, 2022].

[36] T. Bean, "bzx integrates chainlink keepers," 2021. [Online]. Available: https://bzx.network/blog/bzx-integrates-chainlink-keepers, [Accessed: Jan. 4, 2022].

[37] B. M. Musser, "xtoken lending, powered by chainlink," 2021. [Online]. Available: https://medium.com/xtoken/xlend-powered-by-chainlink-12f6c7ff21e7, [Accessed: Dec. 21, 2021].

[38] Chainlink, "Smart contract automation: How to get started with chainlink keepers," 2021. [Online]. Available: https://blog.chain.link/smart-contract-automation/, [Accessed: Jan. 4, 2022].

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

[40] X. Liu, W. Wang, D. Niyato, N. Zhao, and P. Wang, "Evolutionary game for mining pool selection in blockchain networks," *IEEE Wireless Communications Letters*, vol. 7, no. 5, pp. 760–763, 2018.

[41] S. Feng, Z. Xiong, D. Niyato, P. Wang, S. S. Wang, and Y. Zhang, "Cyber risk management with risk aware cyber-insurance in blockchain networks," in *2018 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2018, pp. 1–7.

[42] J. Kang, Z. Xiong, D. Niyato, D. Ye, D. I. Kim, and J. Zhao, "Toward secure blockchain-enabled internet of vehicles: Optimizing consensus management using reputation and contract theory," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 3, pp. 2906–2920, 2019.

[43] J. Kang, Z. Xiong, D. Niyato, D. Ye, D. I. Kim, and J. Zhao, "Towards secure blockchain-enabled internet of vehicles: Optimizing consensus management using reputation and contract theory," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 3, pp. 2906–2920, 2019.

[44] P. Bolton and M. Dewatripont, *Contract theory*. MIT press, 2005.

[45] T. Liu, J. Li, F. Shu, H. Guan, Y. Wu, and Z. Han, "Incentive mechanism design for two-layer wireless edge caching networks using contract theory," *IEEE Transactions on Services Computing*, vol. 14, no. 5, pp. 1426–1438, 2018.

[46] A. Asheralieva and D. Niyato, "Combining contract theory and lyapunov optimization for content sharing with edge caching and device-to-device communications," *IEEE/ACM Transactions on Networking*, vol. 28, no. 3, pp. 1213–1226, 2020.

[47] X. Gao, D. Niyato, P. Wang, K. Yang, and J. An, "Contract design for time resource assignment and pricing in backscatter-assisted rf-powered networks," *IEEE Wireless Communications Letters*, vol. 9, no. 1, pp. 42–46, 2019.

[48] T. Liu, J. Li, F. Shu, M. Tao, W. Chen, and Z. Han, "Design of contract-based trading mechanism for a small-cell caching system," *IEEE Transactions on Wireless Communications*, vol. 16, no. 10, pp. 6602–6617, 2017.

[49] T. Liu, J. Li, F. Shu, H. Guan, S. Yan, and D. N. K. Jayakody, "On the incentive mechanisms for commercial edge caching in 5g wireless networks," *IEEE Wireless Communications*, vol. 25, no. 3, pp. 72–78, 2018.

[50] J. Kang, Z. Xiong, D. Niyato, S. Xie, and J. Zhang, "Incentive mechanism for reliable federated learning: A joint optimization approach to combining reputation and contract theory," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 10 700–10 714, 2019.

[51] J. Li, T. Liu, D. Niyato, P. Wang, J. Li, and Z. Han, "Contract-based approach for security deposit in blockchain networks with shards," in *2019 IEEE International Conference on Blockchain (Blockchain)*. IEEE, 2019, pp. 75–82.

[52] Z. Su, Y. Wang, Q. Xu, M. Fei, Y.-C. Tian, and N. Zhang, "A secure charging scheme for electric vehicles with smart communities in energy blockchain," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4601–4613, 2018.

[53] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[54] I. Industries, "What is industry 4.0?" 2020. [Online]. Available: https://www.ibm.com/industries/industrial/industry-4-0?p1=Search&p4=43700052366136168&p5=e&cm_mmc=Search_Google-_-1S_1S-_-WW_NA-_-industrial, [Accessed: Aug. 7, 2021].

[55] E. Industries, "Unlock the value of industry 4.0," 2020. [Online]. Available: https://www.ericsson.com/en/internet-of-things/industry4-0?gclid=CjwKCAjw7e_0BRB7EiwAlH-goLZiXr7G3osA_J5S8nIWUpveCfwo6iA7YBYz3surpMLgT_tUxsViUxoC52IQAvD_BwE&gclsrc=aw.ds, [Accessed: Jan. 5, 2021].

[56] S. Feng, W. Wang, D. Niyato, D. I. Kim, and P. Wang, "Competitive data trading in wireless-powered internet of things (iot) crowdsensing systems with blockchain," in *2018 IEEE International Conference on Communication Systems (ICCS)*. IEEE, 2018, pp. 289–394.

[57] A. Asheralieva and D. Niyato, "Distributed dynamic resource management and pricing in the iot systems with blockchain-as-a-service and uav-enabled mobile edge computing," *IEEE Internet of Things Journal*, vol. 7, no. 3, pp. 1974–1993, 2020.

[58] Z. Xiong, S. Feng, W. Wang, D. Niyato, P. Wang, and Z. Han, "Cloud/fog computing resource management and pricing for blockchain networks," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4585–4600, 2018.

[59] O. Novo, "Scalable access management in iot using blockchain: a performance evaluation," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4694–4701, 2018.

[60] C. Xu, K. Wang, P. Li, S. Guo, J. Luo, B. Ye, and M. Guo, "Making big data open in edges: A resource-efficient blockchain-based approach," *IEEE Transactions on Parallel and Distributed Systems*, vol. 30, no. 4, pp. 870–882, 2018.

[61] Y. Zhao, J. Zhao, L. Jiang, R. Tan, and D. Niyato, "Mobile edge computing, blockchain and reputation-based crowdsourcing iot federated learning: A secure, decentralized and privacy-preserving system," 2019. [Online]. Available: https://arxiv.org/abs/1906.10893, [Accessed: Sept. 18, 2021].

[62] K. Gai, Y. Wu, L. Zhu, Z. Zhang, and M. Qiu, "Differential privacy-based blockchain for industrial internet-of-things," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 4156–4165, 2019.

[63] S. Popov, "The tangle," 2018. [Online]. Available: https://assets.ctfassets.net/r1dr6vzfxhev/2t4uxvsIqk0EUau6g2sw0g/45eae33637ca92f85dd9f4a3a218e1ec/iota1_4_3.pdf, [Accessed: Aug. 6, 2019].

[64] I. authors, "Iot chain," 2019. [Online]. Available: https://iotchain.io/static/wp_full_en.pdf, [Accessed: Dec. 4, 2020].

[65] Atonomi, "Atonomi: Bringing trust and security to iot," 2020. [Online]. Available: https://atonomi.io/, [Accessed: Jan. 20, 2021].

[66] C. Lab, "Chain of things," 2020. [Online]. Available: https://www.chainofthings.com/, [Accessed: Sept. 7 2021].

[67] M. Ball, A. Rosen, M. Sabin, and P. N. Vasudevan, "Proofs of useful work," 2018. [Online]. Available: https://eprint.iacr.org/2018/559, [Accessed: Sept. 3, 2019].

[68] F. Zhang, I. Eyal, R. Escriva, A. Juels, and R. V. Renesse, "REM: Resource-efficient mining for blockchains," in *26th USENIX Security Symposium (USENIX Security 17)*. USENIX Association, 2017, pp. 1427–1444.

[69] QuantumMechanic, "Proof of stake instead of proof of work," 2011. [Online]. Available: https://bitcointalk.org/index.php?topic=27787.0, [Accessed: Sept. 23, 2021].

[70] S. King and S. Nadal, "Ppcoin: Peer-to-peer crypto-currency with proof-of-stake," 2012. [Online]. Available: https://decred.org/research/king2012.pdf, [Accessed: June 8, 2021].

[71] I. Bentov, A. Gabizon, and A. Mizrahi, "Cryptocurrencies without proof of work," in *International Conference on Financial Cryptography and Data Security*. Springer, 2016, pp. 142–157.

[72] I. Bentov, R. Pass, and E. Shi, "Snow white: Provably secure proofs of stake," *IACR Cryptology ePrint Archive*, vol. 2016, p. 919, 2016.

[73] A. Kiayias, A. Russell, B. David, and R. Oliynykov, "Ouroboros: A provably secure proof-of-stake blockchain protocol," in *Annual International Cryptology Conference*. Springer, 2017, pp. 357–388.

[74] B. David, P. Gaži, A. Kiayias, and A. Russell, "Ouroboros praos: An adaptively-secure, semi-synchronous proof-of-stake blockchain," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, 2018, pp. 66–98.

[75] Blockchain.com, "Transaction rate per second," 2021. [Online]. Available: https: https://www.blockchain.com/charts/transactions-per-second, [Accessed: April 4, 2021].

[76] J. Ray, "Sharding introduction r&d compendium," 2019. [Online]. Available: https://eth.wiki/en/sharding/sharding-introduction-r-d-compendium, [Accessed: Oct. 17, 2018].

[77] V. Inc., "Visa inc. to announce fiscal second quarter 2019 financial results on april 24, 2019," 2019. [Online]. Available: https://usa.visa.com/about-visa/newsroom/press-releases.releaseId.16301.html, [Accessed: May 12, 2020].

[78] K. Croman, C. Decker, I. Eyal, A. E. Gencer, A. Juels, A. Kosba, A. Miller, P. Saxena, E. Shi, E. G. Sirer, D. Song, and R. Wattenhofer, "On scaling decentralized blockchains," in *International Conference on Financial Cryptography and Data Security*. Springer, 2016, pp. 106–125.

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

[80] E. Kokoris-Kogias, P. Jovanovic, L. Gasser, N. Gailly, E. Syta, and B. Ford, "Omniledger: A secure, scale-out, decentralized ledger via sharding," in *2018 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2018, pp. 583–598.

[81] V. Zamfir, "A cbc casper tutorial," 2018. [Online]. Available: https://vitalik.ca/general/2018/12/05/cbc_casper.html, [Accessed: Dec. 5, 2019].

[82] E. Foundation, "Ethereum 2.0 phase 0 – the beacon chain," 2021. [Online]. Available: https://github.com/ethereum/eth2.0-specs/blob/dev/specs/phase0/beacon-chain.md, [Accessed: Dec. 4, 2021].

[83] Y. Zhang, N. H. Tran, D. Niyato, and Z. Han, "Multi-dimensional payment plan in fog computing with moral hazard," in *2016 IEEE International Conference on Communication Systems (ICCS)*. IEEE, 2016, pp. 1–6.

[84] Y. Zhang, L. Song, M. Pan, Z. Dawy, and Z. Han, "Non-cash auction for spectrum trading in cognitive radio networks: Contract theoretical model with joint adverse selection and moral hazard," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 3, pp. 643–653, 2017.

[85] Y. Zhang, C. Jiang, L. Song, M. Pan, Z. Dawy, and Z. Han, "Incentive mechanism for mobile crowdsourcing using an optimized tournament model," *IEEE journal on selected areas in communications*, vol. 35, no. 4, pp. 880–892, 2017.

[86] E. Foundation, "Ethereum 2.0 phase 0 – deposit contract," 2021. [Online]. Available: https://github.com/ethereum/eth2.0-specs/blob/dev/specs/phase0/deposit-contract.md, [Accessed: Dec. 31, 2021].

[87] V. Buterin, "Highlighting a problem: stability of the equilibrium of minimum timestamp enforcement," 2018. [Online]. Available: https://ethresear.ch/t/highlighting-a-problem-stability-of-the-equilibrium-of-minimum-timestamp-enforcement/2257, [Accessed: Aug. 8, 2019].

[88] E. Foundation, "Ethereum 2.0 (eth2)," 2020. [Online]. Available: https://ethereum.org/en/eth2/, [Accessed: Jan. 4 2021].

[89] S. Athey, "Single crossing properties and the existence of pure strategy equilibria in games of incomplete information," *Econometrica*, vol. 69, no. 4, pp. 861–889, 2001.

[90] E. Foundation, "Get involved in eth2," 2020. [Online]. Available: https://ethereum.org/en/eth2/get-involved, [Accessed: April 21, 2021].

[91] Q. T. Zhong and Z. Cole, "Analyzing the effects of network latency on blockchain performance and security using the whiteblock testing platform," 2019. [Online].

Available: https://whiteblock.io/wp-content/uploads/2019/07/analyzing-effects-network.pdf, [Accessed: Aug. 8, 2021].

[92] W. Bi, H. Yang, and M. Zheng, "An accelerated method for message propagation in blockchain networks," 2018. [Online]. Available: https://arXivpreprintarXiv:1809.00455, [Accessed: Dec. 12, 2021].

[93] J.-J. Laffont and J. Tirole, "Using cost observation to regulate firms," *Journal of political Economy*, vol. 94, no. 3, Part 1, pp. 614–641, 1986.

[94] C. Beekhuizen, "Validated, staking on eth2: #3 - sharding consensus," 2020. [Online]. Available: https://blog.ethereum.org/2020/03/27/sharding-consensus/, [Accessed: Jan. 9, 2021].

[95] M. M. Khalili, P. Naghizadeh, and M. Liu, "Designing cyber insurance policies: The role of pre-screening and security interdependence," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 9, pp. 2226–2239, 2018.

[96] S. Nakamoto, "Bitcoin," 2009. [Online]. Available: https://bitcoin.org/en/, [Accessed: Aug. 5, 2019].

[97] CoinMarketCap, "Today's cryptocurrency prices by market cap," 2020. [Online]. Available: https://coinmarketcap.com/, [Accessed: Oct. 22, 2020].

[98] V. Buterin, "Proof of stake faq," 2019. [Online]. Available: https://github.com/ethereum/wiki/wiki/Proof-of-Stake-FAQ#what-are-the-benefits-of-proof-of-stake-as-opposed-to-proof-of-work, [Accessed: Aug. 20, 2020].

[99] V. Buterin, "Discouragement attacks," 2018. [Online]. Available: https://github.com/ethereum/research/blob/master/papers/discouragement/discouragement.pdf, [Accessed: Dec. 16, 2018].

[100] D. Lancashire, "On discouragement attacks," 2018. [Online]. Available: https://org.saito.tech/on-discouragement-attacks/, [Accessed: Jan. 20, 2019].

[101] V. Buterin, "Rate-limiting entry/exits, not withdrawals," 2019. [Online]. Available: https://ethresear.ch/t/rate-limiting-entry-exits-not-withdrawals/4942, [Accessed: Aug. 15, 2021].

[102] C. Beekhuizen, "Validated: Staking on eth2 #0," 2019. [Online]. Available: https://blog.ethereum.org/2019/11/27/Validated-Staking-on-eth2-0/, [Accessed: Aug. 20, 2020].

[103] I. Bentov, C. Lee, A. Mizrahi, and M. Rosenfeld, "Proof of activity: Extending bitcoin's proof of work via proof of stake [extended abstract] y," *ACM SIGMETRICS Performance Evaluation Review*, vol. 42, no. 3, pp. 34–37, 2014.

[104] P4Titan, "Slimcoin: A peer-to-peer crypto-currency with proof-of-burn," 2014. [Online]. Available: https://github.com/slimcoin-project/slimcoin-project.github.io/raw/master/whitepaperSLM.pdf, [Accessed: May 11, 2019].

[105] M. Milutinovic, W. He, H. Wu, and M. Kanwal, "Proof of luck: an efficient blockchain consensus protocol," in *proceedings of the 1st Workshop on System Software for Trusted Execution*. ACM, 2016, p. 2.

[106] Hyperledger, "Hyperledger fabric," 2020. [Online]. Available: https://www.hyperledger.org/projects/fabric, [Accessed: Jan. 20, 2021].

[107] IBM, "Ibm blockchain services," 2020. [Online]. Available: https://www.ibm.com/blockchain/services, [Accessed: Jan. 20, 2021].

[108] IoTex, "Internet of trusted things," 2020. [Online]. Available: https://www.iotex.io/, [Accessed: Jan. 20, 2021].

[109] K.-K. R. Choo, Z. Yan, and W. Meng, "Blockchain in industrial iot applications: Security and privacy advances, challenges and opportunities," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 4119–4121, 2020.

[110] Z. Guan, G. Si, X. Zhang, L. Wu, N. Guizani, X. Du, and Y. Ma, "Privacy-preserving and efficient aggregation based on blockchain for power grid communications in smart communities," *IEEE Communications Magazine*, vol. 56, no. 7, pp. 82–88, 2018.

[111] S. Pal, T. Rabehaja, M. Hitchens, V. Varadharajan, and A. Hill, "On the design of a flexible delegation model for the internet of things using blockchain," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 5, pp. 3521–3530, 2019.

[112] W. Liang, Y. Fan, K.-C. Li, D. Zhang, and J.-L. Gaudiot, "Secure data storage and recovery in industrial blockchain network environments," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 10, pp. 6543–6552, 2020.

[113] H. Yang, Y. Liang, J. Yuan, Q. Yao, A. Yu, and J. Zhang, "Distributed blockchain-based trusted multidomain collaboration for mobile edge computing in 5g and beyond," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 11, pp. 7094–7104, 2020.

[114] M. Keshk, B. Turnbull, N. Moustafa, D. Vatsalan, and K.-K. R. Choo, "A privacy-preserving-framework-based blockchain and deep learning for protecting smart power networks," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 8, pp. 5110–5118, 2019.

[115] E. Foundation, "Ethereum 2.0 (serenity) phases," 2020. [Online]. Available: https://docs.ethhub.io/ethereum-roadmap/ethereum-2.0/eth-2.0-phases/, [Accessed: Oct. 15, 2021].

[116] PWC, "Insurance 2020 & beyond: Reaping the dividends of cyber resilience," 2019. [Online]. Available: https://www.pwc.com/gx/en/industries/financial-services/publications/insurance-2020-cyber.html, [Accessed: Jan. 20, 2019].

[117] M. M. Khalili, P. Naghizadeh, and M. Liu, "Embracing risk dependency in designing cyber-insurance contracts," in *55th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. IEEE, 2017, pp. 926–933.

[118] M. M. Khalili, P. Naghizadeh, and M. Liu, "Designing cyber insurance policies: Mitigating moral hazard through security pre-screening," in *International Conference on Game Theory for Networks*. Springer, 2017, pp. 63–73.

[119] M. M. Khalili, M. Liu, and S. Romanosky, "Embracing and controlling risk dependency in cyber-insurance policy underwriting," *Journal of Cybersecurity*, vol. 5, no. 1, p. tyz010, 2019.

[120] X. Lu, D. Niyato, H. Jiang, P. Wang, and H. V. Poor, "Cyber insurance for heterogeneous wireless networks," *IEEE Communications Magazine*, vol. 56, no. 6, pp. 21–27, 2018.

[121] Z. Xiong, J. Zhao, D. Niyato, P. Wang, and Y. Zhang, "Design of contract-based sponsorship scheme in stackelberg game for sponsored content market," in *2019 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2019, pp. 1–6.

[122] J. Li, D. Niyato, C. S. Hong, K.-J. Park, L. Wang, and Z. Han, "A contract-theoretic cyber insurance for withdraw delay in the blockchain networks with shards," in *IEEE International Conference on Communications (ICC)*. IEEE, 2020, pp. 1–7.

[123] V. Buterin, "The censorship problem," 2015. [Online]. Available: https://blog.ethereum.org/2015/06/06/the-problem-of-censorship/, [Accessed: Feb. 17, 2021].

[124] hackingresear.ch, "Discouragement attacks," 2020. [Online]. Available: https://hackingresear.ch/discouragement-attacks/, [Accessed: Jan. 20, 2021].

[125] V. Buterin, "A griefing factor analysis model," 2018. [Online]. Available: https://ethresear.ch/t/a-griefing-factor-analysis-model/2338, [Accessed: Jun. 29, 2018].

[126] Ericsson, "Unlock the value of industry 4.0," 2020. [Online]. Available: https://www.ericsson.com/en/internet-of-things/industry4-0?gclid=EAIaIQobChMIq424odGU6wIVxkXVCh2p-Q8tEAMYASAAEgLVePD_BwE&gclsrc=aw.ds, [Accessed: Feb. 17, 2021].

[127] McKinsey, "What's new with the internet of things?" 2020. [Online]. Available: https://www.mckinsey.com/industries/semiconductors/our-insights/whats-new-with-the-internet-of-things, [Accessed: Feb. 17, 2021].

[128] Statista, "Number of internet of things (iot) connected devices worldwide in 2018, 2025 and 2030," 2019. [Online]. Available: https://www.statista.com/statistics/802690/worldwide-connected-devices-by-access-technology/, [Accessed: Feb. 17, 2021].

[129] Statista, "Data volume of internet of things (iot) connections worldwide in 2019 and 2025," 2020. [Online]. Available: https://www.statista.com/statistics/1017863/worldwide-iot-connected-devices-data-size/, [Accessed: Feb. 17, 2021].

[130] I. Intelligence, "How iot devices & smart home automation is entering our homes in 2020," 2020. [Online]. Available: https://www.businessinsider.com/iot-smart-home-automation, [Accessed: Feb. 17, 2021].

[131] X. Li, R. Lu, X. Liang, X. Shen, J. Chen, and X. Lin, "Smart community: an internet of things application," *IEEE Communications magazine*, vol. 49, no. 11, pp. 68–75, 2011.

[132] A. Meola, "How smart city technology & the internet of things will change our apartments, grids and communities," 2020. [Online]. Available: https://www.businessinsider.com/iot-smart-city-technology, [Accessed: Jan. 4, 2021].

[133] H. Goverment, "Smart city of the future," 2019. [Online]. Available: https://houstontx.gov/smartcity/, [Accessed: Jan. 15, 2021].

[134] A. Gerber, "Building connected cities with new and existing iot technologies," 2019. [Online]. Available: https://developer.ibm.com/technologies/iot/articles/iot-lp201-iot-connected-cities/, [Accessed: Jan. 15, 2020].

[135] M. Conoscenti, A. Vetro, and J. C. De Martin, "Blockchain for the internet of things: A systematic literature review," in *2016 IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA)*. IEEE, 2016, pp. 1–6.

[136] A. Back, M. Corallo, L. Dashjr, M. Friedenbach, G. Maxwell, A. Miller, A. Poelstra, J. Timón, and P. Wuille, "Enabling blockchain innovations with pegged sidechains," 2014. [Online]. Available: http://kevinriggen.com/files/sidechains.pdf, [Accessed: May 7, 2021].

[137] A. Singh, K. Click, R. M. Parizi, Q. Zhang, A. Dehghantanha, and K.-K. R. Choo, "Sidechain technologies in blockchain networks: An examination and state-of-the-art review," *Journal of Network and Computer Applications*, vol. 149, p. 102471, 2020.

[138] BiTA, "Blockchain in transport alliance," 2017. [Online]. Available: https://www.bita.studio/, [Accessed: Jan. 15, 2021].

[139] IBM, "Ibm food trust," 2019. [Online]. Available: https://www.ibm.com/products/food-trust/pricing, [Accessed: Jan. 15, 2021].

[140] IBM, "Ibm blockchain transparent supply," 2019. [Online]. Available: https://www.ibm.com/blockchain/solutions/transparent-supply, [Accessed: Jan. 15, 2021].

[141] H.-N. Dai, Z. Zheng, and Y. Zhang, "Blockchain for internet of things: A survey," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8076–8094, 2019.

[142] G. Konstantopoulos, "Million-user dapps on ethereum: An introduction to application-specific sidechains," 2017. [Online]. Available: https://medium.com/loom-network/million-user-dapps-on-ethereum-an-introduction-to-application-specific-sidechains-c0fdc288c5e5, [Accessed: Dec. 12, 2021].

[143] M. Harishankar, S. V. Iyer, A. Laszka, C. Joe-Wong, and P. Tague, "Payplace: A scalable sidechain protocol for flexible payment mechanisms in blockchain-based marketplaces," 2020. [Online]. Available: https://arxiv.org/abs/2003.06197, [Accessed: Aug. 15, 2021].

[144] M. Li, H. Tang, A. R. Hussein, and X. Wang, "A sidechain-based decentralized authentication scheme via optimized two-way peg protocol for smart community," *IEEE Open Journal of the Communications Society*, vol. 1, pp. 282–292, 2020.

[145] Y. Wei, J. Yu, T.-M. Lok, and L. Gao, "A novel mobile data contract design with time flexibility," *IEEE Transactions on Mobile Computing*, vol. 18, no. 5, pp. 986–999, 2018.

[146] Liquid, "How do transaction fees on liquid work?" 2020. [Online]. Available: https://help.blockstream.com/hc/en-us/articles/900001386846-How-do-transaction-fees-on-Liquid-work-, [Accessed: Oct. 15, 2021].

[147] E. Maskin and J. Riley, "Optimal auctions with risk averse buyers," *Econometrica: Journal of the Econometric Society*, pp. 1473–1518, 1984.

[148] J. P. Klein and M. L. Moeschberger, *Survival analysis: techniques for censored and truncated data*. Springer, 2006.

[149] O. Morgenstern and J. Von Neumann, *Theory of games and economic behavior*. Princeton university press, 1953.

[150] A. Cretarola and G. Figà-Talamanca, "Modeling bitcoin price and bubbles," in *Blockchain and Cryptocurrencies*. IntechOpen, 2018, pp. 3–17.

[151] B. Oksendal, *Stochastic differential equations: an introduction with applications*. Springer Science & Business Media, 2013.

[152] R. C. Merton, "Optimum consumption and portfolio rules in a continuous-time model," in *Stochastic Optimization Models in Finance*. Elsevier, 1975, pp. 621–661.

[153] K. Itô, "109. stochastic integral," *Proceedings of the Imperial Academy*, vol. 20, no. 8, pp. 519–524, 1944.

[154] E. Maskin and J. Riley, "Optimal multi-unit auctions," *International library of critical writings in economics*, vol. 113, pp. 5–29, 2000.

[155] YCharts, "Ethereum average transaction fee," 2020. [Online]. Available: https://ycharts.com/indicators/ethereum_average_transaction_fee, [Accessed: Oct. 15, 2021].