IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 15, 2019, accepted September 3, 2019, date of publication September 5, 2019, date of current version September 20, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2939778

Two-Phase Cooperative Bargaining Game Approach for Shard-Based Blockchain Consensus Scheme

SUNGWOOK KIM
Department of Computer Science, Sogang University, Seoul 121-742, South Korea
e-mail: swkim01@sogang.ac.kr

This work was supported in part by the Ministry of Science (MSIT) and ICT, South Korea, under the Information Technology Research Center (ITRC) support program, under Grant IITP-2019-2018-0-01799, supervised by the Institute for Information and Communications Technology Planning and Evaluation (IITP), and in part by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education under Grant NRF-2018R1D1A1A09081759.

ABSTRACT In the last few years, blockchain technologies have come to the forefront of the research and industrial communities as they bring potential benefits for many industries. Even though the blockchain is a safe, reliable, and innovative mechanism, current blockchain solutions have fatal drawbacks of non-supervision and huge computational overhead. Therefore, they cannot be directly applied for real world operations. To resolve this problem, a novel model is presented in this study where the key idea is to split the transactions among multiple shards while processing them in parallel. To achieve the maximum system efficiency through the shard mechanism, we focus on the cooperative game theory. Based on the egalitarian bargaining solution, total transactions per each time period are divided for each shard. According to the proportional bargaining solution, assigned transactions in each individual shard are validated by blockchain nodes in a distributed manner. The main advantage of our two-phase bargaining game model is to provide an axiom-based strategic solution for the shard-based consensus problem while dynamically responding to the current blockchain network conditions. The numerical simulation results show that the effectiveness and efficiency of our game based approach by comparing the existing state-of-the-art blockchain control schemes. In the conclusion, we present our conclusions and provide important future research directions based on the combination of blockchain with other technologies.

INDEX TERMS Blockchain, shard-based consensus mechanism, cooperative game theory, egalitarian bargaining solution, proportional bargaining solution.

I. INTRODUCTION

IN the past decade, with the popularity of digital cryptocurrencies, e.g., Bitcoin, blockchain technology has attracted tremendous attention from both academia and industry. Since then, blockchain has grown beyond cryptocurrencies to support many existing business and industrial processes. In addition, they enable the creation of new business models that impact many industrial sectors such as finance, healthcare, manufacturing, and logistics. A blockchain, also called distributed ledger, is essentially an append-only data structure maintained through a set of regular blockchain nodes (BNs). As the backbone of a public and distributed ledger system, the blockchain technology guarantees the tamper-proof ledger, transparent transactions, and trustless but secure trading in a decentralized network [1], [2].

In the original design, BNs in the blockchain agree on an ordered set of blocks, each containing multiple transactions. Therefore, the blockchain can be viewed as a log of ordered transactions. Usually, keeping records of transactions is a core function of all businesses. A block is the current part of a blockchain which records the recent information. Once a block gets completed, it goes into the blockchain as permanent database creating a new block. Blocks are linked to each other like a chain in proper, chronological order. In the database context, blockchain can be viewed as a solution to distributed transaction management; BNs keep replicas of the data and agree on an execution order of transactions. While traditional database and blockchain have some similarities, there is a major difference. Traditional

The associate editor coordinating the review of this manuscript and approving it for publication was Tawfik Al-Hadhrami.

database assumes a trusted environment and employs well known concurrency control techniques to order transactions. However, blockchain assumes that multiple BNs behave in arbitrary manner to store, spread and preserve the blockchain data. Therefore, it is necessary to offer a stronger security to avoid system failure [3].

The blockchain is completely decentralized without any supervision nor central management. Therefore, consensus protocol is one of the most important and revolutionary aspects of blockchain technology. As a term, consensus means that the BNs on the network agree on the same state of a blockchain, in a sense making it a self-auditing ecosystem. This is an absolutely crucial aspect of the technology, carrying out two key functions. First, consensus protocol allows a blockchain to be updated while ensuring that every block in the chain is true. Second, consensus protocol keeps BNs participating in the consensus process while providing incentives. The aim of consensus rules is to guarantee a single chain is used and followed. The key requirement to achieve a successful consensus is a unanimous acceptance among BNs on the network for a single data value, even in the event of some of BNs failing or being unreliable in any way [4].

To ensure data consistency, speed of consensus finality, robustness to arbitrarily behaving BNs and network scalability, blockchain consensus protocols should offer the agreement on the global blockchain data state among a large number of trustless BNs with no identity authentication and low messaging overhead. Most of the existing blockchain consensus protocols attempt to satisfy this goal at the cost of limited processing throughput. However, a number of existing protocols incur the huge consumption of computing power to achieve decentralized consensus among poorly synchronized, trustless BNs. In addition, these existing consensus protocols may impose high latency for transaction confirmation to guarantee a consensus finality. Usually, the performance of blockchain networks significantly relies on the performance of adopted consensus algorithm. Therefore, a lot of attention has been attracted to design an efficient consensus algorithm for practical blockchain operations. However, in spite of tremendous interests from the research communities, one significant shortcoming of blockchain consensus protocols is their low transaction throughput and poor scalability [5], [6].

Recently, there have been significant efforts to improve the transaction throughput and scalability of blockchain consensus protocols. One key outcome of this line of research is sharding solution. Simply stated, sharding solution is a way of partitioning; it fairly and randomly divides the blockchain network into smaller committees, called a shard, and each of which processes a disjoint set of transactions in parallel with other shards. Each BN is not responsible for processing the entire transactional load, but only maintains information related to its corresponding shard. As each shard is reasonably small, we can spread out the computational and storage workload across the blockchain network while increasing the overall transaction throughput of the system. Therefore, the sharding solution is a key outcome to achieve the scale-out throughput of blockchain system [6].

Although the idea of sharding solution is promising, most existing sharding mechanisms fail to clarify i) how to fairly distribute the entire transactional load among multiple shards, and ii) how BNs will be incentivized to honestly participate and discharge their shard duties. For the fair distribution of entire transactions, a prior planning is necessary. If the concept of fairness is not considered explicitly at the design stage of entire transaction load division algorithm, it can result in very unfair working load distribution among shards. In addition, to participate in sharding tasks, i.e., shard generation and transaction validation, it imposes a cost on BNs. Therefore, rational and selfish BNs may not actively participate in sharding tasks. In summary, one key research gap in the sharding solution is a lack of understanding of the strategic behavior of rational BNs and an intelligent load distribution strategy [6]. Such an understanding is critical for designing an appropriate shard-based blockchain control algorithm, which will foster the cooperation inter shards and the cooperation among BNs within the same shard. Our goal in this paper is to address this research gap. However, achieving this goal is difficult in the distributed blockchain infrastructure whereas shards and individual BNs are free to act in a selfish manner. Therefore, we need a new control paradigm to address this problem.

Game theory is the discipline that studies how game players make strategic decisions. It was initially developed in economics to describe and analyze interactive decisions. Specifically, a game model is the mathematical formalization of conflict situations, and it is used to predict the outcome of complex interaction among rational players. Classically, there are two branches of game theory; non-cooperative and cooperative games, which differ in how they formalize interdependence among the players. In non-cooperative game theory, a game is a detailed model of all the moves available to the players. In contrast, cooperative game theory abstracts away from this level of detail, and describes only the outcomes that result when the players come together in different combinations. Usually, cooperative game centers its interest on particular sets of strategies known as solution concepts based on what is required by norms of ideal rationality [7]. Therefore, it can make sense for players to work together to mutual advantage. In this study, we focus on cooperative game models to develop an efficient shard-based blockchain consensus scheme.

A. MOTIVATION AND CONTRIBUTION

The aim of this study is to propose a novel shard-based blockchain consensus scheme based on the cooperative game paradigm. To tackle the transaction distribution problem, we adopt the basic concept of *Egalitarian Bargaining Solution* (*EBS*). To foster participation in the consensus process, we use the key idea of *Proportional Bargaining Solution* (*PBS*). To investigate the hierarchical interactions between shards and BNs, we formulate a new two-phase bargaining game model, which is designed to strategically control the shard-based consensus process. Based on the cooperative game solution guideline, shards and BNs work together cooperatively to guarantee fairness and efficiency while negotiating their conflict interests. The contributions of this study can be briefly summarized as follows:

- **Cooperative game approach:** Depending on the coordinated manner, multiple shards and individual BNs act cooperatively and collaborate with each other to strike an appropriate system performance among conflicting interests. Therefore, control decisions are made intelligently and rationally in a cooperative manner.
- **The synergy of two bargaining solutions:** we explore the sequential interaction of two bargaining solutions, and jointly design an integrated two-phase cooperative game model. During the hierarchical interaction, the synergy effect lies in its responsiveness to the reciprocal combination of different bargaining solutions.
- **Dynamic interactive implementation:** we implement a novel shard-based consensus scheme to address the transaction distribution and consensus process. By using the self-adaptability, the proposed game is repeated based on the step-by-step interactive process; it is generic and applicable to iteratively adapt the dynamic blockchain network situations.
- **Axiom based fair-efficient solution:** based on the feasible combination of optimality and practicality, we can achieve the axiom based fair-efficient solutions. According to the reciprocal-negotiation, our solution can provide mutual advantages including adaptability, flexibility, and responsiveness to the current blockchain network conditions.
- **Performance analysis:** we validate our two-phase bargaining game approach by using the simulation analysis. Numerical results can confirm the we can get a desirable solution compared to the existing protocols under the different transaction workload intensities.

B. ORGANIZATION

The roadmap for this paper is as follows. In Section II, we discuss the state-of-the-art protocols for the shard-based blockchain system. Section III provides an overview on the protocol organization of blockchain networks, and explains the necessary preliminaries of *EBS* and *EBS*. And then, we design the two-phase cooperative game model to develop our shard-based blockchain scheme aiming at the fair-efficient system performance. We also provide the main steps of the proposed scheme to enhance readability. Consequently, Section IV shows the numerical simulation results while providing comparison and discussion with existing shard-based blockchain protocols. We conclude this study in Section V as well as an outlook of the potential future research directions in the context of blockchain networks.

II. RELATED WORK

The most related state-of-the-art researches to sharding protocols are summarized in this section. Briefly, we outline the efforts in the literature towards improving the scalability and transaction rate of consensus protocols in blockchain networks. The *Game Theoretic Shard-based Blockchain* (*GTSB*) scheme [6] comprehensively investigates the problem of selfishness in shard-based blockchains. To design a shard-based protocol, this approach evaluates the strategic behavior of BNs by employing concepts from game theory. In particular, the interactions among BNs are analyzed using a static n-player non-cooperative game. The BN strategies in such a game and resulting payoffs are systematically quantified, and then, the Nash equilibria strategy profile is obtained under different reward sharing scenarios; it is impossible to enforce a cooperative Nash equilibrium in this setting unless certain improbable conditions are met. In addition, the *GTSB* scheme provides an incentive mechanism for shard-based blockchain protocols, which would enforce cooperation among BNs by guaranteeing optimal incentive distribution. By considering the fair sharing of rewards, BNs receive benefits only if they have cooperated within their shards. This work is the first step towards a deeper understanding of the effects of non-cooperative behavior in shard-based blockchain protocols [6].

The *Normalized Autonomous Transaction Settlement* (*NATS*) scheme provides an autonomous and lightweight transaction management platform, called NormaChain, for Internet of Things (IoT)-based E-commerce [8]. To achieve high transaction speed and scalability, this scheme replaces the conventional single layer blockchain to a three-layer sharding blockchain network, with each layer assigned to different responsibilities. By designing a special three-layer shard-based blockchain network, the *NATS* scheme can significantly increase transaction efficiency and system scalability. In addition, this scheme adopts a practical byzantine fault tolerance consensus algorithm in replacement of proof-of-work to minimize the overall mining liability of each process. This approach eliminates the dependence of a trusted central authority, and instead expands it to a fully decentralized governance, which distributes the supervision power equally among all parties. More importantly, NormaChain is secure against malicious adversaries [8].

The *Secure and Scale-out based Decentralized Sharding* (*SSDS*) scheme introduces the OmniLedger, which is the first distributed ledger architecture that provides scale-out transaction processing capacity [9]. The OmniLedger i) chooses representative groups of validators statistically and periodically, ii) ensures a negligible probability that any shard is compromised across the long-term system lifetime via periodically forming shards, and iii) correctly and atomically handles cross-shard transactions, or transactions that affect the ledger state held by two or more distinct shards. As a novel scale-out distributed ledger, the OmniLedger enables any user to transact safely with any other users regardless of the shard. With the introduction of a trust-but-verify approach, users effectively process cross-shard transactions as well as real-time validation. Therefore, the *SSDS* scheme enables validators to securely and efficiently switch between shards without stalling between reconfiguration events.

Simulation results show that the *SSDS* scheme's throughput scales linearly in the number of active validators [9].

In summary, the *GTSB* scheme [6], the *NATS* scheme [8] and the *SSDS* scheme [9] have introduced unique challenges to efficiently control the shard-based blockchain protocols. Therefore, they have attracted a lot of attentions, recently. However, none of existing studies explore the cooperative game approach. In this paper, we consider a new two-phase cooperative game model to handle the sharding problem in blockchain networks. To the best of our knowledge, our cooperative game approach is the first to investigate the selfish behavior of BNs, and its effect, in shard-based blockchain system. Compared to these existing schemes in [6], [8], [9], we demonstrate the superiority of our two-phase game model through the simulation analysis.

III. THE PROPOSED SHARD-BASED BLOCKCHAIN CONTROL SCHEME

In this section, we demonstrate the problem formulation by including the system overview of shard-based blockchain networks, cooperative bargaining solutions, and designing goals. And then, we develop our shard-based blockchain scheme based on the two-phase game model. Finally, we describe concretely the proposed scheme in the eight-step procedures.

A. THE SHARD-BASED BLOCKCHAIN SYSTEM INFRASTRUCTURE

A blockchain is an append-only, immutable distributed database that records a time-sequenced history of transactions. Transactions are typically grouped into blocks, and each block cryptographically linked to the previous one, forming a chain. The blockchain protocol enables the construction and maintenance of consistent copies of blocks in a distributed fashion [6]. In this study, we consider a simple blockchain network infrastructure, which consists of multiple BNs $\mathbb{N} = \{BN_1, BN_2, \ldots, BN_m\}$; they may participate in the blockchain system, and have different computation capabilities. Further, we assume that BNs are honest and rational, but selfish. Usually, the blockchain operation is implemented in the time-driven approach, and the time is divided into a fixed-sized period, called epochs. At the end of each epoch, the blockchain accepts transactions in blocks and commits a new block of transactions. The transaction set $\mathcal{T}^t$ is defined as the total transactions generated at the t^{th} time epoch [6].

The total transaction processing capacity of blockchain system does not increase with added BNs. In fact, it gradually decreases due to increased consensus overheads. To solve the consensus overhead problem, the effective approach is to build scale-out multiple databases, whose capacity scales horizontally with the number of participating BNs. A subset of BNs, which is a shard, can handle a part of $\mathcal{T}$ in a parallel manner. Finally, a special shard, called adjusting shard, combines all parts of $\mathcal{T}$ with confirmation. In our blockchain infrastructure, we assume that there are $k + 1$ shards, i.e., $\mathbb{S} = \{\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_{k+1}\}$ where $\mathcal{S}_{1\leq i\leq k}$ is a normal shard and $\mathcal{S}_{k+1}$ is an adjusting shard. The number of shards $(k + 1)$ is a variable quantity and can grow linearly with the size of the blockchain network. According to the shard-based consensus mechanism, $\mathcal{T}$ is divided into individual subblocks $\mathbb{B} = \{B_1, B_2, \ldots, B_k\}$; $\mathcal{B}$ is composed of k disjoint sets of transactions where $\mathcal{T} = \bigcup_{1\leq i\leq k} \{B_i\}$. Each disjoint set $B_{1\leq i\leq k}$ is assigned to a corresponding shard according to the one-to-one mapping : $B_{1\leq i\leq k} \rightarrow \mathcal{S}_{1\leq i\leq k}$ [6].

Sharding is a distributed consensus protocol executed among a set of shards. At the end of each epoch, multiple shards are randomly generated; each shard $\mathcal{S}_{1\leq i\leq k}$ validates and agrees on the separate subblock $B_{1\leq i\leq k}$ by BNs in the $\mathcal{S}_i$. The adjusting shard, i.e., $\mathcal{S}_{k+1}$, is a special consensus shard, which is responsible for combining the all $B_{1\leq i\leq k}$ collected from the $\mathcal{S}_{1\leq i\leq k}$. Therefore, the $\mathcal{S}_{k+1}$ takes the consented subblocks, merges them to form a final full block $\mathbb{B}$, and broadcasts it to the rest of the blockchain network. In each epoch, the sharding protocol proceeds in the following steps in consecutive order [6], [10];

i) shard formation: unbiased random numbers in the range $[1..k + 1]$ are generated to seed the BN-to-shard assignment. Given by the number, BNs form their shards, which are approximately equally-sized BNs' groups. And then, BNs in each individual shard identify other BNs to fully connect with each other.
ii) shard-based consensus: BNs in the $\mathcal{S}_i$ run a standard byzantine agreement protocol to agree on the B_i. Each $\mathcal{S}_{1\leq i\leq k}$ sends its consented subblock $B_{1\leq i\leq k}$ to the adjusting shard $\mathcal{S}_{k+1}$.
iii) final block shaping: the adjusting shard $\mathcal{S}_{k+1}$ create the full block $\mathbb{B}$ by uniting the collected $B_{1\leq i\leq k}$, and agrees on the final result. And then, the $\mathcal{S}_{k+1}$ broadcasts the $\mathbb{B}$ to the blockchain network.
iv) process repetition: for the next epoch's consensus, proceed to the step I.

In each time epoch, the consensus processing cost $(\mathcal{C}_{BN})$ of each BN can be characterized by considering the sharding steps. In this study, the $\mathcal{C}_{BN}$ is abstractly quantified to analyze the strategic behavior of BNs. In principle, the $\mathcal{C}_{BN}$ consists of two parts; i) forming a shard, and ii) shard-based consensus. During the shard-forming phase, each BN identifies other BNs in its corresponding shard. Therefore, the shard-forming expense is generally dependent on the number $(\mathcal{Q}_{\mathcal{S}})$ of BNs in the $\mathcal{S}$. In the shard-based consensus phase, each BN validates its respective subblock B and will reach an agreement with other BN members. It is clear that the consensus cost is strongly related to the size of B and $\mathcal{Q}_{\mathcal{S}}$. In addition, all BNs pay a mandatory cost $(M_{\mathcal{C}})$ for the shard-formation according to randomness generation. Finally, in the $\mathcal{S}_i$, we can characterize the BN_j's cost $\left(\mathcal{C}_{BN_j}^{\mathcal{S}_i}\right)$ to participate in the sharding protocol as follows;

$$\mathcal{C}_{BN_j}^{\mathcal{S}_i} = \left(M_{\mathcal{C}} + \left(\left(\mathcal{Q}_{\mathcal{S}_i} - 1\right) \times \mathcal{C}^I\right)\right) + \left(\left(\mathcal{C}^V \times \mathcal{G}\left(B_i\right)\right) + \left(\left(\mathcal{Q}_{\mathcal{S}_i} - 1\right) \times \mathcal{C}^C \times \mathcal{G}\left(B_i\right)\right)\right), \quad s.t., \ BN_j \in \mathcal{S}_i \tag{1}$$

where $\mathcal{C}^I$ and $\mathcal{C}^V$ are the charges for each BN's identification, and for each transaction's validation, respectively. $\mathcal{Q}_{\mathcal{S}_i}$ represents the number of BNs in the $\mathcal{S}_i$. $\mathcal{C}^C$ is the charge to reach a consensus for each transaction. $\mathcal{G}(B_i)$ is the function that outputs the number of transactions within the B_i. In the equation (1), the first part is related to the shard-forming, and the second part is associated with the consensus process.

In this study, the interactions among BNs to verify transactions and participate in the consensus process are formulated as a two-phase game model. First, we formally define the upper-game model $\mathbb{G}^{up} = \left\{\mathbb{P}, \mathcal{T}, \mathfrak{W}^{\mathcal{T}}, \mathbb{S}, B_{1\leq i\leq k}, \mathbb{U}^{\mathbb{S}}, T\right\}$ to solve the transaction distribution problem.

- $\mathbb{P} = \{\mathcal{S}_1, \ldots, \mathcal{S}_k\}$ is the finite set of upper-phase game players, and they are normal shards where $\mathbb{P} = \mathbb{S} \setminus \mathcal{S}_{k+1}$.
- $\mathcal{T}$ is the total transactions generated to be confirmed at the current epoch, and $\mathfrak{W}^{\mathcal{T}}$ is the total payment for the $\mathcal{T}$ verification in the shard-based process.
- $\mathbb{S} = \left\{\mathfrak{P}_{\mathcal{S}_1}, \ldots \mathfrak{P}_{\mathcal{S}_i} \ldots, \mathfrak{P}_{\mathcal{S}_k}\right\}$ is a set of strategies; $\mathfrak{P}_{\mathcal{S}_i}$ means the $\mathcal{T}$'s division ratio for $\mathcal{S}_i$ where $\sum_{\mathcal{S}_i \in \mathbb{P}} \mathfrak{P}_{\mathcal{S}_i} = 1$.
- $B_{1\leq i\leq k}$ is the subblock, which is assigned to the $\mathcal{S}_i$; the size of B_i is $\mathcal{T} \times \mathfrak{P}_{\mathcal{S}_i}$.
- $\mathbb{U}^{\mathbb{S}}$ is the payoff received by the $\mathbb{S}$.
- T denotes time, which is represented by a sequence of epochs for the transaction distribution process in the sharding protocol.

Second, we formally define the lower-game model $\mathbb{G}^{low}_{\mathcal{S}_{1\leq i\leq k}} = \left\{\mathbb{P}^{\mathcal{S}_i}, B_i, \mathfrak{W}^{B_i}, \boldsymbol{S}, \mathbb{U}^{BN_j}_{\mathcal{S}_i}, T\right\}$ for the shard-based consensus mechanism within the $\mathcal{S}_i$. With multiple shards $\mathcal{S}_{1\leq i\leq k}$, total k $\mathbb{G}^{low}_{\mathcal{S}_i}$ games are operated independently and dispersively in a parallel manner.

- $\mathbb{P}^{\mathcal{S}_i} = \{\ldots BN_l \ldots\}$ is the finite set of game players in the $\mathbb{G}^{low}_{\mathcal{S}_i}$ where $BN_l \in \mathbb{N}$ represents a BN member in the shard $\mathcal{S}_i$.
- B_i is a subblock, which is assigned to the $\mathcal{S}_i$ at the $\mathbb{G}^{up}$ game process.
- $\mathfrak{W}^{B_i}$ is the payment for the B_i verification process where $\mathfrak{W}^{B_i} = \mathfrak{W}^{\mathcal{T}} \times \mathfrak{P}_{\mathcal{S}_i}$.
- $\boldsymbol{S} = \left\{\ldots \mathcal{R}^{\mathcal{S}_i}_{BN_l}(B_i) \ldots\right\}$ is a set of strategies; $\mathcal{R}^{\mathcal{S}_i}_{BN_l}(B_i)$ represents the given reward for the BN_l where $\sum_{BN_l \in \mathbb{P}^{\mathcal{S}_i}} \mathcal{R}^{\mathcal{S}_i}_{BN_l}(B_i) = \mathfrak{W}^{B_i}$.
- $\mathbb{U}^{BN_j}_{\mathcal{S}_i}$ is the payoff received by the BN_j at the $\mathbb{G}^{low}_{\mathcal{S}_i}$ game process.
- T denotes time, which is represented by a sequence of epochs for the shard-based consensus process.

B. THE MAIN CONCEPT OF BARGAINING SOLUTIONS

To characterize the basic concepts of bargaining solutions, we preliminarily define some mathematical expressions. $\mathbb{N}$ will denote the set of positive integers and $\mathbb{R}\,(\mathbb{R}_+, \mathbb{R}_{++})$ denote the set of all (non-negative, positive) real numbers and $\mathbb{R}^n\left(\mathbb{R}^n_+, \mathbb{R}^n_{++}\right)$ be the n-fold Cartesian product of $\mathbb{R}\,(\mathbb{R}_+, \mathbb{R}_{++})$. Vector inequalities in $\mathbb{R}^n$ are denoted by $\geq, >, \gg$. For $x, y \in \mathbb{R}^n$, we write $x \geq y$ if $x_i \geq y_i$ for all i, $x > y$ if $x \geq y$ and $x \neq y$, and $x \gg y$ if $x_i > y_i$ for all i. By $\mathbb{R}^n$, $n \in \mathbb{N}$, we denote the n-dimensional Euclidean space, and by $\mathbb{R}^n_+$ and $\mathbb{R}^n_{++}$, we denote the set of nonnegative and strictly positive vectors in $\mathbb{R}^n$, respectively, i.e., $\mathbb{R}^n_+ = \{x \in \mathbb{R}^n \,|\, x \geq 0\}$ and $\mathbb{R}^n_{++} = \{x \in \mathbb{R}^n \,|\, x \gg 0\}$. Let $\|\cdot\|$ be the Euclidian norm in $\mathbb{R}^n$. The set $\mathbb{N} = \{1, \ldots, n\}$, $n \in \mathbb{N}$, will denote the player set. A bargaining problem is characterized by a set S of feasible utility allocations, measured in von Neumann-Morgenstern scales, and a point d, called *threat point* or *disagreement point* or *status quo*, which is the outcome of the game if the players do not agree on a utility allocation in the feasible set. Thus, the *status quo* d can be unilaterally enforced by any player where $d \in S \subset \mathbb{R}^n$. Let Σ be the class of all n-person bargaining problems. A solution on a class of bargaining problems $D \subset \Sigma$ is a mapping $\mathcal{F}{:}D \to \mathbb{R}^n$ such that $\mathcal{F}(S, d) \in S$ for all $(S, d) \in D$ [11].

Consider now a bargaining situation in which the players have claims that are not compatible with each other. Assume that the claims are verifiable and that all players agree that they should be taken into account by any fair solution to the problem at issue. An n-person bargaining problem with claims is a triple (S, d, c) where i) $(S, d) \in \Sigma$, ii) $c \in \mathbb{R}^n \setminus S$ and $c > d$. Let Σ^c be the class of all n-person bargaining problems with claims. A solution on a class of bargaining problems with claims $\mathbf{D}^c \subset \Sigma^c$ is a mapping F: $\mathbf{D}^c \to \mathbb{R}^n$ such that $F(S, d, c) \in S$ for all $(S, d, c) \in \mathbf{D}^c$. Consider a reference function given by a mapping $g{:}\Sigma^c \to \mathbb{R}^n$. Based on the reference function, a reference point will serve as an origin from which relative utility gains or losses are measured; the reference point is not necessarily to be a feasible utility allocation. Let $e = (1, \ldots, 1) \in \mathbb{R}^n$ and define the diagonal Δ in $\mathbb{R}^n$ by $\Delta = \{X \,|\, X = (\lambda \cdot e)$ for some $\lambda \in \mathbb{R}\}$. The *EBS* with respect to g ($\mathcal{E}^g(S, d, c)$) equalizes the gains from the reference point when $\mathcal{G}(S, d, r) \in S$, and equalizes the losses from the reference point when $\mathcal{G}(S, d, r) \notin S$. Formally, $\mathcal{E}^g(S, d, c)$ is defined as follows [11];

$$\mathcal{E}^{\mathcal{G}}(S, d, r) = \mathcal{G}(S, d, r) + \left(\bar{\lambda} \cdot e\right)$$
$$s.t., \begin{cases} \bar{\lambda} = max\,\{\lambda \in \mathbb{R} \,|\, (\mathcal{G}(S, d, r) + (\lambda \cdot e)) \in S\} \\ (S, d, r) \in \Sigma^c \end{cases} \tag{2}$$

In order to characterize the *PBS*, we consider a different kind of reference function: Let $g{:}\Sigma^c \to \mathbb{R}^n \times \mathbb{R}^n_+$ where $\mathcal{G}(S, d, r) = (\mathcal{G}^r(S, d, r), \mathcal{G}^p(S, d, r))$. The main feature of *PBS* is that the $\mathcal{G}(S, d, r)$ assigns to any bargaining problem with claims not only a reference point $\mathcal{G}^r(S, d, r)$, but also a vector of weights $\mathcal{G}^p(S, d, r)$. These weights can be interpreted as relative bargaining powers of the game players. Formally, the *PBS* with respect to g ($\mathcal{P}^g(S, d, c)$) is defined as follows [11];

$$\mathcal{P}^g(S, d, r) = \mathcal{G}^r(S, d, r) + \left(\bar{\lambda} \cdot \mathcal{G}^p(S, d, r)\right)$$
$$s.t.,$$

$$\begin{cases} \bar{\lambda} = max\left\{\lambda \in \mathbb{R} \mid \left(\mathcal{G}^r\left(S,d,r\right) + \left(\lambda \cdot \mathcal{G}^p\left(S,d,r\right)\right)\right) \in S\right\} \\ \left\|\mathcal{G}^p\left(S,d,r\right)\right\| = 1 \\ \left(S,d,r\right) \in \bar{\Sigma}_g^c = \left\{\left(S,d,r\right) \in \Sigma^c \mid \mathcal{G}^p\left(S,d,r\right) \in \mathbb{R}_{++}^n\right\} \end{cases} \tag{3}$$

The axioms involved in the characterization of *EBS* and *PBS* are stated below. The *EBS* satisfies the axioms of ***WPO***, ***SY***$_{\mathcal{G}}$, ***TRANS*** and ***RMON***$_{\mathcal{G}}$. The *PBS* satisfies the axioms of ***WPO***, ***SY***$_{\mathcal{G}}$, ***COV*** and ***RMON***$_{\mathcal{G}}$ [11].

- ***Weak Pareto optimality (WPO):*** A set $A \subset \mathbb{R}^n$ is called *comprehensive*, if $x \in A$ and $x > y$ imply that $y \in A$. For $A \subset \mathbb{R}^n$, let $\mathrm{WPO}\left(A\right) = \left\{x \in A \mid y \in \mathcal{R}^n, y \gg x \Longrightarrow y \notin A\right\}$. Therefore, $F\left(s,d,c\right) \in WPO\left(\mathrm{S}\right)$ for all $\left(S,d,c\right) \in \mathbf{D}^c$.
- ***Symmetry with respect to*** $\mathcal{G}$ ***(SY***$_{\mathcal{G}}$***):*** If $\pi{:}N \to N$ is a permutation, then π defines the mapping $\pi{:}\mathbb{R}^n \to \mathbb{R}^n$, which we denote by the same symbol, via $\left(\pi\left(x\right)\right)_i = x_{\pi^{-1}(i)}, i \in N, x \in \mathbb{R}^n$. For a set $A \subset \mathbb{R}^n$ let $\pi\left(A\right) = \left\{y \mid \exists x \in A \text{with} y = \pi\left(x\right)\right\}$. We call $A \subset \mathbb{R}^n$ *symmetric* if $\pi\left(A\right) = A$ for all permutations π. If $\left(S,d,c\right) \in \mathbf{D}^c$ is such that S is *symmetric* and $g\left(S,d,c\right) \in \Delta$, then $F\left(S,d,c\right) \in \Delta$.
- ***Covariance with respect to translations (TRANS):*** A mapping $L{:}\mathbb{R}^n \to \mathbb{R}^n$ is called a *positive affine transformation* if there exist $a \in \mathbb{R}_{++}^n$ and $b \in \mathbb{R}^n$ such that for all $x \in \mathbb{R}^n$ and all i, $\left(L\left(x\right)\right)_i = a_i x_i + b_i$. Let $\mathcal{L}$ be the class of all positive affine transformations on $\mathbb{R}^n$. For a set $A \subset \mathbb{R}^n$ and $L \in \mathcal{L}$, let $L\left(A\right) = \left\{y \mid \exists x \in A \text{with} y = L\left(x\right)\right\}$. For all $\left(S,d,c\right) \in \mathbf{D}^c$ if $L \in \mathcal{L}$ is a translation and if $L\left(S,d,c\right) \in \mathbf{D}^c$, then $F\left(L\left(S,d,c\right)\right) = L\left(F\left(S,d,c,\right)\right)$.
- ***Restricted monotonicity with respect to*** $\mathcal{G}$ ***(RMON***$_{\mathcal{G}}$***):*** Let $\left(S,d,c\right), \left(S',d',c'\right) \in \mathbf{D}^c$ with $g\left(S',d',c'\right) = g\left(S,d,c\right)$ and $\mathrm{S} \subset \mathrm{S}'$. Then $F\left(S,d,c\right) \le F\left(S',d',c'\right)$.
- ***Covariance with respect to positive affine transformations (COV):*** For all $\left(S,d,c\right) \in \mathbf{D}^c$ and $L \in \mathcal{L}$, if $L\left(S,d,c\right) \in \mathbf{D}^c$, then $F\left(L\left(S,d,c\right)\right) = L\left(F\left(S,d,c,\right)\right)$.

C. OUR TWO-PHASE GAME MODEL FOR THE SHARDING ALGORITHM

In this study, rational BNs can decide against participation in the consensus execution by considering their given rewards. The total payoff received by each BN is the difference between the obtained reward and the spent cost in the epoch. Commonly, BNs bear some costs for fully participating in the consensus process while attempting to improve their benefits. In summary, the key point in the shard-based consensus protocol is to find out the best way for the reward allocation method. In our proposed scheme, two bargaining games work together to design an appropriate reward allocation method. In the upper game $\mathbb{G}^{up}$ at each epoch, generated shards are game players, and their utility functions are formulated according to the received payment $\left(\mathcal{O}_B^{\mathbb{S}}\right)$ minus the incurred cost $\left(\mathcal{C}_B^{\mathbb{S}}\right)$. For the shard $\mathbb{S}_i$, the utility function $\left(\mathbb{U}^{\mathbb{S}_i}\left(B_i\right)\right)$ with the assigned subblock B_i is defined as follows;

$$\mathbb{U}^{\mathbb{S}_i}\left(B_i\right) = \mathcal{O}_{B_i}^{\mathbb{S}_i} - \mathcal{C}_{B_i}^{\mathbb{S}_i}$$
$$s.t., \begin{cases} \mathcal{O}_{B_i}^{\mathbb{S}_i} = \dfrac{[\![B_i]\!]}{[\![\mathcal{T}]\!]} \times \mathfrak{W}^{\mathcal{T}}, \quad i.e., \ B_i = \mathcal{T} \times \mathfrak{P}_{\mathbb{S}_i} \\ \mathcal{C}_{B_i}^{\mathbb{S}_i} = \displaystyle\sum_{BN_l \in \mathbb{S}_i} \left(\mathcal{C}_{BN_l}^{\mathbb{S}_i} \times exp\left(\frac{TC_{BN_l} - CC_{BN_l}}{TC_{BN_l}}\right)^{\gamma}\right) \end{cases} \tag{4}$$

where $[\![B_i]\!]$, $[\![\mathcal{T}]\!]$ are the total numbers of B_i's and $\mathcal{T}$'s transactions, respectively. TC_{N_l} and CC_{N_l} are the total computation capacity and the currently available computation capacity of N_l, respectively. γ is a control parameter to adjust the incurred cost. For the $\mathbb{S}_i$, the reference function $\mathcal{G}\left(\mathbb{S}_i\right)$ is decided using the information of BNs in the $\mathbb{S}_i$; it is the expected gain without participating in the consensus process at the current time epoch. With respect to $\mathcal{G}\left(\mathbb{S}_i\right)$, our egalitarian bargaining solution is given by;

$$\mathcal{E}_{\mathbb{S}}^{\mathcal{G}}\left(\mathbb{P}, \mathcal{T}, \mathfrak{W}^{\mathcal{T}}, \mathbb{S}, \mathbb{U}^{\mathbb{S}}\right)$$
$$= \max_{[\mathfrak{B}_s 1, \ldots \mathfrak{B}_s i \ldots, \mathfrak{B}_s k]} \left(\sum_{\mathbb{S}_{1 \le i \le k} \in \mathbb{P}} \left(\mathcal{G}\left(\mathbb{S}_i\right) + \mathbb{U}^{\mathbb{S}_i}\left(B_i\right)\right)\right)$$
$$s.t., \begin{cases} \mathcal{G}\left(\mathbb{S}_i\right) = \displaystyle\sum_{BN_l \in \mathbb{S}_i} \left(\mathfrak{T} \times exp\left(\frac{TC_{BN_l} - CC_{BN_l}}{TC_{BN_l}}\right)\right) \\ \displaystyle\min_{\mathbb{S}_i, \mathbb{S}_j \in \mathbb{P}} \sqrt{\left(\left(\mathcal{G}\left(\mathbb{S}_j\right) + \mathbb{U}^{\mathbb{S}_j}\left(B_j\right)\right) - \left(\mathcal{G}\left(\mathbb{S}_i\right) + \mathbb{U}^{\mathbb{S}_i}\left(B_i\right)\right)\right)^2} \end{cases} \tag{5}$$

where $\mathfrak{T}$ the BN's gain profit to execute its currently working task. For the lower-game $\mathbb{G}_{\mathbb{S}_i}^{low}$ at each epoch, BNs in the $\mathbb{S}_i$, i.e., $BN_l \in \mathbb{P}^{\mathbb{S}_i}$, are game players, and their utility functions $\left(\mathbb{U}^{BN}\left(B\right)\right)$ are formulated as the same manner as the $\mathbb{U}^{\mathbb{S}}\left(B\right)$. For the $BN_l \in \mathbb{P}^{\mathbb{S}_i}$, the utility function $\left(\mathbb{U}^{BN_l}\left(B_i\right)\right)$ with the assigned subblock B_i is defined as follows;

$$\mathbb{U}_{\mathbb{S}_i}^{BN_j}\left(B_i\right) = \mathcal{R}_{BN_l}^{\mathbb{S}_i}\left(B_i\right) - \mathcal{C}_{\mathbb{S}_i}^{BN_l}\left(B_i\right)$$
$$s.t., \begin{cases} \mathcal{R}_{BN_l}^{\mathbb{S}_i}\left(B_i\right), s.t., \displaystyle\sum_{BN_l \in \mathbb{P}^{\mathbb{S}_i}} \mathcal{R}_{BN_l}^{\mathbb{S}_i}\left(B_i\right) = \mathcal{O}_{B_i}^{\mathbb{S}_i} \\ \mathcal{C}_{\mathbb{S}_i}^{BN_l}\left(B_i\right) = \mathcal{C}_{BN_l}^{\mathbb{S}_i} \times exp\left(\dfrac{TC_{BN_l} - CC_{BN_l}}{TC_{BN_l}}\right)^{\gamma} \end{cases} \tag{6}$$

For the BN_l, the reference function $\mathcal{G}\left(BN_l\right)$ is also decided as the same manner as the $\mathcal{G}\left(\mathbb{S}_i\right)$. With respect to $\mathcal{G}\left(BN_l\right)$, our proportional bargaining solution is given by;

$$\mathcal{E}_{\mathbb{S}_i, BN}^{\mathcal{G}}\left(\mathbb{P}^{\mathbb{S}_i}, B_i, \mathfrak{W}^{B_i}, \boldsymbol{S}, \mathbb{U}_{\mathbb{S}_i}^{BN_l}\right)$$
$$= \sum_{BN_l \in \mathbb{P}^{\mathbb{S}_i}} \mathcal{G}(BN_l) + \max_{[\ldots \mathcal{R}_{BN_l} \ldots]} \left(\sum_{BN_l \in \mathbb{P}^{\mathbb{S}_i}} \left(\omega_{BN_l} \times \mathbb{U}_{\mathbb{S}_i}^{BN_l}(B_i)\right)\right)$$

$$s.t., \begin{cases} \mathcal{G}(BN_l) = \mathfrak{T} \times exp\left(\frac{TC_{BN_l} - CC_{BN_l}}{TC_{BN_l}}\right) \\ \omega_{BN_l} = \left(\frac{CC_{BN_l}}{TC_{BN_l}}\right) \Big/ \sum_{BN_k \in \mathbb{P}^{\mathcal{S}_i}} \left(\frac{CC_{BN_k}}{TC_{BN_k}}\right) \end{cases} \quad (7)$$

D. MAIN STEPS OF PROPOSED SHARD-BASED CONSENSUS SCHEME

To operate the shard-based consensus protocol, we face two main control problems; transaction distribution and BN incentivization. In this study, we focus on the key ideas of egalitarian and proportional bargaining solutions; they are mutually dependent on each other to resolve the above two questions under widely different workload intensities. As we have asserted throughout this paper, we design a novel two-phase cooperative game model. Our game based approach enables the judicious mixture of two different solution concepts, and make control decisions to achieve a fair-efficient solution while adapting the current blockchain network situations. Therefore, the principle novelties of our proposed scheme is its adaptability, flexibility, and responsiveness to current system conditions. The main steps of our proposed scheme are described as follows.

Step 1: System factors and control parameters are determined by a simulation scenario (refer to simulation assumptions and Table 1 in Section IV).

Step 2: At each time epoch, $k + 1$ shards are formed. According to the unbiased random number generation, the size of each shard is generally similar.

Step 3: First, the upper-game $\mathbb{G}^{up}$ is executed. Each $\mathcal{S}$ is a game player, and it's payoff is calculated by using the equation (4).

Step 4: Based on our egalitarian bargaining solution concept, the strategies in $\mathfrak{P}_{\mathcal{S}_{1\leq i\leq k}} \in \mathbb{S}$ are decided according to (5).

Step 5: Second, the lower-games $\mathbb{G}^{low}_{\mathcal{S}_{1\leq i\leq k}}$ are executed in a parallel manner to reach the shard-based consensus. Each BN is a game player, and it's payoff is calculated by using the equation (6). To reach a consensus, the standard byzantine agreement protocol is adopted.

Step 6: Based on our proportional bargaining solution concept, the strategies in $\mathcal{R}^{S}_{N} \in \boldsymbol{S}$ are decided according to (7). To maximize the payoff, individual BNs in each shard may decide whether to participate in the consensus process or not.

Step 7: The adjusting shard $\mathcal{S}_{k+1}$ create the full block $\mathbb{B}$ by uniting the subblocks $B_{1\leq i\leq k}$, which are validated by $\mathcal{S}_{1\leq i\leq k}$. And then, the $\mathcal{S}_{k+1}$ broadcasts the $\mathbb{B}$ to the rest of blockchain network system.

Step 8: For the next epoch's consensus process, proceed to the Step 2.

IV. SIMULATION RESULTS AND DISCUSSION

In this section, we develop a simulation model and analyze the blockchain system performance. Specifically, our analysis focuses on three evaluation metrics: i) fairness among shards; ii) fairness among BNs, and iii) participation ratio for the consensus process. And then, we compare the effectiveness of our proposed scheme with the existing *GTSB* [6], *NATS* [8] and *SSDS* [9] schemes. The scenario setups of our simulation are listed below, and simulation parameters are shown in Table 1.

TABLE 1. System parameters used in the simulation experiments.

Paramete r	Value	Description
m	120	the number of BNs in our blockchain system
k	5	the number of shards and subblocks in our blockchain system
$\mathcal{Q}_{\mathcal{S}}$	$1 \leq \mathcal{Q}_{\mathcal{S}} \leq 120$	the number of BNs in the $\mathcal{S}$, which is randomly decided
$M_{\mathcal{C}}$	0.1	mandatory cost for the randomness generation
$\mathcal{C}^{I}$	0.5	the charge for each BN's identification
$\mathcal{C}^{C}$	0.5	the charge to reach a consensus for each transaction
$\mathcal{C}^{V}$	1	the charges for each transaction's validation
γ	2	a control parameter to adjust the incurred cost
$\mathfrak{W}^{\mathcal{T}}$	3,000,000	The total payment for the $\mathcal{T}$
$\mathfrak{T}$	1,000	the BN's gain profit for the currently working task

- Simulated blockchain network system is designed as a hierarchical structure. Specifically, our blockchain structure consists of five shards and one adjustment shard.
- We assume that there are one hundred twenty BNs, and they form their shards according to unbiased random numbers.
- Simply, each BN has the same transaction conform cost $\left(\mathcal{C}^{V}\right)$ for each transaction validation process.
- Total computation capacity of each BN (TC_N) is normalized, e.g., 1, and the current available computation capacity (CC_N) follows the Poisson distribution with a Poisson random variable ($\rho = 0.8$).
- In the shard-based consensus protocol, the workload for 1000 transaction verification is $0.05TC_N$. It increases linearly with the number of transactions.
- The offered transaction numbers ($\mathcal{T}$) are varied from 1000 to 7000.
- The total payment for the $\mathcal{T}$ $\left(\mathfrak{W}^{\mathcal{T}}\right)$ for each epoch is 3M per 1000 transactions, and the BN's gain profit ($\mathfrak{T}$) for the currently working task is 0.001M.
- All cost charges such as $M_{\mathcal{C}}$, $\mathcal{C}^{I}$, $\mathcal{C}^{C}$ and $\mathcal{C}^{V}$ are normalized values; 0.1, 0.5, 0.5 and 1, respectively.
- The blockchain system performance measures obtained on the basis of 100 simulation runs are plotted as functions of the offered transaction workload.

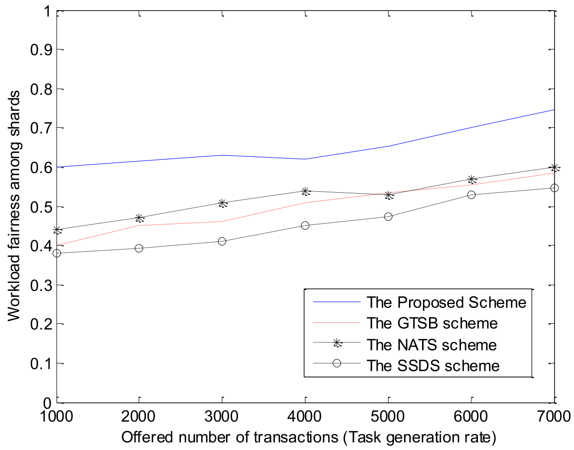


FIGURE 1. Workload fairness among shards.

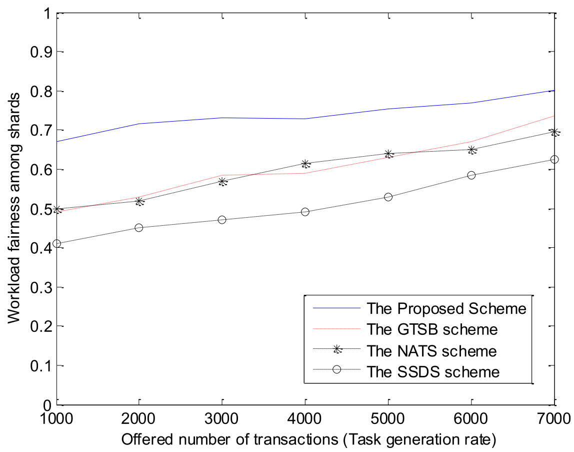


FIGURE 2. Workload fairness among BNs.

According to the simulation metrics, the performance is evaluated mainly to demonstrate the validity of the proposed approach. The simulation parameters are presented in Table 1. Each parameter has its own characteristics.

In Figure 1, we can see the workload fairness among shards. Fairness is a prominent issue to operate the blockchain network system. In this study, we follow the Jain's fairness index to characterize the fairness notion [12]; it has been widely used to measure the fairness in network engineering. Based on the offered numbers of transactions, the relative workload fairness among shards is measured. In our two-phase game approach, we distribute strategically and intelligently the transaction set ($\mathfrak{T}$) per each epoch. During the $\mathbb{G}^{up}$ process, the idea of egalitarian bargaining solution is applied to divide the $\mathfrak{T}$. By considering the reference point of each individual shard, our proposed scheme can ensure the workload fairness. The simulation results in Figure 1 can be interpreted as our egalitarian bargaining solution effectively exploits the load balance in the sharding protocol while the existing *GTSB*, *NATS* and *SSDS* schemes cannot provide such an attractive outcome.

In order to better understand why our two-phase game model is effective to balance the working load in the

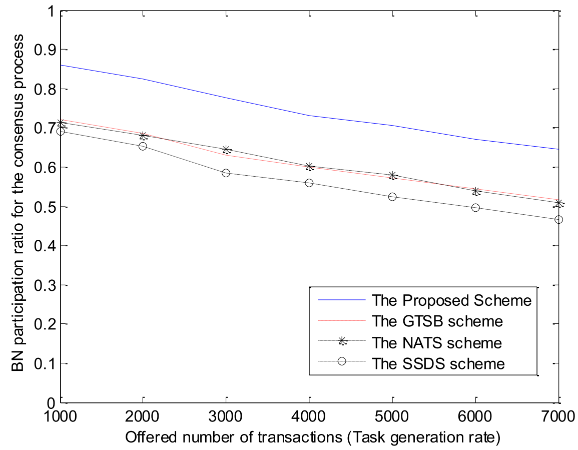


FIGURE 3. BN participation ratio for the consensus process.

blockchain network system, we also plotted the workload fairness ratio among BNs in Figure 2. The trend of fairness among BNs is similar to the trend of fairness among shards. As can be observed, it is shown that our proposed scheme holds significant dominance in the fairness issue than the other existing protocols. The reason for the aforesaid is that our lower-game $\mathbb{G}_{\mathbb{S}}^{low}$ is formulated to force reasonably light-workload BNs into the consensus process. Due to this reason, we can maintain the better workload fairness among BNs from low to high offered workload intensities.

Figure 3 depicts the BN participation ratio in the shard-based consensus process. In our proposed scheme, we develop a hierarchical two-phase bargaining game to efficiently assign the reward for each individual BN by considering the current workload. Through the proportional bargaining method in the $\mathbb{G}_{\mathbb{S}}^{low}$ process, we can adaptively incentivize each BN to participate in the consensus process. Therefore, in our proposed scheme, the actual outcome is properly dealt out compared to the existing schemes. The simulation results displayed in Figure 1 to Figure 3 justify the advantages of the proposed scheme. By the reciprocal combination of two different bargaining solutions, we jointly design a novel two-phase game model to provide an appropriate workload balance while intelligently inducing selfish BNs to participate in the blockchain consensus process. Therefore, the numerical results validate that we can achieve an excellent system performance than the existing *GTSB*, *NATS* and *SSDS* schemes.

V. SUMMARY AND CONCLUSION

Blockchain popularity is gaining massive momentum in the last few years. Recently, a novel blockchain technology, called sharding, has been proposed to achieve scale-out throughput by letting BNs only acquire a fraction of the whole transaction set. However, the shard-based consensus protocol faces essential some challenge - how to distribute the transaction set, and how to share the reward among BNs. In this study, we have addressed the above two questions by considering the synergy of two bargaining solutions. Based on the two-phase cooperative game paradigm, we formulate a new shard-based consensus scheme to provide scale-out system performance. According to the idea of egalitarian bargaining solution, we distribute the transaction set for each shard. By using the main concept of proportional bargaining solution, each individual BN can adaptively share the reward. Our two-phase bargaining game approach can act collectively to force BNs towards more and more collaborations within each shard. To the best of our knowledge, this work is the first step towards a deeper understanding of the effect of cooperative behaviors in the shard-based blockchain network. Finally, we analyze the blockchain system performance using the simulation analysis. Numerical results have demonstrated the feasibility and efficiency of our proposed scheme compared with the existing state-of-the-art literature.

Further studies are still needed in the future. We will design a deep-learning based strategy modification mechanism to further improve the overall system efficiency. And, we would like to jointly integrate more features, such as reputation mechanisms, in order to establish a more sophisticated and effective consensus solution. In addition, we plan to consider other consensus approach from other aspects while including a complete risk analysis. The security and real-time authentication issues also need to be studied in detail. Last, we are keen to implement our protocol to real test-bed and analyze the system performance, which is hopeful to achieve valuable experience for practitioners.

REFERENCES

[1] Z. Liu, N. C. Luong, W. Wang, D. Niyato, P. Wang, Y.-C. Liang, and D. I. Kim, "A survey on blockchain: A game theoretical perspective," *IEEE Access*, vol. 7, pp. 47615–47643, 2019.

[2] J. Al-Jaroodi and N. Mohamed, "Blockchain in industries: A survey," *IEEE Access*, vol. 7, pp. 36500–36515, 2019.

[3] T. T. A. Dinh, R. Liu, M. Zhang, G. Chen, B. C. Ooi, and J. Wang, "Untangling blockchain: A data processing view of blockchain systems," *IEEE Trans. Knowl. Data Eng.*, vol. 30, no. 7, pp. 1366–1385, Jul. 2018.

[4] Y. Wang, S. Cai, C. Lin, Z. Chen, T. Wang, Z. Gao, and C. Zhou, "Study of blockchains's consensus mechanism based on credit," *IEEE Access*, vol. 7, pp. 10224–10231, 2019.

[5] 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. 22328–22370, 2018.

[6] M. H. Manshaei, M. Jadliwala, A. Maiti, and M. Fooladgar, "A game-theoretic analysis of shard-based permissionless blockchains," *IEEE Access*, vol. 6, pp. 78100–78112, 2018.

[7] S. Cano-Berlanga, J.-M. Giménez-Gómez, and C. Vilella, "Enjoying cooperative games: The R package GameTheory," *Appl. Math. Comput.*, vol. 305, pp. 381–393, Jul. 2017.

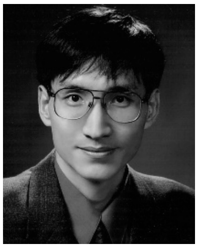
[8] C. Liu, Y. Xiao, V. Javangula, Q. Hu, S. Wang, and X. Cheng, "NormaChain: A blockchain-based normalized autonomous transaction settlement system for IoT-based E-commerce," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4680–4693, Jun. 2019.

[9] E. Kokoris-Kogias, P. Jovanovic, L. Gasser, N. Gailly, E. Syta, and B. Ford, "OmniLedger: A secure, scale-out, decentralized ledger via sharding," in *Proc. IEEE Symp. Secur. Privacy*, May 2018, pp. 583–598.

[10] L. Luu, V. Narayanan, C. Zheng, K. Baweja, S. Gilbert, and P. Saxena, "A secure sharding protocol for open blockchains," in *Proc. ACM SIGSAC*, 2016, pp. 17–30.

[11] A. Gerber, "Reference functions and solutions to bargaining problems with claims," *Social Choice Welfare*, vol. 24, pp. 527–541, Jun. 2005.

[12] M. Dianati, X. Shen, and S. Naik, "A new fairness index for radio resource allocation in wireless networks," in *Proc. IEEE WCNC*, vol. 2, Mar. 2005, pp. 712–715.



SUNGWOOK KIM received the B.S. and M.S. degrees from Sogang University, Seoul, South Korea, in 1993 and 1995, respectively, and the Ph.D. degree from Syracuse University, Syracuse, New York, in 2003, supervised by Prof. Pramod K. Varshney, all in computer science. He has held faculty positions with the Department of Computer Science, Choongang University, Seoul. In 2006, he was with Sogang University, where he is currently a Professor with the Department of Computer Science and Engineering and a Research Director of the Network Research Laboratory. His research interests include resource management, online algorithms, adaptive quality-of-service control, and game theory for network design.