A Game Theoretical Analysis of Non-Linear Blockchain System

Lin Chen[1], Lei Xu[2], Zhimin gao[3], Ahmed Imtiaz Sunny[1], Keshav Kasichainula[4], and Weidong Shi[4]

[1] Texas Tech University in Lubbock, Texas, United States of America
chenlin198662@gmail.com,ahmed.sunny@ttu.edu
[2] University of Texas Rio Grande Valley in Edinburg,Texas United States of America
xuleimath@gmail.com
[3] Auburn University of Montgomery in Montgomery,Alabama, United States of America
mtion@msn.com
[4] University of Houston in Houston,Texas, United States of America
{kkasichainula,wshi3}@uh.edu

Abstract

Recent advances in the blockchain research have been made in two important directions. One is refined resilience analysis utilizing game theory to study the consequences of selfish behavior of users (miners), and the other is the extension from a linear (chain) structure to a non-linear (graphical) structure for performance improvements, such as IOTA and Graphcoin. The first question that comes to mind is what improvements that a blockchain system would see by leveraging these new advances. In this paper, we consider three major properties for a blockchain system: α-partial verification, scalability, and finality-duration. We establish a formal framework and prove that no blockchain system can achieve α-partial verification for any fixed constant α, high scalability, and low finality-duration simultaneously. We observe that classical blockchain systems like Bitcoin achieves full verification ($\alpha = 1$) and low finality-duration, Ethereum 2.0 Sharding achieves low finality-duration and high scalability. We are interested in whether it is possible to partially satisfy the three properties.

1 Introduction

Blockchain is a decentralized ledger that provides a system for self-interested parties to carry out transactions without a fully trusted central system. As such, there is no centralized party who can compute an optimal solution or a Nash equilibrium in advance and then tell each user to take certain actions, as this will violate the foundations of blockchain systems. The basic idea behind this is that, instead of having trust in a centralized system or any other specific participant, each participant chooses to trust the majority of the participants and accepts the outcome achieved through consensus among them. [13]

One major reason that hinders the adoption of blockchain is scalability [29]. For example, Bitcoin network can only process less than 10 transactions per second on average [8], while typical payment systems like Visa can process thousands of transactions per second.

Recently, a variety of approaches are proposed to address the scalability issue. Most of them follow the general framework of divide and conquer, e.g., Zilliqa [26], Harmony [25], and Ethereum 2.0 [24], and use a sharding scheme that allows transactions to be processed by a subgroup of nodes (a sharding committee). A sharding scheme usually has a critical issue in terms of resilience, as the correctness of each transaction now solely depends on a subgroup of voters. Consequently, if common consensus protocols like Proof-of-Work (PoW) or Byzantine Fault Tolerance (BFT) is used within subgroups, then the fraction of malicious nodes within *every* subgroup cannot exceed 1/2 or 1/3, which is a significantly

stronger assumption than that of a standard blockchain system. A typical blockchain system only requires that the fraction of malicious nodes do not exceed 1/2 or 1/3 of all nodes. We remark that both Harmony and Ethereum 2.0 claim that if subgroups are generated in a perfect randomized way, then the percentage of honest nodes within each subgroup is almost the same as their percentage in the whole group of nodes; however, this requires a perfect distributed random number generation as a separate procedure, which brings an additional assumption on the security of this additional procedure.

To guarantee that every transaction is correctly executed by only relying on the standard assumption that the majority of the nodes are honest, we must require every transaction to be verified by all the nodes. Consequently, it is a straightforward question of whether scalability is achievable at all, as it appears that any divide and conquer based solution would inevitably reduce the total number of verifications received by a transaction. A "non-linear" blockchain structure recently introduced by IOTA [15] and Graphcoin [4], can neglect such limitations. The basic idea is to allow blocks to be connected as a directed acyclic graph (DAG) instead of a chain. Such a non-linear structure implements a divide and conquer approach implicitly by allowing multiple blocks to be appended simultaneously, as a general graph can be extended in multiple directions. Meanwhile, if we treat different growing directions as soft forks or branches, then it is possible (depending on system parameters) that they can "merge" again in the future (see the following figure 2, where the sequences of blocks that follow block A and B meet at block C). Therefore, a node that tries to append a new block is required to verify a few previous blocks. So, there is a possibility that a block may still be verified by all the nodes, albeit the delay of such verification.

On a high level, there are three crucial properties involved in a general blockchain system: verification, scalability, and finality-duration. In a nutshell, α-partial verification requires every transaction to be verified by at least α fraction of all the nodes (which thus ensures resilience under the standard assumption that the majority of the nodes follow the protocol); scalability means the system throughput, or the total number of transactions executed per unit of time, is proportional to the total number of participating nodes; and finality-duration means the delay in reaching consensus on the correctness of the execution of each transaction. We give a precise definition in Section 3.

Classical blockchain systems like Bitcoin achieves full verification and low finality-duration, but not scalability. This is because Bitcoin requires every block, and hence the transactions within a block, to be verified by all the nodes; meanwhile, it has a constant finality-duration because every block is finalized after a constant number of blocks are appended afterward. However, it does not scale, as the increase in the number of nodes does not allow the system to handle more transactions per unit of time, which has been pointed out in many prior papers [27, 13, 20, 21].On the other hand, blockchain systems like Harmony [25] and Ethereum 2.0 [24] achieve constant finality-duration and scalability, but not full verification. For example, the sharding scheme used in Ethereum 2.0 allows a block to be verified within a shard (which is a subset of nodes).

In this paper, we provide a view on the relationships between full verification (or more precisely, α-partial verification for any constant α), low finality-duration, and scalability. More precisely:

Our contributions. We show that it is impossible to achieve full verification, low finality-duration and scalability simultaneously.

Given the fact that: (i) Bitcoin achieves full verification and low (asymptotically constant) finality-duration, but not scalability; and (ii) Ethereum 2.0 achieves low (constant) finality-duration and scalability, but not full verification, it is natural to explore to what extent a blockchain system can be designed considering the different trade-off scenarios. In particular, does there exist a blockchain system that satisfies both full verification and scalability/finality-duration? Is it possible to have a system that partially satisfies all of the three properties? We give an affirmative answer in this paper. In particular, we prove that by adopting a non-linear blockchain system and employing a game-theoretical analysis, we can construct a system which achieves full verification and a trade-off between scalability and finality-duration. Informally speaking, the following properties hold simultaneously for the constructed

blockchain system:

(i) $O(s)$ new blocks are generated per unit of time on average;

(ii) after $O(s \log s)$ units of time, with a very high probability, each block will be verified by all users in the systems.

Here s is a system parameter that can be set suitably at the genesis block. When $s = 1$, the non-linear system degenerates to a linear system with a fixed block generation rate that is independent of the nodes in the system, while the delay which is an indication of finality duration, is a constant. This coincides with the classical Bitcoin system. Conversely, s can be as high as $O(m)$ where m is the number of nodes. In this case, the system is fully scalable, albeit that only a sufficiently long delay ($O(s \log s)$) can ensure full verification. However, if we set s to be $O(m)$ to enforce scalability and meanwhile enforce the delay to be some constant instead of $O(s \log s)$, then full verification cannot be guaranteed.

We remark that the big-O notation in our statements hides a constant which is roughly the average time for a block to be generated, that is, we measure the delay in terms of the number of blocks; therefore, Bitcoin is considered as low (asymptotically constant) finality-duration as the delay is constant blocks. Our result does not conflict with prior researches that complain about the "high" finality-duration of Bitcoin because of the long time it takes to generate a single block [15, 4]. The research that tries to decrease such a block generation time is parallel to this paper. For example, if a lighter version of PoW can be used in the existing Bitcoin system, then it can also be used directly in our non-linear blockchain system, while our impossibility result, as well as the trade-off between finality-duration and scalability, remain the same.

2 Related Work

The study of e-cash systems dates back to 1983 [5, 19]. However, all such systems require a centrally or quasi-centrally controlling authority. A well-known exception, Bitcoin, was introduced by Nakamoto [13] in 2008, which uses a public ledger known as a *blockchain* to record transactions carried out between users. Following this line of research, various alternative blockchain-based transaction systems are proposed [27, 12, 20, 21], further improving the performance and security of Bitcoin as well as extending the system to deal with applications beyond transactions (e.g., smart contracts). In [22], Sompolinsky and Zohar have introduced an alternative to the longest chain that allows more transactions to take place at a lower cost. Recently, Sompolinsky et al. [23] have shown faster block generation by generalizing blockchain to a direct acyclic graph of blocks. Blockchain-based consensus protocols Fantômette and Avalanche that rely on blockDAG were proposed in [2, 17]. We refer the readers to several surveys on blockchain systems [28, 6, 10, 18, 1, 31, 11, 3, 9]. In particular, [28] provides a comprehensive introduction to the bitcoin network,[3, 9] focus on the systematized study of the blockchain consensus protocols, [6, 10, 18] focus on the security and privacy results on blockchain, [1] focuses on the applications of blockchain. The most relevant survey to this paper is [11], which summarizes recent results on game-theoretical studies of blockchain. However, most of the existing game theoretical research primarily focuses on the traditional linear blockchain system, only a very recent paper by Popov et al. [16] gives the first game-theoretical analysis of IOTA. Their result, however, does not establish the trade-off between scalability and finality-duration.

2.1 Classical and Non-linear Blockchain

Chain-structured blockchain. Most of the existing blockchain systems, e.g., Bitcoin, Ethereum, Hyperledger, follow the classical structure where blocks form a chain as illustrated by Fig 1.

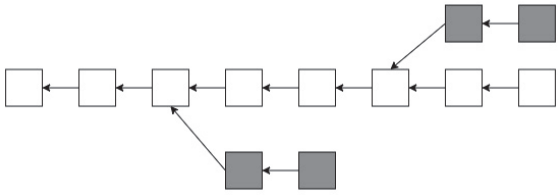

Figure 1: Chain-structured blockchain. The chain grows from left to right. White squares form the main chain, and gray squares form the side chains that are discarded eventually.

Non-linear (graph-structured) blockchain. Popov introduced the concept of *tangle* [15] which allows a blockchain to adopt a directed acyclic graph (DAG) architecture. We summarize the abstract model of a non-linear blockchain in Section 3. We briefly review IOTA, which is the most well-known non-linear blockchain system so far. On a high level, IOTA allows each transaction be an individual node linked in the distributed ledger. We may interpret a transaction as a block in such a system. In the tangle, each user needs to select one transaction from the pool as well as two previous blocks (transactions) in the system. The user verifies these two transactions and mines a new block referring to them. Then this new block (transaction) is broadcasted to the tangle network. Figure 2 gives a simple example of a non-linear blockchain.

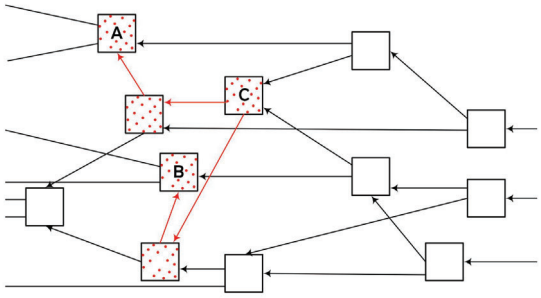

Figure 2: A non-linear blockchain. White squares are verified transactions/blocks.

3 The Abstract Model

We describe an abstract model of a non-linear blockchain which is general enough to incorporate existing well-known non-linear blockchain systems like IOTA and Graphcoin.

A non-linear blockchain *NLB* is defined by a quadruple

$$NLB : (\mathscr{A}, \mathscr{C}, \mathscr{R}, \mathscr{E}), \text{where}$$

- $\mathscr{A}$ defines the rules of building and adding a new block to the blockchain. Since we are considering non-linear blockchain, $\mathscr{A}$ allows multiple blocks to be added simultaneously.
- $\mathscr{C}$ defines the way to check a block, including validity verification, such as whether the block has the correct format and whether transactions included in the block are valid, and whether the block is finalized.
- $\mathscr{R}$ defines the way how the award is assigned to a user who adds a new block to the DAG. A *NLB* needs to encourage users to participate in the construction of the blockchain by giving rewards to those who add new blocks.

- $\mathscr{E}$ defines the rules to eliminate conflicting blocks. Similar to a linear blockchain, it is possible that multiple participants have different local copies of the blockchain, and $\mathscr{E}$ determines which version should be kept.

Next, we provide formal definitions of the three metrics of a blockchain system that we mentioned earlier.

Definition 1 (Partial verification)**.** *For any fixed $\alpha \in (0,1]$, a blockchain system satisfies the property of α-partial verification if every block is verified by at least α fraction of the total nodes in the system before it is finalized.*

In particular, if $\alpha = 1$, then each block is verified by all the nodes and we call it *full verification*. If a blockchain system satisfies full verification, then resilience follows directly from standard assumptions on the percentage of honest nodes among all nodes, e.g., if the blockchain uses PoW or BFT as the consensus protocol, then it is resilient, once the majority or 2/3 of nodes follow the protocol. This is also true for α-partial verification if the α-fraction of the nodes are randomly selected from all nodes. Indeed, many recently developed blockchain systems that claim to achieve scalability, e.g., Ethereum 2.0 [24] implements such an idea by letting a subset of nodes (i.e., a *Shard*) compute and verify a smart contract. The size of such a subset divided by the total number of nodes gives the percentage α.

Definition 2 (Scalability)**.** *The throughput of a blockchain system is the number of blocks n_b that can be added to the system in a fixed time. A blockchain system scales with the number of nodes m in the system if $n_b \to \infty$ when $m \to \infty$. Particularly, a blockchain system fully scales with the number of nodes m if $n_b = \Omega(m)$.*

It should be clear that the definition of scalability or full scalability does not depend on the length of the time period chosen for throughput. It captures the possibility of speeding up blockchain generation with more participating nodes; consequently, classical blockchain systems like Bitcoin does not scale.

Definition 3 (Finality-duration)**.** *The finality-duration of a blockchain system is the time difference between the time point when a block is appended and the time point when a block receives full verification.*

We say the finality-duration of a blockchain system is low (or constant) if the finality-duration is independent of the nodes in the system; consequently, classical blockchain systems like Bitcoin has a low (asymptotically constant) finality-duration because after a fixed number of blocks are appended, all the nodes start following the main chain, thus blocks on the main chain will receive full verification.

4 Impossibility Result

Theorem 1. *There does not exist a blockchain system that simultaneously satisfies (i) scalability; (ii) low finality-duration; and (iii) α-partial verification for an arbitrary constant $\alpha \in (0,1]$.*

The proof follows from a counting argument on the total number of verifications.

Proof. Suppose, on the contrary, that there exists such a block-chain. Then by definition, every block or transaction will receive verifications from at least α fraction of the nodes within a constant delay. Let c_0 be the constant delay. Consider an arbitrary node x and let τ_x be the fixed time it takes for node x to perform one verification. Let the throughput of the blockchain be n_b, then by definition of scalability, $n_b = n_b(m) \to \infty$ when $m \to \infty$. Note that all the n_b blocks generated shall be verified by at least α fraction of nodes within the delay of c_0, which means on average, every node should perform αn_b verifications within c_0. However, node x can only perform c_0/τ_x verifications, which is a constant. Since $n_b \to \infty$, when m is sufficiently large, $\alpha n_b > c_0/\tau_x$ for any fixed α. Therefore, it is impossible for an arbitrary node x to complete all the verifications. Hence, the three properties, scalability, low finality-duration and α-partial verifications, cannot be satisfied simultaneously. □

Remark. If α is not a constant, e.g., $\alpha = 10/N$ where N is the total number of nodes, then $\alpha \to 0$ and it is indeed possible to guarantee scalability and low finality-duration simultaneously.

5 Satisfying Impossibility-Triangle in a Liquid Way

As we have mentioned before, Bitcoin achieves full verification and low finality-duration at the cost of scalability, Ethereum 2.0 achieves scalability and low finality-duration, but cannot guarantee a constant α for partial verification. These systems obey our impossibility triangle by conforming to two properties and disregarding the third one. But what if we want all of them in partial? More precisely, can we have a more fine-grained system that can can smoothly transform from one extreme to another by controlling a simple system parameter? In this section, we construct a non-linear blockchain system, which builds upon the basic structure of IOTA, and show that it achieves full verification, and meanwhile has a trade-off between scalability and finality-duration. In particular, a larger scalability implies longer finality-duration, and this can be controlled through a single system parameter Δ as we will define later.

5.1 Non-linear Blockchain (NLB) Construction

We first propose a concrete construction of NLB that achieves both security and scalability under the agent model (Based on the agent model, every participant is an agent who tries to maximise his/her profit). Without loss of generality, we assume that each block only includes one transaction. In the following, we broaden the terms and use them interchangeably. We first define some concepts.

Definition 4 (Block distance, descendant, and ancestor)**.** *Given two blocks A and B, we define the distance between the two blocks as the length of the shortest directed path from A to B, which is denoted as $d(A,B)$. If there is no such a directed path, we define $d(A,B) = \infty$. If $0 < d(A,B) < \infty$, we say B is a descendant of A, and A is an ancestor of B. For a block B and each $1 \leq k \leq \ell$, let $Anc(B,k) = \{X | d(X,B) = k\}$ and $Des(B,k) = \{d(B,X) = k\}$, where ℓ is a given parameter.*

The new NLB is constructed as follows:

- $\mathscr{A}$. The new *NLB* assumes that there is a pool of new transactions from which a user can select one to construct a new block, which refers to two previous blocks[1]. The user then does lightweight mining to fix this information in the newly constructed block. Lightweight mining is a common approach used in blockchain systems supporting lightweight users (see, e.g. [30]). It means that the user needs shorter computation time, compared with the standard mining process, to find a value for the block that makes it a valid block. This is usually done by a loose requirement on the hash result of the block together with the mined value. Suppose that the newly built block is B, the user also verifies blocks in $Anc(B,k), 1 \leq k \leq \ell$, where ℓ is a pre-defined system parameter that determines how many previous blocks the producer of a new block should verify.

- $\mathscr{C}$. To check a block B, the algorithm first checks whether the block format is correct, including the verification of the mining outcome. The algorithm also checks whether B is finalized or not, which is determined by
$$num_{Anc} \leftarrow |\cup_{k=1}^{\ell} Anc(B,k)|.$$
If num_{Anc} is larger than the system pre-defined threshold, B is finalized.

- $\mathscr{R}$. Each block has a reward value and the system imposes an upper bound on the maximal reward offered by a transaction, so that the largest and smallest reward among transactions (and blocks)

[1]Our analysis in this paper also works if a new block refers to any fixed constant (greater than or equal to 2) number of blocks. For ease of presentation, we take this number to be 2 throughout this paper.

can differ by a factor at most Δ. The producer of the new block also receives rewards from previous blocks. Specifically, each block is associated with a uniform verification cost vrf, which is divided into ℓ parts such that $vrf_1 < vrf_2 < \cdots < vrf_\ell$ and $\sum_{k=1}^{\ell} vrf_k = vrf$. For each $1 \leq k \leq \ell$, the producer of block B gets reward $vrf_k/|Des(X,k)|$ for each $X \in Anc(B,k)$. This means that the verification reward of vrf_k from block X is evenly distributed among all descendants in $Des(B,k)$. Note that the reward is not uniformly divided and it will be only collected when the new block is finalized.

- $\mathscr{E}$. The constructed NLB adopts the largest-weighted descendants principle (LWD) to eliminate disagreement, i.e., when there are blocks containing conflicting transactions, the one whose descendants have a largest total weight will be selected. Note that this will not prevent multiple non-conflict blocks from being added in parallel. Formally, for each block B, let $Des(B) = \cup_{h\geq 1}\{X|d(B,X)\}$. If there are two conflicting blocks B, B' and $|Des(B)| > |Des(B')|$, then B prevails, that is, users will abandon B' together with all its descendants in the sense that a new block will not refer to any of these blocks.

Note that when considering the private costs for the mining task of the agents, the only difference is that the profit of each block is no longer its reward, instead, the new profit of each agent should be the reward of each task subtracting the (agent-dependent) cost. All results in our paper still hold if the largest profit and smallest profit differ by at most Δ times for all agents. In our paper, the blockchain system is designed such that the largest and smallest reward can differ by at most Δ times. Incorporating costs of agents, however, this cannot be ensured by system design. Specifically, if an agent has an excessive private cost then the ratio can be unbounded. But in practice, it is plausible to assume that the private cost is usually small compared with the reward of blocks.

5.2 Scalability and Finality-duration Analysis

We first give a high-level summary of the workflow of the proposed NLB system. Transactions are generated over time and form a pool. Each transaction is associated with a distinct transaction reward and a fixed verification reward vrf. Each time, a miner will select one transaction from the pool and append a block, which refers to two previous blocks. Here the miner needs to decide two things: (i) which transaction to include, and (ii) which two previous blocks to refer to. As we assume that miners are rational players, they will strategically make their decisions to maximize their profits, and this section is devoted to analyze the scalability and security of the system under an arbitrary Nash equilibrium.

We formalize the problem as follows. Let the pool consist of n transactions, with the transaction reward being $p_1, p_2, \cdots, p_n$. Let m be the number of miners, with computational powers being $u_1, u_2, \cdots, u_m$. As we mentioned, each miner will mine a new block by including one transaction from the pool. If multiple miners say, miners in the subset of S, all choose the same transaction, then they compete, and only one of them will succeed, and the probability that some miner $i \in S$ succeeds is $\frac{u_i}{\sum_{h\in S} u_h}$. If, however, all miners choose different transactions, then each of them can append a new block. In the following section, we will analyze the scalability and finality-duration of the constructed NLB separately.

5.2.1 Scalability

For scalability, we are interested in how many different transactions from the pool can be selected by the miners simultaneously. Note that the more different transactions are chosen, the higher scalability is. When miners choose transactions simultaneously, we are considering the worst-case because if miners are selecting transactions at different times, later ones may be able to avoid conflicts with earlier ones. Let n be the number of available transactions and m be the number of miners, the following Theorem 2 implies that the system is scalable even in the worst case such that when there are sufficiently many

transactions, the throughput will be $O(m/\Delta)$ where Δ is a system parameter part of $\mathscr{R}$. By controlling Δ, we can control the scalability of the system. In particular, when we set Δ to be a constant, the system becomes fully scalable with the number of nodes. In the remainder of this section is devoted to proving Theorem 2.

Theorem 2. *With probability at least* $1-\max\{e^{-\Theta(m)}, e^{-\Theta(n)}\}$*, the number of blocks mined by m miners in an arbitrary Nash equilibrium is at least* $\min\{c_1 m/\Delta, c_2 n\}$ *for some universal constants* c_1, c_2.

Notice that a Nash equilibrium always exists by allowing mixed strategies [14]. Towards the proof, we introduce some notations. For simplicity, let all the transaction rewards be $p_1 \geq p_2 \geq \cdots \geq p_n$. By the design of our system we require that $p_1/p_n \leq \Delta$. Note that the strategy of a miner is to select one transaction. We consider the general mixed strategy of a miner where he/she can specify a probability for each transaction.

Consider an arbitrary Nash equilibrium and let $\pi^{(i)} = (\pi_1^{(i)}, \pi_2^{(i)}, \cdots, \pi_n^{(i)})$ be the strategy of miner i in the equilibrium, where $\pi_j^{(i)}$ is the probability that he chooses transaction j. It is obvious that $\sum_{j=1}^n \pi_j^{(i)} = 1$ for any $1 \leq i \leq m$. Let $X_j^{(i)}$ be the 0-1 random variable that indicates whether miner i chooses transaction j. Then $X_j^{(i)} = 1$ with probability $\pi_j^{(i)}$ and $X_j^{(i)} = 0$ with probability $1-\pi_j^{(i)}$.

Consider the above Nash equilibrium. Intuitively, if only a small number of transactions are selected, then miners must have devoted their probabilities to a few transactions. Therefore, to show that a sufficient number of distinct transactions are selected in expectation, by miners, we need to show that the miners are distributing their probabilities in a fair way among transactions, as is implied by the following lemma.

Lemma 1. *If there exists some transaction* j_1 *such that* $\sum_{i=1}^m \pi_{j_1}^{(i)} \geq 12\Delta$*, then for every transaction j, it holds that* $\sum_{i=1}^m \pi_j^{(i)} \geq 1/2$.

Proof. Suppose, on the contrary, the lemma is not true, that is, there exists some transactions j_1 and j_2 such that $\sum_{i=1}^m \pi_{j_1}^{(i)} \geq 12\Delta$ and $\sum_{i=1}^m \pi_{j_2}^{(i)} < 1/2$. Consider the set of miners that choose transaction j_1 with positive probability. For simplicity, let these miners be miner $1, 2, \cdots, k$ such that $u_1 \geq u_2 \geq \cdots \geq u_k$. We show in the following that miner k can change his strategy to get a strictly higher profit, contradicting the fact that this is a Nash equilibrium, and consequently, the lemma is proved. More precisely, we argue that player k can get strictly larger profit (in expectation) by increasing his probability of choosing transaction j_2 and meanwhile decreasing his probability of choosing j_1.

The expected profit that miner k can get from transaction j_1 and j_2 using his current strategy is equal to

$$p_{j_1}\mathbb{E}[\Gamma_1] + p_{j_2}\mathbb{E}[\Gamma_2],$$

where for $h = 1, 2$, we have

$$\Gamma_h = \begin{cases} 0, & \text{if } \sum_{i=1}^n u_i X_{j_h}^{(i)} = 0 \\ \frac{u_k X_{j_h}^{(k)}}{\sum_{i=1}^m u_i X_{j_h}^{(i)}}, & \text{Otherwise.} \end{cases}$$

If k changes his strategy by choosing j_1 with the probability of 0 and choosing j_2 with the probability of $\pi_{j_1}^{(k)} + \pi_{j_2}^{(k)}$, then the expected profit he can get from j_1 and j_2 is equal to $p_{j_2}\mathbb{E}[\tilde{\Gamma}_2]$, where

$$\tilde{\Gamma}_2 = \begin{cases} 0, & \text{if } \sum_{i\neq k} u_i X_{j_2}^{(i)} + u_k \tilde{X}_{j_2}^{(k)} = 0 \\ \frac{u_k \tilde{X}_{j_2}^{(k)}}{\sum_{i\neq k} u_i X_{j_2}^{(i)} + u_k \tilde{X}_{j_2}^{(k)}}, & \text{Otherwise.} \end{cases}$$

and $\tilde{X}_{j_2}^{(k)}$ is the 0-1 random variable that takes the value 1 with the probability of $\pi_{j_1}^{(k)} + \pi_{j_2}^{(k)}$. In the following we show that

$$p_{j_2}\mathbb{E}[\tilde{\Gamma}_2 - \Gamma_2] > p_{j_1}\mathbb{E}[\Gamma_1],$$

which implies the correctness of the lemma. We prove the following two claims.

Claim 1. $\mathbb{E}[\Gamma_1] < \frac{\pi_{j_1}^{(k)}}{2\Delta}$.

Proof. Let $X = \sum_{i \neq k} X_{j_1}^{(i)}$ and $\mu = \mathbb{E}[X] \geq 5\Delta/\theta - \mathbb{E}[X_{j_1}^{(k)}] = 6\Delta/\theta - \pi_{j_1}^{(k)}$. For any $\delta \in (0,1)$, we have

$$\begin{aligned}\mathbb{E}[\Gamma_1] = \quad & \mathbb{E}[\Gamma_1 | X > (1-\delta)\mu] \Pr(X > (1-\delta)\mu) \\ & + \mathbb{E}[\Gamma_1 | X \leq (1-\delta)\mu] \Pr(X \leq (1-\delta)\mu)\end{aligned}$$

Given that $\Gamma_1 \leq 1$, we know that

$$\begin{aligned} & \mathbb{E}[\Gamma_1 | X \leq (1-\delta)\mu] \\ = \quad & \mathbb{E}[\Gamma_1 | X \leq (1-\delta)\mu, X_{j_1}^{(k)} = 1] \Pr(X_{j_1}^{(k)} = 1) \\ & + \mathbb{E}[\Gamma_1 | X \leq (1-\delta)\mu, X_{j_1}^{(k)} = 0] \Pr(X_{j_1}^{(k)} = 0) \\ = \quad & \pi_{j_1}^{(k)} \cdot \mathbb{E}[\Gamma_1 | X \leq (1-\delta)\mu, X_{j_1}^{(k)} = 1] \leq \pi_{j_1}^{(k)}.\end{aligned}$$

Meanwhile, by $u_1 \geq u_2 \geq \cdots \geq u_k$, we have

$$\Gamma_1 \leq \frac{X_{j_1}^{(k)}}{\sum_{i=1}^{k} X_{j_1}^{(i)}}.$$

According to Chernoff bound, we know that

$$\Pr(X \leq (1-\delta)\mu) \leq e^{\frac{-\delta^2\mu}{2}}.$$

Therefore,

$$\begin{aligned}\mathbb{E}[\Gamma_1] \quad = \quad & \mathbb{E}[\Gamma_1 | X > (1-\delta)\mu] \Pr(X > (1-\delta)\mu) \\ & + \mathbb{E}[\Gamma_1 | X \leq (1-\delta)\mu] \Pr(X \leq (1-\delta)\mu) \\ \leq \quad & \mathbb{E}[\frac{X_{j_1}^{(k)}}{(1-\delta)\mu + X_{j_1}^{(k)}}] + e^{\frac{-\delta^2\mu}{2}} \cdot \pi_{j_1}^{(k)}\end{aligned}$$

Consider the function $\varphi(x) = \frac{x}{(1-\delta)\mu + x}$, it is easy to verify that $-\varphi(x)$ is convex in $x \in (0, +\infty)$, therefore, by Jensen's inequality

$$\begin{aligned}\frac{\mathbb{E}[X_{j_1}^{(k)}]}{(1-\delta)\mu + \mathbb{E}[X_{j_1}^{(k)}]} = \varphi(\mathbb{E}[X_{j_1}^{(k)}]) \quad & \geq \quad \mathbb{E}[\varphi(X)] \\ & = \quad \mathbb{E}[\frac{X_{j_1}^{(k)}}{(1-\delta)\mu + X_{j_1}^{(k)}}].\end{aligned}$$

Now consider the function xe^{-x}. It is easy to verify that the function decreases when $x \geq 1$, therefore $e^{-x} \leq \frac{1}{ex}$ for $x \geq 1$, hence, for $\frac{\delta^2\mu}{2} \geq 1$, we have

$$\begin{aligned}\mathbb{E}[\Gamma_1] \quad \leq \quad & \frac{\mathbb{E}[X_{j_1}^{(k)}]}{(1-\delta)\mu + \mathbb{E}[X_{j_1}^{(k)}]} + e^{\frac{-\delta^2\mu}{2}} \cdot \pi_{j_1}^{(k)} \\ \leq \quad & \frac{\pi_{j_1}^{(k)}}{(1-\delta)\mu + \pi_{j_1}^{(k)}} + \frac{2}{e\delta^2\mu} \cdot \pi_{j_1}^{(k)}.\end{aligned}$$

Using $\mu \geq 12\Delta - \pi_{j_1}^{(k)} \geq 11\Delta$ and taking $\delta = 1/2$, we have

$$\mathbb{E}[\Gamma_1] \leq \frac{\pi_{j_1}^{(k)}}{6\Delta} + \frac{2\pi_{j_1}^{(k)}}{e \cdot 1/4 \cdot 11\Delta} < \frac{\pi_{j_1}^{(k)}}{2\Delta}. \qquad \square$$

Claim 2. $\mathbb{E}[\tilde{\Gamma}_2 - \Gamma_2] \geq \pi_{j_1}^{(k)}/2$.

Proof. Let $Y = \sum_{i \neq k} X_{j_2}^{(i)}$. Then

$$\begin{aligned}
\Pr(Y = 0) = \Pr(X_{j_2}^{(i)} = 0, \forall i \neq k) &= \prod_{i \neq k} (1 - \pi_{j_2}^{i}) \\
&\geq 1 - \sum_{i \neq k} \pi_{j_2}^{(i)} \\
&\geq 1 - (1/2 - \pi_{j_2}^{(k)}) \\
&= 1/2 + \pi_{j_2}^{(k)}.
\end{aligned}$$

Note that $X_{j_2}^{(k)}$ and Y are independent, hence we have the following,

$$\begin{aligned}
& \mathbb{E}[\tilde{\Gamma}_2 - \Gamma_2] \\
\geq\ & \mathbb{E}[\tilde{\Gamma}_2 - \Gamma_2 | Y = 0, \tilde{X}_{j_2}^{(k)} > 0] \Pr(Y = 0, \tilde{X}_{j_2}^{(k)} > 0) \\
\geq\ & (\pi_{j_1}^{(k)} + \pi_{j_2}^{(k)})(1/2 + \pi_{j_2}^{(k)}) \cdot \mathbb{E}[\tilde{\Gamma}_2 - \Gamma_2 | Y = 0, \tilde{X}_{j_2}^{(k)} > 0].
\end{aligned}$$

Further notice that $\tilde{X}_{j_2}^{(k)}$ and $X_{j_2}^{(k)}$ are independent, thus

$$\begin{aligned}
& \mathbb{E}[\tilde{\Gamma}_2 - \Gamma_2 | Y = 0, \tilde{X}_{j_2}^{(k)} > 0] \\
=\ & \mathbb{E}[\tilde{\Gamma}_2 - \Gamma_2 | Y = 0, \tilde{X}_{j_2}^{(k)} > 0, X_{j_2}^{(k)} = 0] \\
& \cdot \Pr(X_{j_2}^{(k)} = 0 | Y = 0, \tilde{X}_{j_2}^{(k)} > 0) \\
& + \mathbb{E}[\tilde{\Gamma}_2 - \Gamma_2 | Y = 0, \tilde{X}_{j_2}^{(k)} > 0, X_{j_2}^{(k)} > 0] \\
& \cdot \Pr(X_{j_2}^{(k)} > 0 | Y = 0, \tilde{X}_{j_2}^{(k)} > 0) \\
=\ & \mathbb{E}[\frac{u_k \tilde{X}_{j_2}^{(k)}}{Y + u_k \tilde{X}_{j_2}^{(k)}} | Y = 0, \tilde{X}_{j_2}^{(k)} > 0, X_{j_2}^{(k)} = 0] \cdot \Pr(X_{j_2}^{(k)} = 0) \\
& + \mathbb{E}[\frac{u_k \tilde{X}_{j_2}^{(k)}}{Y + u_k \tilde{X}_{j_2}^{(k)}} - \frac{u_k X_{j_2}^{(k)}}{Y + u_k X_{j_2}^{(k)}} | Y = 0, \tilde{X}_{j_2}^{(k)} > 0, X_{j_2}^{(k)} > 0] \\
& \cdot \Pr(X_{j_2}^{(k)} > 0) \\
=\ & 1 - \pi_{j_2}^{(k)}
\end{aligned}$$

Now we have

$$\mathbb{E}[\tilde{\Gamma}_2 - \Gamma_2] \geq (\pi_{j_1}^{(k)} + \pi_{j_2}^{(k)})(1/2 + \pi_{j_2}^{(k)})(1 - \pi_{j_2}^{(k)})$$

For simplicity, let $\tau = \pi_{j_1}^{(k)} + \pi_{j_2}^{(k)} \in [0,1]$ and $x = \pi_{j_2}^{(k)}$, we have

$$f(x) = \mathbb{E}[\tilde{\Gamma}_2 - \Gamma_2] - 1/2 \cdot \pi_{j_1}^{(k)} = \tau(1/2 + x)(1 - x) - 1/2 \cdot (\tau - x).$$

Notice that f is a quadratic function in x whose quadratic term has negative coefficient, therefore if $f(0) \geq 0$ and $f(1) \geq 0$, then for any $x \in [0,1]$ we have $f(x) \geq 0$. It is easy to see that $f(0) = 0$, and $f(1) = 1/2 \cdot (1-\tau) \geq 0$, hence, the claim is proved. □

Given the two claims and the fact that $p_{j_2} \leq \Delta p_{j_1}$, $p_{j_2}\mathbb{E}[\tilde{\Gamma}_2 - \Gamma_2] > p_{j_1}\mathbb{E}[\Gamma_1]$ follows and the lemma is proved. The proof of the two claims is quite involved. □

Lemma 1 shows that: Either no transaction has received a total amount of probability that is larger than 12Δ, or every transaction receives a total amount of probability at least $1/2$. Note that the two cases are not mutually exclusive. Nevertheless, we show in the following that in both cases, miners will select sufficiently many transactions with very high probability. The proofs of the following lemmas are mathematically involved.

Lemma 2. *If $\sum_{i=1}^{m} \pi_j^{(i)} < 12\Delta$ holds for every transaction j, and $\Delta \leq m/12$, then the probability that only $k \leq \frac{m}{12e\Delta} = \Theta(m/\Delta)$ different transactions are selected by m miners is at most $(1/e)^{\Theta(m)}$.*

Towards the proof of Lemma 2, we need the following Lemma 3.

Lemma 3. *Let $k, 1/\delta \in \mathbb{N}_+$ such that $k\delta = \zeta \in (0,1)$. Let $a_1, a_2, \cdots, a_n$ be n numbers such that $a_i \in [0,\delta]$, $\sum_{i=1}^{n} a_i = 1$. Let $\mathscr{A} = \{S | S \subseteq \{1,2,\cdots,n\}, |S| = k\}$. Then we have*

$$\sum_{S \in \mathscr{A}} (\sum_{j \in S} a_j)^m \leq e^k \zeta^{m-k}.$$

Proof. Without loss of generality we assume that $a_1 \geq a_2 \geq \cdots a_n$. We define

$$f(a_1, a_2, \cdots, a_n) = \sum_{S \in \mathscr{A}} (\sum_{j \in S} a_j)^m.$$

For arbitrary $i < j$ and some small $\varepsilon \geq 0$ such that $a_{i-1} \geq a_i + \varepsilon$, $a_j - \varepsilon \geq a_{j+1}$, we prove in the following that

$$f(\cdots, a_{i-1}, a_i + \varepsilon, a_{i+1}, \cdots, a_{j-1}, a_j - \varepsilon, a_{j+1}, \cdots) \geq f(a_1, a_2, \cdots, a_n)$$

For simplicity we write $f(a_1, a_2, \cdots, a_n) = f_{i,j}(a_i, a_j)$. We have

$$\begin{aligned} & f_{i,j}(a_i + \varepsilon, a_j - \varepsilon) \\ = & \sum_{\substack{S: S \in \mathscr{A} \\ i,j \notin S}} (\sum_{h \in S} a_h)^m + \sum_{\substack{S: S \in \mathscr{A} \\ i \in S, j \notin S}} (a_i + \varepsilon + \sum_{h \in S, h \neq i} a_h)^m \\ & + \sum_{\substack{S: S \in \mathscr{A} \\ i \notin S, j \in S}} (a_j - \varepsilon + \sum_{h \in S, h \neq j} a_h)^m + \sum_{\substack{S: S \in \mathscr{A} \\ i,j \in S}} (\sum_{h \in S} a_h)^m \end{aligned}$$

Consider the function $g(x) = x^m$. Due to its convexity, for $x + \varepsilon > x > y > y - \varepsilon$, we have

$$g(x+\varepsilon) + g(y-\varepsilon) \geq g(x) + g(y).$$

Hence, let $\mathscr{A}' = \{S | S \subseteq \{1,2,\cdots,n\} \setminus \{i,j\}, |S| = k-1\}$, then for any $S' \in \mathscr{A}'$, we have

$$\begin{aligned} & (\sum_{h \in S'} a_h + a_i + \varepsilon)^m + (\sum_{h \in S'} a_h + a_j - \varepsilon)^m \\ \geq & (\sum_{h \in S'} a_h + a_i)^m + (\sum_{h \in S'} a_h + a_j)^m, \end{aligned}$$

therefore, $f_{i,j}(a_i + \varepsilon, a_j - \varepsilon) \geq f_{i,j}(a_i, a_j)$. Now we can iteratively change $a_1, a_2, \cdots, a_n$ into $a'_1, a'_2, \cdots, a'_n$ such that $a'_i = \delta$ for $1 \leq i \leq 1/\delta$, and $a'_i = 0$ otherwise, and get

$$f(a_1, a_2, \cdots, a_n) \leq f(a'_1, a'_2, \cdots, a'_n).$$

It is not difficult to compute that

$$f(a_1', a_2', \cdots, a_n') = \binom{1/\delta}{k}(k\delta)^m \leq \frac{(1/\delta)^k}{k!} \cdot \zeta^m \leq (\frac{e}{\zeta})^k \zeta^m,$$

where for the last inequality we make use of the Stirling's approximation that $k! \geq (k/e)^k$. Hence, the lemma is proved. □

Now we come to the proof of Lemma 2.

Proof. Consider the event that at most k different transactions are chosen by m miners. Let $\mathscr{A} = \{S | S \subseteq \{1, 2, \cdots, n\}, |S| = k\}$ be the superset of all subsets of cardinality k. For any $S \in \mathscr{A}$, let σ_S be any mapping that maps $\{1, 2, \cdots, m\}$ to S, and Ω_S be the set of all such mappings. Consider the event that miner i selects transaction $\sigma_S(i)$, we know that the event happens with the probability $\Pr(\sigma_S) = \prod_{i=1}^m \pi_{\sigma_S(i)}^{(i)}$. Taking summation over all possible mappings, the event that only transactions in S are selected is at most

$$\sum_{\sigma_S \in \Omega_S} \Pr(\sigma_S) = \sum_{\sigma_S \in \Omega_S} \prod_{i=1}^m \pi_{\sigma_S(i)}^{(i)} = \prod_{i=1}^m \sum_{j \in S} \pi_j^{(i)}.$$

The last equality follows as when we expand $\prod_{i=1}^m \sum_{j \in S} \pi_j^{(i)}$, any summand corresponds to $\prod_{i=1}^m \pi_{\sigma(i)}^{(i)}$ for some mapping σ. Using the the inequality of arithmetic and geometric means, we have

$$\sum_{\sigma_S \in \Omega_S} \Pr(\sigma_S) = \prod_{i=1}^m \sum_{j \in S} \pi_j^{(i)} \leq (\frac{\sum_{i=1}^m \sum_{j \in S} \pi_j^{(i)}}{m})^m = (\sum_{j \in S} \frac{\sum_{i=1}^m \pi_j^{(i)}}{m})^m.$$

Taking summation over all possible $S \in \mathscr{A}$, the event that at most k transactions are selected is at most

$$\sum_{S \in \mathscr{A}} \sum_{\sigma_S \in \Omega_S} \Pr(\sigma_S) \leq \sum_{S \in \mathscr{A}} (\sum_{j \in S} \frac{\sum_{i=1}^m \pi_j^{(i)}}{m})^m.$$

Now let $a_j = \frac{\sum_{i=1}^m \pi_j^{(i)}}{m}$. It is easy to see that $\sum_{j=1}^n a_j = 1$. Furthermore, since $\sum_{i=1}^m \pi_j^{(i)} < 12\Delta$, we have $a_j \leq 12\Delta/m$. We apply Lemma 3 by taking $\delta = \frac{1}{\lceil m/(12\Delta) \rceil} \leq 12\Delta/(m + 12\Delta)$, $\zeta = k\delta \leq e^{-1}$, we know that

$$\sum_{S \in \mathscr{A}} \sum_{\sigma_S \in \Omega_S} \Pr(\sigma_S) \leq e^k \zeta^{m-k} \leq e^{-m/2}.$$

Hence, the lemma is proved. □

Note that if $\Delta > m/12$, $\frac{m}{12e\Delta} < 1$. As miners complete at least 1 transaction, the lemma is trivially true.Now we consider the other case and have the following.

Lemma 4. *If $\sum_{i=1}^m \pi_j^{(i)} \geq 1/2$ holds for every transaction j, then the probability that no more than n/e^2 transactions are selected is at most $(1/e)^{\Theta(n)}$.*

Proof. Consider the event that at most θn transactions are selected by m miners for some $\theta \in (0, 1)$. Note that in this case $m = \sum_{j=1}^n \sum_{i=1}^m \pi_j^{(i)} \geq n/2$, hence $n \leq 2m$. Again we define $\mathscr{A} = \{S | S \subseteq \{1, 2, \cdots, n\}, |S| = (1-\theta)n\}$ as the superset of all the subsets of cardinality $(1-\theta n)$. The probability that miner i does *not* select any transaction in some $S \in \mathscr{A}$ is $1 - \sum_{j \in S} \pi_j^{(i)}$. Given that miners select transactions independently, the probability that all miners do not select transactions in S is

$$\begin{aligned} \prod_{i=1}^m (1 - \sum_{j \in S} p_j^{(i)}) &\leq (\frac{m - \sum_{j \in S} \sum_{i=1}^m \pi_j^{(i)}}{m})^m \\ &\leq (\frac{m - \frac{(1-\theta)n}{2}}{m})^m \leq e^{-\frac{(1-\theta)n}{2}}, \end{aligned}$$

where the first inequality follows by inequality of arithmetic and geometric means, the second inequality follows by the fact that $\sum_{i=1}^{m} \pi_j^{(i)} \geq 1/2$, and the third inequality follows by $(1-1/x)^x \leq e^{-1}$ for $x \geq 1$. Taking the summation over all possible $S \in \mathscr{A}$, the probability that at most θn transactions are selected is at most

$$\begin{aligned}
\sum_{S \in \mathscr{A}} \prod_{i=1}^{m} (1 - \sum_{j \in S} p_j^{(i)}) &\leq \binom{n}{\theta n} e^{-\frac{(1-\theta)n}{2}} \\
&\leq \frac{n^{\theta n}}{(\theta n)!} \cdot e^{-\frac{(1-\theta)n}{2}} \\
&\leq (\frac{e}{\theta})^{\theta n} \cdot e^{-\frac{(1-\theta)n}{2}}.
\end{aligned}$$

Taking $\theta = e^{-2}$, simple calculation shows that the right side of the inequality above is e^{cn} for $c \leq -0.02$, and the lemma is proved. □

Given Lemma 2 and Lemma 4, Theorem 2 follows directly.

5.2.2 Finality-duration

For ease of presentation, we let $s = \min\{c_1 m/\Delta, c_2 n\}$ (recall Theorem 2). We will characterize finality-duration in terms of s. Recall that a miner needs to make two decisions: (i) which transaction to include in the new block, and (ii) which two previous blocks to refer to. The two decisions are independent. In the previous subsection we have discussed (i), and in this subsection we focus on (ii), as this affects how the DAG grows.

It should be clear that since the verification reward of a transaction (block) is evenly distributed among miners who append a block of the same distance to it, a miner always prefers a block with no descendants. At any particular time t, we call a block without descendants as a leaf at t, and denote by L_t the set of leaves. We are interested in the size of L_t. Notice that in the classical blockchain system, $|L_t|$ is 1 since it is a chain. However, in a non-linear model, $|L_t|$ is not necessarily 1. Principally, $|L_t|$ could grow arbitrarily large, but what we will show in this section is that, $|L_t|$ is always bounded when miners are using their equilibrium strategies. In this case, although we are considering a non-linear model, it is "almost linear", as implied by Theorem 3. Based on this result, we further leverage the techniques from random walk to prove that, for every block, after a delay of $O(s \log s)$ units of time, all blocks will be its descendant (Theorem 4), consequently, if we set $\ell = \Theta(s \log s)$ in our design, every block will be verified by all users, and security follows.

As we mentioned before, each new block will refer to two leaves in L_t. As every block offers the same total amount of verification reward, every leaf appears the same to the miners (unless they are in conflict with previous blocks and then miners will be biased based on the LWD rule). Therefore, a new block will randomly select two leaves to refer to. Assuming leaves are not conflicting with previous blocks, we show that $|L_t|$ will be $O(s)$ in the long run with an extremely high probability. First, it is easy to see that if $|L_t| \leq s$, then $L_{t+1} \geq s$ as the s new blocks will be leaves at $t+1$. The following lemma shows that if $|L_t|$ is sufficiently large, then with very high probability it will reduce to $O(s)$ after enough time.

Lemma 5. *Let ε be an arbitary small constant. If $|L_t| \geq 1/\varepsilon^3$ and $|L_t| \geq 4s$, then with sufficiently high probability (at least $1 - O(\varepsilon)$), $|L_{t+1}| = |L_t| - X + s \leq |L_t| - \frac{(1-3\varepsilon)s}{2}$, i.e., L_t decreases by at least $\Omega(s)$.*

Proof. Consider an arbitrary L_t. For any $\tau_i \in L_t$ and any new block b, the probability that b refers to τ_i is $2/|L_t|$, hence, the probability that none of the s new blocks refer to τ_i is $(1-2/|L_t|)^s$, i.e., the probability that an arbitrary τ_i has descendant(s) at $t+1$ is $\theta = 1-(1-2/|L_t|)^s$. Let a random variable X_i denote whether the event that τ_i has descendant(s) at $t+1$, then $\Pr(X_i = 1) = \theta$ and $\Pr(X_i = 0) = 1-\theta$. Denote

by $X = \sum_{i=1}^{|L_t|} X_i$ the total number of leaves in L_t that has descendant(s) at $t+1$. According to Chernoff bound, with the probability of at least $1-2e^{-\varepsilon^2|\mathbb{E}X|/3}$, $(1-\varepsilon)\mathbb{E}X \leq X \leq (1+\varepsilon)\mathbb{E}X$. Now we estimate $\mathbb{E}X$ by verifying

$$2s - \frac{2s(s-1)}{|L_t|} \leq \mathbb{E}X = |L_t| - |L_t|(1-\frac{2}{|L_t|})^s \leq 2s.$$

Hence, if $|L_t| \geq 1/\varepsilon^3$ and $|L_t| \geq 4s$, then (at least $1-2e^{-1/(3\varepsilon)} = 1-O(\varepsilon)$), $X \geq (1-\varepsilon)\mathbb{E}X \geq 3/2 \cdot (1-\varepsilon)s$, consequently,

$$|L_{t+1}| = |L_t| - X + s \leq |L_t| - \frac{(1-3\varepsilon)s}{2}. \qquad \square$$

The above lemma shows that if $|L_t|$ is large, then with high probability $|L_t|$ shall decrease, however, what we are interested in is the probability that $|L_t| \leq O(s)$ for all $t \geq 0$. Towards this, we need to cast the problem as a *random walk*. Lemma 5 shows that with the probability of $(1-O(\varepsilon))^3 = 1-O(\varepsilon)$, $|L_t|$ can decrease by $\frac{3(1-3\varepsilon)s}{2} \geq s$, while with probability of at most $O(\varepsilon)$, $|L_t|$ can increase by at most s. This can be interpreted as a random walk which walks right (increase) by s steps with the probability of $1-O(\varepsilon)$, and walks left (decrease) by s steps with the probability of $O(\varepsilon)$. The following lemma is proved for a general random walk.

Lemma 6 ([7], pp.272)**.** *Consider a random walk starting at $RW_0 = 0$, $\Pr(RW_{i+1} - RW_i = s) = p$, $\Pr(RW_{i+1} - RW_i = -s) = q$ where $p+q = 1$ and $s \in \mathbb{Z}_{>0}$. If $p > q$, then*

$$\lim_{n\to\infty} \Pr(RW_i \geq 0, \forall 1 \leq i \leq n) = \frac{p-q}{p}.$$

If $p < q$, the above limit is 0.

Now we are ready to prove the following theorem.

Theorem 3. *Let ε be a small constant such that $s > 1/\varepsilon^3$. With very high probability (at least $1-O(\varepsilon)$), $|L_t| \leq 5s$ for all $t \geq 0$.*

Proof. Recall that $|L_0| = 0$. Let t^* be the smallest time where $|L_{t^*}| \geq 4s$, then $|L_{t^*}| \leq 5s$. Now we take t^* as a starting time, $|L_{t^*}|$ as a starting point and take the random walk interpretation. Using Lemma 6, we have that

$$\begin{aligned} \lim_{n\to\infty} \Pr(|L_t| \leq |L_{t^*}|, \forall 1 \leq t \leq n) &\leq \frac{1-O(\varepsilon)-O(\varepsilon)}{1-O(\varepsilon)} \\ &= 1-O(\varepsilon). \end{aligned}$$

Therefore, the probability that $|L_t|$ is bounded by $5s$ for all $t \geq 0$ is at least $1-O(\varepsilon)$. $\square$

Lemma 7. *Let ε be a small constant such that $s > 1/\varepsilon^3$. For any transaction at t that is not in conflict with prior transactions, with sufficiently high probability (at least $1-O(\varepsilon)$) every block appended at or after $t + O(s\log s)$ will be its descendant.*

Proof. According to Theorem 3, we focus on the event that $|L_t| \leq 5s$ for all $t \geq 0$, which happens with $1-O(\varepsilon)$ probability.

For $h \geq t$, let Ψ_h be the subset of blocks in L_h which has a directed path from some fixed block $\tau_0 \in L_t$, which is a random subset. Let $\psi_h = \mathbb{E}(|\Psi_h|)$. Consider L_{h+1}. For any block $\tau_i \in L_{h+1}$, let X_i be

a binary random variable indicating whether τ_i refers to some block in Ψ_h, and hence admits a directed path from τ_0. Then we know

$$\begin{aligned}\Pr(X_i = 1) &= \frac{\binom{|\Psi_h|}{2} + |\Psi_h|(|L_h| - |\Psi_h|)}{\binom{|L_h|}{2}} \\ &= \frac{|\Psi_h|(2|L_h| - |\Psi_h| - 1)}{|L_h|(|L_h| - 1)}.\end{aligned}$$

We consider $|\Psi_{h+1}|$. It is obvious that if $|\Psi_h| = |L_h|$, then every block in L_{h+1} refers to some block in Ψ_h and thus admits a directed path from τ_0, hence, $|L_{h+1}| = |\Psi_{h+1}|$, and similarly we have $|L_{h+j}| = |\Psi_{h+j}|$ for all $j \geq 1$. Otherwise, we assume $1 \leq |\Psi_h| \leq |L_h| - 1$. Then $2|L_h| - |\Psi_h| - 1 \geq |L_h|$, and we have

$$\mathbb{E}(X_i) = \mathbb{E}\left(\frac{|\Psi_h|(2|L_h| - |\Psi_h| - 1)}{|L_h|(|L_h| - 1)}\right) \geq \frac{\psi_h}{|L_h| - 1}.$$

Note that $|\Psi_{h+1}| = \sum_i X_i$. It is easy to calculate that

$$\psi_{h+1} = \mathbb{E}(|\Psi_{h+1}|) \geq \psi_h \left(1 + \frac{1}{|L_h| - 1}\right).$$

This means, starting from $\psi_t = 1$, for each ψ_h where $h \geq t$, either $\psi_h = |L_h|$ and thus $\psi_{h'} = |L_{h'}|$ for all $h' \geq h$, or $\psi_{h+1} \geq \left(1 + \frac{1}{|L_h|-1}\right)\psi_h$. Since $|L_h| \leq 5s$, ψ_h increases sufficiently close to $|L_h| \leq 5s$ when $h \geq t + O(s \log s)$, and the theorem is proved. □

Given the above lemma, if we set ℓ, the verification depth to be $\ell \geq O(s \log s)$, then any transaction at t will be verified by all the users after $O(s \log s)$ units of time with high probability. The following theorem is thus true.

Theorem 4. *If $s > 1/\varepsilon^3$ and $\ell \geq O(s \log s)$, then with probability of at least $1 - O(\varepsilon)$, any transaction at t will be verified by all the users after $O(s \log s)$ units of time.*

Remark. Recall that the scalability of the system increases as Δ increases, while $s = \min\{c_1 m/\Delta, c_2 n\}$, and hence the finality-duration $O(s \log s)$ decreases as Δ increases. Theorem 4 shows trade-off between the scalability and finality-duration.

6 Conclusion

We provide the first systematic analysis on blockchain systems with respect to three major parameters, verification, scalability, and finality-duration. We establish an impossibility result showing no blockchain system can simultaneously achieve the three properties. We complement the existing blockchain systems by establishing the first NLB that achieves both full verification and scalability. We also reveal, for the first time, the trade-off between scalability and finality-duration in NLB. It is not clear whether a better trade-off exists or not.

References

[1] M. S. Ali, M. Vecchio, M. Pincheira, K. Dolui, F. Antonelli, and M. H. Rehmani. Applications of blockchains in the internet of things: A comprehensive survey. *IEEE Communications Surveys Tutorials*, 21(2):1676–1717, 2019.

[2] S. Azouvi, P. McCorry, and S. Meiklejohn. Betting on blockchain consensus with fantomette. *CoRR*, abs/1805.06786, 2018.

[3] S. Bano, A. Sonnino, M. Al-Bassam, S. Azouvi, P. McCorry, S. Meiklejohn, and G. Danezis. Sok: Consensus in the age of blockchains. In *Proceedings of the 1st ACM Conference on Advances in Financial Technologies, AFT 2019*, pages 183–198, Zurich, 2019. ACM.

[4] X. Boyen, C. Carr, and T. Haines. Blockchain-free cryptocurrencies. 2017.

[5] D. Chaum. Blind signatures for untraceable payments. In *Advances in Cryptology: Proceedings of CRYPTO '82*, pages 199–203, California, 1982.

[6] M. Conti, S. K. E, C. Lal, and S. Ruj. A survey on security and privacy issues of bitcoin. *IEEE Communications Surveys Tutorials*, 20(4):3416–3452, 2018.

[7] W. Feller. *An introduction to probability theory and its applications: volume I*, volume 3. John Wiley & Sons, New York, 3rd edition, 1968.

[8] B. França. Homomorphic mini-blockchain scheme, 2015.

[9] J. A. Garay and A. Kiayias. Sok: A consensus taxonomy in the blockchain era. In *Topics in Cryptology - CT-RSA 2020 - The Cryptographers' Track at the RSA Conference 2020, Proceedings*, volume 12006 of *Lecture Notes in Computer Science*, pages 284–318, San Francisco, 2020. Springer.

[10] M. C. K. Khalilov and A. Levi. A survey on anonymity and privacy in bitcoin-like digital cash systems. *IEEE Communications Surveys Tutorials*, 20(3):2543–2585, 2018.

[11] Z. Liu, N. C. Luong, W. Wang, D. Niyato, P. Wang, Y. Liang, and D. I. Kim. A survey on applications of game theory in blockchain. *CoRR*, abs/1902.10865, 2019.

[12] I. Miers, C. Garman, M. Green, and A. D. Rubin. Zerocoin: Anonymous distributed e-cash from bitcoin. In *2013 IEEE Symposium on Security and Privacy, SP 2013*, pages 397–411, California, 2013. IEEE Computer Society.

[13] S. Nakamoto. Bitcoin: A peer-to-peer electronic cash system, 2008.

[14] J. Nash. Non-cooperative games. *Annals of mathematics*, pages 286–295, 1951.

[15] S. Popov. The tangle. *cit. on*, page 131, 2016.

[16] S. Popov, O. Saa, and P. Finardi. Equilibria in the tangle. *Computers & Industrial Engineering*, 136:160–172, 2019.

[17] T. Rocket. Snowflake to avalanche: A novel metastable consensus protocol family for cryptocurrencies. 2018.

[18] T. Salman, M. Zolanvari, A. Erbad, R. Jain, and M. Samaka. Security services using blockchains: A state of the art survey. *IEEE Communications Surveys Tutorials*, 21(1):858–880, 2019.

[19] T. Sander and A. Ta-Shma. Auditable, anonymous electronic cash extended abstract. In *Advances in Cryptology - CRYPTO '99, 19th Annual International Cryptology Conference, Proceedings*, volume 1666 of *Lecture Notes in Computer Science*, pages 555–572, California, 1999. Springer.

[20] E. B. Sasson, A. Chiesa, C. Garman, M. Green, I. Miers, E. Tromer, and M. Virza. Zerocash: Decentralized anonymous payments from bitcoin. In *2014 IEEE Symposium on Security and Privacy, SP 2014*, pages 459–474, California, 2014. IEEE.

[21] Y. Sompolinsky, Y. Lewenberg, and A. Zohar. SPECTRE: A fast and scalable cryptocurrency protocol. *IACR Cryptology ePrint Archive*, 2016:1159, 2016.

[22] Y. Sompolinsky and A. Zohar. Secure high-rate transaction processing in bitcoin. In *Financial Cryptography and Data Security - 19th International Conference, FC 2015, , Revised Selected Papers*, volume 8975 of *Lecture Notes in Computer Science*, pages 507–527, Puerto Rico, 2015. Springer.

[23] Y. Sompolinsky and A. Zohar. Phantom, ghostdag, 2020.

[24] The Ethereum Team. On sharding blockchains, 2019.

[25] The Harmony Team. Harmony - technical whitepaper, 2018.

[26] The Zilliqa Team. The zilliqa technical whitepaper, 2017.

[27] N. P. Triantafyllidis and T. Oskar van Deventer. Developing an ethereum blockchain application. 2016.

[28] F. Tschorsch and B. Scheuermann. Bitcoin and beyond: A technical survey on decentralized digital currencies. *IEEE Communications Surveys & Tutorials*, 18(3):2084–2123, 2016.

[29] M. Vukolic. The quest for scalable blockchain fabric: Proof-of-work vs. BFT replication. In *Open Problems in Network Security - IFIP WG 11.4 International Workshop, iNetSec 2015, Revised Selected Papers*, volume 9591 of *Lecture Notes in Computer Science*, pages 112–125, Zurich, 2015. Springer.

[30] L. Xu, L. Chen, Z. Gao, S. Xu, and W. Shi. EPBC: efficient public blockchain client for lightweight users. In *Proceedings of the 1st Workshop on Scalable and Resilient Infrastructures for Distributed Ledgers, SERIAL@Middleware 2017*, pages 1:1–1:6, Nevada, 2017. ACM.

[31] R. Yang, F. R. Yu, P. Si, Z. Yang, and Y. Zhang. Integrated blockchain and edge computing systems: A survey, some research issues and challenges. *IEEE Communions Surveys & Tutorials*, 21(2):1508–1532, 2019.