Profitable Manipulations of Cryptographic Self-Selection Are Statistically Detectable

Linda Cai
Princeton University, NJ, USA

Jingyi Liu
Princeton University, NJ, USA

S. Matthew Weinberg
Princeton University, NJ, USA

Chenghan Zhou
Stanford University, Palo Alto, CA, USA

Abstract

Cryptographic Self-Selection is a common primitive underlying leader-selection for Proof-of-Stake blockchain protocols. The concept was first popularized in Algorand [7], who also observed that the protocol might be manipulable. [11] provide a concrete manipulation that is strictly profitable for a staker of any size (and also prove upper bounds on the gains from manipulation).

Separately, [3, 23] initiate the study of *undetectable* profitable manipulations of consensus protocols with a focus on the seminal Selfish Mining strategy [9] for Bitcoin's Proof-of-Work longest-chain protocol. They design a Selfish Mining variant that, for sufficiently large miners, is strictly profitable yet also indistinguishable to an onlooker from routine latency (that is, a sufficiently large profit-maximizing miner could use their strategy to strictly profit over being honest in a way that still appears to the rest of the network as though everyone is honest but experiencing mildly higher latency. This avoids any risk of negatively impacting the value of the underlying cryptocurrency due to attack detection).

We investigate the detectability of profitable manipulations of the canonical cryptographic self-selection leader selection protocol introduced in [7] and studied in [11], and establish that for any player with $\alpha < \frac{3-\sqrt{5}}{2} \approx 0.38$ fraction of the total stake, *every strictly profitable manipulation is statistically detectable*. Specifically, we consider an onlooker who sees only the random seed of each round (and does not need to see any other broadcasts by any other players). We show that the distribution of the sequence of random seeds when any player is profitably manipulating the protocol is inconsistent with any distribution that could arise by honest stakers being offline or timing out (for a natural stylized model of honest timeouts).

2012 ACM Subject Classification Theory of computation → Algorithmic game theory and mechanism design; Applied computing → Digital cash

Keywords and phrases Blockchain, Cryptocurrency, Proof-of-Stake, Strategic Mining, Statistical Detection

Digital Object Identifier 10.4230/LIPIcs.AFT.2024.30

Related Version *Full Version*: `https://arxiv.org/abs/2407.16949` [5]

Funding *Linda Cai*: Supported by a Chainlink Fellowship, and a Ripple UBRI grant.
Jingyi Liu: Supported by a Ripple UBRI grant.
S. Matthew Weinberg: Supported by a Ripple UBRI grant, and NSF CAREER CCF-1942497.
Chenghan Zhou: Supported by a Ripple UBRI grant.

1 Introduction

Since Nakamoto introduced Bitcoin in 2008, blockchain technology has made a significant impact on digital transactions by establishing a decentralized system in which transactions are validated through consensus among peers, rather than by a central authority. This innovation, while popularizing decentralized currencies, has also brought to light substantial challenges, particularly the extensive computational and energy demands of its proof-of-work (PoW) consensus mechanism. Notably, the energy consumption associated with Bitcoin mining exceeds that of many countries, raising significant environmental concerns. Furthermore, the necessity for large-scale mining hardware introduces considerable centralization risks to cryptocurrencies [2], many of which are inherently designed to be decentralized.

In response to these challenges, the blockchain community has been exploring Proof of Stake (PoS), which has been implemented in many prominent crypto-currencies (e.g. Ethereum, Algorand, Cardano). In each round, PoS selects block leaders (who get to propose a block to be included) based on the stake, reducing energy usage and aiming to prevent Sybil attacks by randomly assigning leadership chances proportionally to coin holdings. However, the leader selection process in PoS presents additional challenges. For example, the pseudorandomness resulting from PoW is in some sense "external" to the blockchain (the next miner is selected proportionally to their computational power, independently, and nothing in the blockchain itself can influence this). Replicating this property in PoS blockchains has proved challenging without trusting an external randomness beacon (which is often a non-starter in blockchain applications, whose entire purpose is to remove the need for such trust).[1] On the other hand, pseudorandom numbers generated using the blockchain itself can often be predicted by the miners, opening up the possibility of profitable deviations [4].

One promising idea in addressing the leader selection challenge is cryptographic self-selection, initially proposed by Algorand [7]. Cryptographic self-selection is a protocol to select a block-proposer for round $r + 1$ as a function of communication during round r. We overview Algorand's canonical proposal shortly, and briefly note here that it is known to admit profitable deviations for arbitrarily small participants [11].[2] We subsequently discuss cryptographic self-selection in further detail, but at this point merely wish to note that: (a) nonmanipulable randomness sources are a major open problem within the blockchain community, due to applications for PoS, (b) these problems are important to both researchers [4, 7, 12, 11] and practitioners [1] [3], and (c) the particular approach initially proposed in [7] is a canonical testbed due to its elegance and simplicity (which we overview shortly).

Separately, recent work of [3, 23] propose a novel concern for profitable manipulations: detectability. Specifically, while it may be challenging to *trace* a strategic manipulation to a particular actor in a permissionless system,[4] profitable manipulations would likely be *detect-*

[1] To slightly elaborate on this point: trusting a centralized external randomness beacon (such as NIST) is certainly a non-starter, because NIST then has control over the block producers. One could instead have an external distributed process to generate random numbers independent of this blockchain. But if this blockchain has monetary value, then securely implementing that distributed process is its own challenge. The story is getting more subtle with Verifiable Delay Functions that might act as a cryptographic external randomness beacon, although their security assumptions are hardware-based and not as battle-tested as standard cryptography, so there will always be a desire for solutions based on standard cryptography.

[2] This is in contrast to block-witholding manipulations in PoW longest-chain protocols [21, 16, 9], although alternate strategic manipulations of some PoW protocols are profitable for arbitrarily small miners [13, 15, 23].

[3] For example, this blog post by the Ethereum foundation on manipulating its RanDAO: link.

[4] This is not to say that tracing strategic manipulations is impossible – indeed, law enforcement regularly

able. This observation serves as a basis to mitigate concerns with profitable manipulations in practice – perhaps the manipulator will earn moderate additional cryptocurrency via manipulation, but its detection may cause the value of these tokens to tank when measured in USD. Their work highlights that detectability of strategic manipulations plays a significant role in their usability in practice – undetectable deviations avoid the risk of devaluing the underlying cryptocurrency, while detectable ones can be disincentivized through outside-the-model means.

Our paper lies at the intersection of these two agendas: we investigate the detectability of profitable manipulations in cryptographic self-selection. Surprisingly, our main result finds that for any participant with less than $\frac{3-\sqrt{5}}{2} \approx 0.38$ fraction of the total stake, *all profitable manipulations of Algorand's canonical cryptographic self-selection protocol are statistically detectable*.

We now provide additional context and details for our result.

Leader Selection in PoS Blockchains. PoS consensus protocols typically take one of two forms: they may be a *longest-chain protocol*, or a *Byzantine Fault Tolerant (BFT)-based protocol*. Both formats are well-represented in practice, and Ethereum is in some sense a hybrid of the two. In a longest-chain protocol, strategic manipulations are more straightforward – they typically come by inducing forks, and causing the attacker to have their own blocks represent a greater fraction of blocks in the longest chain [9, 21, 16, 4, 12]. Manipulations of BFT-based protocols are more subtle. BFT-based protocols proceed one block at a time, reaching a strong consensus on block r and finalizing it forever before getting to work on block $r+1$. As such, these protocols are nonmanipulable at the per-block level (unless the attacker has sufficient stake to cause significantly more damage by violating consensus entirely). Instead, these protocols typically have a randomly-selected "leader" dictate the contents of the block and the per-round BFT protocol aims to reach consensus on the leader's block. But, these protocols still need an effective method to select a leader for each round independently and proportional to their stake.

Fortunately for mechanism designers, leader selection protocols are often modular components of the broader blockchain protocol, and can be studied in isolation from the (significantly more complex) BFT protocols that handle per-round consensus.

Algorand's Canonical Leader Cryptographic Self-Selection. [7] propose an elegant leader selection protocol, which we describe for simplicity in the case where each account holds the same number of coins (we rigorously overview their protocol in the general case in Section 2, but omit the generalization now in the interest of clarity). First, pick a uniformly random seed, Q_1, for round one. Then in round r, ask each account holder i to first digitally sign Q_r and then hash[5] their digital signature to get a credential $\textsc{Cred}_i^r$. Whoever broadcasts the smallest credential is the leader for round r.

Their protocol has several desirable properties. First, assuming that every player honestly digitally signs and hashes in each round (and that the hash function behaves like a random oracle), the leader in each round is indeed a uniformly random coin, independent of all previous rounds. Second, it is not predictable too far into the future: because player i cannot

traces attacks in permissionless systems: link.

[5] The formal concept is a Verifiable Random Function, which we define in Section 2.1. Intuitively, the hash is a uniformly random number drawn specifically for player i in a manner that no other player can precompute (because they can't digitally sign on behalf of player i).

digitally sign on behalf of player j, player i has absolutely no idea what seed might result next round (if another player is the leader). Finally, the manner in which it can possibly be manipulated is extremely structured: the only strategies available to a player are to broadcast or not broadcast their credentials.

Still, their protocol is not perfect – [7] already acknowledge that it might be manipulable, and [11] establish a strictly profitable strategy for arbitrarily small players. [11]'s strategy is fairly simple, and we overview it in Section 4.

Detecting Strategic Behavior in Cryptographic Self-Selection. How would one detect that a participant is strategically manipulating a protocol? In PoW longest-chain protocols, [3] propose to look at the pattern of forks – strategic behavior often results in long runs of consecutive forks whereas routine latency instead would result in independently distributed forks. This particular detection method is not applicable to a BFT-based protocol, as BFT-based protocols have no forks once a block is finalized.

In the spirit of [3], we aim to detect strategies using the minimal amount of information possible, and in particular we only use information that is available to anyone following the blockchain. Specifically, anyone following the blockchain *must know who is the leader of round* r and *must know their credential* $\textsc{Cred}_i^r$ that proves they are the leader.

If every participant in the network were honest, and there were n coins in the network, we would expect in a given round that the winning credential is distributed according to the minimum of n independent draws of the Hash function. So across a large number of rounds, an observer could check the sequence of winning credentials and see if they empirically match i.i.d. draws of the minimum of n independent draws of the Hash function.

This is perhaps too strong of an assumption on honest parties, however. In particular, it assumes either that every single coin is online and participating in the protocol or that the observer otherwise knows that exactly (say) k coins are online. An observer instead might know that there exists some number k of online coins participating in the protocol, but not know k. Then, they would expect to see sequences of credentials that empirically match i.i.d. draws from the minimum of k independent draws of the Hash function, for some k.

So consider an attacker who controls multiple accounts. They can selectively refrain from broadcasting in round r (and might benefit from doing so, if another account will win round r anyway and their chosen credential gives them a better shot of winning round $r+1$), but doing so will skew the distribution of round r's credential larger and the distribution of round $r+1$'s credential smaller. This is profitable, but when done naively detectable (we analyze [11]'s particularly simple strategy, which follows from this intuition, in Section 3). The challenge for the attacker is whether it is possible to profit (by biasing their winning credential to be lower in some rounds), without being detectable (by biasing the winning credential in other rounds to be higher). Our main result shows that this is impossible: for any $\alpha < \frac{3-\sqrt{5}}{2} \approx 0.38$,[6] and any participant with an α fraction of the total stake, any strategy that leads a $> \alpha$ fraction of the rounds produces a distribution over sequences of winning credentials that is not consistent with any number of online honest coins.[7]

Finally, one might even consider having a fixed k of online coins to be too stringent of a null hypothesis – perhaps the number of active coins fluctuates from round to round. We also establish that our main result degrades smoothly in the deviation an observer is comfortable

[6] Note, for example, that an adversary with $\alpha > \frac{3-\sqrt{5}}{2} > 1/3$ of the stake could alternatively directly violate the underlying consensus protocol, which would do significantly more damage than a strategic manipulation.

[7] Like most prior work (e.g. [16, 4, 12, 3]), we consider an attacker who does not have excessively strong network connectivity – see Section 3 for the formal setup.

attributing to honest-but-occasionally-offline behavior. If the observer believes the online coins to fluctuate within $1 \pm \delta$ of an unknown baseline, and the true online coins indeed fluctuate within $1 \pm \delta$ of some ground truth baseline, undetectable manipulations lead at most an additional 2δ fraction of rounds.

Roadmap and Discussion. We study the detectability of strategic manipulations in cryptographic self-selection, Algorand's canonical leader selection protocol [7]. We establish that any profitable deviation is detectable, and also quantitatively extend our results to even further relaxed null hypotheses of what might result from honest-but-offline behavior.

Detectability of profitable manipulations is a desirable property of consensus protocols, as it provides an outside-the-model avenue to deter deviant behavior. While [3] derive profitable, undetectable deviations from longest-chain PoW consensus protocols, we instead show that cryptographic self-selection admits no profitable manipulations. Our work now establishes that some canonical protocols admit undetectable profitable deviations while others do not, and further motivates detectability of profitable manipulations as a standard question to be asked of novel consensus protocols.

In Section 1.1, we overview related work in further detail. In Section 2 and Section 3 we overview our model and our statistical detection methods in significantly more detail. Section 4 overviews the profitable strategy of [11] through the lens of detectability in order to familiarize the reader with the techniques. Section 5 formally states and proves our main result and its robust extension.

1.1 Related Work

Detection of Strategic Attacks in Proof-of-Work Protocols. Several methods of detecting selfish mining in proof-of-work protocol have been proposed. [8] presents a heuristic to detect selfish mining based on changes in the height of forks in a blockchain network and their simulation result implies a connection between the presence of selfish mining attack and higher rate of forks, with a mean height of higher than 2.

[18] proposes a statistical test for each miner based on the null hypothesis that under honest mining, the probability of observing two successive blocks mined by the same miner is given by type II binomial distribution of order 2, and the presence of selfish mining will cause deviation from such distribution, causing a higher probability of observing successive blocks mined by the same miner. The authors conduct empirical tests on five cryptocurrencies based on Proof-of-Work – Bitcoin, Litecoin, Ethereum, Monacoin and Bitcoin Cash and claim to be the first research work that reveals the presence of selfish mining in real cryptocurrency systems, although they acknowledge that other reasons can also lead to abnormal successive block discovery rates. We further note that their detection method relies on knowing which addresses or wallets are controlled by the same user, while our detection scheme does not rely on such knowledge.

Other works such as [22] use neural networks that achieve good accuracy of detecting selfish mining on simulated datasets.

In contrast to these detection methods, [3] proves the existence of a statistically undetectable and strictly profitable selfish mining strategy for miners with 38.2% of the total hash rate. Under this strategy, the attacker hides their block with carefully constructed probabilities such that the eventual structure of the blockchain under this selfish mining attack has the same distribution as the structure of the blockchain constructed by only honest miners with a different latency parameter. Thus statistical tests that only look at the pattern of the blockchain itself such as fork heights cannot detect their attack.

Strategic Manipulation of Consensus Protocols. Following seminal work of [9], there is now a long body of work studying strategic manipulations in consensus protocols [9, 21, 16, 6, 15, 13, 12, 11, 24, 23, 3]. These works are all thematically related to ours in that we also study strategic manipulation of consensus protocols. Of these, only [11] bears any technical similarities, as the others all study longest-chain variants.

In terms of motivating cryptographic self-selection, [4] establish that longest-chain variants with fully-internal pseudorandomness are all vulnerable to a selfish-mining-style attack based on predicting future randomness.

Research works on the detection of strategic attacks mostly focus on the longest-chain Proof-of-Stake protocols such as [20]. To the best of our knowledge, our work is the first to propose a detection method for manipulating leader selection protocols in BFT-based blockchains.

Relevant Proof-of-Stake Protocols in Practice. Several large blockchains employ Proof-of-Stake over Proof-of-Work, and there is not yet convergence on a dominant consensus paradigm. For example, Cardano [17] uses a longest-chain variant considered in [4], Algorand uses cryptographic self-selection [14, 7] considered in [11] (although Algorand seems to have since updated their leader selection to induce a round robin aspect – every k rounds, the winner's credential sets the seeds for the subsequent k rounds. See [14].), and Ethereum uses a hybrid of the two (although manipulations of Ethereum are much closer to manipulations of cryptographic self-selection than of longest-chain protocols – see here). In terms of relevance for practice, our results (a) highlight a desirable property of [7]'s original cryptographic self-selection that is desirable in practice, and (b) serve as a canonical example to highlight manners in which a protocol might avoid undetectable profitable deviations.

2 Model and Preliminaries

2.1 Proof-of-Stake Consensus Protocols with Finality

Proof-of-stake protocols with finality look more like classical Consensus algorithms from Distributed Systems than Bitcoin's Longest-Chain protocol. That is, these protocols repeatedly run a secure consensus algorithm to agree on a block of authorized transactions, add this block of transactions to the ledger, and proceed. Unlike the Longest-Chain protocol, these blocks are added to the ledger and remain in the ledger forever. In order to maintain security guarantees, the consensus algorithm for each block is often complex.

To mitigate this complexity (both computational, communication, and conceptual), many protocols select a *leader* ℓ_t who plays a special role in the consensus protocol. Intuitively, all participants try to copy the leader's proposed block. Similarly to a Longest-Chain protocol, the leader ℓ_t dictates the contents of the block. That is, the contents of Block t are fully dictated by the leader ℓ_t, just like in a Longest-Chain protocol (and the only difference is how consensus is reached so that the rest of the network agrees on what block was indeed dictated). As a result, we model the payoff of players to be the fraction of rounds in which they are the block leader, since creating a block is the only way they can gain profit (e.g., through block rewards and MEV).

In order to mitigate grinding attacks, the leader-selection protocol typically needs to ensure that when participants are behaving honestly, the probability that each participant (or pool of participants) gets to be the leader in each round is proportional to each participant's stake (hence the name "proof-of-stake"). Moreover, there should be limited room for a participant to gain extra profit by deviating from the protocol.

Cryptographic self-selection (used by Algorand [7, 14]) is an elegant solution for the leader-selection protocol, which does not rely on the existence of frequent and high quality randomness beacon to generate a random seed for each round. Instead, it uses information about the previous rounds to generate a seed for the current round. We now briefly describe the protocol and the parts that are relevant for constructing a statistical detection method of strategic deviation from the protocol. The reader could refer to [11, 7] for a more detailed description of the protocol and encryption schemes.

There are two key components to the cryptographic self-selection protocol: Verifiable Random Functions (VRFs) and balanced scoring functions.

▶ **Definition 1** (Verifiable Random Function (VRF) [19]). *A Verifiable Random Function is a public-key cryptographic function that generates public key and secret key pairs, denoted* $(\mathsf{sk}, \mathsf{pk})$*, and efficiently evaluates an input* x *using a function* f_{sk} *that is dependent on the secret key. The function produces an output* y *and a proof of correctness, which can be verified efficiently by anyone who has the public key. The following security properties are guaranteed:*

- *Pseudorandomness: Given the public key* pk *and a sequence of input-output pairs* $(x_1, y_1), \ldots, (x_n, y_n)$ *with their corresponding proofs, it is computationally infeasible to predict* $y = f_{sk}(x)$ *for any* $x \neq x_1, \cdots, x_n$ *without the secret key* sk*. In fact, the distribution of* y *looks indistinguishable from the uniform distribution on* $[0, 1]$*.*
- *Unique Provability: For any input* x*, there is exactly one output* y *that can be verified as the correct computation of* $f_{\mathsf{sk}}(x)$*.*

A balanced scoring function takes in the pseudorandom output generated from the VRF associated with an account (i.e. parametrized by the account's secret key) and the amount of stake in that account, and yields a score. The account with the minimum score is selected as the leader. A balanced scoring function always selects a leader proportional to the account's stake assuming the outputs of the VRFs are truly random. In particular, this implies that splitting one's stake between multiple accounts and/or merging stake with another entity does not impact the probability of being selected as the minimum. This forms the basis for selecting a leader-selection protocol that selects leaders independently in each round proportional to their stake.

▶ **Definition 2** (Balanced Scoring Function [11]). *A* scoring function $S(\cdot, \cdot)$ *takes as input a credential* X_i *and a quantity of stake* α_i *and outputs a score* $S(X_i, \alpha_i)$*. A scoring function is* balanced *if for all* n *and all player stakes* $\alpha_1, \cdots, \alpha_n$*,*

$$\mathop{\mathbf{Pr}}_{X_1,\cdots,X_n \leftarrow U([0,1])}\left[\operatorname*{argmin}_{i\in[n]} S(X_i, \alpha_i) = j\right] = \frac{\alpha_j}{\sum_{i=1}^{n} \alpha_i}.$$

▶ **Proposition 3** ([10]). *Let* $S(\cdot, \cdot)$ *be any balanced scoring function. Then, for all* $n \in \mathbb{N}$ *and* $(\alpha_i)_{1\leq i\leq n}$*, the random variables* $S(X, \sum_{i=1}^{n} \alpha_i)$ *and* $\min_{1\leq i\leq n}\{S(X_i, \alpha_i)\}$ *are identically distributed for* $X, X_1, \ldots X_n \sim U([0, 1])$*.*

▶ **Definition 4** (Cryptographic Self-Selection Protocol (CSSP), [11]). *The Cryptographic Self-Selection Protocol (CSSP) operates as follows:*

1. *Each account* i*, with stake* α_i*, sets up a VRF* $f_{\mathsf{sk}_i}(\cdot)$ *with a pair of secret key and public key* $(\mathsf{sk}_i, \mathsf{pk}_i)$*. Participants agree on some Balanced Scoring Function* $S(\cdot, \cdot)$*.*
2. Q_r *denotes the seed used during round* r*.* Q_1 *is a uniformly random draw from* $[0, 1]$*, and* Q_r *will be determined during round* $r - 1$ *(see below).*

3. *In round r, each account i computes their credential $\textsc{Cred}_i^r = f_{\mathsf{sk}_i}(Q_r)$ using their VRF f_{sk_i}. Each account-holder should broadcast their credential (this is not enforced – an account-holder may choose not to broadcast, if desired).*
4. *The leader ℓ_r is the account-holder i who broadcasts the credential with the lowest score $S(\textsc{Cred}_i^r, \alpha_i)$.*
5. *The seed for the next round, Q_{r+1}, is set as the credential of the leader of round r, namely $\textsc{Cred}_{\ell_r}^r$.*

The actions of a player (who may control multiple accounts) in round r of a CSSP are simply to decide which (if any) of their credentials to broadcast. The payoff to player i is the fraction of rounds in which they are the leader. Formally, if $L_p(r)$ is the indicator variable for whether an account controlled by player p is the leader in round r, then the payoff to player p is $\liminf_{r\to\infty} \frac{\sum_{r'\le r} L_p(r')}{r}$.

CSSP is a formalization of the leader-selection protocol initiated by Algorand [7]. Note that leader-selection is but one aspect of a Proof-of-Stake protocol (the core of the protocol is reaching consensus on the block proposed by the leader). Fortunately, the leader-selection protocols are modular, and can be studied in isolation from the (significantly more complex) consensus algorithms that use them.

2.2 Strategic Play in CSSP

Studies of strategic manipulation in consensus protocols first and foremost aim to understand whether one should expect strategic players to choose to be honest. As such, the overwhelming majority of prior work considers a single strategic player against a profile of honest players. [11] establish that this single strategic player is *not* incentivized to be honest, and we ask whether a strategy that realizes these gains is always detectable. In concurrent and independent work, [10] establish tight bounds on the profitability of manipulations. They do not consider detectability, and therefore the work is orthogonal to ours.

In a CSSP, honest behavior corresponds to broadcasting all credentials in every round. A strategic player may selectively choose which credentials to broadcast in each round. It should initially seem counterintuitive that strategic behavior is profitable – hiding a credential *in round r* certainly cannot help a player *win round r*. However, hiding a credential in round r might help a player *win round* $r' > r$ by influencing the seed $Q_{r'}$.

Strategy Space in CSSP ([11]). Consider a CSSP parameterized by α, the fraction of stake controlled by the strategic player, and $\beta \in [0, 1]$, the network connectivity strength of the strategic player. The strategic player is called β-strong if it learns β fraction of the credentials broadcast by the honest players before they must broadcast themselves. Specifically, $\beta = 1$ represents a player that learns all credentials of the honest players before they broadcast (because they are extremely well-connected in the network) and $\beta = 0$ represents a player that learns none of the credentials of the honest players.

[11] make refinement of the strategy space of the CSSP game by showing that any strategy of the strategic player is equivalent to a strategy that only broadcasts at most one credential per round, splits their stake into as many accounts as possible, and considers only two honest players B and C, the former with $\beta(1-\alpha)$ fraction of the stake, and the latter with $(1-\beta)(1-\alpha)$ fraction of the stake. Their proof shows that for any strategy s in CSSP, you can find another strategy s' in the refined strategy space with the same payoff. Since our focus is on both the profitability and detectability of a strategy, we need to show that such

refinement also preserves the detectability of a strategy. Our detection methods (will be introduced in Section 5) only assume an access to the broadcast credential with the minimum score (i.e. the leader's credential) in each round. Thus two strategies that induce the same minimum broadcast credential in each round are either both detectable or both undetectable. Since for any undetectable strategy s in CSSP, there is also an undetectable strategy s' in the refined strategy space, by having s' broadcast the same credential in each round as the minimum credential that s broadcasts (and broadcasts none if s broadcasts none), it is without loss of generality to consider only refined strategies from now on. The refined strategy space is described below:

▶ **Definition 5** (Refined CSSP, [11]). *The strategic player first splits their stake in as many account as possible. This set of accounts, denoted as A, is then fixed for all rounds. In each round r of CSSP, the strategic player:*

1. *Is aware of the seed Q_r and the honesty of player B and C.*
2. *Has access to the credential $\textsc{Cred}_B^r$ of honest player B, but not to the credential $\textsc{Cred}_C^r$ of honest player C. The player only knows the fact that $\textsc{Cred}_C^r$ is distributed uniformly on $[0,1]$.*
3. *Can compute credentials $\textsc{Cred}_i^r$ and scores $S(\textsc{Cred}_i^r, \alpha_i)$ for accounts $i \in A$, and can compute the score $S(\textsc{Cred}_B^r, \beta(1-\alpha))$.*
4. *For each $\ell \in A \cup \{B\}$, can imagine that perhaps $Q_{r+1} = \textsc{Cred}_\ell^r$, and then pre-computes hypothetical credentials $\textsc{Cred}_i^{r+1}$ for each $i \in A$ in case we were to have $\ell_r = \ell$.*
5. *Can extend this pre-computation to any round k and sequence of accounts $i_0, \ldots, i_k$ (with $i_0 \in A \cup \{B\}$ and $i_\ell \in A$ for $0 < \ell \leq k$), and compute $\textsc{Cred}_{i_k}^{r+k}$ based on the hypothetical possibility that $\ell_{r+\ell} = i_\ell$ for each $l \in \{0, \ldots, k-1\}$.*
6. *Selects an account $i^* \in A$ and broadcasts its credential $\textsc{Cred}_{i^*}^r$, or chooses not to broadcast any credential.*

The Refined CSSP is the precise mathematical game we study, for a particular balanced scoring function S. In addition, Proposition 6 implies that the game induced by CSSP is the same for any balanced scoring function used in the protocol. They further imply that if we have a statistical detection method for strategic behavior under one CSSP protocol using a particular balanced scoring function S, we can apply the same detection method to a variant of the protocol using another balanced scoring function S'. Therefore, undetectable profitable strategies exist for any leader-selection protocol based on Algorand's cryptographic self-selection if and only if they exist in the Refined CSSP for any particular S of our choosing.

▶ **Proposition 6** ([11, 10]). *The game induced by CSSP with a balanced scoring function is independent of the particular balanced scoring function used. Formally, for two distinct balanced scoring functions S, S', the games induced by CSSP are identical. Specifically, for all players i, there is a bijective mapping f from strategies of player i in the CSSP with S to strategies of player i in the CSSP with S', where all players broadcast the same set of credentials in each round. For all i, the payoff to player i in the CSSP with S under strategy profile s is exactly the same as the payoff to i in the CSSP with S' under strategy profile $\langle f_i(s_i)\rangle_i$.*

To simplify our analysis, we choose $S(X, \alpha) := -\ln(X)/\alpha$, which induces an exponential distribution with rate α, i.e. when X is drawn from $U([0,1])$, $S(X,\alpha)$ is drawn from $\mathsf{Exp}(\alpha)$. The exponential distribution has the nice property that for a set of random variables $X_1, \ldots, X_n$ drawn from $\mathsf{Exp}(\alpha_1), \ldots, \mathsf{Exp}(\alpha_n)$ respectively, the minimum score $\min_{i\in[n]} X_n$ is distributed according to $\mathsf{Exp}(\sum_{i=1}^n \alpha_i)$. This implies that if the total sum of active stakes is 1 and all players are honest, then the minimum score broadcast is distributed according

to Exp(1). Additionally, Lemma 29 and Lemma 30 imply that the scoring function is a balanced scoring function, and therefore by Proposition 6 there is a bijective mapping from strategies of player i in the CSSP game with balanced scoring function S' and strategies of player i in the CSSP game with balanced scoring function $S = -\ln(X)/\alpha$, which achieves the same outcome (i.e., the same leader is selected each round) and the same profit for player i. Thus if we can detect any profitable strategy under scoring function S, we can use the same method to detect a profitable strategy under scoring function S' since the same strategy is also profitable under S. It is therefore without loss of generality to assume $S(X, \alpha) = -\ln(X)/\alpha$ for all the analysis that follows. Appendix A contains all the relevant properties of an exponential distribution that we will employ to detect strategic deviation.

3 Statistical Detection Methods for Strategic Deviations

Our paper concerns detection of strategic manipulations in CSSP, so we must first clarify what information is available to the onlooker who wishes to distinguish between the case when all participants are honest (but perhaps suffer latency issues), or a strategic player is manipulating the protocol.

We consider the minimal amount of information necessary for an onlooker just to follow the state of the blockchain: the credentials of the leader from each round.[8] We will show that this information alone suffices to detect any profitable manipulation in case the strategic player has $\beta = 0$.

Before continuing, we briefly note that the $\beta = 0$ case corresponds to a "poorly connected" attacker who cannot learn the broadcasts of other players before deciding their own. This matches the $\gamma = 0$ case when analyzing Proof-of-Work protocols, which is considered standard/canonical. We also note that, if desired, a leader selection protocol could take steps to induce $\beta = 0$ (for example, participants could cryptographically commit to their credential with a large deposit, and then only receive their deposit back upon revealing). Our main result does leverage $\beta = 0$ (and we will highlight where), and it is an interesting technical question to understand the case of $\beta = 1$.[9] But, we hope this brief note reminds the reader that $\beta = 0$ is considered the canonical setting. We now proceed with a formal description of the information observed.

▶ **Definition 7** (Observed distribution). *Let an observer pick a uniformly random round r from the set of all rounds $\{1, \ldots, R-1\}$. Let Z, Z_{+1} be the random variables denoting the score of the winning credential in consecutive rounds r and $r+1$ respectively. i.e., $Z = S(\textsc{Cred}^r_{\ell_r}, \alpha_{\ell_r})$ and $Z_{+1} = S(\textsc{Cred}^{r+1}_{\ell_{r+1}}, \alpha_{\ell_{r+1}})$. Then D_Z and $D_{Z_{+1}}$ represent the distributions of the winning credentials in round r and $r+1$ when $R \to \infty$, and we define $F_Z, F_{Z_{+1}}$ to be the cumulative density function (c.d.f.) of $D_Z, D_{Z_{+1}}$ respectively.*

Let us now briefly discuss a null hypothesis for the observed distribution. One null hypothesis might be that in every round r, every player is online and suffers no latency issues (that is, every account-holder i learns of the seed Q_r, computes $\textsc{Cred}^r_i$ and broadcasts it within the allotted time-window), and behaves honestly. If this were the case, we would expect the distribution of winning scores to be i.i.d. from the distribution $S(U([0,1]), 1) = \mathsf{Exp}(1)$ (by Lemma 27 and Lemma 28), and in particular we would expect (Z, Z_{+1}) to be distributed according to $\mathsf{Exp}(1) \times \mathsf{Exp}(1)$.

[8] Note that the detection method of [3] is tailored to Longest-Chain protocols and in particular looks at the distribution of orphans. As there are no orphans in consensus protocols with finality, we need a fundamentally different detection method.

[9] We further explore [11]'s 1-LOOKAHEAD strategy in Section 4, which leverages $\beta = 1$ and is detectable.

This is perhaps too strong a null hypothesis, though – some participants may go offline for extended periods of time, and there is no reason the rest of the network should a priori be aware of this. Additionally, some participants may be online but suffer latency issues that prevent them from broadcasting their credential in time. We therefore consider a weaker null hypothesis which instead posits that there exists *some* stake γ which is online and honest each round, *except the precise value of* γ *is unknown*. Under this null hypothesis, we would expect there to *exist some* γ for which the distribution of winning scores is i.i.d from $S(U([0,1]),\gamma) = \mathsf{Exp}(\gamma)$, and therefore we would expect there to exist some γ for which (Z, Z_{+1}) is distributed according to $\mathsf{Exp}(\gamma) \times \mathsf{Exp}(\gamma)$. For simplicity of notation, we w.l.o.g. let 1 denote the "true" online stake, which might be less than the total stake. Therefore, we consider the null hypothesis to also be satisfied when $\gamma > 1$.[10]

Our main result establishes that no profitable strategy for a $\beta = 0$ strategic player induces an observed distribution that passes the null hypothesis. We also consider an even more robust null hypothesis in Section 5 where there exists some unknown γ for which the fraction of active stake in each round lies in $[(1-\delta)\cdot\gamma, (1+\delta)\cdot\gamma]$, but stick to the simpler null hypothesis first for cleanliness of our main result.

We now elaborate below on two types of statistical tests. Note that in each round, we only have access to the realization, rather than the underlying distribution of the minimum score, so we only have an empirical estimate of $(F_Z, F_{Z_{+1}})$. Still, the number of rounds of history for a Proof-of-Stake-with-Finality blockchain protocol is extremely large. For example, Ethereum produces new blocks every twelve seconds, or 7200 blocks/day. We also remind the reader that all prior analysis on profitability, and the unique prior work on detectability, consider profitability and detectability in steady-state. This is sensible given the intended lifespan of a blockchain and the rate at which blocks are produced.

Detection Method 1: Distribution of Minimum Score. We first focus simply on the distribution of the winning credential across rounds, without looking at correlation of credentials between rounds. Proposition 8 explicitly confirms that under the null hypothesis, D_Z should be $\mathsf{Exp}(\gamma)$ for some γ.

▶ **Proposition 8.** *When the total online stake is constant across rounds and all players honestly broadcast their credentials, there exists a number* γ *such that* D_Z *is distributed identically to* $\mathsf{Exp}(\gamma)$.

Proof. Let λ be the actual amount of total online stakes. By Proposition 3, $\min_i\{S(X_i,\alpha_i)\}$ is distributed identically to $S(X, \sum_i \alpha_i)$ when X, X_i are i.i.d. from $U([0,1])$. Since when all players are honest, $\textsc{Cred}_i^r$ is distributed identically to $U([0,1])$. Therefore, at each round r, the score of the leader $S(\textsc{Cred}_{\ell_r}^r, \alpha_{\ell_r})$ is distributed identically to $S(X,\lambda)$, where $X \sim U([0,1])$. By our choice of the scoring function, $S(\textsc{Cred}_{\ell_r}^r, \alpha_{\ell_r})$ is distributed identically to $\mathsf{Exp}(\lambda)$. Thus, the c.d.f. of D_Z is

$$F_Z(z) = \lim_{R\to\infty} \sum_{r=1}^{R} \frac{1}{R} F_{S(\textsc{Cred}_{\ell_r}^r, \alpha_{\ell_r})}(z) = \lim_{R\to\infty} \sum_{r=1}^{R} \frac{1}{R}(1 - e^{-\lambda z}) = 1 - e^{-\lambda z}$$

which implies that D_Z is distributed identically to $\mathsf{Exp}(\lambda)$. Taking $\gamma = \lambda$ concludes our proof. ◀

[10] That is, if the strategic player causes it to appear as though a $\gamma > 1$ fraction of the total stake is online and honest, then clearly something is wrong and an onlooker should detect this. But if the true online stake is only 1/3 of the total stake and the strategic player causes it to appear as though $2 \cdot 1/3$ of the total stake is online and honest, this is plausible to an onlooker who doesn't know the true fraction of online stake.

A more robust null hypothesis allows for the possibility that the fraction of online players varies across rounds, but not by much. In this setting, the total amount of online stake lies in some range $[(1-\delta)\lambda, (1+\delta)\lambda]$ for some small $\delta > 0$ and $\lambda > 0$. The exact distribution D_Z is impossible to compute without knowing the actual online stake $\lambda_1, \cdots, \lambda_R$ in each round. Nevertheless, the observer expects that the c.d.f. of D_Z is within a certain range parameterized by γ that represents her estimation of λ.

▶ **Proposition 9.** *When the fraction of online stake lies within a multiplicative* $1 \pm \delta$ *factor across all rounds, and all players honestly broadcast their credentials, there exists a number* γ *such that* $\mathsf{Exp}((1+\delta)\gamma) \preceq D_Z \preceq \mathsf{Exp}((1-\delta)\gamma)$.[11]

Proof. Let λ be such that the online stake in each round is within $[(1-\delta)\cdot\lambda, (1+\delta)\cdot\lambda]$. Such λ is guaranteed to exist by hypothesis. At each round r, since the fraction of online stake in round r is λ_r, we know that $S(\textsc{Cred}^r_{\ell_r}, \alpha_{\ell_r})$ is distributed according to to $\mathsf{Exp}(\lambda_r)$. Thus, the c.d.f. of D_Z is

$$F_Z(z) = \lim_{R\to\infty} \sum_{r=1}^{R} \frac{1}{R} F_{S(\textsc{Cred}^r_{\ell_r}, \alpha_{\ell_r})}(z) = \lim_{R\to\infty} \sum_{r=1}^{R} \frac{1}{R}(1 - e^{-\lambda_r z})$$

Because $\lambda_r \in [(1-\delta)\lambda, (1+\delta)\lambda]$, for all r,

$$\mathsf{Exp}((1-\delta)\lambda) \preceq \mathsf{Exp}(\lambda_r) \preceq \mathsf{Exp}((1+\delta)\lambda)$$

Plugging this in $F_Z(z)$ and substituting λ with γ, we are able to conclude that

$$\mathsf{Exp}((1+\delta)\gamma) \preceq D_Z \preceq \mathsf{Exp}((1-\delta)\gamma)$$ ◀

We conclude this detection method by reminding the reader that because the observer does not know the actual amount of online stake, γ, a strategic player could make it appear as though the total online stake is some $\lambda \neq \gamma$ (and we specifically remind the reader that $\gamma > \lambda$ would and should still satisfy our null hypothesis). Our main contribution in this paper is to show that it is impossible to be profitable and preserve the distribution of broadcast scores to be consistent with any fraction of online stake.

Detection Method 2: Correlation of Consecutive Minimum Scores. Our second detection method leverages the fact that the credentials of the leader are independent from round to round when all players follow the protocol. In particular, we examine the correlation between the minimum scores in consecutive rounds. Under honest mining behavior, all credentials are drawn i.i.d. from $U([0,1])$, thus the probability of seeing the score of the leader's credential to be z_{+1} in round $r+1$ should not be changed given the score of the leader's credential z_r in round r. Formally,

$$F_{Z,Z_{+1}}(z, z_{+1}) = F_Z(z) \times F_{Z_{+1}}(z_{+1})$$

3.1 Necessary Conditions for Undetectable Strategic Attacks

In this section, we analyze the effect of the adversary's strategy on D_Z and define the concept of a *statistically undetectable strategy*. The honest players would always broadcast their credentials with the minimum scores (equivalently, broadcast all credentials they have), while the adversary commits to a strategy π that does not necessarily broadcast the credential with minimum score.

[11] Here, $D_1 \preceq D_2$ denotes that D_2 first-order stochastically dominates D_1.

▶ **Definition 10.** *Let the scoring function* $S(\textsc{Cred}, \alpha) = -\ln(\textsc{Cred})/\alpha$*, where* α *is the stake of the adversary. Pick a round* r *uniformly at random from the set of all rounds* $[R]$*, where the total active stake in round* r *is* λ_r*. Let* $X_r(\lambda_r), X_{r+1}(\lambda_{r+1})$ *be the random variables that denote the minimum score of the honest players' broadcast credentials in round* r *and* $r+1$ *respectively; let* $Y_r(\pi), Y_{r+1}(\pi)$ *be the random variables that denote the score of the adversary's broadcast credential in round* r *and* $r+1$ *respectively when they commit to strategy* π*. Thus, the score of the leader in round* r *and* $r+1$*, could be written as* $Z_r(\pi, \lambda_r) = \min\{X_r(\lambda_r), Y_r(\pi)\}$ *and* $Z_{r+1}(\pi, \lambda_{r+1}) = \min\{X_{r+1}(\lambda_{r+1}), Y_{r+1}(\pi)\}$*.*

We also define the distribution of these random variables with respect to a uniformly random round r.

▶ **Definition 11.** *Let* $\mathcal{X}$ *be a random variable over a uniformly random round.* $D_{\mathcal{X}}$ *is defined to be the distribution of* $\mathcal{X}$*, where the corresponding cumulative density function*

$$F_{\mathcal{X}}(x) = \lim_{R\to\infty} \mathop{\mathbf{Pr}}_{r\leftarrow U\{1,\ldots,R\}}[\mathcal{X} \le x].$$

An observer may choose to examine the distribution of all possible random variables, and even joint distribution of random variables over a random round. For instance, she might examine the score in the previous round, or the joint distribution of the scores in the next 10 rounds. Formally, let $\mathcal{Z}$ denote the set of scores of broadcast credentials that the observer chooses to examine. A strategy π is robust to any statistical detection if for any set $\mathcal{Z}$, the joint distributions of seeing all scores in $\mathcal{Z}$ over a random round are identical when the adversary uses strategy π with online stake 1 and when the adversary honestly follows the protocol with online stake γ. That is, no matter which set of scores the observer chooses to examine, she could not distinguish the distribution when the adversary honestly follows the protocol and there is a γ fraction of online stake, or when they use strategy π and there is a 1 fraction of online stake (recall that we w.l.o.g. let 1 denote the fraction of online stake for simplicity of notation).

The two detection methods proposed in the previous section give us two necessary conditions for a strategic attack to be undetectable, since we expect the distribution of the minimum scores and the correlation between consecutive minimum scores to follow certain patterns when all players are honest. The first detection method that uses the distribution of minimum scores corresponds to the case when the observer chooses to examine the score at each specific round, i.e., $\mathcal{Z} = \{Z_r\}$.

▶ **Definition 12.** *Let* λ_r *be the real participating stake in round* r*. The observer knows that the sequence of participating stakes falls into a certain class* $\mathcal{C}_R$ *representing a sequence of active stakes (e.g. fluctuation must be within* $1 \pm \delta$ *fraction). The strategy* π *is statistically undetectable to the distribution test if for some* $\{\gamma_r\}_{r\in[R]} \in \mathcal{C}_R$ *and all* z_r*,*

$$\lim_{R\to\infty} \mathop{\mathbf{Pr}}_{r\leftarrow U(\{1,\ldots,R\})}[Z_r(\pi, \lambda_r) \le z_r] = \lim_{R\to\infty} \mathop{\mathbf{Pr}}_{r\leftarrow U(\{1,\ldots,R\})}[Z_r(\pi_{honest}, \gamma_r) \le z_r].$$

The second test on correlation between consecutive minimum scores corresponds to the case when the observer chooses to examine the scores in two consecutive rounds, i.e., $\mathcal{Z} = \{Z_r, Z_{r+1}\}$. In order to distinguish with the first test, we leave the constraint on D_Z to Definition 12 and only focus on the correlation between D_Z and $D_{Z_{+1}}$ in Definition 13.

▶ **Definition 13.** *A strategy π is statistically undetectable to the correlation test if $Z(\pi)$ and $Z_{+1}(\pi)$ are independent. That is, for any z_r and z_{r+1},*

$$\begin{aligned}&\lim_{R\to\infty} \mathop{\mathbf{Pr}}_{r\leftarrow U(\{1,\ldots,R\})}[Z_r(\pi) \le z_r \ \wedge \ Z_{r+1}(\pi) \le z_{r+1}]\\ =&\lim_{R\to\infty} \mathop{\mathbf{Pr}}_{r\leftarrow U(\{1,\ldots,R\})}[Z_r(\pi) \le z_r] \cdot \mathop{\mathbf{Pr}}_{r\leftarrow U[1,R]}[Z_{r+1}(\pi) \le z_{r+1}]\end{aligned}$$

4 A Canonical Example

In CSSP, the winning credential of the current round is used as the seed of the next round. This leaves the possibility that an adversary would be strategic in their winning credentials and effectively bias the distribution of seeds. For instance, [11] demonstrate that such protocols are indeed vulnerable to such deviations. In order to acquaint the reader with both the CSSP and statistical detectability, we will show that [11]'s canonical 1-Lookahead manipulation is statistically detectable using *either* our distribution test or our correlation test.

Here is some brief intuition for 1-Lookahead: because the winning credential is the seed of the next round, the adversary is able to compute credentials for all wallets assuming that a credential in this round is the winning credential. Thus, if the adversary has multiple credentials with low scores to choose from, they could choose to broadcast only the one which maximizes the expected number of rounds won among the current and one-after round. We repeat the formal definition of 1-Lookahead below:

▶ **Definition 14** (1-Lookahead strategy)**.** *Let the total stake be fixed and normalized to* 1 *with the adversary owning an α fraction of the total stake. The goal of the 1-Lookahead strategy is to maximize the expected number of rounds won among the present and subsequent rounds, and proceeds as follows:*

1. *Let r be the current round and A be the set of all accounts of the adversary, B be the lone honest account that is broadcast when the adversary decides with total stake $\beta(1-\alpha)$.*
2. *Let $W(Q_r) \subseteq A$ denote the accounts i satisfying $S(\textsc{Cred}_i^r, \alpha_i) < S(\textsc{Cred}_B^r, \beta(1-\alpha))$. Observe that $W(Q_r)$ might be empty, and that when $\beta = 0$, $W(Q_r) = A$.*
3. *If $W(Q_r)$ is empty, the adversary cannot win this round, so they move on to the next round and go back to step 1.*
4. *If $W(Q_r)$ is non-empty, for all potential winning accounts $i \in W(Q_r)$ and all potential next-round accounts $j \in A$, compute credential $\textsc{Cred}_{i,j}^{r+1} = f_{\mathsf{sk}_j}(\textsc{Cred}_i^r)$, which is the credential of account j in round $r+1$ in the event that account i happens to win round r.*
5. *Let $j(i) = \arg\min_{j\in A} S(\textsc{Cred}_{i,j}^{r+1}, \alpha_j)$ – this is the account whose credential is most likely to win in round $r+1$ if account i wins round r.*
6. *For each $i \in W$, define P_i^{r+1} to be the probability that the adversary wins with account i in round r* and *wins with account $j(i)$ in round $r+1$.*[12] *That is (below, think of $X^r := S(\textsc{Cred}_C^r, (1-\beta)(1-\alpha))$):*

$$P_i^{r+1} = \mathop{\mathbf{Pr}}_{X^r\leftarrow \mathsf{Exp}((1-\beta)(1-\alpha))}[S(\textsc{Cred}_i^r, \alpha_i) < X^r] \cdot \mathop{\mathbf{Pr}}_{X^{r+1}\leftarrow \mathsf{Exp}(1-\alpha)}[S(\textsc{Cred}_{i,j(i)}^{r+1}, \alpha_{j(i)}) < X^{r+1}].$$

[12] For example, if $\beta = 1$, the probability that the adversary wins with account i in round r is 1. No matter β, the probability that the adversary wins with account $j(i)$ in round $r+1$ is just the probability that this credential beats a draw from $\mathsf{Exp}(1-\alpha)$.

7. *Let* $i^* = \arg\max_{i \in W} \left(\mathbf{Pr}_{X^r \leftarrow \mathsf{Exp}((1-\beta)(1-\alpha))}[S(\textsc{Cred}_i^r, \alpha_i) < X^r] + P_i^{r+1} \right)$. i^* *is the account that maximizes the expected number of consecutive rounds (among* $r, r+1$*) that the adversary wins.*
8. *Broadcast* $\textsc{Cred}_{i^*}^r$ *at round* r*. If* $\textsc{Cred}_{i^*}^r$ *is the credential with minimum score in round* r*, broadcast* $\textsc{Cred}_{j(i^*)}^{r+1}$ *at round* $r+1$*. If* $\textsc{Cred}_{i^*}^r$ *does not win round* r *continue.*
9. *Return to Step 1.*

While the honest strategy always broadcasts the credential with minimum score and maximizes the probability of winning the current round r, 1-LOOKAHEAD instead optimizes the expected number of consecutive rounds won (but only considering the next round – this is why the strategy is termed 1-LOOKAHEAD). For example, when $\beta = 1$, and the adversary has multiple accounts that can win this round, they may as well broadcast the credential whose seed gives them the best chance of winning the subsequent round. For $\beta < 1$, the math is trickier, but the strategy always strictly outperforms honesty.

Because the purpose of this section is to gain comfort with the concept of detectability, we focus on the simplest version of 1-LOOKAHEAD, which is when $\beta = 1$ (which corresponds to the most powerful adversary). The arguments in the subsequent subsections proceed roughly as follows:

- Section 4.1 shows how we might start reasoning about the distribution test (Definition 12). In particular, Section 4.1 identifies that we can view the distribution of minimum score $D_{Z_r(\pi_{1-\textsc{Lookahead}})}$ as a mixture of distributions associated with transitions in a two-state Markov Chain, and reasons through what each of these three distributions are. Intuitively, these three distributions are "what is the minimum broadcast score, conditioned on r being a reset round (i.e. the adversary did not bias Q_r in $r-1$) and the adversary having at least two winning accounts?", "what is the minimum broadcast score, conditioned on r being a round where the adversary biased Q_r in $r-1$?", and "what is the minimum broadcast score, conditioned on r being a reset round and the adversary has at most one winning account?".
- Section 4.2 then establishes that no mixture of these distributions can result in an exponential distribution, and therefore $1-$LOOKAHEAD fails the distribution test and is detectable. Intuitively, this follows simply because exponential distributions have a precise rate of tail decay, and the above distributions have no reason to match this precise tail, nor to cancel the differences out.
- Section 4.3 considers the correlation test, and establishes that 1-LOOKAHEAD also fails the correlation test. Intuitively, this is because during reset rounds we expect to see a larger than normal winning score (because the adversary may hide coins during a reset round), but during biased rounds we expect to see a lower than normal winning score (because the adversary has biased the seed to make their own score lower than normal). So consecutive rounds are in fact negatively correlated.

Note that failing either of the two tests suffice for a strategy to be statistically detectable – we include both to acquaint the reader with various detection methods (a priori, a strategy might pass one test but fail another).

4.1 Broadcast Distribution on a Markov Chain

We observe that at each round r, the distribution of credentials only depend on the distribution of the seed Q_r. This allows us to characterize the CSSP as a stationary Markov chain. For instance, when all players follow the protocol of CSSP and broadcast their credentials that result in the lowest score, the distribution of Q_r is uniformly random from $[0, 1]$ for all r.

Thus, the Markov chain describing the honest CSSP has only one state that transits to itself with probability 1. In particular, the game effectively resets when the distribution of Q_r is unbiased. We call such a round to be a "reset round".

▶ **Definition 15** (Reset Round [11]). *A round r is a reset round if for all possible strategies π, the distribution of $\{\mathbf{Pr}[Y_{r'}(\pi) \leq X_{r'}]\}_{r' \geq r}$ conditioned on Q_{r-1} and all historical information prior to round $r-1$, is identical to the distribution of $\{\mathbf{Pr}[Y_{r'}(\pi) \leq X_{r'}]\}_{r' \geq r}$ after replacing Q_r with a uniformly random draw from $[0,1]$.*

The 1-LOOKAHEAD strategy, on the other hand, effectively biases the distribution of the seed in favor of the adversary by comparing the best credential for next round. Therefore, the Markov chain of CSSP when the adversary plays 1-LOOKAHEAD is different from the Markov chain of CSSP when every player follows the protocol.

▶ **Lemma 16.** *A CSSP process, with the adversary owning α fraction of the stakes with $\beta = 1$ and using 1-LOOKAHEAD, is equivalent to the stationary Markov chain with two states $\Pi = \{C, H\}$, where the transition probability is*

$$\begin{aligned}
\mathbf{Pr}[\Pi_{r+1} = C | \Pi_r = C] =& 1 - \alpha^2 \\
\mathbf{Pr}[\Pi_{r+1} = H | \Pi_r = C] =& \alpha^2 \\
\mathbf{Pr}[\Pi_{r+1} = C | \Pi_r = H] =& 1 \\
\mathbf{Pr}[\Pi_{r+1} = H | \Pi_r = H] =& 0
\end{aligned}$$

Standard calculation shows that the stationary distribution of the above Markov chain would be $s_C = \frac{1}{1+\alpha^2}$ and $s_H = \frac{\alpha^2}{1+\alpha^2}$. The following Lemma shows the overall distribution of the leader's credential's score, which is computed by summing the distribution conditioned on each type of transition respectively.

▶ **Lemma 17.** *The overall distribution of D_{Z_r} for 1-LOOKAHEAD strategy is*

$$\begin{aligned}
D_{Z_r} =& \frac{1}{1+\alpha^2} \left(\sum_{\ell=1}^{\infty} \mathsf{Exp}_\ell(1) \left[\sum_{\omega \geq \ell} \frac{\alpha^\omega (1-\alpha)}{\omega} \right] + (1-\alpha)\mathsf{Exp}(1) \right) + \\
& \frac{1}{1+\alpha^2} \sum_{\omega=2}^{\infty} \alpha^\omega (1-\alpha) \mathsf{Exp}(1 + (\omega-1)\alpha),
\end{aligned} \tag{1}$$

where $\mathsf{Exp}_\ell(1) := \mathsf{Exp}_{\ell-1}(1) + \mathsf{Exp}(1)$ with $\mathsf{Exp}_0(1) := 0$.

Equation (1) shows that D_{Z_r} could be viewed as mixture of exponential and Erlang distributions (sum of identical exponential distributions) with different rates. We briefly sketch the argument in the proof of Lemma 17, which is quite technical. Since the score of credential in each account i with stake α_i is distributed identical to an exponential $\mathsf{Exp}(\alpha_i)$ by the properties of exponential distributions (Lemma 28 and Lemma 29), by Lemma 30, the ℓ^{th} minimum score of credentials that an adversary owns is distributed identical to a Erlang distribution that is sum of ℓ identical exponential distributions, denoted as Exp_ℓ. Since the adversary's action in each round is confined to choosing which credential to broadcast, the adversary, and hence the overall score distribution must be a mixture of Exp and Exp_ℓs with different rates.

This key property about D_{Z_r} leads to our main results in this section. In section 4.2, we show that that 1-LOOKAHEAD is statistically detectable under distribution test because the mixture of distributions is not an exponential distribution; In section 4.3, we show that 1-LOOKAHEAD is detectable under correlation test because of stochastical dominance relationships between exponential distributions with different rates.

4.2 Broadcast Distribution of 1-Lookahead Cannot be an Exponential Distribution

It is known that the sum of ℓ independent exponential variables with mean 1 each is an Erlang distribution of parameterized by $\ell, 1$. That means, the probability density function (p.d.f.) of $\mathsf{Exp}_\ell(z;1)$ is $\frac{z^{\ell-1}e^{-z}}{(\ell-1)!}$. Plugging in this and the p.d.f. of exponential distributions, we obtain the p.d.f. of D_{Z_r} to be

$$f_{Z_r} = \frac{1}{1+\alpha^2}\left(\sum_{\ell=1}^{\infty}\frac{z^{\ell-1}e^{-z}}{(\ell-1)!}\left[\sum_{\omega\geq l}\frac{\alpha^\omega(1-\alpha)}{\omega}\right]+(1-\alpha)e^{-x}\right)+ \frac{1}{1+\alpha^2}\sum_{\omega=2}^{\infty}\alpha^\omega(1-\alpha)(1+(\omega-1)\alpha)e^{-(1+(\omega-1)\alpha)} \tag{2}$$

If 1-LOOKAHEAD is a statistically undetectable strategy, there exists a parameter $\gamma > 0$ such that f_Z equals to the p.d.f. of $\mathsf{Exp}(\gamma)$. However, the following Lemma shows that this is impossible.

▶ **Lemma 18.** *There is no $\gamma > 0$ such that $D_Z(\pi_{1-\text{LOOKAHEAD}}) = \mathsf{Exp}(\gamma)$.*

Proof Sketch. Assume by contradiction that equation (2) is an exponential distribution. i.e., $f_Z = \gamma e^{-\gamma z}$ where $\gamma > 0$ is the amount of active stakes. Rewriting $e^{(1-\gamma)z}$ and $e^{(1-\omega)\alpha}$ according to the Taylor expansion of $e^x = \sum_{\ell=1}^{\infty}\frac{1}{(\ell-1)!}x^{\ell-1}$, the coefficient for the $x^{\ell-1}$ must agree on all $\ell \geq 1$. This means that for all $\ell \geq 1$,

$$\gamma^2(1-\gamma)^{\ell-1} = \frac{\alpha^\ell(1-\alpha)}{1+\alpha^2}\left[\frac{1}{\ell}+\sum_{\omega=2}^{\infty}\alpha^{\omega-1}(1-\omega)^{\ell-1}(1+(\omega-1)\alpha)^2\right]$$

We now take the absolute value on both sides, and show that the absolute value on the left hand side and the right hand side does not grow at the same rate with l. Therefore, we can conclude that f_{Z_r} cannot be an exponential distribution. ◀

4.3 Distribution of Consecutive Two Rounds are Negatively Correlated in 1-Lookahead

In this section, we apply the correlation test to 1-LOOKAHEAD and show that the distribution of consecutive two rounds are negatively correlated. In a high level, when the adversary successfully hides some credentials in round r and bias Q_{r+1} in round r, they have to do so by strategically hiding credentials with minimum scores. The distribution of scores in such a round stochastically dominates the honest distribution. However, the adversary only chooses to hide credentials because they can obtain credentials with lower scores in round $r+1$. Therefore, the distribution of scores in such a round is stochastically dominated by the honest distribution. This establishes a negative correlation between the scores in subsequent rounds. The Lemma states as follows:

▶ **Lemma 19.** *When the adversary uses 1-LOOKAHEAD strategy, the distribution of consecutive two rounds, $D_{Z_r(\pi_{1-\text{LOOKAHEAD}})}, D_{Z_{r+1}(\pi_{1-\text{LOOKAHEAD}})}$ are negatively correlated. That is, for any numbers a, b,*

$$\Pr_{r\leftarrow U[1,R]}[Z_{r+1} > b | Z_r > a] < \Pr_{r\leftarrow U[1,R]}[Z_{r+1} > b]$$

We defer the formal proof of Lemma 19 to the full version of the paper [5].

5 Profitable Strategies are Detectable

In this section, we will show that when the online stake remains constant throughout the protocol, *every* profitable strategy of an adversary with $\beta = 0$ is detectable (and this holds for all α).

Let us first highlight a few complexities of detecting profitable manipulations. First, there are certainly undetectable non-profitable manipulations (for example, the adversary could simply never broadcast – this results in i.i.d. scores across rounds according to $\mathsf{Exp}(1-\alpha)$, and is indistinguishable from if the adversary were 'non-strategically offline'). Second, note that a strategic adversary can look as far into the (hypothetical) future (assuming they win consecutive rounds) as they like when deciding which accounts to broadcast, and could try to carefully curate them to match a particular distribution. In general, CSSP induces a Markov Decision Process for the adversary, where each state is a countably long list of real numbers. 1-Lookahead witnesses that the MDP always has a strategy that outperforms honest, and we seek to understand whether any such strategy also satisfies a collection of complex constraints (and more over, there is not a single collection of constraints to satisfy – the adversary can pick any γ and satisfy the undetectability constraints to appear as i.i.d. $\mathsf{Exp}(\gamma)$).

Given the complexity of the strategy space in CSSP, our proof is surprisingly simple. Firstly, we make use of the following observation: since the adversary does not know the credentials owned by the honest miner before broadcasting their own,[13] in order to improve their probability of winning throughout the protocol, the adversary must on average broadcast credentials with smaller scores compared to when they are honest. Simultaneously, as discussed in Section 3, the observer expects the empirical score distribution to follow an exponential distribution (of undetermined rate γ). Hence, in order to maintain undetectability, the adversary's credentials must be distributed as an exponential with rate greater than 1.

However, [11] shows that unless the adversary controls almost half of the network, the adversary loses to honest participants in a non-trivial fraction of rounds. After such an event, the adversary loses their advantage gained before from strategic manipulation, and must participate as if the protocol has restarted. We call such rounds where adversary regains the perspective of a uniformly random seed "reset rounds". In a reset round, we show that the adversary must broadcast credentials with scores at least as large as when honest. This leads to a contradiction – the tail of the "reset round" credential score distribution is already too fat for the credential score distribution of the adversary to be an exponential of rate greater than 1.

Our result also extends to the setting where the active stake fluctuates within $1 \pm \delta$ factor, where we show that any undetectable strategies can only achieve limited profitability bounded by 2δ.

5.1 Detectability for Steady Online Stake

Throughout Section 5.1, we will assume that the online stake in each round remains constant, and equal to 1 (by normalization). Among all the online stake, the adversary holds α stake, while the honest participants hold $1-\alpha$ stake. The outside observer knows that the total online state is steady across rounds, but does not know how much total stake is online.

We first prove that a profitable and undetectable adversary has a score distribution that is strictly dominated by $\mathsf{Exp}(\alpha)$. We will use notations related to the minimum score of broadcast credentials that are formally defined in Definition 10.

[13] This is the key simplifying aspect of our proof that leverages $\beta = 0$.

▶ **Theorem 20.** *When the online stake remains constant throughout the protocol, for any adversary who holds α stake and employs a profitable and undetectable strategy π, the adversary's broadcast score $Y_r(\pi)$ from a random round r is distributed identically to* $\mathsf{Exp}(\alpha+\epsilon)$ *for some $\epsilon > 0$.*

Proof. Let $X_r(1)$ and $Y_r(\pi)$ be the minimum score of broadcast credential among honest miners and the adversary respectively, at a uniformly random round r. Then the overall minimum score at that round is $\min\{X_r(1), Y_r(\pi)\}$. Since the adversary must broadcast before observing the honest miner's credentials in round r, $X_r(1)$ is independent of $Y_r(\pi)$. By Definition 12 and Proposition 8, in order for the adversary's strategic attack to remain undetectable, $\min\{X_r(1), Y_r(\pi)\}$ must distribute according to $\mathsf{Exp}(\gamma)$ for some $\gamma > 0$. Since $X_r(1) \sim \mathsf{Exp}(1-\alpha)$ and by independence between $X_r(1)$ and $Y_r(\pi)$, we have that for any $z > 0$,

$$\begin{aligned}
&\mathbf{Pr}[\min\{X_r(1), Y_r(\pi)\} \geq z] = e^{-\gamma z} \\
&\implies \mathbf{Pr}[X_r(1) \geq z]\,\mathbf{Pr}[Y_r(\pi) \geq z] = e^{-\gamma z} \\
&\implies e^{-(1-\alpha)z}\,\mathbf{Pr}[Y_r(\pi) \geq z] = e^{-\gamma z} \\
&\implies \mathbf{Pr}[Y_r(\pi) \geq z] = e^{-(\gamma-(1-\alpha))z} = e^{-(\alpha+(\gamma-1))z}.
\end{aligned}$$

Thus $Y_r(\pi) \sim \mathsf{Exp}(\alpha + (\gamma - 1))$. The expected fraction of rounds that the adversary wins if they are honest is α. Thus to be strictly profitable, the adversary needs to win with fraction $> \alpha$, which requires $\mathbf{Pr}[Y_r(\pi) < X_r(1)] > \alpha$. Since $X_r(1)$ and $Y_r(\pi)$ are exponential random variables with rate $(1-\alpha)$ and $\alpha + (\gamma - 1)$, by Lemma 29,

$$\alpha < \mathbf{Pr}[Y_r(\pi) < X_r(1)] = \frac{\alpha + (\gamma - 1)}{\gamma}.$$

The above equation implies that $\gamma > 1$. Thus we conclude $Y_r(\pi) \sim \mathsf{Exp}(\alpha+(\gamma-1)) = \mathsf{Exp}(\alpha+\epsilon)$ for some $\epsilon > 0$. ◀

Now, we show that for a non-trivial fraction of rounds, the adversary's score is drawn from a distribution that dominates $\mathsf{Exp}(\alpha)$. We will need to reason about the adversary's score in reset rounds (defined in Section 4 at Definition 15), where the distribution of the seed Q_r is unbiased. For the reader's convenience, the formal definition of the reset round is restated here.

▶ **Definition 15** (Reset Round [11])**.** *A round r is a reset round if for all possible strategies π, the distribution of $\{\mathbf{Pr}[Y_{r'}(\pi) \leq X_{r'}]\}_{r' \geq r}$ conditioned on Q_{r-1} and all historical information prior to round $r-1$, is identical to the distribution of $\{\mathbf{Pr}[Y_{r'}(\pi) \leq X_{r'}]\}_{r' \geq r}$ after replacing Q_r with a uniformly random draw from $[0,1]$.*

[11] shows that the number of reset rounds are non-negligible.

▶ **Lemma 21** ([11], Theorem 4.1)**.** *For $\alpha < \frac{3-\sqrt{5}}{2} \approx 0.38$, the fraction of rounds that is a reset round is strictly greater than 0.*

Meanwhile, we show that in a reset round, the adversary's output score distribution stochastically dominates $\mathsf{Exp}(\alpha)$.

▷ Claim 22. Given that round r is a reset round, adversary's output distribution in round r must (weakly) stochastically dominate $\mathsf{Exp}(\alpha)$.

Proof. Let $Y_r(\pi)$ be the broadcast minimum coin of the adversary using strategy π at a random round r. Notice that if r is a reset round, then $Y_r(\pi)$ would be distributed according to $\mathsf{Exp}(\alpha)$ had π been an honest strategy. Let $C_1, \ldots, C_j$ be the score of credentials of all accounts that the adversary owns at round r, where $C_1 \leq C_2 \cdots \leq C_j$. Then for any $z > 0$,

$$\mathbf{Pr}[Y_r(\pi) < z \mid r \text{ is a reset round}] \leq \mathbf{Pr}[C_1 < z \mid r \text{ is a reset round}] = 1 - e^{-\alpha z}$$

since $C_1 \sim \mathsf{Exp}(\alpha)$ after a reset round. Thus $Y_r(\pi)$'s distribution given that r is a reset round must (weakly) stochastically dominate $\mathsf{Exp}(\alpha)$. ◁

Combining Theorem 20, Lemma 21 and Claim 22, we show a contradiction between above two properties that we have derived about a profitable adversary's score distribution. This shows no profitable adversary strategy is undetectable.

▶ **Theorem 23.** *When the online stake remains constant throughout the protocol and $\alpha < \frac{3-\sqrt{5}}{2}$, there is no profitable and statistically undetectable strategy.*

Proof. Given any adversary strategy π, let p_π be the fraction of rounds that is a reset round, by Lemma 21, $p_\pi > 0$. Let $Y_r(\pi)$ be the broadcast minimum coin of the adversary at a random round r. Let $Y_{rs}(\pi)$, $Y_{non-rs}(\pi)$ be the broadcast minimum coin of the adversary at a random reset round and at a random non reset round respectively, as defined in Definition 15. Then $Y_r(\pi)$ is a mixture of random variable $Y_{rs}(\pi)$ and $Y_{non-rs}(\pi)$ Specifically, $Y = p_\pi \cdot Y_{rs}(\pi) + (1 - p_\pi) \cdot Y_{non-rs}(\pi)$. Then for any $z > 0$,

$$\mathbf{Pr}[Y_r(\pi) \geq z] = p \cdot \mathbf{Pr}[Y_{rs}(\pi) \geq z] + (1 - p) \cdot \mathbf{Pr}[Y_{non-rs}(\pi) \geq z].$$

By Theorem 20, for any undetectable strategy, it must be the case that $\mathbf{Pr}[Y_r(\pi) \geq z] \leq e^{-(\alpha+\epsilon)z}$. However, by Claim 22 and Lemma 21, $p > 0$ and $\mathbf{Pr}[Y_{rs}(\pi) \geq z] \geq e^{-\alpha z}$, hence $\mathbf{Pr}[Y_r(\pi) \geq z] \geq p \cdot e^{-\alpha z}$. Since p is not dependent on z, it is impossible for $\mathbf{Pr}[Y_r(\pi) \geq z]$ to be both at most $e^{-(\alpha+\epsilon)z}$ and at least $p \cdot e^{-\alpha z}$ for all $z > 0$. ◀

5.2 Extension to Fluctuation in Online Stakes

In practice, limited fluctuation in participating stake of the protocol may be expected. In this subsection, we consider the case where the online stake in any round fluctuates within a $1 \pm \delta$ multiplicative factor of the baseline online stake. By normalization, we assume the ground truth baseline online stake is 1, while the observer anticipates the online stakes to be in $[(1 - \delta)\gamma, (1 + \delta)\gamma]$ for some $\gamma > 0$. We show that any adversary who profits beyond 2δ probability of winning is detectable. Our key observation is that Theorem 20 can be generalized to adversaries who enjoy profits beyond the online stake fluctuation range (as discussed below in Theorem 25).

▶ **Definition 24.** *An adversary with stake α is Δ-profitable if their probability of being the leader in a random round r is more than $\alpha + \Delta$.*

▶ **Theorem 25.** *When the online stake fluctuation is known to lie in all rounds within a multiplicative $1 \pm \delta$ band of its baseline, for any adversary who holds α stake and employs a 2δ-profitable and undetectable strategy π, the adversary's broadcast score $Y_r(\pi)$ from a random round r is stochastically dominated by $\mathsf{Exp}(\alpha + \epsilon)$ for some $\epsilon > 0$.*

The proof of Theorem 25 can be found in the full version of the paper [5].

▶ **Theorem 26.** *When the online stake fluctuation is known to lie in all rounds within a multiplicative* $1 \pm \delta$ *band of its baseline, no undetectable strategy is* 2δ*-profitable when* $\alpha < \frac{3-\sqrt{5}}{2}$.

Proof. The proof is identical to that of Theorem 23, where we establish that the adversary cannot produce a strategy whose broadcasts are dominated by $\mathsf{Exp}(\alpha+\epsilon)$. The only difference is that we use Theorem 25 instead of Theorem 20 to conclude that this is necessary in order to be undetectable and strictly profitable. ◀

References

1 Musab A. Alturki and Grigore Roşu. Statistical model checking of randao's resilience to pre-computed reveal strategies. In Emil Sekerinski, Nelma Moreira, José N. Oliveira, Daniel Ratiu, Riccardo Guidotti, Marie Farrell, Matt Luckcuck, Diego Marmsoler, José Campos, Troy Astarte, Laure Gonnord, Antonio Cerone, Luis Couto, Brijesh Dongol, Martin Kutrib, Pedro Monteiro, and David Delmas, editors, *Formal Methods. FM 2019 International Workshops*, pages 337–349, Cham, 2020. Springer International Publishing.

2 Nick Arnosti and S. Matthew Weinberg. Bitcoin: A natural oligopoly. In Avrim Blum, editor, *10th Innovations in Theoretical Computer Science Conference, ITCS 2019, January 10-12, 2019, San Diego, California, USA*, volume 124 of *LIPIcs*, pages 5:1–5:1. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2019. doi:10.4230/LIPICS.ITCS.2019.5.

3 Maryam Bahrani and S. Matthew Weinberg. Undetectable selfish mining. In *EC '24: The 25rd ACM Conference on Economics and Computation, New Haven, CT, USA, July 8 - 11, 2024*. ACM, 2024.

4 Jonah Brown-Cohen, Arvind Narayanan, Alexandros Psomas, and S. Matthew Weinberg. Formal barriers to longest-chain proof-of-stake protocols. In *Proceedings of the 2019 ACM Conference on Economics and Computation, EC 2019, Phoenix, AZ, USA, June 24-28, 2019.*, pages 459–473, 2019. doi:10.1145/3328526.3329567.

5 Linda Cai, Jingyi Liu, S. Matthew Weinberg, and Chenghan Zhou. Profitable manipulations of cryptographic self-selection are statistically detectable, 2024. arXiv:2407.16949.

6 Miles Carlsten, Harry A. Kalodner, S. Matthew Weinberg, and Arvind Narayanan. On the instability of bitcoin without the block reward. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, October 24-28, 2016*, pages 154–167, 2016. doi:10.1145/2976749.2978408.

7 Jing Chen and Silvio Micali. Algorand: A secure and efficient distributed ledger. *Theor. Comput. Sci.*, 777:155–183, 2019. doi:10.1016/J.TCS.2019.02.001.

8 Vanessa Chicarino, Célio Albuquerque, Emanuel Jesus, and Antônio Rocha. On the detection of selfish mining and stalker attacks in blockchain networks. *Annals of Telecommunications*, 75(3):143–152, 2020. doi:10.1007/s12243-019-00746-2.

9 Ittay Eyal and Emin Gün Sirer. Majority is not enough: Bitcoin mining is vulnerable. In *Financial Cryptography and Data Security*, pages 436–454. Springer, 2014.

10 Matheus V. X. Ferreira, Aadityan Ganesh, Jack Hourigan, Hannah Huh, S. Matthew Weinberg, and Catherine Yu. Computing optimal manipulations in cryptographic self-selection proof-of-stake protocols. In *EC '24: The 25rd ACM Conference on Economics and Computation, New Haven, CT, USA, July 8 - 11, 2024*. ACM, 2024.

11 Matheus V. X. Ferreira, Ye Lin Sally Hahn, S. Matthew Weinberg, and Catherine Yu. Optimal strategic mining against cryptographic self-selection in proof-of-stake. In David M. Pennock, Ilya Segal, and Sven Seuken, editors, *EC '22: The 23rd ACM Conference on Economics and Computation, Boulder, CO, USA, July 11 - 15, 2022*, pages 89–114. ACM, 2022. doi:10.1145/3490486.3538337.

12 Matheus V. X. Ferreira and S. Matthew Weinberg. Proof-of-stake mining games with perfect randomness. In Péter Biró, Shuchi Chawla, and Federico Echenique, editors, *EC '21: The 22nd ACM Conference on Economics and Computation, Budapest, Hungary, July 18-23, 2021*, pages 433–453. ACM, 2021. doi:10.1145/3465456.3467636.

13 Amos Fiat, Anna Karlin, Elias Koutsoupias, and Christos H. Papadimitriou. Energy equilibria in proof-of-work mining. In *Proceedings of the 2019 ACM Conference on Economics and Computation, EC 2019, Phoenix, AZ, USA, June 24-28, 2019.*, pages 489–502, 2019. doi:10.1145/3328526.3329630.

14 Yossi Gilad, Rotem Hemo, Silvio Micali, Georgios Vlachos, and Nickolai Zeldovich. Algorand: Scaling byzantine agreements for cryptocurrencies. In *Proceedings of the 26th Symposium on Operating Systems Principles, Shanghai, China, October 28-31, 2017*, pages 51–68. ACM, 2017. doi:10.1145/3132747.3132757.

15 Guy Goren and Alexander Spiegelman. Mind the mining. In *Proceedings of the 2019 ACM Conference on Economics and Computation, EC 2019, Phoenix, AZ, USA, June 24-28, 2019.*, pages 475–487, 2019. doi:10.1145/3328526.3329566.

16 Aggelos Kiayias, Elias Koutsoupias, Maria Kyropoulou, and Yiannis Tselekounis. Blockchain mining games. In *Proceedings of the 2016 ACM Conference on Economics and Computation, EC '16, Maastricht, The Netherlands, July 24-28, 2016*, pages 365–382, 2016. doi:10.1145/2940716.2940773.

17 Aggelos Kiayias, Alexander Russell, Bernardo David, and Roman Oliynykov. Ouroboros: A provably secure proof-of-stake blockchain protocol. In *Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part I*, pages 357–388, 2017. doi:10.1007/978-3-319-63688-7_12.

18 Sheng-Nan Li, Carlo Campajola, and Claudio J. Tessone. Twisted by the pools: Detection of selfish anomalies in proof-of-work mining. *CoRR*, abs/2208.05748, 2022. doi:10.48550/arXiv.2208.05748.

19 Silvio Micali, Michael Rabin, and Salil Vadhan. Verifiable random functions. In *40th annual symposium on foundations of computer science (cat. No. 99CB37039)*, pages 120–130. IEEE, 1999.

20 Michael Neuder, Daniel J. Moroz, Rithvik Rao, and David C. Parkes. Defending against malicious reorgs in tezos proof-of-stake. In *Proceedings of the 2nd ACM Conference on Advances in Financial Technologies*, AFT '20, pages 46–58, New York, NY, USA, 2020. Association for Computing Machinery. doi:10.1145/3419614.3423265.

21 Ayelet Sapirshtein, Yonatan Sompolinsky, and Aviv Zohar. Optimal selfish mining strategies in bitcoin. In *Financial Cryptography and Data Security - 20th International Conference, FC 2016, Christ Church, Barbados, February 22-26, 2016, Revised Selected Papers*, pages 515–532, 2016. doi:10.1007/978-3-662-54970-4_30.

22 Zhaojie Wang, Qingzhe Lv, Zhaobo Lu, Yilei Wang, and Shengjie Yue. Forkdec: Accurate detection for selfish mining attacks. *Security and Communication Networks*, 2021:5959698, 2021. doi:10.1155/2021/5959698.

23 Aviv Yaish, Gilad Stern, and Aviv Zohar. Uncle maker: (time)stamping out the competition in ethereum. In Weizhi Meng, Christian Damsgaard Jensen, Cas Cremers, and Engin Kirda, editors, *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security, CCS 2023, Copenhagen, Denmark, November 26-30, 2023*, pages 135–149. ACM, 2023. doi:10.1145/3576915.3616674.

24 Aviv Yaish, Saar Tochner, and Aviv Zohar. Blockchain stretching & squeezing: Manipulating time for your best interest. In David M. Pennock, Ilya Segal, and Sven Seuken, editors, *EC '22: The 23rd ACM Conference on Economics and Computation, Boulder, CO, USA, July 11 - 15, 2022*, pages 65–88. ACM, 2022. doi:10.1145/3490486.3538250.

A Relavent Properties for Exponential Distributions

▶ **Lemma 27** ([11], Lemma 2.1.). *Let* $S(x, \alpha_i) := \frac{-\ln(x)}{\alpha_i}$. *When* x *is drawn uniformly from* $[0, 1]$, $S(x, \alpha_i)$ *is identically distributed to* $\mathsf{Exp}(\alpha_i)$.

▶ **Lemma 28** ([11], Lemma A.1.). *Let $X_1, \cdots, X_n$ be independent random variables where X_i is drawn from $\mathsf{Exp}(\alpha_i)$ for some $\alpha_i > 0$. Then $\min_{i\in[n]}\{X_n\}$ is identically distributed to $\mathsf{Exp}\left(\sum_{i=1}^{n}\alpha_i\right)$.*

▶ **Lemma 29** ([11], Lemma A.2.). *Let X_1, X_2 be two independent random variables drawn from $\mathsf{Exp}(\alpha_1), \mathsf{Exp}(\alpha_2)$ respectively, where $\alpha_1, \alpha_2 > 0$. Then $\mathbf{Pr}[X_1 < X_2] = \frac{\alpha_1}{\alpha_1+\alpha_2}$.*

▶ **Lemma 30** ([11], Lemma 4.3.). *Let $X_1, X_2, \ldots$ be i.i.d. copies of an exponentially distributed random variable such that $\min_{n\in\mathbb{N}} X_n$ is exponentially distributed with rate α. Then, for all $i \in \mathbb{N}$, the random variable $Y_i = \min^{(i)}_{n\in\mathbb{N}} X_n$ is identically distributed to $Z_i = Z_{i-1} + Exp(\alpha)$ where $Z_0 := 0$.*

▶ **Lemma 31** ([11], Lemma 4.4.). *Let $Y_1, Y_2, \cdots$ be i.i.d. copies of an exponentially random variable such that $\min_{n\in\mathbb{N}} Y_n$ is exponentially distributed with rate α. Let X be exponentially distributed with rate $1-\alpha$. Let $W = \{i \in \mathbb{N} : Y_i < X\}$. Then $\mathbf{Pr}[|W| = \ell] = \alpha^\ell(1-\alpha)$.*