Optimal Strategic Mining Against Cryptographic Self-Selection in Proof-of-Stake

MATHEUS V. X. FERREIRA, Harvard University, USA
YE LIN SALLY HAHN, Princeton University, USA
S. MATTHEW WEINBERG, Princeton University, USA
CATHERINE YU, Princeton University, USA

Cryptographic Self-Selection is a subroutine used to select a leader for modern proof-of-stake consensus protocols. In cryptographic self-selection, each round r has a seed Q_r. In round r, each account owner is asked to digitally sign Q_r, hash their digital signature to produce a credential, and then broadcast this credential to the entire network. A publicly-known function scores each credential in a manner so that the distribution of the lowest scoring credential is identical to the distribution of stake owned by each account. The user who broadcasts the lowest-scoring credential is the leader for round r, and their credential becomes the seed Q_{r+1}. Such protocols leave open the possibility of manipulation: a user who owns multiple accounts that each produce low-scoring credentials in round r can selectively choose which ones to broadcast in order to influence the seed for round $r + 1$. Indeed, the user can pre-compute their credentials for round $r + 1$ for each potential seed, and broadcast only the credential (among those with low enough score to be leader) that produces the most favorable seed.

We consider an adversary who wishes to maximize the expected fraction of rounds in which an account they own is the leader. We show such an adversary always benefits from deviating from the intended protocol, regardless of the fraction of the stake controlled. We characterize the optimal strategy; first by proving the existence of optimal positive recurrent strategies whenever the adversary owns last than $\frac{3-\sqrt{5}}{2} \approx 38\%$ of the stake. Then, we provide a Markov Decision Process formulation to compute the optimal strategy.

CCS Concepts: • **Theory of computation → Algorithmic game theory and mechanism design**; • **Security and privacy → Cryptography**; • **Information systems → Digital cash**.

Additional Key Words and Phrases: blockchain, cryptocurrency, proof-of-Stake, cryptography, strategic mining

ACM Reference Format:
Matheus V. X. Ferreira, Ye Lin Sally Hahn, S. Matthew Weinberg, and Catherine Yu. 2022. Optimal Strategic Mining Against Cryptographic Self-Selection in Proof-of-Stake. In *Proceedings of the 23rd ACM Conference on Economics and Computation (EC '22), July 11–15, 2022, Boulder, CO, USA.* ACM, New York, NY, USA, 26 pages. https://doi.org/10.1145/3490486.3538337

EC '22, July 11–15, 2022, Boulder, CO, USA

ACM ISBN 978-1-4503-9150-4/22/07…$15.00
https://doi.org/10.1145/3490486.3538337

1 INTRODUCTION

Proposed by Nakamoto in 2008, Bitcoin was one of the major innovations in peer-to-peer networks for electronic transactions [15]. Bitcoin is a decentralized currency without an administrator where anyone is free to join and submit or validate transactions in a public ledger. To modify the state of the ledger, users must publicly broadcast transactions. Those transactions are included in a block by miners and validated via a Proof-of-Work (PoW). The significant computational resources required to validate a block coupled with the reward for validating blocks create an economic incentive for miners to validate blocks correctly.

Unfortunately, Bitcoin is not without limitations. The proof-of-work consensus mechanism was designed to be energy-intensive—the global energy consumption from all Bitcoin miners exceeds that of all but 26 countries [1]. Moreover, the economies of scale from designing and purchasing large quantities of specialized hardware for proof-of-work mining has demonstrated that Bitcoin is more prone to centralization than initially thought [2]. In attempt to address the limitations of proof-of-work, many alternative blockchain designs have been proposed [6, 11, 13, 17].

In particular, proof-of-stake blockchains replace proof-of-work by a randomized mechanism to select a miner as the leader who proposes a new block. To avoid sybil attacks, where an adversary impersonates multiple identities, the mechanism hopes to choose a miner with probability proportional to their fraction of owned coins (commonly referred to as their stake in the system). The main challenge for such systems is to sample a miner without sacrificing decentralization and, at the same time, preserve the security and economic properties of the blockchain.

Several proposals for "Bitcoin-like" longest-chain proof-of-stake protocols have been made, but several drawbacks to this approach still exist. For example, [8] considers a longest-chain proof-stake blockchain that requires a randomness beacon [12, 16], and proves that this qualitatively preserves the mining incentives of Bitcoin, but the need for a trusted external randomness beacon is prohibitive in most settings. Without a trusted external randomness beacon, proof-of-stake implementations often rely on using the own blockchain as a source of pseudorandomness. Because miners can often predict the randomness in such protocols, Brown-Cohen et al. [3] showed that a large class of longest-chain proof-of-stake protocols are vulnerable to profitable deviations that they term "predictable selfish mining." Thus, the current state of longest-chain proof-of-stake cryptocurrencies must either: (a) propose a trusted randomness beacon, or (b) propose clever applications of cryptography to minimize the ability of strategic miners to manipulate the blockchain pseudorandomness. While active research agendas aim to address both (a) and (b), these are currently notable barriers to longest-chain proof-of-stake protocols.

One alternative design that relies on neither a trusted beacon, nor even an underlying longest-chain protocol, is cryptographic self-selection. This procedure is adopted in blockchains like Algorand [6, 10]. In cryptographic self-selection, each round r has a seed Q_{r-1}, used to sample the leader ℓ_r to propose the r-th block in the blockchain. In the ideal case where Q_{r-1} is unbiased, a clever cryptographic construction ensures that a miner owning a $\alpha \in [0, 1]$ fraction of the coins has a probability α of becoming the leader.

Unlike longest-chain blockchains, blockchains using cryptographic self-selection are immutable because once the leader validates B_r, B_r cannot be modified. Nevertheless, one limitation of such protocols is that the leader for round r may have some influence over the seed Q_{r+1} for round $r + 1$. For example, Chen and Micali [6] note that it is possible for an adversary to bias seeds in future rounds in Algorand's cryptographic self-selection. In this work, we study quantitatively the limits of how much these deviations might benefit an economically-motivated adversary. Specifically, we assume that the adversary wishes to maximize the fraction of rounds during which they are the leader. If the adversary were honest, this would be exactly an α fraction. We seek to understand

$f(\alpha) \geq \alpha$, the maximum fraction of rounds that an adversary with an α fraction of the stake can lead in expectation.[1]

1.1 Overview of Results and Roadmap

Our main contributions are as follows:

- First, we provide a formal stochastic process that captures the game played by strategic players who want to be leader as often as possible in a cryptographic self-selection protocol. We provide a detailed, formal description of the game in Section 2, and prove several basic facts in Section 3.1. These sections provide a clean stochastic process whose analysis directly informs the rewards achievable by strategic players in blockchain based on cryptographic self-selection.
- In this game, it is a priori possible that an *extremely* strong strategy exists that lets a player with an $\alpha < 1$ fraction of the stake win unboundedly-many rounds in a row in expectation. We prove that when $\alpha < \frac{3-\sqrt{5}}{2} \approx 0.38$, this is not possible, and the optimal fraction of rounds won by a strategic player with $\alpha < \frac{3-\sqrt{5}}{2}$ is < 1. Note that in many protocols, including [6], that use cryptographic self-selection, owning $\alpha > 1/3$ of the stake is already enough to subvert consensus, so the $\alpha < 1/3$ is the most relevant range where strategic mining is a concern. We prove this in Section 4.
- We pose a simple strategy, the 1-Lookahead strategy, which *strictly outperforms the honest strategy for all* α. We fully analyze the expected reward of this strategy in Section 5.
- Finally, we describe how the optimal strategy can be found by solving a series of MDPs. As the MDPs are infinite, this unfortunately does not immediately give an efficient algorithm. This appears in Section 6.

1.2 Related Work

Seminal work of [7] established that strategic mining in Bitcoin is possible. Hundreds of followup works pushed their ideas in various directions. One notable work, which is similar in spirit to ours, is [19], who nail down the optimal achievable reward for a miner who has an α-fraction of the computational power. Follow-up works such as [3, 8] consider similar questions for proof-of-stake instead of proof-of-work, but to the best of our knowledge, *all prior work in this agenda considers longest-chain protocols.* In comparison to this line of work, ours is the first (to our knowledge) to consider a formal model of strategic behavior in cryptographic self-selection, which is used in protocols based on Byzantine consensus.

Chen and Micali [6] develop the theoretical protocol for Algorand, and propose that manipulations of the cryptographic self-selection protocol may be possible. They do not propose a concrete manipulation, but do upper bound the maximum fraction of rounds that certain kinds of adversaries can possibly be leader. In comparison to their work, our work proposes a formal model to capture the entire strategy space in cryptographic self-selection protocols, and our 1-Lookahead strategy also provides the first concrete profitable manipulation.

[1] This is the same objective in prior work [3, 7, 8, 19]. In prior work, this objective function was motivated by the block reward associated with creation of each block. Even if a protocol has no block reward there is still *some* economic incentive associated with creating a block. This could be due to transaction fees [9, 18], or side contracts that the leader is able to execute in deciding what to include. We do not explicitly model the direct connection between being a leader in a round and the monetary reward, and treat this per-block incentive as exogenous.

2 BACKGROUND AND SETUP

In this section, we provide our formal model and some preliminary observations. Our model captures the cryptographic self-selection protocol of [6], but we remind the reader that our model *only* concerns leader selection—this process is independent of block creation, consensus, etc.

2.1 Blockchain Protocols with Finality

Many modern proof-of-stake blockchain protocols, such as Algorand [6, 10], differ significantly from the longest-chain protocols like Bitcoin. All blockchain protocols maintain a ledger, which is a sequence of blocks $B_0, B_1, \dots, B_t, \dots$. Protocols with *finality* differ from Bitcoin in that there are *no forks*. Once round t has concluded, there is a single well-defined B_t, which will stay fixed throughout eternity.

To produce the block for round t, blockchain protocols with finality run an underlying consensus protocol. These consensus protocols often require a *leader* ℓ_t, selected based on $B_1, \dots, B_{t-1}$, who gets to propose the block which could potentially be ratified as B_t. Note that because there are no forks, the leader ℓ_t is well-defined.

2.2 Cryptographic Self-Selection to Determine a Leader

One problem that any blockchain protocol with finality must resolve is how to determine ℓ_t as a function of $B_1, \dots, B_{t-1}$, and this must be done with care. For example, if there are N coins indexed from 1 to N, one naive proposal might simply declare the owner of coin $t \pmod{N+1}$ to be the leader at round t. This is vulnerable to a predictability attack: an attacker knows exactly which coins they need to own in which rounds, and can be solely responsible for block proposal for many rounds in a row. Another naive proposal might declare the owner of coin $\text{HASH}(B_{t-1}) \pmod{N+1}$ to be the leader at round t. This is vulnerable to a grinding attack: ℓ_{t-1} has many options for the contents they include in B_{t-1}, and can try arbitrarily many contents until $\text{HASH}(B_{t-1})$ results in a coin they own. In general, the goal of a leader-selection protocol is to pick a leader in a manner so that:

- When every user honestly follows the intended protocol, the distribution of each ℓ_t is proportional to the stake, and i.i.d. across rounds. That is, each user with an α fraction of the total stake is selected as the leader in round t with probability α, independently across rounds.
- A self-interested user has little ability to predict future rounds in which they could become the leader, or to increase the fraction of rounds in which they are the leader.

Cryptographic self-selection is a clever approach, used by Algorand [6, 10] to select a leader. Before defining the full protocol, we need two basic tools.

2.2.1 Tools for Cryptographic Self-Selection. One useful cryptographic tool towards cryptographic self-selection is a *verifiable random function* [14]. For the purposes of this paper, we'll use an *ideal* verifiable random function.

Definition 2.1 (Ideal Verifiable Random Function)**.** An Ideal Verifiable Random Function (Ideal VRF) satisfies the following properties:

- **Setup.** There is an efficient randomized process to produce a secret key $\textsc{sk}$ and a public key $\textsc{pk}$ that parameterize the function $f_{\textsc{sk}}(\cdot)$.
- **Private Computability.** There is an efficient algorithm $\mathcal{A}$ such that for all $\textsc{sk}$, $\mathcal{A}(x, \textsc{sk}) = f_{\textsc{sk}}(x)$ (that is, f can be efficiently computed with knowledge of the secret key $\textsc{sk}$).
- **Perfect Randomness.** For all inputs $x \neq y$, the random variables $f_{\textsc{sk}}(x)$ and $f_{\textsc{sk}}(y)$ are independent, and uniformly drawn from $[0, 1]$, conditioned on knowledge of $\textsc{pk}$. In particular,

this implies that $f_{\textsc{sk}}(x)$ is distributed uniformly on $[0,1]$ to any user who sees only PK, even after that user has seen any number of pairs $(x_1, f_{\textsc{sk}}(x_1)), \ldots, (x_k, f_{\textsc{sk}}(x_k))$, where $x_i \neq x,\ \forall i$.

- **Verifiable.** There exists an efficient algorithm V that takes as input x, y, PK and outputs YES if and only if $y = f_{\textsc{sk}}(x)$.

Intuitively, an Ideal VRF allows a holder of SK to draw a random number uniformly from $[0,1]$ in a way that is unpredictable to anyone without knowledge of SK, yet also in a verifiable manner. The distinction between an Ideal VRF and VRF lies in perfect randomness: it is generally not possible to have the random variables $\{f(x_1), \ldots, f(x_k)\}$ be *statistically* indistinguishable from independent uniformly random draws from $[0,1]$ conditioned on PK. VRFs used in practice instead provide that the distribution of $\{f(x_1), \ldots, f(x_k)\}$ be *computationally* indistinguishable from independent uniformly random draws from $[0,1]$, conditioned on PK.[2]

We omit a formal definition of (non-Ideal) VRFs, which is cumbersome and not relevant to our results. In particular, all proposed deviations work *even when the protocol has access to an Ideal VRF*, and therefore they also work when the protocol instead uses a VRF.

To have a simple example of a (non-Ideal) VRF in mind, consider any digital signature scheme and hash function. On input x, first, digitally sign x to obtain $\mathrm{SIG}(x)$, and then hash it (this is the VRF used in [6]). Indeed, with the secret key, a user can efficiently compute a digital signature of any x and hash it. Similarly, correct computation of the hash function can be efficiently verified by anyone, and correct computation of the digital signature can be efficiently verified by anyone with the public key. Any input $\mathrm{SIG}(x)$ to the hash function is mapped to a uniformly-random draw from $[0,1]$, independently of all other inputs, and the digital signature scheme ensures that anyone without the secret key cannot guess $\mathrm{SIG}(x)$, even with knowledge of x, and any number of input/output pairs to SIG.

A second tool we will need is a concept that enables the leader to be selected proportional to the stake, rather than uniformly at random among all accounts.

Definition 2.2 (Balanced Scoring Function)**.** A scoring function $S(\cdot,\cdot)$ is *balanced* if for all $n \in \mathbb{N}$ and all $\langle \alpha_1, \ldots, \alpha_n \rangle \in \mathbb{R}^n_{>0}$:

$$\Pr_{X_1,\ldots,X_n \leftarrow U([0,1]^n)}\left[\arg\min_{i\in[n]} \{S(X_i,\alpha_i)\} = j\right] = \frac{\alpha_j}{\sum_{\ell=1}^n \alpha_\ell}.$$

Observe that, if ties in arg min are broken lexicographically, this implies that for all α, the distribution of $S(X,\alpha)$ when X is drawn uniformly from $[0,1]$ must have no point-masses.

Intuitively, one can think of $\arg\min_i\{X_i\}$ as the winner of a random process when each $X_1, \ldots, X_n$ is drawn independently from the uniform distribution on $[0,1]$, denoted by $U([0,1])$, and each player is equally likely to win. A balanced scoring function allows us to redistribute the probability of winning to be proportional to α_i instead.

2.2.2 Cryptographic Self-Selection Protocol. Now, we define the cryptographic self-selection protocol, the leader-selection protocol analyzed throughout our paper.

Definition 2.3 (Cryptographic Self-Selection Protocol A)**.** The Cryptographic Self-Selection Protocol A (CSSPA) is the following:

- Every account i sets up an Ideal VRF with secret key $\textsc{sk}_i$ and public key $\textsc{pk}_i$. $\alpha_i \in [0,1]$ refers to the fraction of the total stake that account i owns.

[2] Also, any (pseudo) random number generator used in practice produces output that is uniformly random over $\{0,1\}^\lambda$ for large λ, rather than over $[0,1]$. For simplicity of exposition, we think of $\lambda \to \infty$. This again does not affect our results, except for error that is exponentially small in λ (due to the tiny possibility of ties).

- Q_r denotes the seed for round r. The initial seed is a uniformly random number in $[0, 1]$ constructed via a coin tossing protocol [4].
- In round r, each user i computes their credential $\textsc{Cred}_r^i := f_{\text{SK}_i}(Q_r)$. Every user can either broadcast, or not broadcast. B_r denotes the set of users who broadcast in round r.[3]
- There is a publicly-known balanced scoring function S. The leader ℓ_r for round r is

$$\arg\min_{i\in B_r}\{S(\textsc{Cred}_r^i, \alpha_i)\}.$$

- $Q_{r+1} := \textsc{Cred}_r^{\ell_r}$. That is, the seed for round $r + 1$ is the credential of the leader for round r.

We note a few quick observations about CSSPA:

- Aside from network/security/cryptography attacks, which are not the focus of this paper, the action space of a single account in each round is binary: broadcast your credential, or don't. A single player may own multiple accounts. Therefore, the actions a single player may take in our game is to: a) decide how to divide their stake among multiple accounts, and b) pick which subset of credentials to broadcast.
- We'll refer to the *honest strategy* as one which announces all credentials in every round.
- Assuming all players are honest, each leader is drawn i.i.d. and proportional to $\vec{\alpha}$. This follows immediately from the definition of Ideal VRF and balanced scoring function.
- Assuming that all players are honest, the protocol is robust to Sybil attacks. That is, a player who truly controls an α_i fraction of the total stake can put all of their funds into a single account, or split their funds arbitrarily over any number of accounts. No matter how they divide their funds, the probability that an account owned by this player is selected as leader is exactly α_i.
- Much analysis of CSSPA can be done agnostically to the particular balanced scoring function. For example, Proposition 2.1 establishes that our analysis holds for a wide class of "canonical" balanced scoring functions. In particular, our analysis chooses a particularly simple balanced scoring function for the benefit of tractability, but our analysis holds for the balanced scoring function used in [6] as well via Proposition 2.1.

Throughout our paper, we'll use the balanced scoring function $S(x, \alpha_i) := \frac{-\ln(x)}{\alpha_i}$. This allows us to leverage basic facts about independent draws from exponential distributions.

Definition 2.4 (Exponential Distribution)**.** The exponential distribution with rate α is the distribution with Cumulative Density Function (CDF) $F_\alpha(x) := 1 - e^{-\alpha x}$, for all $x \geq 0$. We refer to $\textsc{Exp}(\alpha)$ as one independent sample from the exponential distribution with rate α. For simplicity of notation in later calculations, we will denote by $\textsc{Exp}(0)$ to be a point-mass at $+\infty$.

Exponential distributions have many relevant properties that we remind the reader of in Appendix A.

Lemma 2.1. Define S so that $S(x, \alpha_i) := \frac{-\ln(x)}{\alpha_i}$. Then $S(\cdot, \cdot)$ is a balanced scoring function. Moreover, when x is drawn uniformly from $[0, 1]$, $S(x, \alpha_i)$ is distributed according to $\textsc{Exp}(\alpha_i)$.

Proof. We first show that $S(x, \alpha_i)$ is distributed according to $\textsc{Exp}(\alpha_i)$ when x is drawn uniformly from $[0, 1]$. This follows essentially because $\frac{-\ln(x)}{\alpha_i}$ is the inverse reverse-CDF of $\textsc{Exp}(\alpha_i)$. To see the claim, we compute the probability that $S(x, \alpha_i) > y$, for any y:

[3] In order to focus on the relevant aspects, we assume that any broadcast is received by all other users. This is consistent with prior work that focuses on the underlying incentives, and not distributed computing [3, 5, 7, 8].

$$\begin{aligned}\Pr_{x\leftarrow U([0,1])}[S(x,\alpha_i) > y] &= \Pr_{x\leftarrow U([0,1])}[\frac{-\ln(x)}{\alpha_i} > y]\\ &= \Pr_{x\leftarrow U([0,1])}[\ln(x) < -\alpha_i \cdot y]\\ &= \Pr_{x\leftarrow U([0,1])}[x < e^{-\alpha_i \cdot y}]\\ &= e^{-\alpha_i \cdot y}\end{aligned}$$

This means that the CDF of $S(x, \alpha_i)$, when x is drawn uniformly from $[0, 1]$, is exactly $1 - e^{-\alpha_i x}$, and therefore this distribution is equal to $\textsc{Exp}(\alpha_i)$. Now, the fact that $S(\cdot, \cdot)$ is a balanced scoring function follows from Corollary A.1 (which states that the minimum of $X_1, \ldots, X_n$, when each X_i is drawn independently from $\textsc{Exp}(\alpha_i)$ is equal to X_i with probability α_i, for all i). □

We conclude this section by formally establishing that our analysis extends to a broad class of scoring functions.

Definition 2.5. A scoring function S is *canonical* if:

- For all α, $S(\cdot, \alpha)$ is monotone decreasing on domain $(0, 1)$.
- For all n, and $\alpha_1, \ldots, \alpha_n$, the random variables $S(X, \sum_{i=1}^{n} \alpha_i)$ and $\min_{i=1}^{n}\{S(X_i, \alpha_i)\}$ are identically distributed when each $X, X_1, \ldots, X_n$ are i.i.d. from $U([0, 1])$.
- S is *continuous* in α. That is: for all x and α, if $\lim_{\beta\to\alpha} S(x, \beta)$ exists, then $S(x, \alpha) = \lim_{\beta\to\alpha} S(x, \beta)$.

Before proceeding, we give quick context for each bullet. Balanced scoring functions where $S(\cdot, \alpha)$ is not monotone decreasing exist, but assuming that $S(\cdot, \alpha)$ is monotone decreasing is w.l.o.g. Indeed, for any α, let F_α denote the CDF of the random variable $S(X, \alpha)$ when X is drawn uniformly from $[0, 1]$. Now consider redefining $S'(x, \alpha) := F_\alpha^{-1}(1 - x)$. Then the distribution of $S(X, \alpha)$ and $S'(X, \alpha)$ are identical, but $S'(\cdot, \alpha)$ is monotone decreasing. We conjecture that all balanced scoring functions satisfy the second two bullets, but we suspect that rigorously establishing this will require significant analysis. As this is not the focus of our paper, we instead treat these bullets as reasonable assumptions. Indeed, the balanced scoring function used by Algorand is canonical.

Proposition 2.1. The game induced by CSSPA with a canonical balanced scoring function is independent of the particular canonical balanced scoring function used. Formally, for two distinct canonical balanced scoring functions S, S', the games induced by CSSPA are identical. Specifically, for all players i, there is a bijective mapping f from strategies of player i in the CSSPA with S to strategies of player i in the CSSPA with S'. For all i, the payoff that player i receives in the CSSPA with S under strategy profile $\vec{s}$ is equal to the payoff that i receives in the CSSPA with S' under strategy profile $\langle f_i(s_i)\rangle_i$.

A complete proof of Proposition 2.1 appears in Appendix B.

3 OUR MODEL: STRATEGIC MINING IN CRYPTOGRAPHIC SELF-SELECTION

This section formally defines our model and, in particular, the optimization problem considered by a strategic player. Like prior work [5, 7, 8], we consider a *single* strategic player who is best responding to a profile of honest players. The purpose of this analysis, like in prior work, is to understand the maximum disruption that can be caused when a $1 - \alpha$ fraction of the stake is owned by honest players, and an α fraction of the stake is owned by strategic players.[4] We now formalize the strategy space of the strategic player.

[4] That is, the worst-case scenario, among all scenarios where a $1 - \alpha$ fraction of the stake is held by honest players, is when there is a single-strategic player with an α fraction of the stake. Our goal, like prior work, is to understand this scenario.

Definition 3.1 (Strategy Space in CSSPA). CSSPA is parameterized by α, the fraction of stake owned by the strategic player, $\vec{\alpha}$ the distribution of remaining stake among honest players, and $\beta \in [0,1]$, the network connectivity strength of the strategic player. We'll refer to this as a β-strong player. When $\beta = 1$, we'll simply refer to the player as *strong*, and when $\beta = 0$ we'll refer to the player as *weak*. The strategic player knows $\alpha, \beta, \vec{\alpha}$.

In round r, the strategic player in CSSPA has the following information and makes the following decisions, in order:

(1) The strategic player can distribute their total stake of α arbitrarily among as many accounts as they desire. Refer to this set as A.
(2) The strategic player knows Q_r, and knows that all other players are honest.
(3) For a set of accounts B such that $B \cap A = \emptyset$, and $\sum_{j \in B} \alpha_j = \beta \cdot (1-\alpha)$, the strategic player learns $\textsc{Cred}_r^j$, for all accounts $j \in B$. The strategic player *does not learn* $\textsc{Cred}_r^j$ for any $j \notin A \cup B$ (that is, the player only knows that each $S(\textsc{Cred}_r^j, \alpha_j)$ will be drawn from $\textsc{Exp}\left(\alpha_j\right)$, independently).
(4) Observe that the strategic player can compute $\textsc{Cred}_r^i$ and also $S(\textsc{Cred}_r^i, \alpha_i)$, for all accounts $i \in A$.
(5) Observe further that for all $j \in A \cup B$, $\textsc{Cred}_r^j$ is a possible seed for Q_{r+1}. So the player can *also* pre-compute a hypothetical $\textsc{Cred}_{r+1}^i$, assuming $Q_{r+1} = \textsc{Cred}_r^j$, for each account $i \in A$ and $j \in A \cup B$. But observe that the strategic player *cannot* execute this computation for $i \notin A$ (because they cannot compute the ideal VRF for accounts $\notin A$).
(6) More generally, for any k, and any list of accounts $\langle i_0, \ldots, i_k \rangle$ such that $i_0 \in A \cup B$, and each $i_j \in A$ for all $j > 0$, the player can *also* pre-compute what $\textsc{Cred}_{r+k}^{i_k}$ would be, assuming that $\ell_{r+j} = i_j$ for all $j \in \{0, \ldots, k-1\}$.
(7) The strategic player selects a subset $A_r \subseteq A$, and broadcasts all credentials in A_r.

We will consider optimal strategies for all $\alpha, \beta, \vec{\alpha}$. Note that the role of β differentiates how much information they know about other players' credentials before deciding which credentials of their own to broadcast. Before getting into our main analysis, we prove some basic facts about optimal strategies in this model.

3.1 Basic Facts on Optimal Strategies

First, we define the reward achieved by a particular strategy π, which the strategic player aims to optimize. A priori, the reward can depend on α, β, and the distribution of the remaining $(1-\alpha)$ fraction of stake, $\vec{\alpha}$.

Definition 3.2 (Reward of a Strategy). A strategy π prescribes an action to take during each round. $X_r^{\alpha,\beta,\vec{\alpha}}(\pi)$ is an indicator random variable for the event that the strategic player is the leader during round r, when the game with parameters $\alpha, \beta, \vec{\alpha}$ is played. The reward of a strategy π is simply the expected fraction of rounds where the strategic player is the leader. We drop the superscript and write $X_r(\pi)$ whenever $\alpha, \beta, \vec{\alpha}$ is clear from context. Formally:

$$\textsc{Rev}_{\alpha,\beta,\vec{\alpha}}(\pi) = \mathbb{E}\left[\liminf_{R \to \infty} \frac{\sum_{r=1}^{R} X_r^{\alpha,\beta,\vec{\alpha}}}{R}\right] \tag{1}$$

The expectation is taken over the randomness in the Ideal VRFs in every round, assuming that all non-strategic miners are honest. We use the notation $\textsc{val}(\alpha, \beta, \vec{\alpha}) := \sup_\pi\{\textsc{Rev}_{\alpha,\beta,\vec{\alpha}}(\pi)\}$. We say that a strategy π is ε-optimal for parameters $\alpha, \beta, \vec{\alpha}$ if $\textsc{Rev}(\pi) \geq \textsc{val}(\alpha, \beta, \vec{\alpha}) - \varepsilon$.

Next, we produce a series of refinements concerning ε-optimal strategies, which will allow us to greatly simplify the analysis of strategies in CSSPA. First, we observe that the strategic player need not consider any set with $|A_r| > 1$.

Observation 3.1. For any strategy π, there is another strategy π' that results in exactly the same leaders as π in every round (and therefore has $\textsc{Rev}_{\alpha,\beta,\vec{\alpha}}(\pi') = \textsc{Rev}_{\alpha,\beta,\vec{\alpha}}(\pi)$) for all $\alpha, \beta, \vec{\alpha}$), and always selects an A_r with $|A_r| \leq 1$.

Proof. Observe that the strategic player can compute $S(\textsc{Cred}_r^i, \alpha_i)$ for all $i \in A$. If they broadcast a set $A_r \neq \emptyset$, then the leader will be $i^* := \arg\min_{i \in A_r}\{S(\textsc{Cred}_r^i, \alpha_i)\}$ if and only if $S(\textsc{Cred}_r^{i^*}, \alpha_{i^*}) < S(\textsc{Cred}_r^j, \alpha_j)$ for all $j \notin A$. Observe that this is exactly what would happen if the strategic player instead broadcast only $\{i^*\}$ instead. So π' will broadcast only $\{i^*\}$, and this results in the same leader as using π.

If instead the strategic player chooses to broadcast $A_r = \emptyset$, then π' will broadcast $\emptyset$ as well. This clearly results in the same leader as using π, because the actions are identical.

The leader is the same in both cases, and π' only ever broadcasts (at most) a single credential. □

Next, we show that optimal strategies split their stake among as many accounts as possible.

Lemma 3.1. Consider a strategy π where strategic player divides their stake into n wallets with stake $\alpha_i > 0$, for $i \in [n]$. Then there is a strategy π' where the strategic player instead divides their stake into $2n$ wallets with stake $\alpha'_i > 0$, for all $i \in [2n]$, and $\textsc{Rev}(\pi') = \textsc{Rev}(\pi)$.

Proof. The strategy π' defines $2n$ wallets with stake

$$\alpha'_i = \begin{cases} \frac{\alpha_i}{2} & \text{for } i \leq n, \\ \frac{\alpha_{i-n}}{2} & \text{for } n < i \leq 2n. \end{cases}$$

Observe that, conditioned on Q_r, $S(\textsc{Cred}_r^i, \alpha_i)$ is distributed according to $\textsc{Exp}\,(\alpha_i)$, independently for all i. Similarly, $S(\textsc{Cred}_r^j, \alpha'_j)$ is distributed according to an $\textsc{Exp}\left(\alpha'_j\right)$, independently for all j. Define now the random variable $j(i) := \arg\min\{S(\textsc{Cred}_r^i, \alpha'_i), S(\textsc{Cred}_r^{n+i}, \alpha'_{n+i})\}$, and denote by $Y_r^i := S(\textsc{Cred}_r^{j(i)}, \alpha'_{j(i)})$. Then by Lemma A.1, Y_r^i is distributed according to $\textsc{Exp}\left(\alpha'_i + \alpha'_{n+i}\right) = \textsc{Exp}\,(\alpha_i)$, independently for all $i \in [n]$. Therefore, Y_r^i and $S(\textsc{Cred}_r^i, \alpha_i)$ are *identically distributed.* Therefore, we can couple executions of π and π' so that $Y_r^i = S(\textsc{Cred}_r^i, \alpha_i)$ for all r, i, and also so that $S(\textsc{Cred}_r^j, \alpha_j)$ is identical for all $r, j \notin A$.

Consider now the strategy π' that does the following. If π does not broadcast a credential, then π' also does not broadcast a credential. If π broadcasts i^*, then π' broadcasts $j(i^*)$. Observe now that the score of the credential broadcast by π and π' is identical (due to the coupling), and the scores of credentials broadcast by the honest players are also identical. Therefore, $\ell_r = i^*$ under π if and only if $\ell_r = j(i^*)$ under π'. Moreover, Q_{r+1} is identical under both executions. We have therefore coupled the executions of π and π' so that $X_r^{\alpha,\beta,\vec{\alpha}}(\pi) = X_r^{\alpha,\beta,\vec{\alpha}}(\pi')$ for all r, and therefore $\textsc{Rev}^{\alpha,\beta,\vec{\alpha}}(\pi) = \textsc{Rev}^{\alpha,\beta,\vec{\alpha}}(\pi')$. □

Next, we argue that is w.l.o.g. to consider two honest players, one with a fraction $\beta \cdot (1-\alpha)$ of the stake, and the other with fraction $(1-\beta) \cdot (1-\alpha)$ of the stake.

Observation 3.2. For any α, β, define $\vec{\alpha}'$ to have two honest players, one with $\alpha'_1 = \beta \cdot (1-\alpha)$, and another with $\alpha'_2 = (1-\beta) \cdot (1-\alpha)$. Then for any strategy π, $\textsc{Rev}^{\alpha,\beta,\vec{\alpha}}(\pi) = \textsc{Rev}^{\alpha,\beta,\vec{\alpha}'}$.

Proof. Let $Y_B := \min_{j \in B}\{S(\textsc{Cred}_r^j, \alpha_j)\}$, and $Y_C := \min_{j \notin (A \cap B)}\{S(\textsc{Cred}_r^j, \alpha_j)\}$. Then by Lemma A.1, Y_B is distributed according to $\textsc{Exp}\,(\beta \cdot (1-\alpha))$, and Y_C is distributed according to $\textsc{Exp}\,((1-\beta) \cdot (1-\alpha))$.

Therefore, we can couple Y_B and Y_C in the execution with $\vec{\alpha}$ with $S(\textsc{Cred}_r^1, \alpha_1')$ and $S(\textsc{Cred}_r^2, \alpha_2')$ in the execution with $\vec{\alpha}'$.

Observe, now, that the seed for round $r+1$ in the execution with $\vec{\alpha}$ will be the minimum of Y_B, Y_C, and the score of the credential broadcast by the strategic player. In the execution with $\vec{\alpha}'$, the seed for round $r+1$ will be the minimum of $S(\textsc{Cred}_r^1, \alpha_1')$, $S(\textsc{Cred}_r^2, \alpha_2')$, and the score of the credential broadcast by the strategic player. Therefore, Q_{r+1} is the same in both executions. Moreover, we also have $X_r^{\alpha,\beta,\vec{\alpha}}(\pi) = X_r^{\alpha,\beta,\vec{\alpha}'}(\pi)$. We have therefore coupled the executions with $\vec{\alpha}$ and $\vec{\alpha}'$ so that $X_r^{\alpha,\beta,\vec{\alpha}}(\pi) = X_r^{\alpha,\beta,\vec{\alpha}'}(\pi)$ for all r, and therefore $\textsc{Rev}^{\alpha,\beta,\vec{\alpha}}(\pi) = \textsc{Rev}^{\alpha,\beta,\vec{\alpha}'}(\pi)$. □

We make one final observation, which will simplify later definitions (it is not necessary for our analysis, but greatly simplifies Definition 4.1).

Observation 3.3. For any strategy π, there is another strategy π' satisfying $\textsc{Rev}^{\alpha,\beta,\vec{\alpha}}(\pi) = \textsc{Rev}^{\alpha,\beta,\vec{\alpha}}(\pi')$ for all $\alpha, \beta, \vec{\alpha}$, and also such that in any round r where the player learns in step (3) that $\min_{j\in B}\{S(\textsc{Cred}_r^j, \alpha_j)\} < \min_{j\in A}\{S(\textsc{Cred}_j^r, \alpha_j)\}$, π' does no computation in steps (4)-(6).

Proof. Observe that if $\min_{j\in B}\{\textsc{Cred}_r^j\} < \min_{j\in A}\{\textsc{Cred}_r^j\}$, then the seed Q^{r+1} will be equal to the minimum credential among all honest nodes, no matter what the strategic player chooses to broadcast. So no matter what they do this round, they cannot affect Q^{r+1}. Because the strategic player's actions during round r have no impact on the game, they can shift any computation they originally planned to do in round r later to round $r+1$. This results in a strategy π' that results in identical seeds in every round as π, but that does no computation during rounds where their action has no impact. □

We now state the strategy space of the refined CSSPA.

Definition 3.3 (Refined CSSPA). The refined CSSPA is parameterized by α, the fraction of stake owned by the strategic player, and $\beta \in [0,1]$, the network connectivity strength of the strategic player. There are two honest players B and C. B owns a $\beta \cdot (1-\alpha)$ fraction of the stake, and C owns a $(1-\beta)\cdot(1-\alpha)$-fraction of the stake.

In round r, the strategic player in CSSPA has the following information and makes the following decisions, in order:

(1) The strategic player can distribute their total stake of α arbitrarily among as many accounts as they desire. Refer to this set as A.
(2) The strategic player knows Q_r, and knows that all other players are honest.
(3) The strategic player learns $\textsc{Cred}_r^B$. The strategic player *does not learn* $\textsc{Cred}_r^C$ (that is, the player only knows that $S(\textsc{Cred}_r^C, (1-\beta)\cdot(1-\alpha))$ will be drawn from $\textsc{Exp}\left((1-\beta)\cdot(1-\alpha)\right)$, independently).
(4) Observe that the strategic player can compute $\textsc{Cred}_r^i$ and also $S(\textsc{Cred}_r^i, \alpha_i)$, for all accounts $i \in A \cup \{B\}$. For any k, and any list of accounts $\langle i_0, \ldots, i_k\rangle$ such that $i_0 \in A \cup \{B\}$ and $i_j \in A$ for all $j > 0$, the player can *also* pre-compute what $\textsc{Cred}_{r+k}^{i_k}$ would be, assuming that $\ell_{r+j} = i_j$ for all $j \in \{0, \ldots, k-1\}$. If the strategic player learned in Step (3) that $S(\textsc{Cred}_r^B, \beta\cdot(1-\alpha)) < \min_{j\in A}\{S(\textsc{Cred}_r^j, \alpha_j)\}$, then the player does no computation.
(5) The strategic player selects an account i^* to broadcast, or decides not to broadcast.

We let $\textsc{Rev}_{\alpha,\beta}(\pi)$, $\textsc{val}(\alpha,\beta)$ denote the reward of a strategy π in the refined CSSPA, and the optimal reward, respectively.

Based on the observations in this section, we conclude the following:

Corollary 3.1. For all $\alpha, \beta, \vec{\alpha}, \pi$: $\textsc{Rev}_{\alpha,\beta}(\pi) = \textsc{Rev}_{\alpha,\beta,\vec{\alpha}}(\pi)$. Therefore, $\textsc{val}(\alpha,\beta) = \textsc{val}(\alpha,\beta,\vec{\alpha})$ as well.

4 EXISTENCE OF OPTIMAL RECURRENT STRATEGIES

Recall we bootstrap the initial seed Q_0 to be drawn from $U[0,1]$ via a distributed coin tossing protocol. Hence Q_0 is an unbiased seed since it does not favor any player. Formally, we say a seed Q_{r-1} is *unbiased* if substituting Q_{r-1} by a fresh independent sample from $U[0,1]$ results in the same distribution for $X_r(\pi), X_{r+1}(\pi), \ldots$ conditioned on all the queries to f_{SK_i} for all i up to round $r-1$. Another interpretation is that the adversary did not query any f_{SK_i} on Q_{r-1} before round r begins which suggests the adversary is indifferent about replacing Q_{r-1} for a fresh sample from $U[0,1]$.

The adversary has a probability at most α of becoming the leader for round r if Q_{r-1} is unbiased because the probability an honest miner samples the lowest scoring credential is equal to $1-\alpha$—the adversary can only reduce their chances of being a leader by not broadcasting their credentials.

How can the adversary build a biased Q_r provided Q_{r-1} is unbiased? For some intuition, suppose $\beta = 1$, and the adversary has the lowest scoring credential for round r. In other words, the adversary observes the credentials of all honest miners and knows that if they broadcast some credential $\textsc{Cred}_r^{i^*}$, $i^* \in A$ becomes the leader for round r. However, the adversary also has the option to not broadcast any credential, in which case, some account B becomes the leader. Note that the adversary already knows $\textsc{Cred}_r^B$ before deciding if they will broadcast $\textsc{Cred}_r^{i^*}$ or not (the assumption $\beta = 1$ implies the adversary is well connected and get to see all other credentials before taking any action). Then, the adversary queries f_{SK_i} on $\textsc{Cred}_r^{i^*}$ and $\textsc{Cred}_r^B$ for all $i \in A$ and observes which seed would be more favorable for round $r+1$ (would allow the adversary to sample credentials with the lowest scores for round $r+1$). This concludes our example, and in Section 5, we provide a complete description of one such strategy. As a takeaway, the *the adversary can bias the seed Q_r unless the credential with the lowest score comes from an account $j \notin A$.*

It will be convenient to ask when the game reaches a round $\tau \geq 1$ where $Q_{\tau+1}$ is unbiased given that Q_0 is unbiased.

Definition 4.1 (Stopping Time). We call a round τ a *stopping time* if for all possible strategies π, the distribution of $\{X_r(\pi)\}_{r>\tau}$, conditioned on Q_τ and all information the adversary has during round τ, is identical to the distribution of $\{X_r(\pi)\}_{r>\tau}$ after replacing $Q_{\tau+1}$ with a uniformly random draw from $[0,1]$. That is, τ is a stopping time if the game effectively resets at round $\tau+1$, because the adversary was unable to bias the distribution of $Q_{\tau+1}$.

We now state the main way in which stopping times arise.

Observation 4.1. Let τ be a round such that the adversary does not query any VRF on $Q_{\tau+1}$ during any round $\leq \tau$. Then τ is a stopping time.

Proof. Because the adversary has not queried $Q_{\tau+1}$ on any VRF, this means that the adversary currently believes that every future query to any VRF on $Q_{\tau+1}$ is independently drawn from $U([0,1])$ (by definition of VRF). Replacing $Q_{\tau+1}$ with any other seed that has not been queried by the adversary has exactly the same distribution. In particular, with probability 1, a uniformly random draw from $[0,1]$ has not been queried by the adversary in any previous round, and therefore τ is a stopping time. □

Definition 4.2 (Positive Recurrence). Let $\tau \geq 1$ be the first stopping time induced by π. We say π is *positive recurrent* if $\mathbb{E}[\tau] < \infty$.

Let $\tau_0, \tau_1, \ldots$ be a sequence of stopping times. Since we can assume the adversary's strategy resets whenever a stopping time is reached, $\tau_1 - \tau_0, \tau_2 - \tau_1, \ldots$ and $\sum_{r=\tau_0+1}^{\tau_1} X_r(\pi), \sum_{r=\tau_1+1}^{\tau_2} X_r(\pi), \ldots$ are sequences of i.i.d. random variables. The following result simplifies the expression for revenue for positive recurrent strategies:

Lemma 4.1. Let π be positive recurrent. Then $\textsc{Rev}(\pi) = \frac{\mathbb{E}\left[\sum_{r=1}^{\tau} X_r(\pi)\right]}{\mathbb{E}[\tau]}$ where τ is a stopping time.

Proof. Let $\tau_0 = 0, \tau_1, \tau_2, \ldots$ be the sequence of stopping times and let $N(t)$ be the index for the most recent stopping time by time t. Then

$$\textsc{Rev}(\pi) = \mathbb{E}\left[\liminf_{T\to\infty} \frac{\frac{1}{N(T)}\left(\sum_{r=\tau_{N(T)}}^{T} X_r(\pi) + \sum_{i=1}^{N(T)} \sum_{r=\tau_{i-1}+1}^{\tau_i} X_r(\pi)\right)}{\frac{1}{N(T)}\left((T - \tau_{N(T)}) + \sum_{i=1}^{N(T)} (\tau_i - \tau_{i-1})\right)}\right]$$

Since $N(T) \to \infty$ as $T \to \infty$, the statement follows from the strong law of large numbers (Lemma A.4). □

Lemma 4.1 provides a nice characterization for the revenue of positive recurrent strategies which will be critical when studying optimal strategies. In the rest of this section, we aim to show a sufficient condition for the existence of optimal positive recurrent strategies by proving the following informal claim: *for any strategy π, let $\tau \geq 1$ be the first round where* $\arg\min_{i\in[n]}$ *$S(\textsc{Cred}^i_\tau, \alpha_i) \notin A$, then τ is a stopping time and* $\mathbb{E}[\tau] < \infty$ *for $\alpha < \frac{3-\sqrt{5}}{2}$* ≈ 0.38.

Definition 4.3 (Forced stopping time)**.** Consider round r with seed Q_r. If $\arg\min_{i\in[n]} \mathrm{S}(\textsc{Cred}^i_r, \alpha_i) \notin A$, we say r is a *forced stopping time* with respect to Q_r.

Lemma 4.2. If r is a forced stopping time, then r is a stopping time.

Proof. The leader $\ell_r \notin A$, because both B and C always broadcast their credentials, and one of them has the lowest score. Let j^* refer to the account in $\{B, C\}$ with minimum score. Then $Q_r = \textsc{Cred}^{j^*}_r$ regardless of the adversary's action.

Now, observe that the probability that Q_{r+1} has been any previous credential in any previous round $< r$ is 0 (because all credentials are drawn uniformly from $[0, 1]$ when drawn). Moreover, *because $\ell_r \notin A$*, the adversary cannot possibly have known Q_{r+1} prior to round r. This is because the adversary cannot compute the VRF of ℓ_r, and ℓ_r only broadcasts Q_{r+1} during round r. Finally, the adversary did not query Q_{r+1} after learning Q_{r+1} during round r because either the minimum account was B (in which case, by definition of step (4), the adversary did not query B), or the minimum account was C (in which case, the adversary does not have a step to query any VRFs during round r after learning C. Therefore, the adversary certainly did not query Q^{r+1} after learning Q^{r+1}.

The only remaining possibility is that the adversary had previously decided to query Q^{r+1} at a point when all they know is that Q^{r+1} is drawn independently from $U([0, 1])$, conditioned on inducing the minimum credential for round r. As this distribution is continuous (even after any conditioning), the probability that it outputs any particular credential is 0. Therefore, assuming that the adversary queries a finite number of inputs across all previous rounds, the probability that it has previously queried any VRF on Q^{r+1} during any previous round is also 0.

Therefore, Q^{r+1} has not been queried by the adversary in rounds $\leq r$, and r is a stopping time. □

4.1 The Branching Process

Next, we aim to show that the expected value of the forced stopping time is finite whenever the adversary owns at most 38% of the stake. Fix the seed Q_{r-1} and let $j^* = \arg\min_{j\notin A} \mathrm{S}(\textsc{Cred}^j_r, \alpha_j)$, the honest account with lowest score when the seed is Q_{r-1}. Let $W(Q_{r-1})$ denote all the accounts that could become leaders during round r when the seed is Q_{r-1}:

$$W(Q_{r-1}) = \{j^*\} \cup \{i \in A : \mathrm{S}(\textsc{Cred}^i_r, \alpha_i) < \mathrm{S}(\textsc{Cred}^{j^*}_r, \alpha_{j^*})\}$$

The distribution of $|W(Q_{r-1})|$ is related to the growth distribution in a Galton Watson branching process [20]. To see this, consider a tree $\text{Tree}(Q_0)$ where each node stores a seed. We give a recursive definition for $\text{Tree}(Q_0)$. Initialize the tree to contain only the root Q_0, which we color black. Then while $\text{Tree}(Q_0)$ contains some black node Q:

- If $|W(Q)| \geq 2$, for each $i \in W(Q)$, we append the edge $(Q, f_{\text{SK}_i}(Q))$ to $\text{Tree}(Q_0)$. Color Q red and color $f_{\text{SK}_i}(Q)$ black.
- If $|W(Q)| = 1$, color Q red.

Intuitively, a node is colored red without appending new edges whenever that node is a forced stopping time. A node is colored red after appending a new edge if it is not a forced stopping time (and then we need to recurse on each possible subgame induced by each possible seed).

The height of $\text{Tree}(Q_0)$ gives an upper bound for how long it takes for a game starting with seed Q_{r-1} to reach a forced stopping time. To see this, consider an *omniscient* adversary, who knows all secret keys (and therefore can query all VRFs in any round). Even this omniscient adversary can bias the next $k \geq 1$ rounds, if and only if $|W(Q_{r-1})| \geq 2$ (the adversary has at least two options for the seed Q_r) and there is a value for $Q_r \in \{\textsc{Cred}^i_r : i \in W(Q_{r-1})\}$ such that the omniscient adversary can bias the next $k - 1$ rounds. In other words, the omniscient adversary can bias k rounds if and only if there is a path $Q_0, Q_1, \ldots, Q_k$ in the tree.

A real adversary cannot search over the entire tree for the longest path, since the real adversary cannot compute the VDFs of accounts they do not own in future rounds (they can still compute VDFs for their own accounts in hypothetical future rounds, which provides statistical information about what the tree might look like in future rounds, but they do not know the precise tree as the omniscient adversary does). However, the performance of the omniscient adversary is clearly an upper bound on the performance of the real adversary, so T provides an upper bound for the number of rounds the adversary can bias. Hence, showing that the expected height of T is finite implies that any strategy played by a strategic miner is positive recurrent.

First, we will characterize the distribution of $|W(Q_{r-1})|$. Formally, we will show that

$$Pr\left[|W(Q_{r-1})| - 1 = j\right] = \alpha^j(1-\alpha).$$

The notation $\min^{(i)}\{S\}$ refers to the i^{th}-smallest element of S for $i \geq 1$ and $\min^{(0)}\{S\} := 0$. As a technical tool, we recall a useful property for exponential distributions: for all $k \geq 1$, $\min^{(k)}_{i\in[n]} \text{S}(\textsc{Cred}^i_r, \alpha_i)$ is identically distributed to $\textsc{Exp}\,(\alpha) + \min^{(k-1)}_{i\in[n]} \text{S}(\textsc{Cred}^j_r, \alpha_k)$ where $\textsc{Exp}\,(\alpha)$ refers to an independent sample from the exponentially distributed with rate α. We defer the proof to Appendix C.

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 $Z_i = Z_{i-1} + \textsc{Exp}\,(\alpha)$ where $Z_0 := 0$.

Remark. Lemma 4.3 provides an useful tool to reduce the computational cost of sampling only the best credentials for our adversary. If one wants to sample the k lowest scores among accounts in A, a naive approach would require us to take $|A|$ samples from $\textsc{Exp}\left(\frac{\alpha}{|A|}\right)$, sort in increasing order and output the first k credentials. However, from Lemma 4.3, it suffices to sample and output the sequence $X_1 = \textsc{Exp}\,(\alpha), X_2 = X_1 + \textsc{Exp}\,(\alpha), \ldots X_k = X_{k-1} + \textsc{Exp}\,(\alpha)$.

We now prove the probability that the adversary has j options for the seed of round r given Q_{r-1} is $\alpha^j(1-\alpha)$.

Lemma 4.4. Let $X_1, X_2, \ldots$ be i.i.d. exponentially distributed random variables such that $\min_{n\in\mathbb{N}} X_n$ is exponentially distributed with rate α. Let W be exponentially distributed with rate $1-\alpha$. Let $S = \{i \in \mathbb{N} : X_i < W\}$. Then $Pr\left[|S| = j\right] = \alpha^j(1-\alpha)$.

Proof. Let $Z_i = \min^{(i)}_{n\in\mathbb{N}} X_n$. Let E_i denote the event $Z_i < W$ and let E^c_i be its complement. Then $|S| = j$ if and only if $Z_1 < Z_2 < \ldots < Z_j < W < Z_{j+1}$. Then

$$\begin{aligned} Pr\left[|S| = j\right] &= Pr\left[E^c_{j+1} \cap (\cap^j_{i=1} E_i)\right] \\ &= Pr\left[E^c_{j+1}|E_j\right] \prod_{i=1}^{j} Pr\left[E_i|E_{i-1}\right] \\ &= Pr\left[W < Z_{j+1}|W > Z_j\right] \prod_{i=1}^{j} Pr\left[W > Z_i|W > Z_{i-1}\right]. \end{aligned}$$

From Lemma 4.3, Z_{i+1} is identically distributed to $Z_i + \textsc{Exp}(\alpha)$ for all $i \in \mathbb{N}$. Then

$$\begin{aligned} Pr\left[|S| = j\right] = Pr&\left[W < Z_j + \textsc{Exp}(\alpha)\,|W > Z_j\right] \\ &\times \prod_{i=1}^{j} Pr\left[W > Z_{i-1} + \textsc{Exp}(\alpha)\,|W > Z_{i-1}\right] \\ = &\,Pr\left[W < \textsc{Exp}(\alpha)\right] \prod_{i=1}^{j} Pr\left[W > \textsc{Exp}(\alpha)\right] && \text{From Lemma A.3,} \\ = &\,(1-\alpha) \prod_{i=1}^{j} \alpha = \alpha^j(1-\alpha) && \text{From Lemma A.2} \end{aligned}$$

□

Corollary 4.1. Let Q_{r-1} de drawn from $U[0,1]$ and $W(Q_{r-1}) = \{i \in A : \mathrm{S}(\textsc{Cred}^i_r, \alpha_i) < \min_{j\notin A} \mathrm{S}(\textsc{Cred}^j_r, \alpha_j)\}$. Then $Pr\left[|W(Q_{r-1})| = j\right] = \alpha^j(1-\alpha)$.

Proof. Recall $\min_{j\notin A} \mathrm{S}(\textsc{Cred}^j_r, \alpha_j)$ is identically distributed to $\textsc{Exp}(1-\alpha)$ and $\min_{i\in A} \mathrm{S}(\textsc{Cred}^i_r, \alpha_i)$ is identically distributed to $\textsc{Exp}(\alpha)$ (Lemma A.1). From Lemma 4.4, $Pr\left[|W(Q_{r-1})| = j\right] = \alpha^j(1-\alpha)$ as desired. □

4.2 Extinction in the Branching Process

Next, we derive necessary conditions for the expected height of $\mathrm{Tree}(Q_0)$ to be finite. This result will will imply the existence of optimal positive recurrent strategies.

Lemma 4.5. Let Q_0 be an unbiased seed and let τ be the first forced stopping time. Then $Pr\left[\tau \geq k\right] \leq \left(\frac{\alpha(2-\alpha)}{1-\alpha}\right)^k$.

Proof. Clearly τ is upper bounded by the height of $\mathrm{Tree}(Q_0)$, then the event $\tau \geq k$ implies the height of $\mathrm{Tree}(Q_0)$ is at least $k+1$. For all $k \geq 0$ and $Q \in [0,1]$, let $E_{k,Q}$ denote the event that $\mathrm{Tree}(Q)$ has height at least $k+1$. Note $Pr\left[E_{0,Q}\right] = 1$. Then, for $k \geq 1$, the event $E_{k,Q}$ holds if and only if $|W(Q)| \geq 2$ and for some child $Q' \in W(Q)$, the sub-tree $\mathrm{Tree}(Q')$ has height at least $k-1$. Let $A_k = Pr\left[E_{k,Q_0}\right]$. Then,

$$\begin{aligned}
Pr\left[\tau \geq k\right] &\leq Pr\left[E_{k,Q_0}\right] = A_k \\
&= \sum_{i=1}^{\infty} Pr\left[|W(Q_0)| = i+1\right] Pr\left[\cup_{j \in W(Q_0)\cup\{j_0\}} E_{k-1,\textsc{Cred}_0^j}\right] \\
&= (1-\alpha)\sum_{i=1}^{\infty} \alpha^i Pr\left[\cup_{j \in W(Q_0)\cup\{j_0\}} E_{k-1,\textsc{Cred}_0^j} \mid |W(Q_0)| = i+1\right] && \text{From Lemma 4.4,} \\
&\leq (1-\alpha)\sum_{i=1}^{\infty}(i+1)\alpha^i Pr\left[E_{k-1,U[0,1]}\right] && \text{From the union bound,} \\
&= \frac{\alpha(2-\alpha)}{1-\alpha} A_{k-1}
\end{aligned}$$

The last line observes the geometric series converges to $\frac{\alpha(2-\alpha)}{1-\alpha}$. To conclude, we proof by induction that $A_k \leq \left(\frac{\alpha(2-\alpha)}{1-\alpha}\right)^k$. The base case is clear: $A_0 \leq 1$. For $k \geq 1$, the inductive assumption gives

$$A_k = \frac{\alpha(2-\alpha)}{1-\alpha} A_{k-1} \leq \left(\frac{\alpha(2-\alpha)}{1-\alpha}\right)^k$$

as desired. This proves the statement. □

Theorem 4.1. Consider any strategy π and let Q_0 be an unbiased seed. Let $\tau \geq 1$ the first forced stopping time. If $\alpha < \frac{3-\sqrt{5}}{2} \approx 0.38$, $\mathbb{E}\left[\tau\right] < \frac{1-\alpha}{1-3\alpha+\alpha^2} < \infty$. Hence π is positive recurrent.

Proof. Recall that for positive discrete random τ, $\mathbb{E}\left[\tau\right] = \sum_{i=0}^{\infty} Pr\left[\tau > i\right]$. From Lemma 4.5,

$$\mathbb{E}\left[\tau\right] < \sum_{i=0}^{\infty}\left(\frac{\alpha(2-\alpha)}{1-\alpha}\right)^i = \frac{1-\alpha}{1-3\alpha+\alpha^2} < \infty$$

The last inequality observes the geometric series converges for $\alpha < \frac{3-\sqrt{5}}{2}$. □

As an application of Theorem 4.1, we derive a theoretical upper bound on the revenue for any strategy. Figure 1 compares the curve for the theoretical upper bound with the revenue of the honest strategy.

Theorem 4.2. For $\alpha < \frac{3-\sqrt{5}}{2} \approx 0.38$, and all β, $\textsc{val}(\alpha, \beta) \leq \frac{\alpha(2-\alpha)}{1-\alpha}$.

Proof. From Theorem 4.1, for $\alpha < \frac{3-\sqrt{5}}{2}$, there is an optimal positive recurrent strategy π. Let $\tau \geq 1$ be a forced stopping time. From Lemma 4.2, τ is a stopping time and

$$\begin{aligned}
\textsc{Rev}(\pi) &= \frac{\mathbb{E}\left[\sum_{r=1}^{\tau} X_r(\pi)\right]}{\mathbb{E}\left[\tau\right]} && \text{Lemma 4.1,} \\
&\leq \frac{\mathbb{E}\left[\tau - 1\right]}{\mathbb{E}\left[\tau\right]} \\
&= 1 - \frac{1}{\mathbb{E}\left[\tau\right]} && \text{From linearity of expectation,} \\
&\leq \frac{\alpha(2-\alpha)}{1-\alpha} && \text{From Theorem 4.1.}
\end{aligned}$$

The first inequality observes that if the adversary cannot choose Q_τ, then the adversary does not create block B_τ. □

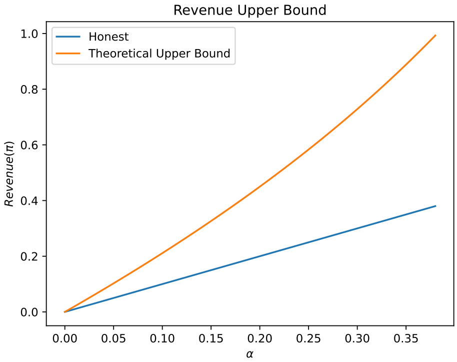

Fig. 1. Revenue upper bound.

The analysis in this section shows the following: for all $\alpha < \frac{3-\sqrt{5}}{2}$, even an omniscient adversary with an α fraction of the total stake can win at most an $\alpha \cdot \frac{2-\alpha}{1-\alpha} < 1$ fraction of the rounds in expectation.

5 THE 1-LOOKAHEAD STRATEGY

This section defines the 1-Lookahead strategy for a strong adversary ($\beta = 1$), which outperforms the honest strategy for any value of α. Recall that the adversary divides their stake equally among an arbitrarily large number of accounts A. Note that this is a concrete strategy that can be used in CSSPA, and therefore its reward gives a lower bound on $\textsc{val}(\alpha, 1)$.

Definition 5.1 (1-Lookahead strategy)**.** The strategy proceed as follows:

1. Let r be the current round. Let $W(Q_{r-1}) = \{i \in A : \mathrm{S}(\textsc{Cred}^i_r, \alpha_i) < \min_{i \notin A} \mathrm{S}(\textsc{Cred}^i_r, \alpha_i)\}$ be the collection of potential winners for the adversary.
2. If $|W(Q_{r-1})| = 0$, broadcast no credentials. Terminate round r and return to Step 1.
3. If $|W(Q_{r-1})| \geq 1$, for each potential winner $i \in W(Q_{r-1})$, for each account $j \in A$, sample credential $\textsc{Cred}^{i,j}_{r+1} = f_{\mathrm{SK}_j}(\textsc{Cred}^i_r)$. Let $j(i) = \min_{j \in A} \textsc{Cred}^{i,j}_{r+1}$.
4. Let $i^* = \arg\min_{i \in W(Q_{r-1})} \min_{j \in A} \mathrm{S}(\textsc{Cred}^{i,j}_{r+1}, \alpha_j)$.
5. Broadcast $\textsc{Cred}^{i^*}_r$ at round r and $\textsc{Cred}^{j(i^*)}_{r+1}$ at round $r + 1$.
6. Return to Step 1.

Theorem 5.1. $\textsc{Rev}_{\alpha,1}(\text{1-}\textsc{Lookahead}) = \frac{1-\alpha}{1+\alpha} \sum_{i=1}^{\infty} \alpha^i \left(1 + \frac{i\alpha}{1+(i-1)\alpha}\right)$.

Proof. For our strategy, consider the stopping time τ where $\tau = 1$ when $|W(Q_0)| = 0$ and $\tau = 2$ when $|W(Q_0)| \geq 1$. Observe that this is indeed a stopping time as: (a) when $|W(Q_0)| = 0$, $\tau = 1$ is a forced stopping time, and (b) while the adversary queries $f_{\mathrm{SK}_j}(Q_1)$ for multiple possible values of Q_1, *they do not query $f_{\mathrm{SK}_j}(Q_2)$ for any of the possible values for Q_2*. Therefore, from the perspective of the adversary, the distribution of any VRF Q_2 is just $U([0, 1])$, and this distribution is identical if we replace Q_2 with a fresh seed. Therefore, $\tau = 2$ is indeed a stopping time when $W(Q_0) \geq 1$.

Now let's compute $E[\tau]$. The probability $|W(Q_0)| \geq 1$ is equal to the probability $\min_{i \in A}\{\mathrm{S}(\textsc{Cred}^i_r, \alpha_i)\} < \min_{j \notin A}\{\mathrm{S}(\textsc{Cred}^j_r, \alpha_j)\}$. The first term is exponentially distributed with rate α (Lemma A.1) while the second is exponentially distributed with rate $1 - \alpha$. Hence the probability the adversary has at least one winner is α (Lemma A.2). Then $\mathbb{E}[\tau] = (1 - \alpha) + 2\alpha = 1 + \alpha$. Because the 1-Lookahead

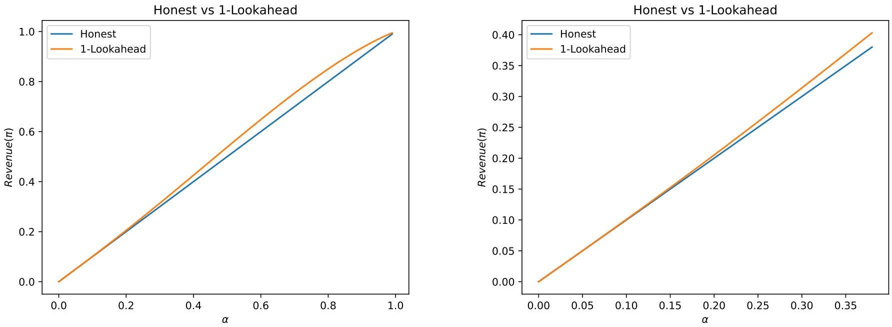

Fig. 2. Revenue for the 1-Lookahead strategy.

strategy is positive recurrent Lemma 4.1 implies

$$\textsc{Rev}(1\text{-}\textsc{Lookahead}) = \frac{\mathbb{E}\left[\sum_{r=1}^{\tau} X_r(\pi)\right]}{\mathbb{E}\left[\tau\right] = 1+\alpha}.$$

Let's compute the numerator. If $|W(Q_0)| \geq 1$, the adversary wins round r since they always reveal a winning credential. Moreover, for round r+1, they reveal a credential with score $\min_{i \in W(Q_0), k \in A} \textsc{Cred}_{r+1}^{i,k}$ which is exponentially distributed with rate $\alpha \cdot |W(Q_0)|$ (Lemma A.1). From Lemma A.2, the probability the adversary wins round $r+1$ given $|W(Q_0)| = j$ is $\frac{\alpha j}{1+\alpha(j-1)}$. Hence,

$$\mathbb{E}\left[\sum_{r=1}^{\tau} X_r(\pi) || W(Q_0)| = j\right] = \mathbb{1}_{j\geq 1} + \frac{\alpha j}{1+\alpha(j-1)}.$$

From Corollary 4.1 the probability $|W(Q_0)| = j$ is $\alpha^j(1-\alpha)$. Then

$$\mathbb{E}\left[\sum_{r=1}^{\tau} X_r(\pi)\right] = \sum_{j=1}^{\infty} Pr\left[W(Q_0) = j\right] \mathbb{E}\left[\sum_{r=1}^{\tau} X_r(\pi) | W(Q_0) = j\right]$$

$$= \sum_{j=1}^{\infty} \alpha^j (1-\alpha)\left(1 + \frac{\alpha j}{1+\alpha(j-1)}\right)$$

as desired. This concludes the proof. □

Figure 2 shows the revenue of the 1-Lookahead strategy (Theorem 5.1) against the revenue of the honest strategy. Observe that it is always more profitable than the honest strategy, as expected.

6 MARKOV DECISION PROCESS FOR OPTIMAL STRATEGIES

This section shows how the optimal strategy can be computed by querying a Markov Decision Process (MDP) solver whenever $\alpha < \frac{3-\sqrt{5}}{2} \approx 0.38$. Let us recall the available information for the adversary before choosing an action. Once the game starts, the miner can compute all possible values for $Q_1, Q_2, \ldots, Q_k$ assuming $\ell_1 \in A \cup \{B\}$, $\ell_2, \ell_3 \ldots, \ell_{k-1} \in A$. As a special case, the 1-Lookahead strategy only computes the possible values of Q_2 before choosing which credential to broadcast in the first round.

The information available for the adversary can be encoded in a tree where each node is a seed. For a seed Q, constant $k \geq 0$, define the Q rooted tree $\text{Tree}_k(Q)$ recursively as follows:

- If $k = 0$, let $\text{Tree}_k(Q)$ contain only the root Q.

- If $k \geq 1$, let Q be the root of $\mathrm{Tree}_k(Q)$. Moreover,
  - For each $i \in A$, add edge $(Q, \mathrm{Tree}_{k-1}(f_{\mathrm{SK}_i}(Q)))$ to $\mathrm{Tree}_k(Q)$ where $\mathrm{Tree}_{k-1}(f_{\mathrm{SK}_i}(Q))$ becomes a $f_{\mathrm{SK}_i}(Q)$ rooted sub-tree in $\mathrm{Tree}_k(Q)$ connected by the edge $(Q, f_{\mathrm{SK}_i}(Q))$.
  - For each $i \notin A$, once user i already broadcast $f_{\mathrm{SK}_i}(Q)$, add edge $(Q, \mathrm{Tree}_{k-1}(f_{\mathrm{SK}_i}(Q)))$ to $\mathrm{Tree}_k(Q)$.

Let $\mathrm{Tree}(Q)$ be the graph obtained when we take $k \longrightarrow \infty$ in $\mathrm{Tree}_k(Q)$. Recall the basic facts for an optimal strategy π from Section 3.1: (1) π divides its stake α among an infinite amount of wallets; (2) π broadcast at most one credential each round. Then, without loss of generality, a strategy π maps $\mathrm{Tree}(Q)$ to at most one credential from $\{f_{\mathrm{SK}_i}(Q)\}_{i \in A}$, corresponding to the credential π broadcast in a round with seed Q. If the strategy outputs no credential, we write $\pi(\mathrm{Tree}(Q)) = \perp$.

Definition 6.1 (Value Function). Let π be a positive recurrent strategy, and let ρ be a positive constant. For a tree $\mathrm{Tree}(Q)$, define

$$\mathrm{V}_{\pi}^{\rho}(\mathrm{Tree}(Q)) := \mathbb{E}\left[\sum_{r=1}^{\tau}(X_r(\pi) - \rho)|Q_0 = Q\right]$$

where τ is stopping time. Taking the expected value with respect to $\mathrm{Tree}(Q)$ gives

$$\mathrm{V}_{\pi}^{\rho} := \mathbb{E}\left[\mathrm{V}_{\pi}^{\rho}(\mathrm{Tree}(Q))\right].$$

We can derive a recursive formula for the value function as follows:

Proposition 6.1. For any positive recurrent strategy π, positive constant ρ, tree $\mathrm{Tree}(Q)$,

$$\mathrm{V}_{\pi}^{\rho}(\mathrm{Tree}(Q)) = \mathbb{E}\left[(X_1(\pi) - \rho) + \mathrm{V}_{\pi}^{\rho}(Q_1)|Q_0 = Q\right]$$

Theorem 6.1. Let π and π' be positive recurrent strategies. Then

- $\mathrm{V}_{\pi}^{\rho} = 0$ if and only if $\rho = \textsc{Rev}(\pi)$.
- $\textsc{Rev}(\pi') < \textsc{Rev}(\pi)$ if and only if $\mathrm{V}_{\pi}^{\textsc{Rev}(\pi')} > \mathrm{V}_{\pi}^{\textsc{Rev}(\pi)}$.

Proof. From Lemma 4.1 and the assumption π is positive recurrent, $\textsc{Rev}(\pi) = \frac{\mathbb{E}\left[\sum_{r=1}^{\tau} X_r(\pi)\right]}{\mathbb{E}[\tau]}$. Clearly $1 - \textsc{Rev}(\pi) = \frac{\mathbb{E}\left[\sum_{r=1}^{\tau}(1 - X_r(\pi))\right]}{\mathbb{E}[\tau]}$. Then

$$\begin{aligned} 0 &= \frac{\mathbb{E}\left[\sum_{r=1}^{\tau} X_r(\pi)\right]\mathbb{E}\left[\sum_{r=1}^{\tau}(1 - X_r(\pi))\right]}{\mathbb{E}\left[\tau\right]} - \frac{\mathbb{E}\left[\sum_{r=1}^{\tau} X_r(\pi)\right]\mathbb{E}\left[\sum_{r=1}^{\tau}(1 - X_r(\pi))\right]}{\mathbb{E}\left[\tau\right]} \\ &= (1 - \textsc{Rev}(\pi))\mathbb{E}\left[\sum_{r=1}^{\tau} X_r(\pi)\right] - \textsc{Rev}(\pi)\mathbb{E}\left[\sum_{r=1}^{\tau}(1 - X_r(\pi))\right] \\ &= \mathbb{E}\left[\sum_{r=1}^{\tau}(X_r(\pi) - \textsc{Rev}(\pi))\right] \qquad \text{From linearity of expectation,} \\ &= \mathrm{V}_{\pi}^{\textsc{Rev}(\pi)} \end{aligned}$$

The chain of inequalities proofs $\mathrm{V}_{\pi}^{\rho} = 0$ when $\rho = \textsc{Rev}(\pi)$ as desired. For the other direction, observe V_{π}^{ρ} is a strictly decreasing function of ρ. Hence there is a unique value for ρ where V_{π}^{ρ} vanishes to zero. This proves the first bullet. The second bullet follows from the fact V_{π}^{ρ} is strictly monotone decreasing in ρ. □

Corollary 6.1. Let $\pi^* \in \arg\max_{\tilde{\pi}} \textsc{Rev}(\tilde{\pi})$. Then $\pi \in \arg\max_{\tilde{\pi}} \textsc{Rev}(\tilde{\pi})$ is optimal if and only if $\pi \in \arg\max_{\tilde{\pi}} \mathrm{V}_{\tilde{\pi}}^{\textsc{Rev}(\pi^*)}$.

Proof. First we prove that if $\pi \in \arg\max_\pi \textsc{Rev}(\pi)$, then $\pi \in \arg\max_\pi \mathrm{V}_\pi^{\textsc{Rev}(\pi^*)}$. From Theorem 6.1, for all strategy $\tilde{\pi}$,

$$\mathrm{V}_\pi^{\textsc{Rev}(\pi)} = 0 = \mathrm{V}_{\tilde{\pi}}^{\textsc{Rev}(\tilde{\pi})} \geq \mathrm{V}_{\tilde{\pi}}^{\textsc{Rev}(\pi)}$$

where the first and second equality are the first bullet in the theorem; the inequality is the second bullet and the fact $\textsc{Rev}(\pi) = \textsc{Rev}(\pi^*) \geq \textsc{Rev}(\tilde{\pi})$. Since the inequality holds for any $\tilde{\pi}$, we have $\pi \in \arg\max_{\tilde{\pi}} \mathrm{V}_{\tilde{\pi}}^{\textsc{Rev}(\pi^*)}$. This proves the first part.

For the second part, we proof that if $\pi \in \arg\max_{\tilde{\pi}} \mathrm{V}_{\tilde{\pi}}^{\textsc{Rev}(\pi^*)}$, then $\pi \in \arg\max_{\tilde{\pi}} \textsc{Rev}(\tilde{\pi})$. We already proved that $\mathrm{V}_{\pi^*}^{\textsc{Rev}(\pi*)} \geq \mathrm{V}_\pi^{\textsc{Rev}(\pi^*)}$. The assumption implies $\mathrm{V}_\pi^{\textsc{Rev}(\pi^*)} \geq \mathrm{V}_{\pi^*}^{\textsc{Rev}(\pi^*)}$. Hence $\mathrm{V}_\pi^{\textsc{Rev}(\pi^*)} = \mathrm{V}_{\pi^*}^{\textsc{Rev}(\pi^*)} = 0$ which proves π is optimal (Theorem 6.1). □

The following is equivalent to Bellman's principle of optimality.

Lemma 6.1. Let $\pi \in \arg\max_{\tilde{\pi}} \textsc{Rev}(\tilde{\pi})$ and assume $\alpha < \frac{3-\sqrt{5}}{2} \approx 0.38$. Then for all $\text{Tree}(Q)$:

$$\pi \in \arg\max_{\tilde{\pi}} \mathrm{V}_{\tilde{\pi}}^{\textsc{Rev}(\pi)}(\text{Tree}(Q)).$$

Proof. Let π_r refer to the action of strategy π at round r. From Corollary 6.1, the fact π is optimal implies

$$\begin{aligned}
\mathrm{V}_\pi^{\textsc{Rev}(\pi)} &= \max_{\tilde{\pi}} \mathrm{V}_{\tilde{\pi}}^{\textsc{Rev}(\pi)} \\
&= \mathbb{E}\left[\max_{\tilde{\pi}} \mathrm{V}_{\tilde{\pi}}^{\textsc{Rev}(\pi)}(\text{Tree}(Q_0))\right] \\
&= \mathbb{E}\left[\max_{\tilde{\pi}} \mathbb{E}\left[(X_1(\tilde{\pi}) - \textsc{Rev}(\pi)) + \mathrm{V}_{\tilde{\pi}}^{\textsc{Rev}(\pi)}(\text{Tree}(Q_1))|Q_0\right]\right] \\
&= \mathbb{E}\left[\max_{\tilde{\pi}_1} \mathbb{E}\left[(X_1(\tilde{\pi}) - \textsc{Rev}(\pi)) + \max_{\tilde{\pi}:\tilde{\pi}_1} \mathrm{V}_{\tilde{\pi}}^{\textsc{Rev}(\pi)}(\text{Tree}(Q_1))|Q_0\right]\right]
\end{aligned}$$

The second equality is Proposition 6.1. The third equality observes that the optimal strategy for the sub-game starting with seed Q_1 is independent of the action taken at round 1. Hence $\mathrm{V}_\pi^{\textsc{Rev}(\pi)}(\text{Tree}(Q)) = \max_{\tilde{\pi}} \mathrm{V}_{\tilde{\pi}}^{\textsc{Rev}(\pi)}(\text{Tree}(Q))$ for all $\text{Tree}(Q)$. □

To compute the optimal strategy $\pi^* \in \arg\max_{\tilde{\pi}} \textsc{Rev}(\tilde{\pi})$, we can use a similar binary search algorithm from Sapirshtein et al. [19]. We pick some $\rho \in [0,1]$ as our guess for $\textsc{Rev}(\pi^*)$ and maximize the Markov Decision Process $\max_{\tilde{\pi}} \mathrm{V}_{\tilde{\pi}}^\rho$. Let π be the strategy the solver outputs. Then one of the following cases tell us if ρ is a lower bound or an upper bound on the optimal revenue:

- The case where $\mathrm{V}_\pi^\rho \geq 0$ witnesses that $\textsc{Rev}(\pi^*) \geq \rho$. To see, recall $\mathrm{V}_\pi^{\textsc{Rev}(\pi)} = 0$ (Theorem 6.1). Because V_π^ρ is a strictly decreasing function of ρ, we conclude $\rho \leq \textsc{Rev}(\pi) \leq \textsc{Rev}(\pi^*)$ as desired.
- Te case where $\mathrm{V}_\pi^\rho < 0$ witnesses that $\textsc{Rev}(\pi^*) < \rho$. To see, it suffices to prove the contrapositive: if $\textsc{Rev}(\pi^*) \geq \rho$, then $\mathrm{V}_\pi^\rho \geq 0$. Assume $\textsc{Rev}(\pi^*) \geq \rho$. Because V_π^ρ is a strictly decreasing function of ρ, we conclude $\mathrm{V}_{\pi^*}^\rho \geq \mathrm{V}_{\pi^*}^{\textsc{Rev}(\pi*)} = 0$ (Theorem 6.1). By assumption, $\mathrm{V}_\pi^\rho = \max_{\tilde{\pi}} \mathrm{V}_{\tilde{\pi}}^\rho \geq \mathrm{V}_{\pi^*}^\rho \geq 0$ as desired.

7 CONCLUSION

We propose a stylized model to study optimal strategic mining in Cryptographic Self-Selection leader election protocols. We consider rational miners that wish to maximize the fraction of blocks they create. The same adversary has been studied in the context of Proof-of-Work blockchains since the discovery of the selfish mining attacks against Bitcoin [7].

Prior work largely classifies existing protocols into two camps: those where sufficiently small miners *cannot* profitably deviate (longest-chain proof-of-work protocols with block reward and longest-chain proof-of-stake protocols with a randomness beacon), and those where arbitrarily small miners can still profitably deviate (longest-chain proof-of-work protocols with transaction fees, longest-chain proof-of-stake protocols without a randomness beacon). Our work classifies blockchains based on cryptographic self-selection with the latter group: we give a closed-form representation for a strategy that outperforms the honest strategy for any amount of stake.

The key open question left by our work is to nail down the optimal fraction of rounds that a β-strong strategic miner with an α fraction of the stake can earn. While our work states that this quantity can in principle be determined by performing binary search over infinitely-sized MDPs, actually significant innovation seems to be required to actually perform this search, or even to approximating it computationally-efficiently.

REFERENCES

[1] 2021. URL https://ccaf.io/cbeci/index. Accessed: 2022-1-10.

[2] N. Arnosti and S. M. Weinberg. Bitcoin: A natural oligopoly. *Management Science*, 2022.

[3] J. Brown-Cohen, A. Narayanan, A. Psomas, and S. M. Weinberg. Formal barriers to longest-chain proof-of-stake protocols. In *Proceedings of the 2019 ACM Conference on Economics and Computation*, pages 459–473, 2019.

[4] C. Cachin, K. Kursawe, F. Petzold, and V. Shoup. Secure and efficient asynchronous broadcast protocols. In *Annual International Cryptology Conference*, pages 524–541. Springer, 2001.

[5] M. Carlsten, H. Kalodner, S. M. Weinberg, and A. Narayanan. On the instability of bitcoin without the block reward. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 154–167, 2016.

[6] J. Chen and S. Micali. Algorand: A secure and efficient distributed ledger. *Theoretical Computer Science*, 777:155–183, 2019.

[7] I. Eyal and E. G. Sirer. Majority is not enough: Bitcoin mining is vulnerable. In *International conference on financial cryptography and data security*, pages 436–454. Springer, 2014.

[8] M. V. Ferreira and S. M. Weinberg. Proof-of-stake mining games with perfect randomness. In *Proceedings of the 22nd ACM Conference on Economics and Computation*, pages 433–453, 2021.

[9] M. V. Ferreira, D. J. Moroz, D. C. Parkes, and M. Stern. Dynamic posted-price mechanisms for the blockchain transaction-fee market. In *Proceedings of the 3rd ACM Conference on Advances in Financial Technologies*, pages 86–99, 2021.

[10] Y. Gilad, R. Hemo, S. Micali, G. Vlachos, and N. Zeldovich. Algorand: Scaling byzantine agreements for cryptocurrencies. In *Proceedings of the 26th symposium on operating systems principles*, pages 51–68, 2017.

[11] K. Karantias, A. Kiayias, and D. Zindros. Proof-of-burn. In *International conference on financial cryptography and data security*, pages 523–540. Springer, 2020.

[12] J. Kelsey, L. T. Brandão, R. Peralta, and H. Booth. A reference for randomness beacons: Format and protocol version 2. Technical report, National Institute of Standards and Technology, 2019.

[13] A. Kiayias, A. Russell, B. David, and R. Oliynykov. Ouroboros: A provably secure proof-of-stake blockchain protocol. In *Annual international cryptology conference*, pages 357–388. Springer, 2017.

[14] S. Micali, M. Rabin, and S. Vadhan. Verifiable random functions. In *40th annual symposium on foundations of computer science (cat. No. 99CB37039)*, pages 120–130. IEEE, 1999.

[15] S. Nakamoto. Bitcoin: A peer-to-peer electronic cash system. *Decentralized Business Review*, page 21260, 2008.

[16] M. O. Rabin. Transaction protection by beacons. *Journal of Computer and System Sciences*, 27(2):256–267, 1983.

[17] L. Ren and S. Devadas. Proof of space from stacked expanders. In *Theory of Cryptography Conference*, pages 262–285. Springer, 2016.

[18] T. Roughgarden. Transaction fee mechanism design. *ACM SIGecom Exchanges*, 19(1):52–55, 2021.

[19] A. Sapirshtein, Y. Sompolinsky, and A. Zohar. Optimal selfish mining strategies in bitcoin. In *International Conference on Financial Cryptography and Data Security*, pages 515–532. Springer, 2016.

[20] H. W. Watson and F. Galton. On the probability of the extinction of families. *The Journal of the Anthropological Institute of Great Britain and Ireland*, 4:138–144, 1875.

A PROBABILITY THEORY BACKGROUND

Lemma A.1. Let $X_1, X_2, \ldots, X_n$ be independent random variables where X_i is a copy from $\textsc{Exp}\left(\alpha_i\right)$ where α_i is a positive constant. Then $\min_{i\in[n]}\{X_i\}$ is identically distributed to $\textsc{Exp}\left(\sum_{i=1}^n \alpha_i\right)$.

Proof. The proof follows from computing the probability $Pr\left[\min X_i \le x\right]$:

$$\begin{aligned}Pr\left[\min X_i \le x\right] &= 1 - Pr\left[\wedge_{i=1}^n\{X_i > x\}\right] = 1 - \prod_{i=1}^n Pr\left[X_i > x\right]\\ &= 1 - \prod_{i=1}^n e^{-\alpha_i x}\\ &= 1 - e^{-x\sum_{i=1}^n \alpha_i}.\end{aligned}$$

The last line witness $\min X_i$ is exponentially distributed with weight $\sum_{i=1}^n \alpha_i$. □

Lemma A.2. Let X and Y be drawn independently from an exponential distributions with rate α_X and α_Y respectively. Then

$$Pr\left[X < Y\right] = \frac{\alpha_X}{\alpha_X + \alpha_Y}.$$

Proof. We have as follows:

$$\begin{aligned}Pr\left[X < Y\right] &= \int_0^\infty f_Y(y) Pr\left[X < y\right] dy\\ &= \int_0^\infty \alpha_Y e^{-\alpha_Y y}(1 - e^{-\alpha_X y})dy\\ &= \int_0^\infty \alpha_Y e^{-\alpha_Y y}dy - \int_0^\infty \alpha_Y e^{-(\alpha_Y+\alpha_X)y}dy\\ &= 1 - \frac{\alpha_Y}{\alpha_X + \alpha_Y}\\ &= \frac{\alpha_X}{\alpha_X + \alpha_Y}\end{aligned}$$

□

Corollary A.1. Let $X_1, \ldots, X_n$ be drawn independently from exponential distributions with rates $\alpha_1, \ldots, \alpha_n$, respectively. Then $\Pr[X_i = \min_{j\in[n]}\{X_j\}] = \frac{\alpha_i}{\sum_{j=1}^n \alpha_j}$.

Proof. We prove this by induction, using Lemmas A.1 and A.2. As a base case, the claim is clearly true when $n = 1$, for all α_1. Now as an inductive hypothesis, assume that the claim is true for some n, and all $\alpha_1, \ldots, \alpha_n$. We now consider the case of $n + 1$ and any $\alpha_1, \ldots, \alpha_{n+1}$.

By Lemma A.1, $\min_{j\in[n+1]\setminus i}\{X_j\}$ is distributed according to an exponential of rate $\sum_{j\neq i}\alpha_j$. By Lemma A.2, the probability that $X_i = \min_{j\in[n+1]}\{X_j\} = \frac{\alpha_i}{\sum_{j=1}^{n+1}\alpha_j}$, as desired. This argument holds for any i, and completes the inductive step. □

Lemma A.3 (Memorylessness Property). Let X be drawn from an exponential distribution (with any rate α), then for any $n, m \ge 0$,

$$Pr\left[X > n + m | X \ge m\right] = Pr\left[X > n\right].$$

Proof. We have as follows:

$$\begin{aligned}
Pr\left[X > n+m | X \geq m\right] &= \frac{Pr\left[X > n+m, X \geq m\right]}{Pr\left[X \geq m\right]} \\
&= \frac{Pr\left[X > n+m\right]}{Pr\left[X \geq m\right]} \\
&= \frac{1-(1-e^{-\alpha(n+m)})}{1-(1-e^{-\alpha m})} \\
&= \frac{e^{-\alpha(n+m)}}{e^{-\alpha m}} \\
&= e^{-\alpha n} \\
&= Pr\left[X > n\right]
\end{aligned}$$

□

Lemma A.4 (Strong Law of Large Numbers). Let X be a random variable. Let $X_1, X_2, \ldots, X_n$ be independent copies of X. Then $Pr\left[\lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} X_i = \mathbb{E}\left[X\right]\right] = 1$.

B OMITTED PROOFS FROM SECTION 2

Lemma B.1. Let $g(\cdot)$ be a monotone increasing function with domain $(0,1)$. Define $S^g(x,\alpha) := g(x^{1/\alpha})$. Then, $S^g(\cdot,\cdot)$ is a canonical balanced scoring function.

Proof. First, we show that $S^g(\cdot,\cdot)$ is a balanced scoring function. To see this, we observe that:

$$\begin{aligned}
\Pr_{X_1,\ldots,X_n\leftarrow U([0,1]^n)}\left[\arg\min_{i\in[n]}\{S^g(X_i,\alpha_i)\} = j\right] &= \Pr_{X_1,\ldots,X_n\leftarrow U([0,1]^n)}\left[\arg\min_{i\in[n]}\{g(X_i^{1/\alpha_i})\} = j\right] \\
&= \Pr_{X_1,\ldots,X_n\leftarrow U([0,1]^n)}\left[\arg\max_{i\in[n]}\{X_i^{1/\alpha_i}\} = j\right] \\
&= \Pr_{X_1,\ldots,X_n\leftarrow U([0,1]^n)}\left[\arg\max_{i\in[n]}\{\ln(X_i^{1/\alpha_i})\} = j\right] \\
&= \Pr_{X_1,\ldots,X_n\leftarrow U([0,1]^n)}\left[\arg\min_{i\in[n]}\{-\ln(X_i^{1/\alpha_i})\} = j\right] \\
&= \frac{\alpha_j}{\sum_{i=1}^{n}\alpha_i}
\end{aligned}$$

Above, the first line follows by definition of $S^g(\cdot,\cdot)$. The second line follows as $g(\cdot)$ is monotone decreasing. The third follows as $\ln(\cdot)$ is a monotone increasing function. The fourth line follows trivially. The final line follows as $-\ln(X_i^{1/\alpha_i})$ is distributed according to an exponential with rate α_i by Lemma 2.1, and Corollary A.1 (which states that the minimum of $Y_1,\ldots,Y_n$, when each Y_i is drawn independently from Exp (α_i) is equal to X_i with probability α_i, for all i)

To see that $S^g(\cdot,\cdot)$ is canonical, we first observe that because $g(\cdot)$ is monotone decreasing, $S^g(\cdot,\alpha)$ is monotone decreasing for all α. To see the second bullet, we simply observe the following facts for any x:

$$\begin{aligned}\Pr_{X\leftarrow U([0,1])}[S^g(X,\sum_{i=1}^n \alpha_i) > g(x)] &= \Pr_{X\leftarrow U([0,1])}[g(X^{1/\sum_{i=1}^n \alpha_i}) > g(x)]\\ &= \Pr_{X\leftarrow U([0,1])}[X^{1/\sum_{i=1}^n \alpha_i} < x]\\ &= \Pr_{X\leftarrow U([0,1])}[X < x^{\sum_{i=1}^n \alpha_i}]\\ &= x^{\sum_{i=1}^n \alpha_i}.\end{aligned}$$

The first line follows by definition of $S^g(\cdot,\cdot)$. The second follows as g is monotone decreasing. The third follows as both $X, x > 0$, and the final line follows as X is drawn uniformly from $[0,1]$. For similar reasons, we have:

$$\begin{aligned}\Pr_{\vec{X}\leftarrow U([0,1])^n}[\min_{i=1}^n\{S^g(X_i,\alpha_i)\} > g(x)] &= \Pr_{\vec{X}\leftarrow U([0,1])^n}[\min_{i=1}^n\{g(X_i^{1/\alpha_i})\} > g(x)]\\ &= \Pr_{\vec{X}\leftarrow U([0,1])^n}[\max_{i=1}^n\{X_i^{1/\alpha_i}\} < x]\\ &= \Pr_{\vec{X}\leftarrow U([0,1])^n}[X_i^{1/\alpha_i} < x,\ \forall i]\\ &= \Pr_{\vec{X}\leftarrow U([0,1])^n}[X_i < x_i^{\alpha},\ \forall i]\\ &= x^{\sum_{i=1}^n \alpha_i}.\end{aligned}$$

Therefore, we see that for all n, and all $\langle\alpha_1,\ldots,\alpha_n\rangle$, the distributions of $S^g(X,\sum_{i=1}^n \alpha_i)$ and $\min_{i=1}^n\{S^g(X_i,\alpha_i)\}$ are identical.

□

For example, the canonical scoring rule we use for our analysis is $S^g(\cdot,\cdot)$ where $g(x) := -\ln(x)$. The canonical scoring rule used in [6] is $S^h(\cdot,\cdot)$ where $h(x) := 1 - x$ (where $\alpha = 1$ denotes that the account owns a single coin).

Lemma B.2. Let S be a canonical balanced scoring function, and define $g(\cdot) := S(\cdot, 1)$. Then $S = S^g$.

Proof. The proof follows from three simple steps: (a) we show that $S(\cdot, 1/n) = S^g(\cdot, 1/n)$ for all integers n. Then, we use this to show that $S(\cdot, c/n) = S^g(\cdot, c/n)$, for all integers c. This concludes the proof for all rational α, which is all we consider.

We now execute the first step. Observe that, because S is canonical, we know exactly what the distribution of $S(X, 1/n)$ must be when $X \leftarrow U([0,1])$. Indeed, we must have, for all x (for notational convenience below defining an inverse, we let $S_{1/n}(x) := S(x, 1/n)$):

$$\begin{aligned}
\Pr_{\vec{X} \leftarrow U([0,1])^n}[\min_{i=1}^{n}\{S(X_i, 1/n)\} > x] &= \Pr_{X \leftarrow U([0,1])}[S(X, 1) > x] \\
\Rightarrow \Pr_{X \leftarrow U([0,1])}[S(X, 1/n) > x]^n &= \Pr_{X \leftarrow U([0,1])}[g(X) > x] \\
\Rightarrow \Pr_{X \leftarrow U([0,1])}[S(X, 1/n) > x] &= \Pr_{X \leftarrow U([0,1])}[g(X) > x]^{1/n} \\
&= \Pr_{X \leftarrow U([0,1])}[X < g^{-1}(x)]^{1/n} \\
&= (g^{-1}(x))^{1/n} \\
\Rightarrow (g^{-1}(x))^{1/n} &= \Pr_{X \leftarrow U([0,1])}[S(X, 1/n) > x] \\
&= \Pr_{X \leftarrow U([0,1])}[X < S_{1/n}^{-1}(x)] \\
&= S_{1/n}^{-1}(x) \\
\Rightarrow S_{1/n}(x) &= g(x^n), \ \forall\, x.
\end{aligned}$$

We now execute the second step, which has nearly identical calculations.

$$\begin{aligned}
\Pr_{X \leftarrow U([0,1])}[S(X, c/n) > x] &= \Pr_{\vec{X} \leftarrow U([0,1])^c}[\min_{i=1}^{c}\{S(X_i, 1/n)\} > x] \\
&= \Pr_{X \leftarrow U([0,1])}[S(X, 1/n) > x]^c \\
&= \Pr_{X \leftarrow U([0,1])}[g(X^n) > x]^c \\
&= \Pr_{X \leftarrow U([0,1])}[X < g^{-1}(x)^{1/n}]^c \\
&= g^{-1}(x)^{c/n} \\
\Rightarrow g^{-1}(x)^{c/n} &= \Pr_{X \leftarrow U([0,1])}[S(X, c) > x] \\
&= \Pr_{X \leftarrow U([0,1])}[X < S_{c/n}^{-1}(x)] \\
&= S_{c/n}^{-1}(x) \\
\Rightarrow S_{c/n}(x) &= g(x^{n/c}), \ \forall\, x.
\end{aligned}$$

□

Proof of Proposition 2.1. By Lemma B.2, we know that both S and S' are of the form S^g and S^h for some monotone decreasing functions g, h. We will use this property to couple outcomes of the two games.

First, we need to define the bijective mapping for each player. The mapping we will use is simple: in each game, split your stake exactly the same way. When choosing which credentials to broadcast, observe that in both games player i has some information available to them (they see the credentials of all accounts they control, plus some other credentials of other players). Then given a strategy for the first game, we can define a strategy for the second game: during round r, broadcast the credential of player i if and only if player i broadcasts its credential in the first game.

Next, we need to couple the two games and claim that under this coupling, for all r, the leader in both games is the same. We will couple the games so that $\textsc{Cred}_r^j$ is the same for all rounds r and accounts j.

Now, observe that because we have mapped strategies of every player to one which distributes their stake identically among accounts, and because we have coupled the games so that $\textsc{Cred}_r^j$ is the same for all rounds j and accounts j, that for all players i, the information available to player i is identical in each game. Therefore, player i will choose to broadcast exactly the same set of credentials. The only remaining step is to confirm that the same leader will be selected in each round because both S and S' are canonical.

Indeed, observe that among the set B_r of broadcast credentials, the winner in the CSSPA with S is exactly:

$$\begin{aligned}\arg\min_{j\in B_r}\{S(\textsc{Cred}_r^j,\alpha_j)\} &= \arg\min_{j\in B_r}\{g((\textsc{Cred}_r^j)^{1/\alpha_j})\}\\ &= \arg\max_{j\in B_r}\{(\textsc{Cred}_r^j)^{1/\alpha_j})\}\end{aligned}$$

The first line follows as $S = S^g$, and the second line follows as $g(\cdot)$ is monotone decreasing. By exactly the same reasoning, we have that the winner in CSSPA with S' is:

$$\begin{aligned}\arg\min_{j\in B_r}\{S'(\textsc{Cred}_r^j,\alpha_j)\} &= \arg\min_{j\in B_r}\{h((\textsc{Cred}_r^j)^{1/\alpha_j})\}\\ &= \arg\max_{j\in B_r}\{(\textsc{Cred}_r^j)^{1/\alpha_j})\}\end{aligned}$$

Therefore, we have shown a mapping between strategies, and a coupling between outcomes, such that in each round the leader in both games is the same. This completes the proof. □

C OMITTED PROOFS FROM SECTION 4

Proof of Lemma 4.3. Consider instead the finite stochastic process $X_1, \ldots, X_n$ where each X_i is an i.i.d. copy from $\textsc{Exp}\left(\frac{\alpha}{n}\right)$. For all $i \in [n]$, define the random variables

$$Y_i = \overset{(i)}{\min}(\{X_1,\ldots,X_n\}), \qquad Z_i = Z_{i-1} + \textsc{Exp}\left(\alpha - \frac{(i-1)\alpha}{n}\right),$$

and let $Y_0 = Z_0 = 0$.

Claim C.1. For all $i \in [n]$, Y_i is identically distributed to Z_i.

Proof. The proof is by induction on $i \geq 0$. Assume for $i \geq 0$, Z_i is identically distributed to Y_i and observe the base case ($i = 0$) follows by definition. Then, it suffices to show that for any absolute constant x,

$$Pr\left[Y_{i+1} > x\right] = Pr\left[Z_{i+1} > x\right] = Pr\left[Z_i + \textsc{Exp}\left(\alpha - \frac{i\alpha}{n}\right) > x\right].$$

Fix $Y_i = z$ where z is any absolute constant. For the case $x < z$, the fact $Y_{i+1} \geq Y_i$ implies

$$Pr\left[Y_{i+1} > x | x < z = Y_i\right] = 1 = Pr\left[Y_i + \textsc{Exp}\left(\alpha - \frac{i\alpha}{n}\right) > x | x < z = Y_i\right].$$

For the case $x \geq z$, let $A = \{j \in [n] | X_j > Y_i\}$ and observe that with probability 1, $|A| = n - i$. Fix $A = S$ for any set $S \subseteq [n]$. Then

$$\begin{aligned}
Pr\left[Y_{i+1} > x | x \geq z = Y_i, A = S\right] &= \prod_{j \in S} Pr\left[X_j > x | x \geq z = Y_i, A = S\right] \\
&= \prod_{j \in S} Pr\left[X_j > Y_i + (x - Y_i) | x \geq z = Y_i, A = S\right] \\
&= \prod_{j \in S} Pr\left[X_j > x - Y_i | x \geq z = Y_i\right] \\
&= \prod_{j \in S} e^{-(x-z)\frac{\alpha}{n}} = e^{-(x-z)(n-i)\frac{\alpha}{n}} \quad \text{Since } |A| = n - i, \\
&= Pr\left[\text{Exp}\left((n-i)\frac{\alpha}{n}\right) > x - Y_i | x \geq z = Y_i\right] \\
&= Pr\left[Y_i + \text{Exp}\left(\alpha - \frac{i\alpha}{n}\right) > x | x \geq z = Y_i\right]
\end{aligned}$$

The first line observes that $Y_{i+1} > x$ if and only if for all $j \in S$, $X_j > x$. The third line observes that $j \in S$ if and only if $X_j > Y_i$ and invokes the memoryless property (Lemma A.3). By the Law of Total Probability and combining the case where $x < z$ and $x \geq z$, we obtain

$$\begin{aligned}
Pr\left[Y_{i+1} > x\right] &= Pr_{Y_i}\left[Pr_A\left[Pr\left[Y_{i+1} > x | Y_i = z, A = S\right] | Y_i = z\right]\right] \\
&= Pr_{Y_i}\left[Pr\left[Y_i + \text{Exp}\left(\alpha - \frac{i\alpha}{n}\right) > x | Y_i = z\right]\right] \\
&= Pr\left[Y_i + \text{Exp}\left(\alpha - \frac{i\alpha}{n}\right) > x\right]
\end{aligned}$$

as desired. □

From Claim C.1 and taking the limit as $n \longrightarrow \infty$ proves the statement. □