Improving Privacy of Anonymous Proof-of-Stake Protocols

Shichen Wu[1,2], Zhiying Song[1,2], Puwen Wei[1,2,3(✉)], Peng Tang[1,2], and Quan Yuan[4]

[1] Key Laboratory of Cryptologic Technology and Information Security, Ministry of Education, Shandong University, Qingdao, China
[2] School of Cyber Science and Technology, Shandong University, Qingdao, China
[3] Quancheng Laboratory, Jinan, China
[4] The University of Tokyo
{shichenw,szyyz}@mail.sdu.edu.cn, {pwei,tangpeng}@sdu.edu.cn, yuanquan@g.ecc.u-tokyo.ac.jp

Abstract. The proof of stake (PoS) mechanism, which allows stakeholders to issue a block with a probability proportional to their wealth instead of computational power, is believed to be an energy-efficient alternative to the proof of work (PoW). The privacy concern of PoS, however, is more subtle than that of PoW. Recent research has shown that current anonymous PoS (APoS) protocols do not suffice to protect the stakeholder's identity and stake, and the loss of privacy is theoretically inherent for any (deterministic) PoS protocol that provides liveness guarantees. In this paper, we consider the concrete stake privacy of PoS when considering the limitations of attacks in practice. To quantify the concrete stake privacy of PoS, we introduce the notion of (T, δ, ϵ)-privacy. Our analysis of (T, δ, ϵ)-privacy on Cardano shows to what extent the stake privacy can be broken in practice, which also implies possible parameters setting of rational (T, δ, ϵ)-privacy for PoS in the real world. The data analysis of Cardano demonstrates that the (T, δ, ϵ)-privacy of current APoS is not satisfactory, mainly due to the deterministic leader election predicate in current PoS constructions. Inspired by the differential privacy technique, we propose an efficient non-deterministic leader election predicate, which can be used as a plugin to APoS protocols to protect stakes against frequency analysis. Based on our leader election predicate, we construct anonymous PoS with noise (APoS-N), which can offer better (T, δ, ϵ)-privacy than state-of-the-art works. Furthermore, we propose a method of proving the basic security properties of PoS in the noise setting, which can minimize the impact of the noise on the security threshold. This method can also be applied to the setting of PoS with variable stakes, which is of independent interest.

Keywords: Blockchain · Proof of stake · Privacy · Verifiable random function.

1 Introduction

Proof of work (PoW) based blockchain protocols, such as bitcoin [1], provide a novel way to achieve consensus among users in a permissionless setting. However, one of the main concerns of PoW is its high energy consumption. To address this issue, Proof of Stake (PoS) protocols have emerged as a promising, energy-efficient alternative. In PoS protocols, users participate in a process to elect a leader who will propose the next block. The probability of a user winning the election is proportional to their wealth or relative stakes at any given time. The rationale behind PoS is that users with higher relative stakes have more economic incentives to keep the PoS system running, and their stakes would lose value if the system fails. In the past decade, a series of solid works focused on the candidates of PoS protocols [2,3,4,5,6,7,8,9,10]. In particular, [5,6,7,8,9,10] have presented PoS with rigorous security proofs and formal security models.

Due to the public nature of the permissionless setting, privacy has become a significant concern for blockchain users. For PoW based blockchains, privacy-preserving solutions such as ZCash[11] and Monero[12] have been developed to provide privacy protection for transactions, including the payer/payee identities and transaction amounts. However, achieving privacy in PoS based blockchains is more challenging. This is because in PoS, privacy not only needs to be ensured for transactions and identities but also for the leaders' stakes. In particular, a PoS user (or stakeholder) needs to provide public verifiable proof of his leadership, which is verified based on his public key and stakes. Even if this proof is realized in a zero-knowledge manner, as in the PoW setting, the number of times an anonymous leader wins the election implies an approximation of his stakes. To protect the stakeholders' identities and stakes, anonymous PoS protocols have been proposed [10,13,14].

Nevertheless, [15] has pointed out that current anonymous PoS protocols do not suffice to protect the stakeholder's identity and stake, and has shown the theoretical impossibility of a PoS blockchain protocol that guarantees both liveness and anonymity. Specifically, they introduce the tagging attack, which can leverage the network delay to distinguish the target stakeholder from others. Once the adversary can launch the tagging attack for the target stakeholder "enough" times, they can reveal the target stakeholder's stake since the frequency of winning the election is determined by participants' stakes in PoS. Theoretically, such leakage is inherent for deterministic PoS protocols when considering the network delay. In fact, the security loss through frequency attack (or frequency analysis) is inherent for any deterministic cryptographic schemes. To mitigate the tagging attack, [15] provides possible countermeasures, such as sanitization protocol and reliable broadcast mechanisms, which aim to ensure that all parties have the same view. These strategies, however, rely on additional assumptions on the network and have limitations on either privacy or practicality and scalability. As mentioned in [15], new technologies are needed to protect stakeholders' privacy against any potential network adversary.

It is worth noting that network attacks in the real world have limitations, as it takes time for an adversary to launch an attack, and the attack is not

always successful. For instance, the success probability of eclipse attacks [16] is about 84%, and the attack may be stopped once the target user restarts the server. That means that the adversary of tagging attack may not be able to collect enough information to determine the target stakeholder's stake due to the limited duration of the attack. Therefore, it is natural to question the extent to which the frequency attacks (including tagging attack) can break the stake privacy of current anonymous PoS protocols and how to enhance the stake privacy of anonymous PoS against frequency attacks while preserving efficiency and scalability.

Our contributions. In this work, we answer this question by proposing an effective method to enhance the privacy of stakes of anonymous PoS protocols. To that end, we first analyze the success probability of estimating the target stakeholder's stake using frequency attacks, such as repeated (reverse) tagging attacks or any attacks that exploit frequency analysis. We note that the estimation accuracy is heavily influenced by the number of attacks and the success probability of leader election in PoS system. The small number of attacks and success probability of leader election could lead to large estimation errors due to the inherent limitation of statistical methods. We then introduce the notion of (T, δ, ϵ)-privacy to quantify the concrete stake privacy of PoS. In particular, we analyze the (T, δ, ϵ)-privacy of Cardano, which is one of the largest PoS systems by market capitalization. Our results show that for the stake pools of Cardano with small relative stake, say $\leq 0.05\%$, the (T, δ, ϵ)-privacy it can achieve is $T = 432000$ slots, $\delta = 10\%$ and $\epsilon = 60.95\%$. That is, if the attack duration is restricted in one epoch (432000 slots), the probability that the adversary can approximate the target stake pool's stake with an error $\delta = 10\%$ is as high as 60.95%.

Furthermore, we find that the crux of the stake estimation by frequency analysis is the deterministic relation between the stakeholder's stake and his success probability of leader election. Inspired by the differential privacy technique [17,18], we propose an efficient non-deterministic leader election function that can randomize this relation by adding noise with a particular distribution such that the resulting stake estimation error can be increased significantly. Based on our noisy leader election function, we provide an anonymous PoS protocol with noise (APoS-N), which can enhance the stake privacy while preserving the stakeholders' long-term benefits. The main idea is to add "random" noise to the stakeholders' stakes, and the leader election function is evaluated using the noisy stake, where the expectation of the noise distribution is 0. Following Ganesh et al.'s framework [13] of constructing anonymous PoS, we can construct APoS-N by implementing the underlying leader election function with our noisy version. Due to the interference of the noise, it is difficult for the adversary to get the target stakeholder's accurate stake in APoS-N, resulting in better (T, δ, ϵ)-privacy being achieved. In addition, the privacy requirements defined by [13] are preserved in our APoS-N, as it follows the framework of [13].

The main challenge, however, is that the basic security properties of the underlying PoS blockchain, i.e., common prefix, chain growth and chain quality, may not hold due to the random noise. For instance, the noisy stakes of either all stakeholders or the adversary may be larger than the original one, which means the original threshold of adversarial relative stakes, say 1/2, could be broken in APoS-N. To address this problem, we improve the security analysis in [7,8], called characteristic string, to adapt to our noise setting. This improvement can minimize the impact of the noise on the security threshold. Our results show that the basic security properties of PoS can still be preserved in APoS-N if the noise is upper-bounded properly. It is worth noting that our proof can be applied to the setting of PoS with variable stakes, such as when the total active stakes of some slots are less than expected due to the absent stakeholders. This result is of independent interest.

Related work Our work is independent of another work by Wang et al. [19]. Their work extends the tagging attack model of [15] to the randomized PoS protocol and presents a practical stake inference attack with sublinear complexity. In our work, the analysis of frequency attack considers the concrete cost and accuracy of stake estimation, which are applicable to any attacks that rely on sampling frequency. Wang et al. [19] also propose a private PoS protocol using differential privacy techniques. However, we note that their protocol has security flaws. Specifically, we present an attack that allows the adversary to amplify his noisy stakes and gain more profits than required, breaking chain quality, which is one of the fundamental security requirements of the underlying PoS [8]. Even worse, this also implies the break of safety discussed in [8]. The presence of security flaws in Wang et al.'s approach is due to a limitation of the UC framework, which makes it difficult to capture all desired security requirements in the ideal functionality explicitly. In contrast, our protocol carefully controls the noisy stakes to preserve the fundamental security requirements of PoS, including common prefix, chain growth and chain quality. By explicitly considering these requirements, our protocol can provide stronger security guarantees than the approach used by Wang et al. [19].

2 Preliminaries

Notations Let $\mathbb{N}$ denote the set of all natural numbers. Let $B(n, p)$ denote the binomial distribution with parameters n and p, where n denotes the total number of independent trials and p denotes the success probability of each trial. $Be(a, b)$ denotes the beta distribution with parameters a and b. $U(a, b)$ represents the uniform distribution on the interval $[a, b]$. We write $X \sim D$ to denote the random variable X following the distribution D.

Ouroboros Praos We briefly recall *Ouroboros Praos* [8] and its anonymous version[13], which are typical PoS protocols with rigorous security proofs.

Ouroboros Praos works as follows: Suppose that n stakeholders $U_1, \ldots, U_n$ interact throughout the protocol. The stakeholders' initial stakes and related public keys, say $\{(stk_i, pk_i)\}_{i=1}^n$, are hardcoded into the genesis block. Let STK denote the total stakes of the PoS system. During the execution, the time is divided into discrete units called slots, and a set of n_e adjacent slots is called an epoch. Stakeholders participate in the leader election protocol in each slot to decide who is eligible to issue a block. In the process of leader election, each stakeholder locally evaluates a special verifiable random function (VRF) on the current slot and a nonce that is determined for an epoch. Let (y, π) denote the output of VRF, where y is pseudorandom and π is the proof. If y is less than a function of their stakes, then that stakeholder wins the election and can generate a new block. The probability of winning an election is proportional to the stakeholder's relative stake. More specifically, the leader election process can be captured by $Lottery\ Protocol^{\mathcal{E},LE}$ [13], where $\mathcal{E}$ is the set of the allowed entry parameters. The core of $Lottery\ Protocol^{\mathcal{E},LE}$ is a leader election predicate $LE(\cdot,\cdot)$. A stakeholder wins an election in a slot sl iff his $LE(stk, y) = 1$, where stk is the stakeholder's stake. The LE predicate has the following form:

$$LE(stk, y) = \begin{cases} 1, \text{ if } y < 2^{\ell_\alpha} \cdot (1 - (1-f)^{\frac{stk_i}{STK}}) \\ 0, \text{ otherwise.} \end{cases}$$

$\frac{stk_i}{STK}$ is the stakeholder's relative stake. ℓ_α denotes the output length of the VRF and f is called the active slots coefficient, which is the probability that a hypothetical party with 100% relative stake would be elected leader in a slot. A critical property of LE is that the probability of a stakeholder becoming a slot leader depends on his stake, whether this stakeholder acts as a single party or splits his stake among several virtual parties. Once a leader proposes a new block, all the stakeholders can check the validity of the block using the leader's public information, say stake, public key, π, etc., and update their local state by following the longest chain rule, which enables the honest users to converge to a unique view.

Anonymous PoS protocols (APoS) [10,13] focus on establishing a privacy-preserving election process that can protect the leader's identity and stakes. In order to hide the stakes, the stakeholders in APoS need to generate commitments to their stakes. Using these commitments and the list of all stakeholders' identities (ID), the stakeholder can execute $Lottery\ Protocol^{\mathcal{E},LE}$ in a zero-knowledge manner, which means all the users can check the validity of the leader election (or the block) without knowing the leader's identity and stake.

The related $Lottery\ Protocol^{\mathcal{E}}$ are described assuming hybrid access to ideal functionalities such as $\mathcal{F}_{Init}^{Com}$, $\mathcal{F}_{crs}$, $\mathcal{F}_{VRF}^{Com}$ and $\mathcal{F}_{ABC}^{\Delta}$. The functionality $\mathcal{F}_{Init}^{Com}$ initially contains a list of stakeholder's ID and their stakes. It computes the commitments to each stakeholder's stake and generates the corresponding public/secret key pairs. The functionality $\mathcal{F}_{crs}$ provides the common reference string for zero-knowledge proofs. To hide the identity of the sender, anonymous PoS protocols [10,13] need to rely on an ideal anonymous broadcast channel, which is captured by the functionality $\mathcal{F}_{ABC}^{\Delta}$. It takes as input a message m from a user

and adds m to all users' buffers, where the adversary can influence the buffer of the user by introducing bounded delays. In particular, the adversary is allowed to send anonymous messages to specific users and impose an upper bound delay Δ on specific messages. Stakeholders use the functionality $\mathcal{F}_{VRF}^{Com}$ to generate the randomness for the leader election. For each stakeholder, $\mathcal{F}_{VRF}^{Com}$ generates a unique key as a private identity for accessing $\mathcal{F}_{VRF}^{Com}$ and a commitment to randomness y, which the stakeholder uses for the leader election. The commitment is used by users to check the validity of the claimed $\mathcal{F}_{VRF}^{Com}$ evaluation. More details of the above functionalities are shown in Appendix C.

Threat Model The threat model in our paper is similar to that of [8], where the adversary $\mathcal{A}$'s capabilities are defined in the following three aspects:
Corruption: $\mathcal{A}$ is able to corrupt a set of stakeholders adaptively without delay and control these corrupted stakeholders to take any actions beyond the protocol, such as withholding blocks or publishing multiple blocks when they are leaders. In each slot, the fraction of the stake controlled by $\mathcal{A}$ cannot be greater than 50%, otherwise, the security of the PoS protocol can be broken directly.
Propagation: $\mathcal{A}$ can arbitrarily manipulate the propagation of honest messages within Δ slots. Specifically, for any honest message m sent in slot i, the adversary $\mathcal{A}$ can choose the time when each honest stakeholder receives m, but all honest must have received m at the end of slot $i + \Delta$. Notice that the messages sent by honest stakeholders could be new blocks, transactions, or other information.
Limitation: $\mathcal{A}$ has limited computing power so that it cannot violate the security properties of any underlying cryptographic component. For corruption and propagation, this means that the adversary cannot make the probability of corrupted stakeholders being elected leader exceed the adversary's stake proportion, nor can it tamper with honest messages, which requires $\mathcal{A}$ to break the security of the underlying VRF or digital signatures.

Security Requirements The basic security properties of PoS follow that of [7]. A PoS protocol Π that implements a robust transaction ledger should satisfy the persistence and liveness. [20,21] demonstrate that persistence and liveness can be derived from the following three properties if the protocol Π uses the blockchain data structure to export the ledger.

- Common Prefix (CP) with parameters $k \in \mathbb{N}$. The chains C_1, C_2 possessed by two honest parties at the onset of the slots $sl_1 < sl_2$ are such that $C_1^{\neg k} \preceq C_2$, where $C_1^{\neg k}$ denotes the chain obtained by removing the last k blocks from C_1, and $\preceq$ denotes the prefix relation.
- Chain Quality (CQ) with parameters $\mu \in (0, 1]$ and $k \in \mathbb{N}$. Consider any portion of the length at least k of the chain possessed by an honest party at the onset of a slot, the ratio of blocks originating from the adversary is at most $1 - \mu$, where μ is the chain quality coefficient.
- Chain Growth (CG) with parameters $\tau \in (0, 1]$ and $s \in \mathbb{N}$. Consider the chains C_1 and C_2 possessed by two honest parties at the onset of two slots

sl_1, sl_2 with sl_2 at least s slots ahead of sl_1. Then it holds that $len(C_2) - len(C_1) \geq \tau \cdot s$, where τ is the speed coefficient and $len(C_i)$ denotes the length of the chain C_i.

On the privacy of anonymous PoS, [13] introduces the private lottery functionality $\mathcal{F}_{Lottery}^{\mathcal{E},LE}$ to capture the privacy requirements of anonymous PoS in the universal composition (UC) setting. Loosely speaking, $\mathcal{F}_{Lottery}^{\mathcal{E},LE}$ can be considered as an ideal-world PoS protocol that can hide the leader's identity and stake. For more information of $\mathcal{F}_{Lottery}^{\mathcal{E},LE}$, we refer to [13]. We emphasize that the privacy defined by $\mathcal{F}_{Lottery}^{\mathcal{E},LE}$ does not rule out the possibility of privacy leakage by tagging attack described below.

3 Attack on Anonymous PoS and Its Limitations

In this section, we introduce frequency attack, which abstracts any attacks (including tagging attack) that estimate the target stakeholder's stake using frequency analysis. Then, we analyze the accuracy of the stake estimation and show its limitations in practice.

3.1 Frequency Attacks against Stake Privacy

The frequency attack against stake privacy is an attack that may use various methods to determine the number of blocks proposed by the target stakeholder within a specific time period and then uses the frequency of proposed blocks to estimate the stakeholder's stake. The adversary can monitor either the physical or network layer to obtain the block frequency. A typical example of frequency attacks is the tagging attack [15], which can manipulate the targeted stakeholder's network delays to create a different view from others, enabling the adversary to distinguish blocks proposed by the targeted stakeholder and associate them with their stake. More precisely, the adversary creates a transaction tx_{Δ} for the purpose of tagging the targeted stakeholder P. By controlling the network delay, the adversary is capable of ensuring that stakeholder P receives tx_{Δ} at time t, while other stakeholders receive it after time $t + \Delta$. Notice that if a stakeholder succeeds in winning an election, then it adds all the transactions in his current view to the new block. For any block B that is produced between t and $t + \Delta$, the adversary is able to check whether tx_{Δ} is in B even if it can achieve privacy-preserving since the adversary is the owner of tx_{Δ}. As no one has tx_{Δ} before $t + \Delta$ except P, tx_{Δ} in B indicates that B is generated by P. By repetitively executing this attack, the adversary can determine the frequency of blocks proposed by P during a specific period. Then, the frequency can be exploited to uncover the relative stake of P, thereby compromising the stake privacy of the PoS system.

Theoretically, the relative stake of P can be approximated by statistical analysis, e.g., point estimation or interval estimation of the probability of success in a binomial distribution. Note that all the statistical methods have their limitations

on the accuracy of the approximation due to the target probabilistic distributions and the number of samples. We show the accuracy of interval estimation, which is crucial to the stake privacy of anonymous PoS in practice. Interval estimation is an effective statistical method to estimate an interval of possible values of the unknown population parameters. For the stake estimation of PoS, which follows the binomial distribution with a small success probability, we adopt the Jeffreys interval rather than the standard interval in order to reduce the severity of the chaotic behavior of the confidence interval's coverage probability [22].

To illustrate the interval estimation for stakes, consider the following case. Suppose that the total number of slots during the attack is C and the target stakeholder's stake is fixed. Let $\mathsf{suc}[C, t]$ denote the event that t blocks proposed by P among C slots are observed by the adversary. Let p denote the relative stake of P. We use X to indicate whether P wins the election in a slot, where $X = 1$ if P wins the election. Otherwise, $X = 0$. We use $\Phi(\cdot)$ to denote the function which takes as inputs a stakeholder's relative stake and outputs the corresponding probability that he can win the election in a slot. The probability of P winning an election is $\Phi(p)$[5]. Since $\Phi(\cdot)$ is usually public and deterministic and p can be easily obtained given $\Phi(p)$, we focus on the estimation of $\Phi(p)$ to simplify our illustration. So X follows the Bernoulli distribution with $\Pr[X = 1] = \Phi(p)$ and t follows the binomial distribution with parameters C and $\Phi(p)$, i.e., $t \sim B(C, \Phi(p))$.

To estimate the unknown $\Phi(p)$, we apply the Jeffreys interval, which is the Bayesian confidence interval obtained using the non-informative Jeffreys prior of the binomial distribution $\Phi(p)$. The Jeffreys prior is a Beta distribution with parameters $(1/2, 1/2)$. The posterior distribution is derived from $\mathsf{suc}[C, t]$, which follows the Beta distribution $Be(t+1/2, C-t+1/2)$. The $100(1-\psi)\%$ equal-tailed Bayesian interval is $[Q(\psi/2; t+1/2, C-t+1/2), Q(1-\psi/2; t+1/2, C-t+1/2)]$, where Q is the quantile function of $Be(t + 1/2, C - t + 1/2)$.

3.2 Interval Estimation for Stakes in Practice

Following the above method, we estimate the stakes of Cardano [23] to show the accuracy of interval estimation in practice. We choose 100 stake pools with total relative stake $p \approx 26.732\%$ as the target stakeholder P with $\Phi(p) = \Pr[X = 1] \approx 1.362\%$. By analyzing the data of two different periods in epoch 325, which are $\mathsf{suc}[3000, 66]$ (1 hour) and $\mathsf{suc}[345600, 4994]$ (96 hours), we get the Jeffreys intervals for $\Phi(p)$, respectively. Figure 1 shows the estimation of $\Phi(p)$ using Jeffreys intervals, where the red line and the blue line represent the probability density functions of $\Phi(p)$ using $\mathsf{suc}[3000, 66]$ (1 hour) and $\mathsf{suc}[345600, 4994]$ (96 hours), respectively. When considering confidence level 95%, the Jeffreys intervals of the red line is [0.01721, 0.02773] with interval length 0.01052. For the blue line, the Jeffreys interval is [0.01406 0.01486] with interval length of 0.0008. So far, it follows the intuition that a large number of blocks that knew by the adversary

[5] In Ouroboros, the probability of P winning an election is defined by $\Phi(p) = 1 - (1 - f)^p$, which is close to $p \cdot f$.

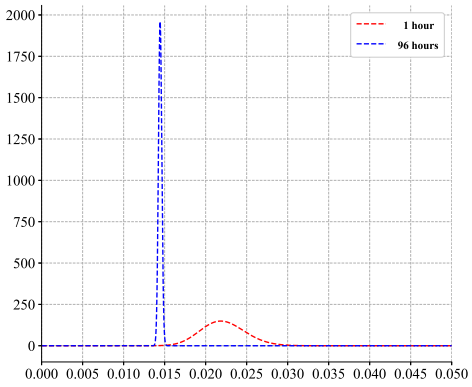

Fig. 1. Probability density function of $\Phi(p)$.

can improve the estimation accuracy for p. However, we stress that the relative stake p of a stake pool in Cardano is only about $0.001\% \sim 0.35\%$ and the corresponding $\phi(0.01\%) \sim \phi(0.35\%)$ is $(0.000513\% \sim 0.0179\%)$. That means even interval length 0.0008 is too large to distinguish stakeholders' $\Phi(p)$ in Cardano. So frequency attack and Jeffreys intervals is not accurate enough to distinguish most stakeholders' stake in Cardano when the attack duration is "short", say 96 hours (345600 slots).

Furthermore, more trials do not necessarily imply a more accurate estimation. [22,24] reveal the degree of severity of the chaotic oscillation behavior of many intervals' coverage probability. Such chaotic oscillation behavior is more obvious for the binomial distribution with relatively small p. For instance, for a binomial distribution $B(C, p)$ with $p = 0.005$ [22], the coverage probability of the 95% confidence interval increases monotonically in C until $C = 591$ to 0.945, and drops to 0.792 when $C = 592$.

The above limitations of statistical analysis show the possibility of protecting the stakes of anonymous PoS in practice.

4 Privacy of PoS against Frequency Attack

In this section, we introduce the notion of (T, δ, ϵ)-privacy to capture the concrete stake privacy of PoS and analyze the (T, δ, ϵ)-privacy of Cardano.

4.1 (T, δ, ϵ)-privacy

In theory, if the adversary has an infinite amount of time to acquire the frequency of proposed blocks, they could precisely ascertain the stake of any stakeholder. However, in practice, the attack time for the adversary to gather information

about the frequency of proposed blocks is usually limited. To conduct a more comprehensive evaluation of the costs and effects of frequency attacks in real-world scenarios, we need to consider the attack time. Let T denote the number of slots that an adversary can perform frequency attack. Hence, for a stakeholder with a relative stake p, the expected number of blocks generated by it during T slots is $T \cdot \Phi(p)$. We capture the concrete stake privacy of a PoS protocol Π by the following experiment, called $\mathsf{Exp}^{\mathcal{A}}_{\Pi,\delta}$.

- The challenger runs the protocol Π among n stakeholders.
- The adversary $\mathcal{A}$ chooses the target stakeholder (or stakeholders) S to launch the frequency attack, where the relative stake of S is p. Suppose that the frequency attack can last for T slots.
- Finally, $\mathcal{A}$ outputs X, which denotes the number of "tagged" and valid blocks generated by S.

We say the adversary $\mathcal{A}$ wins the experiment $\mathsf{Exp}^{\mathcal{A}}_{\Pi,\delta}$ if $(1-\delta) \cdot T \cdot \Phi(p) \leq X \leq (1+\delta) \cdot T \cdot \Phi(p)$, where $\delta \in (0,1)$. Let $\mathsf{Exp}^{\mathcal{A}}_{\Pi,\delta}(1^\lambda) = 1$ denote the event that $\mathcal{A}$ wins, where λ denotes the security parameter.

Definition 1. *(T,δ,ϵ)-privacy: A PoS protocol Π is (T,δ,ϵ)-privacy for a stakeholder with relative p if for any PPT adversary $\mathcal{A}$, $\Pr[\mathsf{Exp}^{\mathcal{A}}_{\Pi,\delta}(1^\lambda) = 1] \leq \epsilon$, where $0 < \epsilon < 1$ and δ is called the privacy error.*

Note that (T,δ,ϵ)-privacy captures to what extent the stake privacy of a PoS protocol can achieve no matter which statistical tool or strategies the adversary would use. Consider the case of Ouroboros Praos, we have $X \sim B(T,\Phi(p))$ and

$$\Pr[\mathsf{Exp}^{\mathcal{A}}_{\Pi,\delta}(1^\lambda) = 1] = \sum_{i=\lfloor (1-\delta)T\Phi(p) \rfloor}^{\lfloor (1+\delta)T\Phi(p) \rfloor} \Pr[X = i] \approx 60.95\%, \tag{1}$$

where the target stakeholder's relative stake $p = 0.3\%$ and $T = 432000$ (the number of slots in an epoch). In fact, due to the law of large numbers, typical PoS protocols usually cannot achieve (T,δ,ϵ)-privacy when T is large enough. Specifically, when $T \to \infty$ and Φ is deterministic, $\Pr[(1-\delta)T\Phi(p) \leq X \leq (1+\delta)T\Phi(p)] \to 1$. As shown in the previous section, when T and p are small, the accuracy of the estimation for target stakes is heavily influenced by the limitation of the underlying statistical analysis. So (T,δ,ϵ)-privacy depends on the duration of frequency attacks and the actual probability of the target stakeholder proposing a block.

4.2 (T,δ,ϵ)-privacy in practice

To measure the impact of frequency attacks on (T,δ,ϵ)-privacy in practice, we make a thorough analysis of the data of Cardano. Note that the underlying PoS protocol of Cardano is Ouroboros, which is also the core of anonymous PoS protocols [10,13]. While employing privacy-preserving techniques, the probability of

stakeholder winning an election in [10,13] does not change. Hence, the block data of Cardano can reflect (T, δ, ϵ)-privacy of [10,13] in practice, although Cardano does not consider anonymity. We assume that the adversary can successfully find all the blocks generated by the target stakeholder during the attack. That is, X is the number of the blocks generated by the target stakeholder during the attack.

We investigate the transactions of Cardano for two months and focus on 600 pools, denoted by $\mathcal{S}_{total}$, which have more than 90% stakes of the entire system. To evaluate the error of frequency attack for stake estimation, we define the frequency attack error as $R = \left|1 - \frac{X}{T\Phi(p)}\right|$. Due to Definition 1, δ is the upper bound of R to break (T, δ, ϵ)-privacy.

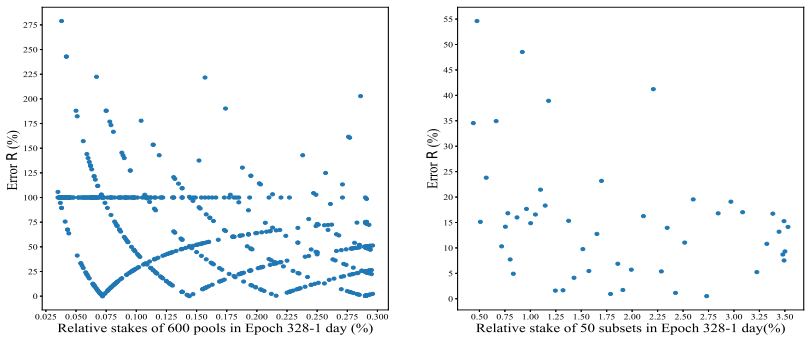

Fig. 2. Each blue dot represents a pool or a subset of pools where the x-coordinate denotes its relative stake and the y-coordinate denotes the corresponding frequency attack error during the first day of epoch 328. In the right figure, 600 pools are divided into 50 subsets, where each blue dot represents a subset of 12 pools.

Figure 2 (left) shows the relation between each pool in $\mathcal{S}_{total}$ and its corresponding frequency attack error R in the first 24 hours of epoch 328. In the horizontal axis, all the pools in $\mathcal{S}_{total}$ are sorted in ascending order of their relative stakes. For instance, there are stake pools with relative stake 0.3%, which have error 100.1%, 57.8% and 1.58%, respectively. By "merging" multiple pools in different ways, we can simulate multiple frequency attacks on different pools using the same transaction data. More precisely, 600 pools are randomly divided into subsets of equal size. In Figure 2 (right), 600 pools are divided into 50 subsets, each of which has 12 pools. The horizontal coordinate and the vertical coordinate denote the total relative stake and the frequency attack error R of a subset, respectively. As illustrated in Figure 2, the larger the relative stake, the less the frequency attack error is. In particular, all the subsets with relative

stake about 3.5% in Figure 2 (right) has error less than 20%, while most pools with relative stake about 0.05% in Figure 2 (left) has error larger than 50%.

One may consider $\delta = 10\%$, since the difference of relative stake of most adjacent pools on the horizontal axis of Figure 2 (left) is about 10%. So it is possible for pools with relative stake less than 0.3% to preserve (T, δ, ϵ)-privacy if $T = 432000$ slots, $\delta = 10\%$ and $\epsilon = 60.95\%$.

Table 1. Proportion of pools (subsets of pools) such that $R > \delta$.

Epoch 328		Proportion s.t. $R > \delta$			
		$\delta = 0.1$	$\delta = 0.2$	$\delta = 0.3$	$\delta = 0.4$
24 hours	600sets	86.5%	78.7%	66.2%	60.3%
	200sets	78.5%	63.5%	43.5%	32.0%
	100sets	72.0%	49.0%	32.0%	18.0%
	50sets	64.0%	18.0%	12.0%	6.0%
48 hours	600sets	78.5%	60.3%	46.7%	32.8%
	200sets	66.0%	39.0%	18.5%	8.0%
	100sets	55.0%	23.0%	9.0%	1.0%
	50sets	36.0%	6.0%	4.0%	0%
72 hours	600sets	74.8%	51.0%	31.8%	19.2%
	200sets	57.5%	26.5%	9.5%	3.0%
	100sets	35.0%	11.0%	3.0%	0%
	50sets	18.0%	2.0%	0%	0%
96 hours	600sets	64.2%	38.2%	19.5%	9.8%
	200sets	48.5%	16.5%	4.5%	1.0%
	100sets	34.0%	7.0%	0%	0%
	50sets	12.0%	0%	0%	0%

Intuitively, the frequency attack error will be decreased when the adversary can extend the duration of frequency attack. Table 1 shows the effect of extending the duration of frequency attack for epoch 328. In Table 1, we show how the proportion of pools (or sets of pools) with $R > \delta$ changes over time in epoch 328. "200sets (resp. 100sets, 50sets)" means that we choose 3 (resp. 6, 12) pools as a set. Table 1 shows that the proportion of the subsets with $R > 0.1$ in the first 24 hour of epoch 328 is greater than 50%, while the proportion of the subsets with $R > 0.4$ drops to about 0% by the first 96 hours of epoch 328.

Similar phenomena occur in different epochs (shown in Table 2 in Appendix D), where the proportion such that $R > 0.1$ in the first 24 hours of each epoch is greater than 83%. To sum up, comparing with larger relative stakes, say $2\% \sim 3\%$, smaller relative stakes, say $0.03\% \sim 0.3\%$, can dramatically reduce the accuracy of stake estimation in a short period of time, say 1 day. In addition, stakes in Cardano has "dense" distribution, where there are many pools with similar relative stakes. The above results implies the possibility for anonymous PoS protocols in practice to achieve (T, δ, ϵ)-privacy when considering $\delta = 0.1 \sim 0.2$. As shown above, the corresponding ϵ of stakeholder with relative stake 0.3%

can reach 60.95% in the first day of an epoch, which is too high for the privacy in practice. Next, we show how to reduce ϵ further.

5 Anonymous Proof-of-Stake with Noise

In this section, we construct the anonymous PoS with noise (APoS-N). In particular, we propose a non-deterministic leader election function, which can be used as a plug-in for PoS based blockchain to enhance stake privacy.

5.1 Adding Noise to Anonymous PoS

As shown in frequency attack, the adversary may determine the target stakeholder's stake by the frequency of "tagged" blocks, i.e., the frequency of $LE(stk, y) = 1$ for the target stakeholder. In order to hide the stakeholders' stake, we change the frequency of $LE(stk, y) = 1$ during a short period by adding noise to the stake stk. The main idea of our techniques is similar to differential privacy [17,18], which can preserve the data's privacy and statistical validity by adding noise with a particular distribution.

More specifically, we modify LE such that the probability of a stakeholder winning election depends on his "noisy" stake. The noise is generated by a noise function $\gamma(\cdot)$, which takes as input random value z and outputs a value following a particular distribution $\mathcal{D}$. The expectation of distribution $\mathcal{D}$ should be 0, e.g., the uniform distribution with expectation 0, so that the frequency of a stakeholder becoming a leader over the long term would not be changed. That means, the frequency of a stakeholder becoming a leader during a long period of time is still proportional to his stake, but during a short period of time, it is hard to estimate the probability of a stakeholder becoming a leader due to the noise. Our modified leader election predicate, called LE^*, is described as follows.

$$LE^*(stk; \eta) = \begin{cases} 1, \text{if } y < 2^{\ell_\alpha} \cdot (1 - (1-f)^{\frac{stk \cdot (1+\gamma(z))}{STK}}) \\ 0, \text{otherwise} \end{cases}$$

where $\eta = y||z$ is generated by querying $\mathcal{F}_{VRF}^{Com}$.

Comparing with the definition of $\mathcal{F}_{VRF}^{Com}$ in [13], we make slight modifications that the randomness η returned by $\mathcal{F}_{VRF}^{Com}$ is divided into two parts, i.e., $\eta = y||z$, where y is the same as that of [13], and z is used for the noise function $\gamma(\cdot)$. Formal description of our $\mathcal{F}_{VRF}^{Com}$ is given below, where Com denotes the commitment scheme.

Functionality $\mathcal{F}_{VRF}^{Com}$

Key Generation

Upon input (KeyGen,sid) from a stakeholder uid, generate a unique key vid and set $U(vid) = uid$. Return (KeyGen, sid, vid) to uid.

VRF Evaluation

Upon receiving a request (Eval, sid, vid, m) from stakeholder uid, check whether $U(vid) \stackrel{?}{=} uid$. If not, ignore the request.

1. If $T(vid, m)$ is undefined, pick random η, r from $\{0,1\}^{\ell_{VRF}}$, where $\eta = y||z$.
2. Set $T(vid, m) = (\eta, Com(\eta; r), r)$.
3. Return (Evaluated, $sid, T(vid, m)$) to stakeholder uid.

VRF Verification
Upon receiving (Verify, sid, m, c) from some user, set $b = 1$ if there exists a vid such that $T(vid, m) = (\eta, c, r)$ for some η and r. Otherwise, set $b = 0$. Output (Verified, sid, m, c, b) to the user.

When instantiated with concrete VRF, the output of the corresponding VRF is longer than that of [13]. Note that $(1-(1-f)^{\frac{stk\cdot(1+\gamma(z))}{STK}}) \approx f \cdot (\frac{stk\cdot(1+\gamma(z))}{STK})$ still holds since $f \cdot (\frac{stk\cdot(1+\gamma(z))}{STK}) << 1$ if the slot and f is small enough.

Following the framework of [13], we present the modified *Lottery Protocol*$^{\mathcal{E},LE^*}$ below, where the main difference is that we use the noisy version of leader election predicate LE^*. Each stakeholder, say U, runs the modified *Lottery Protocol*$^{\mathcal{E},LE^*}$ to join the leader election. More details of related ideal functionalities $\mathcal{F}_{Init}^{Com}$, $\mathcal{F}_{crs}$, and $\mathcal{F}_{\Delta}^{ABC}$ are shown in Appendix C.

Lottery Protocol$^{\mathcal{E},LE^*}$

Suppose the underlying signature scheme consists of (SIG.keygen, SIG.sign, SIG.vrfy), which denote the key generation algorithm, the signing algorithm and the verification algorithm, respectively.

Initialzation

- Send (GetList, sid) to $\mathcal{F}_{Init}^{Com}$ to get the list $\mathcal{L}$ of stakeholders with committed stake and the corresponding signature verification key.
- Send (Setup, sid) to $\mathcal{F}_{crs}$ to get the common reference string crs for zero-knowledge proofs.
- If U is a stakeholder, send (Get-private-Data, sid) to $\mathcal{F}_{Init}^{Com}$ to get $\alpha_{uid}, r_{\alpha,uid}, sk_{uid}$, and send (KeyGen, sid) to $\mathcal{F}_{VRF}^{Com}$ to get vid. Initialize $V(\cdot) = \{\phi\}$.

Lottery and Publishing

- As a stakeholder upon receiving (Lottery, sid, e):
  1. Ignore the request if e is not in $\mathcal{E}$, which is the set of allowed entry parameters.
  2. If $V(e)$ is undefined, send (Eval, sid, vid, e) to $\mathcal{F}_{VRF}^{Com}$ and get (Evaluated, $sid, (\eta, c, r)$). Compute $b = LE^*(\alpha_{uid}, \eta)$, and set $V(e) = (b, \eta, c, r)$.
  3. Return (Lottery, sid, e, b) where $V(e) = (b, \eta, c, r)$.

- As a stakeholder upon receiving (Send, sid, e, m)
  1. Ignore the request if $V(e) = (0, \cdots)$ or is undefined.
  2. If there exists $(1, \eta, c, r)$ such that $V(e) = (1, \eta, c, r)$,
     (a) Generate a signature σ on (e, m) under vk_{uid}.
     (b) Generate a zero-knowledge proof π_{zk} using crs for the following statement.
     $$\{(\alpha_{uid}, r_{\alpha,uid}, vk_{uid}, sk_{uid}, c_{\alpha,uid}, \sigma, \eta, r) : \mathsf{SIG.vrfy}_{vk_{uid}}((e, m), \sigma) = 1 \wedge LE^*(\alpha_{uid}, \eta) = 1 \wedge vk_{uid} = \mathsf{SIG.keygen}(sk_{uid}) \wedge c = Com(\eta; r) \wedge c_{\alpha,uid} = Com(\alpha_{uid}; r_{\alpha,uid}) \wedge (c_{\alpha,uid}, vk_{uid}) \in \mathcal{L}\}$$
  3. Send (Send, sid, (e, m, c, π_{zk})) to $\mathcal{F}_{ABC}^{\Delta}$.
- Upon receving ($\mathsf{Fetch\text{-}New}$,$sid$)
  1. Send ($\mathsf{Receive}$,sid) to $\mathcal{F}_{ABC}^{\Delta}$ and get $\vec{m}$.
  2. For each $(e, m, c, \pi_{zk}) \in \vec{m}$, do :
     (a) Check that $e \in \mathcal{E}$.
     (b) Send $(\mathsf{Verify}, sid, e, c)$ to $\mathcal{F}_{VRF}^{Com}$, and get the response $(\mathsf{Verified}, sid, e, c, b)$. Check that $b = 1$.
     (c) Check the validity of π_{zk}.
     (d) If all the above hold, add (e, m, c, π_{zk}) to $\vec{o}$
  3. Output ($\mathsf{Feach\text{-}New}$, $sid, \vec{o}$).

To implement VRF Evaluation of $\mathcal{F}_{VRF}^{Com}$, [13] proposed the anonymous VRF (AVRF) , which consists of ($\mathsf{AVRF.gen}$, Update, $\mathsf{AVRF.prov}$, $\mathsf{AVRF.vrfy}$), in order to hide the identity of the stakeholder. Comparing with VRF, the special property of AVRF is that the stakeholder updates his public key without changing the corresponding private key, and two evaluations on different messages under the same secret key cannot be linked to a public key, while other properties of VRF can still be preserved. More details of the construction of AVRF are shown in Appendix B.

In our setting, AVRF with key k takes as input the public key pk and the slot sl and outputs the randomness η and the proof π_{AVRF}. For convenience, let $F_k(sl)$ denote randomness output by AVRF with k and slot sl. That is, $F_k(sl) = \eta$. To ensure the validity of an election, it remains to prove that the corresponding AVRF key is in the list $\mathcal{L}$ in a zero-knowledge manner. The overall ZK proof π_{zk} for APoS-N is similar to that of [13] except that we need to consider the ZK poofs for the consistency of the noise function $\gamma(z)$. The implementation of π_{zk} in *Lottery Protocol*$^{\mathcal{E}, LE^*}$ follows the method of [13], with the modification that we need to consider the ZK proof about the noise.

Proof of evaluation of LE^* predicate π_{LE^*}. To implement $\mathcal{F}_{VRF}^{Com}$, each stakeholder needs a signature key pair (vk, sk) and a AVRF key pair (pk, k). The list $\mathcal{L}$ which consists of the tuples (c_{stk}, vk, pk) is recorded in the "genesis" block of each epoch, where c_{stk} denotes the commitment to the corresponding stake stk. If a stakeholder uid with stake stk wants to prove he won the election in slot sl, he needs to provide ZK proof π_{LE^*} for the following statement.

$$\{(\eta, k, stk) : LE^*(stk, \eta) = 1 \wedge F_k(sl) = \eta\},$$

where $LE^*(stk, \eta) = 1$ iff the $y < 2^{\ell_\alpha} \cdot (1-(1-f)^{\frac{stk \cdot (1+\gamma(z))}{STK}})$ and $\eta = y||z$. Next, the stakeholder proves the ownership of stake stk.

Proof of ownership π_{own}. We follow the idea of π_{own} in [13], which uses Merkle tree, denoted as $\mathcal{L}(root)$, to maintain the stakeholder list $\mathcal{L}$. The leaf of $\mathcal{L}(root)$ is of the form $(c_{stk}, vk, pk) \in \mathcal{L}$.

When a stakeholder proposes a new block, he needs to provides the membership proof of his "leaf". More precisely, he proves the knowledge of (vk, stk, pk) such that (c_{stk}, vk, pk) is a leaf of $\mathcal{L}(root)$ for the following statement. The resulting proof is denoted as π_{own}.

$$\{(vk, stk, k, pk, c_{stk}) : (c_{stk}, vk, pk) \in \mathcal{L} \wedge pk = g^k\},$$

where $pk = g^k$ follows the key generation algorithm of the AVRF.

Proof of signature on a new block under winning key π_{sig}. A block must be signed by the proposer to prevent from being tampered by adversaries. But the verification of the signature may reveal the identity of the signer. Hence, we need the ZK proof π_{sig} for the following statement, which shows that there exists a signature σ on a block M without revealing (vk, sk, σ).

$$\{(vk, sk, \sigma) : vk = \mathsf{SIG.keygen}(sk) \wedge \mathsf{SIG.vrfy}_{vk}(\sigma, M) = 1\}.$$

Overall proof of π_{zk}. The overall proof π_{zk} for *Lottery Protocol*$^{\mathcal{E}, LE^*}$ described below is similar to that of [13], where our modifications on the noise function do not affect the description of the main steps of π_{zk} in [13]. For more details of π_{zk}, we refer to [13].

Protocol for π_{zk}

Let $\mathcal{L}(root)$ denote the Merkle tree of $\mathcal{L} = \{(c_{stk_{uid}}, vk_{uid}, pk_{uid})\}$.

- For stakeholder uid with stake stk, his private information is (stk, c_{stk}, c_k, vk, sk, k), where c_k denotes the commitment to k. Generate the signature $\sigma = \mathsf{SIG.sign}(sk, M)$, where M is the part of the block that is signed. To generate a proof π_{zk} for proposing a new block:
  - Compute $pk' = \mathsf{Update}(pk)$ and $\mathsf{AVRF.prov}_k(pk', sl) = (\eta, \pi)$, where $\eta = y \,||\, z$. Compute the commitments $c_\sigma = Com(\sigma)$, $c_{vk} = Com(vk)$, $c_{sk} = Com(sk)$ and $c_\eta = Com(\eta)$. Then publish pk', c_σ, c_{vk} and c_{sk}. Let $Eq(pk', pk, k)$ denote the predicate which outputs 1 iff pk' and pk corresponds to the same secret key k.
  - Generate π_{LE^*} for the following statement:
    $$\{(\eta, \pi, stk, c_{stk}) : LE^*(stk, \eta) = 1 \wedge c_{stk} = Com(stk) \wedge c_\eta = Com(\eta) \wedge \mathsf{AVRF.vrfy}_{pk'}(sl, \eta, \pi) = 1\}$$
  - Generate π_{sig} for the following statement:
    $$\{(vk, sk, \sigma) : vk = \mathsf{SIG.keygen}(sk) \wedge c_{vk} = Com(vk) \wedge c_{sk} = Com(sk) \wedge c_\sigma = Com(\sigma) \wedge \mathsf{SIG.vrfy}_{vk}(\sigma, M) = 1\}$$

- Generate π_{own} for the following statement:
$$\{(vk, stk, k, pk, c_{stk}) : (c_{stk}, vk, pk) \in \mathcal{L}(\mathsf{root}) \\ \wedge c_{vk} = Com(vk) \wedge Eq(pk', pk, k) = 1 \wedge c_{stk} = Com(stk)\}$$

Set π_{zk} to be $(\pi_{LE^*}, \pi_{sig}, \pi_{own})$.

Privacy defined by $\mathcal{F}_{Lottery}^{\mathcal{E},LE}$. The only difference between APoS proposed by [13] and our APoS-N is that we replace LE with our LE^*. Although the ZK proofs π_{zk} for APoS-N need to consider the noise z, π_{zk} is a special case of the description of π_{zk} in [13]. That means the construction of our APoS-N including the related ZK proof still follows the framework of [13] and the security proof for the privacy of APoS defined in [13] can be applied to APoS-N. By Theorem 1 and Corollary 1 in [13], we have the following theorem.

Theorem 1. *Lottery Protocol$^{\mathcal{E},LE^*}$ realizes the $\mathcal{F}_{Lottery}^{\mathcal{E},LE^*}$ functionality in the ($\mathcal{F}_{\Delta}^{ABC}$, $\mathcal{F}_{Init}^{Com}$, $\mathcal{F}_{crs}$, $\mathcal{F}_{VRF}^{Com}$)-hybrid world in the presence of a PPT adversary. APoS-N with Lottery Protocol$^{\mathcal{E},LE^*}$ results in a private PoS protocol.*

5.2 (T, δ, ϵ)-privacy of APoS-N

Since the privacy defined by $\mathcal{F}_{Lottery}^{\mathcal{E},LE}$ does not rule out the possibility of the privacy leakage by frequency attack, we focus on the evaluation of (T, δ, ϵ)-privacy of APoS-N.

Notice that the frequency of changing $\gamma(z)$ will influence the effect of hiding stake. If $\gamma(z)$ is changed too frequently, e.g., $\gamma(z)$ takes as input fresh z in each slot, the interference effects of the noise will tend to be nullified in a short time. Because the expectation of the noise distribution is 0 and more noise samples make the sum of noise approximate to 0 much faster. Hence, we suggest that the same $\gamma(z)$ should be used for a period of time, say an epoch. In particular, we modify the first step of VRF Evaluation of $\mathcal{F}_{VRF}^{Com}$ as follows.

VRF Evaluation (Eval, sid, vid, m)

1. If $T(vid, m)$ is undefined, pick random η, r from $\{0,1\}^{\ell_{VRF}}$, where $\eta = y||z$ and $|y| = \ell_y$. If sid corresponds to the first slot of the corresponding epoch, the related randomness η is denoted as $y_1||z_1$. Otherwise, set $\eta = y||z_1$.

That is, the same randomness z will be used for the whole epoch and refreshed only at the beginning of each epoch. We stress that the above modification does not change the framework of APoS-N, where only minor modification on the concrete instantiations needs to be made. Let Π^* denote the resulting APoS-N. Hence, Theorem 1 still holds for Π^*.

To evaluate (T, δ, ϵ)-privacy of Π^* in practice, we consider the leader election process of a target stakeholder with stake p in an epoch, where the noise $\gamma(z)$ is fixed. Let $\Phi^*(p, \gamma(z))$ denote the probability of a stakeholder with noisy relative stake winning an election. So we have $\Pr[\mathsf{Exp}^{tag}_{\Pi^*,\delta}(1^\lambda) = 1] = \sum_{i=\lfloor(1-\delta)T\Phi(p)\rfloor}^{\lfloor(1+\delta)T\Phi(p)\rfloor} \Pr[X = i]$, where $X \sim B(T, \Phi^*(p, \gamma(z)))$. Suppose $\gamma(z)$ follows the uniform distribution over $[-\gamma_{max}, \gamma_{max}]$. We need to consider the expectation of $\Pr[\mathsf{Exp}^{tag}_{\Pi^*,\delta}(1^\lambda) = 1]$, which is

$$\int_{-\gamma_{max}}^{\gamma_{max}} \Pr[\mathsf{Exp}^{tag}_{\Pi^*,\delta}(1^\lambda) = 1 | \gamma(z) = x] \Pr[\gamma(z) = x] dx.$$

Consider the concrete parameter $p = 0.3\%$, $\gamma_{max} = 0.3$, $T = 432000$ (an epoch) and $\delta = 0.1$. Recall that $\Pr[\mathsf{Exp}^{tag}_{\Pi,\delta}(1^\lambda) = 1] = 60.95\%$ for the APoS protocol Π (without noise) [10,13]. For APoS-N protocol Π^*, the expectation of $\Pr[\mathsf{Exp}^{tag}_{\Pi^*,\delta}(1^\lambda) = 1]$ is as low as 34.01%, which is decreased by 44.2% comparing with that of APoS. That means, the APoS-N protocol Π^* is expected to achieve $(432000, 0.1, 34.01\%)$-anonymity for a stakeholder with relative stake 0.3% in an epoch.

Long term benefits. Although larger noise bound γ_{max} can lead to better (T, δ, ϵ)-privacy, one may concern about the total number of proposed blocks of stakeholders during some periods deviates from their expectations too much due to the large noise. So the stakeholders' benefits in APoS-N may not match their stakes for some periods, which violates the intuition of proof of stake. It is obvious that the long-term block benefits of the stakeholder in APoS-N is similar to that of APoS, since the expectation of the noise distribution is 0. The problem is how long the stakeholder should wait to get what he deserves. Intuitively, the larger the noise the longer the stakeholder should wait. In Figure 3, we simulate the block generation of a stakeholder with relative stake 0.3% over 60 days (12 epochs) in APoS and APoS-N, respectively, where $\gamma_{max} = 0.3$. The red curve and the green curve represent the deviation of the total number of blocks from the expectation, i.e., $\frac{X}{T \cdot \Phi(p)} - 1$, for APoS and APoS-N, respectively. As shown in Figure 3, the difference of the deviations between APoS and APoSN is large during the first 20 days, while it decreases to about 1% after the first $44 \sim 47$ days. That means, the time of the stakeholder with relative stake 0.3% to match his expectation is about $44 \sim 47$ days.

Restriction on individual's maximum relative stake. To prevent the adversary from getting too much undeserved benefits in APoS-N for some period of time, e.g., winning an election with probability $\Phi(p \cdot (1 + \gamma_{max}))$ for an epoch. We restrict the maximum relative stake of each stakeholder. That is, if a stakeholder's relative stake is larger than the maximum value, he should split his stake among multiple virtual parties, where each virtual party's stake p_i is less than the maximum value. Due to the randomness of the each virtual party's noise, it is hard for all the virtual parties to reach $\Phi(p_i \cdot (1 + \gamma_{max}))$ simultaneously. In fact, such strategy is consistent with the saturation mechanism [25] in Cardano.

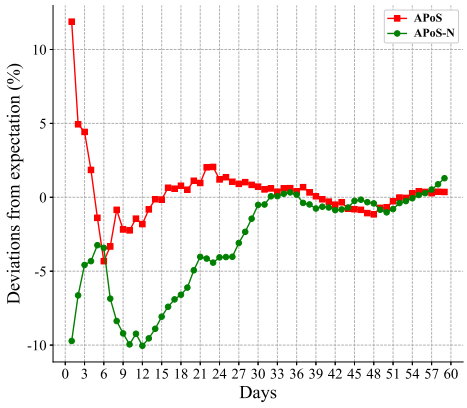

Fig. 3. Deviations from expectation in APoS and APoS-N during 12 epochs.

The saturation mechanism is designed to prevent centralization by diminishing the stake pool's rewards if it reaches the saturation threshold, which is about 0.27%. Hence, no pool in Cardano has more than 0.4% relative stake.

We emphasize that the restriction on the maximum stake for each stakeholder is crucial for not only the stabilization of the benefits but also the security threshold of the adversarial stakes, which will be explained next.

Attack against Wang et al.'s protocol [19] The work of [19] also proposes private PoS protocol using differential privacy technique, where the stake is distorted by adding noise. The noise in their stake distortion mechanisms follows the "same" Laplace distribution. As mentioned in [19], the noisy stake can be negative and a party with a negative stake is treated as having no stake. A direct attack on this mechanism involves an adversary dividing their stakes among multiple corrupted participants so that each participant has very small stakes. When applying their stake distortion mechanisms, some of the corrupted participants' noisy stakes are zero, while others may become larger. However, the expected total noisy stakes of the adversary are larger than his original stake due to the neglect of the negative stake. This gives the adversary a higher payoff, which violates the chain quality and even the safety of the resulting protocol and contradicts Theorem 17 in [19]. In our work, noise follows the uniform distribution and the amplitude of the noise is related to the stake. This allows for careful control of the noisy stake, preserving the fundamental security requirements of the underlying PoS protocol [8], such as chain quality.

6 Common Prefix, Chain Growth and Chain Quality

Since the unpredictable noise changes the relation between stakes and the corresponding probability of proposing blocks in a short period of time, it is at the

risk of breaking the basic security properties of PoS, i.e., common prefix, chain growth and chain quality. Recall that typical PoS protocols [7] are proven secure under the condition that the adversarial stakeholders' relative stakes should be less than a threshold, say 1/2. In some slots of our APoS-N, the adversarial stakeholders' noisy relative stakes may be larger than the threshold. Besides, the total noisy stakes may be also larger or less than the original total stakes STK. That means, the security proof of previous works cannot be applied in our setting. In this section, we will analyze the impact of noise on the security proof of PoS and prove the basic security properties of APoS-N.

6.1 Security Proof of PoS

First, we briefly recall the security proof of PoS [7,8]. The main idea of [7,8] is based on *characteristic string*, which provides a novel method to capture the possible states of slots in a Δ-semi-synchronous environment. Specifically, there are three types of states for each slot i, which are denoted by $\omega_i \in \{\perp, 0, 1\}$. The state of an empty slot i (where no stakeholder wins the election) is represented by $\omega_i = \perp$. For a non-empty slot i, if only one honest stakeholder is elected as leader, the state of the slot is denoted as $\omega_i = 0$, otherwise it is denoted as $\omega_i = 1$. Hence, the execution of a PoS system from slot 1 to slot n corresponds to a characteristic string $\omega = \omega_1\omega_2 \cdots \omega_n$.

Notice that the same characteristic string may imply different kinds of global states of the chain which can be captured by a directed graph. E.g., in a slot

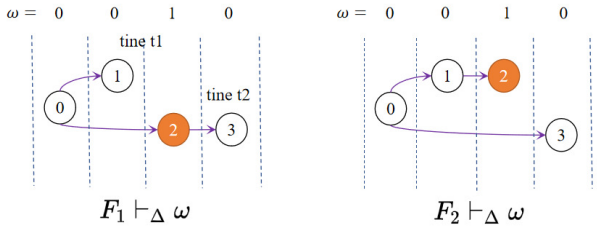

Fig. 4. Fork F_1 and F_2 correspond to the same string ω=0010

with $\omega_i = 1$, the corresponding block may have different parent block as shown in Figure 4, which is determined by the adversary's control over propagation delay Δ. Such a directed graph is called a fork and denoted as $F \vdash_\Delta \omega$. So a characteristic string ω can imply different forks $F \vdash_\Delta \omega$, which corresponds to different global blockchains in Δ-semisynchronous environment. In fact, characteristic string can be interpreted as an effective way to classify the possible states of the global chains.

A path in a fork originating at the root is called a tine, which represents a chain in some stakeholders' view at some slot. A tine t is called a viable tine if its last block (vertex) is honest, which means t is adopted by some honest

stakeholder at some time. Let $len(t)$ denote the length of t and $\ell(t)$ denote the slot number of the last vertex of t. Let $t_1 \cap t_2$ denote the common prefix of t_1 and t_2.

Another important notion of characteristic string is divergence $div_\Delta(\omega)$, which is defined as the maximum value of $len(t_1) - len(t_1 \cap t_2)$ for any viable tine t_1 and t_2 in any fork $F \vdash_\Delta \omega$, where $len(t_1) < len(t_2)$. That is, the divergence $div_\Delta(\omega)$ shows the maximum length of orphaned branches.

[8] provides an efficient method, called reduction mapping, to reduce the analysis of characteristic string in Δ-semi-synchronous environment to that of synchronous setting, which can eliminate the impact of delay Δ in the proof. Specifically, a reduction mapping ρ_Δ can map a Δ-semi-synchronous string ω to a synchronous string ω' while preserving the structure of forks.

Definition 2. *(Reduction mapping) [8] For $\Delta \in \mathbb{N}$, the function $\rho_\Delta : \{0,1,\perp\}^* \to \{0,1\}^*$ is defined inductively as follows: $\rho_\Delta(\epsilon) = \epsilon$, $\rho_\Delta(\perp || \omega) = \rho_\Delta(\omega)$, $\rho_\Delta(1||\omega) = 1||\rho_\Delta(\omega)$,*

$$\rho_\Delta(0||\omega) = \begin{cases} 0 \,||\, \rho_\Delta(\omega) & if \;\; \omega \in \perp^{\Delta-1} \,||\, \{0,1,\perp\}^*, \\ 1 \,||\, \rho_\Delta(\omega) & otherwise. \end{cases}$$

Through the mapping function $\rho_\Delta(\cdot)$, an honest slot in ω that has waited long enough to complete synchronization is still an honest slot ($\omega'_i = 0$), while all other non-empty slots are classified as $\omega'_i = 1$. The core idea of the mapping is that if no other leader appears in $\Delta - 1$ slots after a unique honest-leader slot, all honest stakeholders will accept the unique honest leader. This event is called a Δ-right-isolated. Since only the slot of Δ-right-isolated is 0 after the mapping, the divergence of ω' is an upper bound on the divergence of ω.

By characteristic string, the basic security properties of PoS blockchain can be interpreted as follows [7].

- **Common Prefix (CP) with parameter** $m \in \mathbb{N}$. A characteristic string ω possesses m-CP if, for every fork $F \vdash_\Delta \omega$ and every pair of viable tines t_1 and t_2 of F for which $\ell(t_1) \leq \ell(t_2)$, the tine $t_1^{\neg m}$ is a prefix of t_2. Equivalently, $len(t_1) - len(t_1 \cap t_2) \leq m$.
- **Chain Quality (CQ) with parameter** $s \in \mathbb{N}$. A characteristic string ω possesses s-CQ if, for every fork $F \vdash_\Delta \omega$ and every viable tine t of F, any portion of t spanning s slots contains at least one honest vertex.
- **Chain Growth (CG) with parameters** $\tau \in (0,1]$ **and** $s \in \mathbb{N}$. A characteristic string ω possesses (τ, s)-CG if, for every fork $F \vdash_\Delta \omega$ and every viable tine t of F, any portion of t spanning s slots contains at least τs vertices.

Common prefix means that in any fork defined on ω, any tine that drops last m blocks is the prefix of the longest tine. Due to the definition of divergence, the common prefix property of ω can be obtained by proving that the probability of $\Pr[div_0(\rho_\Delta(\omega)) \geq m]$ is negligible. On the other hand, when the Δ-right-isolated

event happens, there is an honest block whose depth is greater than other blocks. Therefore, as long as the frequency of Δ-right-isolated event, denoted by $Right_{\Delta}$, is large enough, the blockchain increases at least the number of slots in which Δ-right-isolated events occurs, which implies chain growth. Similarly, as long as enough Δ-right-isolated events happen, the probability of generating a substring completely controlled by the adversary at any given time is negligible, which implies chain quality. Thus the basic security properties of PoS can be proven by showing the above properties of the characteristic string.

6.2 Basic Security Properties of APoS-N

Next, we analyze the distribution of string ω in APoS-N. Let α denote the fraction of the stake controlled by honest stakeholders in APoS-N, and $\beta = 1-\alpha$ denote the adversary's stake fraction. Recall that we make restrictions on the individual's maximum stake. (There are thousands of stake pools in Cardano, none of which has more than 0.4% relative stake.) Without loss of generality, we assume the total number of stakeholders is n and each stakeholder has the same relative stake. We have the following two observation:

Observation 1: The total noisy honest relative stake in any slot is almost α. Due to the noise γ_i, a stakeholder U_i's noisy relative stake is $\frac{1}{n} \cdot (1+\gamma_i)$, where $\gamma_i \sim U(-\gamma_{max}, \gamma_{max})$. We denote the total honest relative stake with noise in any slot as $\hat{\alpha} := \sum_{i=1}^{\alpha n}(\frac{1}{n} \cdot (1+\gamma_i))$, where U_i is honest. Then we get the expectation of $\hat{\alpha}$, which is $\mathbb{E}(\hat{\alpha}) = \sum_{i=1}^{\alpha n}(\frac{1}{n} \cdot (1+\mathbb{E}(\gamma_i))) = \alpha$. According to the Chernoff-Hoeffding Bound in Appendix A, we can obtain the approximate range of $\hat{\alpha}$:

$$\Pr[\hat{\alpha} < (1-\theta_1)\alpha] \leq exp(\frac{-\theta_1^2 \cdot \alpha n}{4\gamma_{max}^2}) \tag{2}$$

Observation 2: The probability that all the adversarial relative stakes reach $\frac{1}{n}(1+\gamma_{max})$ simultaneously decreases exponentially with the number of pools. Saturation mechanism ensures that all the adversary's stakes would not be concentrated in one account, otherwise the probability of adversary becoming leader will decrease. Similar to Observation 1, we define the total adversary's relative stake as $\hat{\beta} := \sum_{i=1}^{\beta n}(\frac{1}{n} \cdot (1+\gamma_i))$, where U_i is adversarial, and get $\mathbb{E}(\hat{\beta}) = \beta$. By Chernoff-Hoeffding Bound, we have

$$\Pr[\hat{\beta} > (1+\theta_2)\beta] \leq exp(\frac{-\theta_2^2 \cdot \beta n}{4\gamma_{max}^2}) \tag{3}$$

Based on the above observations, we use θ-**Noise** to denote the event that the noisy honest stake $\hat{\alpha}$ is greater than $(1-\theta)\alpha$ and noisy adversarial stake $\hat{\beta}$ is less than $(1+\theta)\beta$, where $\theta > 0$. Due to equations (2)(3), we can get:

Lemma 1. $\Pr[\theta\text{-}\boldsymbol{Noise}] > 1 - exp(-\Omega(\theta^2 n/\gamma_{max}^2))$.

Let $\mathcal{D}_{\mathcal{A}}^{f,\theta}$ denote the distribution of characteristic strings ω when θ-**Noise** happens. Note that the distribution is influenced by $\mathcal{A}$. We have

$$\begin{aligned} p_\perp = \Pr[\omega_i = \perp] &= \prod_{j\in\mathcal{P}}(1-\phi(\frac{1}{n}\cdot(1+\gamma_j))) > (1-f)^{1-\theta\alpha+\theta\beta} > (1-f)^{1+\theta\beta}, \\ p_0 = \Pr[\omega_i = 0] &= \sum_{h\in\mathcal{H}}(\phi(\frac{1}{n}\cdot(1+\gamma_h))\cdot(1-f)^{\sum_{j\neq h}(\alpha_j(1+\gamma_j)))} \\ &> \phi((1-\theta)\alpha)\cdot(1-f)^{1+\theta\beta}, \\ p_1 = \Pr[\omega_i = 1] &= 1-p_\perp-p_0, \end{aligned}$$

where $\mathcal{P}$ denotes the set of all stakeholders and $\mathcal{H}$ denotes the set of all honest stakeholders. To simplify the analysis of $\mathcal{D}_{\mathcal{A}}^{f,\theta}$, we modify the notion of dominant distribution [8] so that it can capture the worst case in our noise setting.

Definition 3 (Dominant distribution $\mathcal{D}_{\alpha}^{f,\theta}$). *For parameters f, α and θ, define $\mathcal{D}_{\alpha}^{f,\theta}$ to be the distribution on strings $\omega \in \{0,1\,\perp\}^{\mathcal{R}}$ so that $p_\perp = \Pr[\omega_i = \perp] = (1-f)^{1+\theta\beta}$, $p_0 = \Pr[\omega_i = 0] = \phi(\alpha\cdot(1-\theta))\cdot(1-f)^{1+\theta\beta}$, and $\Pr[\omega_i = 1] = 1-p_\perp-p_0$.*

$\mathcal{D}_{\alpha}^{f,\theta}$ dominates the string distribution $\mathcal{D}_{\mathcal{A}}^{f,\theta}$ implies that if the security property can be preserved under distribution $\mathcal{D}_{\alpha}^{f,\theta}$, it will also be preserved under distribution $\mathcal{D}_{\mathcal{A}}^{f,\theta}$. Loosely speaking, for slot i, if $\omega_i = 1$, the adversary can take advantage of this slot to break the security, e.g., splitting the chain into two branches. So the number of 1s of a characteristic string ω reflects the adversary's ability to control the blockchain. In particular, characteristic strings that follows distribution $\mathcal{D}_{\alpha}^{f,\theta}$ contain the maximum number of 1s, which can be interpreted as the worst case of $\mathcal{D}_{\mathcal{A}}^{f,\theta}$.

Let $\bar{f} = 1-(1-f)^{1+\theta\beta}$ and $\bar{\alpha} = \alpha\cdot(1-\theta)$. So $\mathcal{D}_{\alpha}^{f,\theta}$ can be written in the form of $\mathcal{D}_{\bar{\alpha}}^{\bar{f}}$, which can be interpreted as a distribution for the case without noise. Then we use $\mathcal{D}_{\bar{\alpha}}^{\bar{f}}$ to analyze the upper bound of the probability of the characteristic string violating the security under distribution $\mathcal{D}_{\alpha}^{f,\theta}$.

Following the idea of [8], the security properties of APoS-N can be proven by analyzing the probability that a monotone event E occurs over the distribution $\mathcal{D}_{\bar{\alpha}}^{\bar{f}}$ and $\mathcal{D}_{\alpha}^{f,\theta}$. The definition of monotone event is as follows. For the set $\{0,1,\perp\}^{\mathcal{R}}$, $x_1\cdots x_{\mathcal{R}} \preceq y_1\cdots y_{\mathcal{R}}$ iff $x_i = y_i$ or $y_i = 1$ for each i. A subset $E \subseteq \{0,1,\perp\}^{\mathcal{R}}$ is monotone if $x \in E$ and $x \preceq y$ implies that $y \in E$.

For two random variables $Y = (Y_1,\ldots,Y_{\mathcal{R}})$ and $Z = (Z_1,\ldots,Z_{\mathcal{R}})$ over $\{0,1,\perp\}^{\mathcal{R}}$, when $\Pr[Y_i = 1] \leq \Pr[Z_i = 1]$, we can conclude that $Y \preceq Z$ by Lemma 4.18 of [7]. $Y \preceq Z$ implies that Y is dominated by Z and $\Pr[Y \in E] \leq \Pr[Z \in E]$. Let $Y = \mathcal{D}_{\bar{\alpha}}^{\bar{f}}$ and $Z = \mathcal{D}_{\alpha}^{f,\theta}$. By $Pr[Y_i = 1] \leq Pr[Z_i = 1]$, we get $\mathcal{D}_{\alpha}^{f,\theta} \preceq \mathcal{D}_{\bar{\alpha}}^{\bar{f}}$, where the corresponding adversary $\mathcal{A}$ is called $\bar{\alpha}$-dominated. Lemma 2 described below shows the probability of Δ-right-isolated event $Right_\Delta$ of the distribution $\mathcal{D}_{\bar{\alpha}}^{\bar{f}}$ after reduction mapping. By the properties of the reduction, Lemma 2 captures the lower bound on the security of the distribution $\mathcal{D}_{\bar{\alpha}}^{\bar{f}}$.

Lemma 2 (Structure of the Induced Distribution $\mathcal{D}_{\bar{\alpha}}^{\bar{f}}$). *Let $x_1 \cdots x_\ell = \rho_\Delta(\omega)$, where $\omega \in \{0,1,\perp\}^{\mathcal{R}}$ follows $\mathcal{D}_{\bar{\alpha}}^{\bar{f}}$ and $\bar{f} = 1-(1-f)^{1+\theta\beta}$, $\bar{\alpha} = \alpha\cdot(1-\theta)$. There exists a sequence of independent random variables $q_1, q_2, \cdots$ with each $q_i \in \{0,1\}$ so that*

$$\Pr[q_i = 0] = (\frac{p_0}{p_0+p_1})p_{\perp,1}p_{\perp,2}\cdots p_{\perp,\Delta-1} \geq \bar{\alpha}\cdot f\cdot(1-f)^{(1+\theta\beta)\Delta}/\bar{f}$$

and $x_1\cdots x_{\ell-\Delta} = \rho_\Delta(\omega_1\ldots\omega_{\mathcal{R}})^{\neg\Delta}$ is a prefix of $q_1q_2\ldots$, where $p_{\perp,i}$ denotes the probability that $\omega_i = \perp$.

Proof. By the definition of Δ-right-isolated event, we have

$$\Pr[q_i = 0] = (\tfrac{p_0}{p_0+p_1})p_{\perp,1}p_{\perp,2}\cdots p_{\perp,\Delta-1}.$$

Due to $p_\perp \geq (1-f)^{1+\theta\beta}$ and $p_0 \geq \phi(\bar{\alpha})\cdot(1-f)^{1+\theta\beta}$, we can get

$$\Pr[q_i = 0] = (\tfrac{p_0}{p_0+p_1})p_{\perp,1}p_{\perp,2}\cdots p_{\perp,\Delta-1} \geq \bar{\alpha}\cdot f\cdot(1-f)^{(1+\theta\beta)\Delta}/\bar{f}.$$

When taking the minimum value $p_0 = \bar{\alpha}\cdot f\cdot(1-f)^{1+\theta\beta}$ and $p_\perp = (1-f)^{1+\theta\beta}$, q_i follows the binomial distribution with parameter $\approx \bar{\alpha}\cdot f\cdot(1-f)^{(1+\theta\beta)\Delta}/\bar{f}$.

6.3 Common Prefix, Chain Growth and Chain Quality

Based on **θ-Noise** and $\mathcal{D}_{\bar{\alpha}}^{\bar{f}}$, we prove common prefix, chain growth and chain quality of APoS-N. On the proof of common prefix, we need to consider $div_\Delta(\omega)$. Because the divergence represents the maximum length of orphaned branch in the fork F, which can be implied by ω. Let $D_\Delta = \{x | div_\Delta(x) \geq m\}$. Note that if a string x satisfies D_Δ, then the string obtained by changing some "0" of x to "1" will also satisfy D_Δ. So D_Δ is monotone. The following Lemma proven by [7] shows that the probability of a synchronized characteristic string with large divergence in a binomial distribution decreases exponentially.

Lemma 3. *[7] Let $\ell, m \in \mathbb{N}$ and $\epsilon \in (0,1)$. Let $\omega \in \{0,1\}^\ell$ be drawn according to the binomial distribution so that $\Pr[\omega_i = 1] = (1-\chi)/2$. Then $Pr[div_0(\omega) \geq m] \leq exp(\ln\ell - \Omega(m))$, where $\Omega(\cdot)$ depends on χ.*

We extend Lemma 3 to the setting of noisy stake and get Lemma 4 below.

Lemma 4. *Given some $f \in (0,1]$ and $\Delta \geq 1$, let $\bar{f} = 1-(1-f)^{(1+\theta\beta)}$ and $\bar{\alpha} = \alpha\cdot(1-\theta)$ such that $\bar{\alpha}\cdot f\cdot(1-f)^{(1+\theta\beta)\Delta}/\bar{f} = (1+\chi)/2$ for some $\chi > 0$. For any $\omega \in \{0,1,\perp\}^{\mathcal{R}}$ which follows $\mathcal{D}_{\bar{\alpha}}^{\bar{f}}$, we have $\Pr[div_\Delta(\omega) \geq m+\Delta] = exp(\ln\mathcal{R} - \Omega(m))$, where $\Omega(\cdot)$ depends on χ and θ.*

Proof (sketch). Since $div_\Delta(\cdot)$ and $div_0(\cdot)$ are monotone, we have

$$div_\Delta(\omega) \leq div_0(\rho_\Delta(\omega)) \leq div_0(\rho_\Delta(\omega)^{\lceil\Delta} + \Delta \leq div_0(q_1\cdots q_{\mathcal{R}}) + \Delta.$$

For q_i which follows binomial distribution with $\Pr[q_i = 0] \geq \bar{\alpha}\cdot f\cdot(1-f)^{(1+\theta\beta)\Delta}/\bar{f}$, we conclude that $\Pr[div_\Delta(\omega) \geq m] \leq \Pr[div_0(\rho_\Delta(\omega)^{\lceil\Delta}) \geq m-\Delta] = exp(\ln\mathcal{R} - \Omega(m))$ by Lemma 3 and the assumption of Lemma 4.

Theorem 2 (Common Prefix). *Let $m, \mathcal{R}, \Delta \in \mathbb{N}$ and $\chi \in (0,1)$. Let $\mathcal{A}$ be an $\bar{\alpha}$-dominated adversary against APoS-N for some $\bar{\alpha}$ satisfying $\bar{\alpha} \cdot f \cdot (1-f)^{(1+\theta\beta)\Delta}/\bar{f} \geq (1+\chi)/2$ where $\bar{f} = 1-(1-f)^{1+\theta\beta}$ and $\bar{\alpha} = \alpha \cdot (1-\theta)$. When APoS-N executed in a Δ-semisynchronous environment with noise upper bound γ_{max}, the probability that APoS-N preserves the Common Prefix property with parameter m throughout a period of $\mathcal{R}$ slots is greater than $1-exp(\ln \mathcal{R} - \Omega(m-\Delta)) - exp(-\Omega(\theta^2 n/\gamma_{max}^2))$.*

Proof. If and only if $F \vdash_\Delta \omega$ induced by this execution satisfies $div_\Delta(F) \geq m$, the execution of APoS-N will violate the Common Prefix property with parameter m. Since the divergence of ω is greater than that of $F \vdash_\Delta \omega$, When event θ-**Noise** happens, we have

$$Pr[div_\Delta(F) \geq m] \leq Pr_{\mathcal{D}_{\bar{\alpha}}^{\bar{f}}}[div_\Delta(\omega) \geq m] \leq exp(\ln \mathcal{R} - \Omega(m-\Delta)).$$

where the first inequality follows the definition of $div_\Delta(\cdot)$ and definition 3, and the second one always holds due to Lemma 4.

By Lemma 1, the probability of θ-**Noise** is $1 - exp(-\Omega(\theta^2 n/\gamma_{max}^2))$. Therefore, the probability that the APoS-N follows the Common Prefix is greater than $(1 - exp(\ln \mathcal{R} - \Omega(m-\Delta))) \cdot 1 - exp(-\Omega(\theta^2 n/\gamma_{max}^2)) \geq 1 - exp(\ln \mathcal{R} - \Omega(m-\Delta)) - exp(-\Omega(\theta^2 n/\gamma_{max}^2))$. This completes the proof of Common Prefix.

Theorem 3 (Chain Growth). *Let $m, \mathcal{R}, \Delta \in \mathbb{N}$ and $\chi \in (0,1)$. Let $\mathcal{A}$ be an $\bar{\alpha}$-dominated adversary against APoS-N for some $\bar{\alpha} > 0$. When APoS-N executed in a Δ-semisynchronous environment with noise upper bound γ_{max}, the probability that APoS-N preserves the chain growth property with parameter $s \geq 4\Delta$ and $\tau = c\bar{\alpha}/4$ throughout a period of $\mathcal{R}$ slots is greater than $1-exp(-c\bar{\alpha}(s-3\Delta)/(20\Delta)) + ln\mathcal{R}\Delta) - exp(-\Omega(\theta^2 n/\gamma_{max}^2))$, where c denotes the constant $f(1-\bar{f})^\Delta$ and $\bar{f} = 1-(1-f)^{1+\theta\beta}$.*

Proof. Suppose that the longest chain possessed by an honest party at slot sl_1 is $\mathcal{C}_1$, and the longest chain possessed by an honest party at slot $sl_2 \geq sl_1 + s$ is $\mathcal{C}_2$. We note that Δ-right-isolated event $Right_\Delta$ implies that the length of the main chain increases by 1. Due to the definition of Chain Growth, there should be enough Δ-right-isolated during s slots, which can be proven by Chernoff bounds.

When event θ-**Noise** happens, let $\hat{sl}_1, ..., \hat{sl}_h$ be the increasing sequence of all Δ-right-isolated honest slots among the slots in $T := \{sl_1 + \Delta, ..., sl_2 - \Delta\}$. Since $\hat{sl}_1 \geq sl_1 + \Delta$, the leader of $\hat{sl}_1$ will append at least 1 block to a chain $\mathcal{C}$ where $len(\mathcal{C}) \geq len(\mathcal{C}_1)$. By the definition of Δ-right-isolated event and the longest chain rule, the $\mathcal{C}_1$ will be received by all the users. Analogously, the leader of every $\hat{sl}_i$ will append at least 1 block. So we get that $len(\mathcal{C}_2) \geq len(\mathcal{C}_1) + h$.

Let $H_T(x)$ denote the number of Δ-right-isolated uniquely honest slots among the slots in T for $x \in \{0,1\perp\}^{\mathcal{R}}$. Let $E = \left\{x \in \{0,1,\perp\}^{\mathcal{R}} \,|H_T(x) < c\bar{\alpha}s/4\right\}$ where $c = f(1-\bar{f})^\Delta$. So E is monotone and $\mathcal{D}_{\mathcal{A}}^{f,\theta} \preceq \mathcal{D}_{\bar{\alpha}}^{\bar{f}}$ implies

$$\Pr_{x \leftarrow \mathcal{D}_{\mathcal{A}}^{f,\theta}}[H_T(x) < c\alpha s/4] \leq \Pr_{x \leftarrow \mathcal{D}_{\bar{\alpha}}^{\bar{f}}}[H_T(x) < c\alpha s/4].$$

Consider the characteristic string x sampled according to $\mathcal{D}_{\bar{\alpha}}^{\bar{f}}$. For each $t \in T$, let X_t be the indicator random variable for the event that $\hat{sl}_t$ is Δ-right-isolated uniquely honest. In the distribution $\mathcal{D}_{\bar{\alpha}}^{\bar{f}}$, $\mu = \mathbb{E}[X_t] = p_0 p_\perp^{\Delta-1} \geq \bar{\alpha} f (1-\bar{f})^{\Delta}$ where $\bar{\alpha} = \alpha \cdot (1-\theta)$ and $1-\bar{f} = (1-f)^{1+\theta\beta}$. If $|t-t'| \geq \Delta$, the random variables X_t and $X_{t'}$ are independent. Let $T_z = \{t \in T | t \equiv z(\mod \Delta)\}$. Then the family of variables X_t indexed by T_z are independent. $T = \bigcup_{i=0}^{\Delta-1} T_i$ and for each i, $|T_z| > \lfloor (s-2\Delta)/\Delta \rfloor \geq (s-3\Delta)/\Delta$. By the Chernoff bound with $\delta = 1/2$, we have

$$\Pr[\textstyle\sum_{t\in T_z} X_t < \mu|T_z|/2] \leq e^{-\mu|T_z|/20} \leq e^{-\frac{\mu(s-3\Delta)}{20\Delta}}.$$

Notice that $H_T(x) = \sum_{t\in T} X_t$ and $\Delta \cdot \mu|T_z|/2 \geq \mu(s-2\Delta)/2$. We have

$$\Pr_{x\leftarrow\mathcal{D}_{\bar{\alpha}}^{\bar{f}}}[H_T(x) < \mu(s-2\Delta)/2] \leq \Delta \cdot Pr[\textstyle\sum_{t\in T_z} X_t < \mu|T_z|/2] \leq \Delta \cdot e^{-\frac{\mu(s-3\Delta)}{20\Delta}}.$$

When $\mu \geq \bar{\alpha} f (1-\bar{f})^{\Delta}$, we obtain

$$\Pr_{x\leftarrow\mathcal{D}_{\bar{\alpha}}^{\bar{f}}}[H_T(x) < c\alpha(s-2\Delta)/2] \leq \Pr_{x\leftarrow\mathcal{D}_{\bar{\alpha}}^{\bar{f}}}[H_T(x) < \mu(s-2\Delta)/2].$$

As $s \geq 4\Delta$, we have that $s - 2\Delta \geq s/2$. So

$$\Pr_{x\leftarrow\mathcal{D}_{\bar{\alpha}}^{\bar{f}}}[H_T(x) < cs\bar{\alpha}/4] = \Pr_{x\leftarrow\mathcal{D}_{\bar{\alpha}}^{\bar{f}}}[H_T(x) < \tau s] \leq e^{-\frac{\mu(s-3\Delta)}{20\Delta}}.$$

By the union bound over $\mathcal{R}$ slots, the probability of the event that violates Chain Growth with $s = 4\Delta$ and $\tau = c\bar{\alpha}/4$ is no more than

$$\mathcal{R} \cdot exp(-c\bar{\alpha}(s-3\Delta)/(20\Delta)) = exp(-c\bar{\alpha}(s-3\Delta)/(20\Delta) + \ln \mathcal{R}\Delta).$$

By Lemma 1, the probability that APoS-N preserves Chain Growth is greater than $(1 - exp(-c\bar{\alpha}(s-3\Delta)/(20\Delta) + ln\mathcal{R}\Delta)) \cdot (1 - exp(-\Omega(\theta^2 n/\gamma_{max}^2))) \geq 1 - exp(-c\bar{\alpha}(s-3\Delta)/(20\Delta) + ln\mathcal{R}\Delta) - exp(-\Omega(\theta^2 n/\gamma_{max}^2))$.

Theorem 4 (Chain Quality). *Let $m, \mathcal{R}, \Delta \in \mathbb{N}$ and $\chi \in (0,1)$. Let $\mathcal{A}$ be an $\bar{\alpha}$-dominated adversary against APoS-N for some $\bar{\alpha} > 0$ satisfying $\bar{\alpha} \cdot f \cdot (1-f)^{(1+\theta\beta)\Delta}/\bar{f} \geq (1+\chi)/2$ where $\bar{f} = 1-(1-f)^{1+\theta\beta}$. When APoS-N executed in a Δ-semisynchronous environment with noise upper bound γ_{max}, the probability that APoS-N preserves the Chain Quality property with parameters m and $\mu = 1/m$ throughout a period of $\mathcal{R}$ slots is greater than $1 - exp(\ln \mathcal{R} - \Omega(m)) - exp(-\Omega(\theta^2 n/\gamma_{max}^2))$.*

Proof. Proof of Theorem 4 follows that of Theorem 7 in [8] with modification that we need to consider the constraints of α with noise. The core of the proof is that the blocks generated by honest parties are more than the blocks generated by the adversary with sufficient period of time. Note that if the adversary is able to break Chain Quality, this implies the event of constructing a new subchain of length greater than m, where each block in the subchain does not correspond to a $Right_\Delta$ event, happens with negligible probability by the assumption of Theorem 4 and Chernoff bound. Recall the definition of Chain Quality, we call

a slot good if it is Δ-right-isolated uniquely honest, otherwise it is bad if not empty. A block is good (resp.bad) if it comes from a good (resp.bad) slot.

When event θ-**Noise** happens, let $B_1, ...B_m$ be a sequence of consecutive blocks in a chain C_1. Assume that all blocks $B_1, ..., B_m$ are bad (controlled by $\mathcal{A}$). Let G_1 denote the latest good block before B_1 in C_1, and G_2 denote the first good block after B_m in C_1 (if there is no good one, we take the last one in C_1). In this case, all blocks are bad between G_1 and G_2.

Let $\hat{sl}_1$ (resp.$\hat{sl}_2$) denote the good slot in which G_1 (resp.G_2) was created and let T denote the sequence of consecutive slots between $\hat{sl}_1$ and $\hat{sl}_2$, excluding $\hat{sl}_1$, but including $\hat{sl}_2$. By the proof of Theorem 3, in each Δ-right-isolated slot in T, the (unique) honest leader creates a block that has depth increased by at least 1. We use $d(G)$ represents the depth of the block G in the blockchain. Therefore, by the definition of Δ-fork, we conclude that $d(G_2) \geq d(G_1) + g$, where g is the number of good slots in T. However, we also have that $d(G_2) \leq d(G_1) + b$, where b is the number of bad slots in T. Both conditions are satisfied only if $g \leq b$, which is unlikely to happen.

Construct $E = \left\{ x \in \{0, 1, \perp\}^{\mathcal{R}} \,|g(x) \leq b(x) \right\}$, where $g(x)$ (resp.$b(x)$) denotes the number of good (resp.bad) slots on T in the string x. Obviously, E is monotone and $\mathcal{D}_{\mathcal{A}}^{f,\theta} \preceq \mathcal{D}_{\bar{\alpha}}^{\bar{f}}$ implies

$$\Pr_{x \leftarrow \mathcal{D}_{\mathcal{A}}^{f,\theta}}[g(x) \leq b(x)] \leq \Pr_{x \leftarrow \mathcal{D}_{\bar{\alpha}}^{\bar{f}}}[g(x) \leq b(x)].$$

Suppose $|T| = O(m)$. So $g(x) - b(x) \leq 0$ implies that $b(x) \geq O(m)/2$. However, by the assumption that $\bar{\alpha} \cdot f \cdot (1-f)^{(1+\theta\beta)\Delta}/\bar{f} \geq (1+\chi)/2$, it implies that good slots happen with higher probability than bad slots. Therefore,

$$\Pr_{x \leftarrow \mathcal{D}_{\bar{\alpha}}^{\bar{f}}}[g(x) \leq b(x)] \leq exp(-\Omega(m)).$$

Applying union bound over R slots, the event which violates chain quality with parameters m and $u = 1/m$ happens with probability no more than

$$\mathcal{R} \cdot exp(-\Omega(m)) = exp(\ln \mathcal{R} - \Omega(m)).$$

By Lemma 1, the probability that APoS-N follows the Chain Quality is greater than $(1 - exp(\ln \mathcal{R} - \Omega(m))) \cdot (1 - exp(-\Omega(\theta^2 n/\gamma_{max}^2))) \geq 1 - exp(\ln \mathcal{R} - \Omega(m)) - exp(-\Omega(\theta^2 n/\gamma_{max}^2))$.

References

1. Nakamoto, S.: Cryptocurrencies without proof of work (2008)
2. King, S., Nadal., S.: Ppcoin: Peer-to-peer crypto-currency with proof-of-stake
3. Bentov, I., Gabizon, A., Mizrahi, A.: Cryptocurrencies without proof of work. In: FC 2016 International. LNCS, vol. 9604, pp. 142–157. Springer (2016)
4. Bentov, I., Lee, C., Mizrahi, A., Rosenfeld, M.: Proof of activity: Extending bitcoin's proof of work via proof of stake [extended abstract]y. SIGMETRICS Perform. Evaluation Rev. **42**(3), 34–37 (2014)

5. Daian, P., Pass, R., Shi, E.: Snow white: Robustly reconfigurable consensus and applications to provably secure proof of stake. Cryptology ePrint Archive, Paper 2016/919 (2016), https://eprint.iacr.org/2016/919
6. Chen, J., Micali, S.: Algorand: A secure and efficient distributed ledger. Theor. Comput. Sci. **777**, 155–183 (2019)
7. Kiayias, A., Russell, A., David, B., Oliynykov, R.: Ouroboros: A provably secure proof-of-stake blockchain protocol. In: Advances in Cryptology - CRYPTO 2017. LNCS, vol. 10401, pp. 357–388. Springer (2017)
8. David, B., Gazi, P., Kiayias, A., Russell, A.: Ouroboros praos: An adaptively-secure, semi-synchronous proof-of-stake blockchain. In: Advances in Cryptology - EUROCRYPT 2018. LNCS, vol. 10821, pp. 66–98. Springer (2018)
9. Badertscher, C., Gazi, P., Kiayias, A., Russell, A., Zikas, V.: Ouroboros genesis: Composable proof-of-stake blockchains with dynamic availability. In: CCS 2018. pp. 913–930. ACM (2018)
10. Kerber, T., Kiayias, A., Kohlweiss, M., Zikas, V.: Ouroboros crypsinous: Privacy-preserving proof-of-stake. In: 2019 IEEE SP. pp. 157–174. IEEE (2019)
11. Ben-Sasson, E., Chiesa, A., Garman, C., Green, M., Miers, I., Tromer, E., Virza, M.: Zerocash: Decentralized anonymous payments from bitcoin. In: 2014 IEEE SP. pp. 459–474. IEEE Computer Society (2014)
12. Noether, S.: Ring signature confidential transactions for monero. Cryptology ePrint Archive, Paper 2015/1098 (2015), https://eprint.iacr.org/2015/1098
13. Ganesh, C., Orlandi, C., Tschudi, D.: Proof-of-stake protocols for privacy-aware blockchains. In: Advances in Cryptology - EUROCRYPT 2019. LNCS, vol. 11476, pp. 690–719. Springer (2019)
14. Baldimtsi, F., Madathil, V., Scafuro, A., Zhou, L.: Anonymous lottery in the proof-of-stake setting. In: 33rd IEEE Computer Security Foundations Symposium. pp. 318–333. IEEE (2020)
15. Kohlweiss, M., Madathil, V., Nayak, K., Scafuro, A.: On the anonymity guarantees of anonymous proof-of-stake protocols. In: 42nd IEEE SP. pp. 1818–1833. IEEE (2021)
16. Heilman, E., Kendler, A., Zohar, A., Goldberg, S.: Eclipse attacks on bitcoin's peer-to-peer network. In: 24th USENIX Security Symposium. pp. 129–144. USENIX Association (2015)
17. Dwork, C.: Differential privacy. In: Automata, Languages and Programming, 33rd International Colloquium, ICALP 2006, Venice. LNCS, vol. 4052, pp. 1–12. Springer (2006)
18. Dwork, C., McSherry, F., Nissim, K., Smith, A.D.: Calibrating noise to sensitivity in private data analysis. In: Theory of Cryptography, Third Theory of Cryptography Conference. LNCS, vol. 3876, pp. 265–284. Springer (2006)
19. Wang, C., Pujo, D., Nayak, K., Machanavajjhala, A.: Private proof-of-stake blockchains using differentially-private stake distortion. Cryptology ePrint Archive, Paper 2023/787 (2023), https://eprint.iacr.org/2023/787
20. Garay, J.A., Kiayias, A., Leonardos, N.: The bitcoin backbone protocol: Analysis and applications. In: Oswald, E., Fischlin, M. (eds.) Advances in Cryptology - EUROCRYPT 2015. LNCS, vol. 9057, pp. 281–310. Springer (2015)
21. Pass, R., Seeman, L., Shelat, A.: Analysis of the blockchain protocol in asynchronous networks. In: Advances in Cryptology - EUROCRYPT 2017. LNCS, vol. 10211, pp. 643–673 (2017)
22. Brown, L.D., Cai, T.T., DasGupta, A.: Interval Estimation for a Binomial Proportion. Statistical Science **16**(2), 101 – 133 (2001)

23. Cardano pooltool, https://pooltool.io/
24. Agresti, A.: On small-sample confidence intervals for parameters in discrete distributions. Biometrics **57**, 963 – 971 (09 2001)
25. Cardano official website, https://cardano.org/stake-pool-operation/

Appendix

A Hoeffding Bound

Theorem 5. *(Hoeffding bound). Let* $\{X_i\}_{i=1}^n$ *be independent random variables ranging in* $[a,b]$ *where* $a < b$, $X = \sum_{i=1}^n X_i$ *and let* $\mu = \mathbb{E}[x]$, *then for any* t*:*

$$Pr[|X - \mu| > t] \leq 2e^{\frac{-t^2}{n(b-a)^2}}.$$

B AVRF

AVRF consists of (AVRF.gen, Update, AVRF.prov, AVRF.vrfy). Suppose that G is a group of prime order q such that $q = \Theta(2^{2m})$. Let $H(x)$ denote the hash function.

- AVRF.gen(1^{2m}): Choose a generator $g \in G$, sample a random $k \in \mathbb{Z}_q$ and output(pk, k), where the public key $pk = (g, g^k)$.
- Update(pk): Let $v = g^k$. Randomly choose $r \in \mathbb{Z}_q$. Let $g' = g^r, v' = v^r$. Set $pk' = (g', v')$. Output pk'.
- AVRF.prov$_k(pk', x)$: Let $pk' = (g, v)$. Compute $u = H(x)$, $\eta = u^k$ and π', which is the ZK proof of statement $\{(k) : \log_u(\eta) = \log_g(v)\}$. Set $\pi = (u, \pi')$. Output(pk', η, π).
- AVRF.vrfy$_k(x, \eta, \pi)$: Output 1 if $u = H(x)$ and π verifies, and 0 otherwise.

C Functionalities

In this section, we recall functionalities $\mathcal{F}_{crs}$, $\mathcal{F}_{Init}^{Com}$ and $\mathcal{F}_{\Delta}^{ABC}$ defined in [13] [15].

Functionality $\mathcal{F}_{crs}$

The functionality is parameterized by a distribution $\mathcal{D}$.

- Sample crs from the distribution $\mathcal{D}$.
- Upon receiving (Setup,sid) from a party, output(Setup,sid, crs).

Functionality $\mathcal{F}_{Init}^{Com}$

The functionality is parameterized by a signature scheme **Sig** =(SIG.keygen, SIG.sig, SIG.vrfy) and a commitment scheme **Com**.

Initialization
The Functionality $\mathcal{F}_{Init}^{Com}$ contains a list of each stakeholder unique id - uid, their election stake $\mathcal{S}_{uid}$. For each stakeholder uid, the functionality dose :
1. Execute **Com** with fresh randomness r_{uid} to get commitment **Com**($\mathcal{S}_{uid}$,r_{uid});
2. Randomly pick a secret key sk_{uid} and compute public key vk_{uid} = $\mathsf{KeyGen}(sk_{uid})$.
Information

- Upon receiving an input message ($\mathsf{GetPrivateData}$,sid) from a stakeholder uid, output ($\mathsf{GetPrivateData}$, sid, $\mathcal{S}_{uid}$, r_{pid}, sk_{uid}).
- Upon receiving ($\mathsf{GetList}$, sid) from a party, output $\mathcal{L}$ = ($\mathcal{S}_{uid}$, r_{uid}).

Anonymous Broadcast Functionality: $\mathcal{F}_{\Delta}^{ABC}$

All parties can register or deregister at any time. The list $\mathcal{P}$ consists of registered parties $\{P_1, P_2, ..., P_n\}$. The functionality maintains a message buffer M.

Send Message
Upon receiving message (SEND, sid, m) from some party $P_i \in \mathcal{P}$, where $\mathcal{P} = \{P_1, P_2....P_n\}$ denotes the current party set, do:

1. Choose n new unique message-IDs: $mid_1, ..., mid_n$.
2. Initialize $2n$ new variables $D_{mid_1} = D_{mid_1}^{Max}, ..., D_{mid_n} = D_{mid_n}^{Max} = 1$, which are the delays and the maximum delays of the message for each party.
3. Set $M = M||(m, mid_i, D_{mid_i}, P_i)$ for each party $P_i \in \mathcal{P}$.
4. Send$(SEND, m, sid, mid_1, ..., mid_n)$ to the adversary.

Receive Message
Upon receiving message (FETCH,sid) from $P_i \in \mathcal{P}$:

1. For all tuples$(m, mid, D_{mid}, P_i) \in M$, set $D_{mid} = D_{mid} - 1$.
2. Let $M_0^{P_i}$ denote the subvector of M including all tuples of the (m, mid, D_{mid}, P_i) with $D_{mid} = 0$. Delete all $M_0^{P_i}$ from M and send $(sid, M_0^{P_i})$ to P_i

Adversarial Influence
Upon receiving message (DELAY,$sid, (T_{mid_1}, mid_1), \cdots, (T_{mid_\ell}, mid_\ell)$ from the adversary, do the following for each pair (T_{mid}, mid_i):

1. If $D_{mid_i}^{Max} + T_{mid_i} \leq \Delta$ and mid is a message-ID registered in the current M, set $D_{mid_i} = D_{mid_i} + T_{mid_i}$ and set $D_{mid_i}^{Max} = D_{mid_i}^{Max} + T_{mid_i}$; otherwise ignore this pair.

Adversarial multicast
Upon receiving (MSEND,$(m_1, P_1), \cdots, (m_\ell, P_\ell)$) from the adversary with $(P_1, \cdots, P_\ell \in \mathcal{P})$:

1. Choose ℓ new unique message-IDs: $mid_1, \cdots, mid_\ell$.
2. Initialize 2ℓ new variables $D_{mid_1} = D^{Max}_{mid_1}, \cdots D_{mid_\ell} = D^{Max}_{mid_\ell} = 1$.
3. Set $M = M||(m_1, mid_1, D_{mid_1}, P_1)|| \cdots ||(m_\ell, mid_\ell, D_{mid_\ell}, P_\ell)$.
4. Send (MSEND,$sid, m_1, mid_1, \cdots, m_\ell, mid_\ell$) to the adversary.

D Frequency Attack over 12 epochs

We investigate the transactions of Cardano for two months and focus on 600 pools. The proportion of the subsets with $R > \delta$ in different epochs is shown in Table 2.

Table 2. Proportion of 600 pools such that $R > \delta$ over 12 epochs.

Epoch		Proportion s.t. $R > \delta$				Epoch		Proportion s.t. $R > \delta$			
		$\delta = 0.1$	$\delta = 0.2$	$\delta = 0.3$	$\delta = 0.4$			$\delta = 0.1$	$\delta = 0.2$	$\delta = 0.3$	$\delta = 0.4$
325	24 h	86.8%	76.7%	66.0%	58.7%	331	24 h	83.2%	69.8%	57.1%	45.1%
	48 h	73.5%	53.3%	37.5%	22.3%		48 h	74.5%	52.8%	35.5%	23.5%
	72 h	69.0%	46.3%	31.0%	18.3%		72 h	70.3%	43.7%	26.2%	15.2%
	96 h	66.2%	39.8%	22.3%	12.8%		96 h	66.0%	34.0%	20.3%	10.7%
326	24 h	85.0%	68.0%	53.3%	44.5%	332	24 h	85.6%	69.0%	55.5%	45.8%
	48 h	82.5%	66.8%	51.2%	35.8%		48 h	75.7%	55.2%	39.8%	25.7%
	72 h	74.2%	50.0%	34.8%	22.0%		72 h	69.8%	44.7%	30.6%	17.7%
	96 h	62.7%	35.2%	16.8%	9.5%		96 h	66.5%	42.2%	23.8%	12.5%
327	24 h	92.5%	80.5%	72.5%	62.7%	333	24 h	88.8%	73.6%	63.7%	53.0%
	48 h	72.3%	50.5%	33.8%	21.3%		48 h	75.5%	55.5%	37.7%	22.8%
	72 h	71.5%	46.3%	29.2%	17.8%		72 h	66.8%	41.3%	24.1%	14.8%
	96 h	68.0%	41.7%	25.7%	13.3%		96 h	67.2%	38.2%	20.3%	9.2%
328	24 h	86.5%	78.7%	66.2%	60.3%	334	24 h	83.3%	68.5%	54.8%	43.8%
	48 h	78.5%	60.3%	46.7%	32.8%		48 h	74.3%	51.3%	33.2%	21.5%
	72 h	74.8%	51.0%	31.8%	19.2%		72 h	68.7%	42.0%	24.5%	12.8%
	96 h	64.2%	38.2%	19.5%	9.8%		96 h	65.2%	35.8%	18.8%	9.8%
329	24 h	84.3%	70.2%	57.0%	46.8%	335	24 h	82.7%	67.5%	55.3%	47.2%
	48 h	76.5%	56.0%	35.2%	22.8%		48 h	76.7%	58.3%	41.2%	28.5%
	72 h	67.3%	39.3%	22.8%	11.2%		72 h	69.2%	44.2%	25.2%	13.2%
	96 h	68.2%	38.3%	20.2%	10.0%		96 h	66.3%	36.2%	16.9%	9.3%
330	24 h	88.2%	73.3%	63.3%	52.5%	336	24 h	88.3%	79.5%	65.8%	58.3%
	48 h	74.5%	53.2%	34.5%	21.5%		48 h	74.6%	55.8%	35.3%	24.2%
	72 h	70.5%	43.3%	26.7%	15.3%		72 h	68.3%	43.3%	25.5%	14.5%
	96 h	63.5%	37.0%	19.7%	11.0%		96 h	66.8%	36.5%	20.2%	12.4%