OUROBOROS TAKTIKOS

Aaron Schutza
Houston, TX
a.schutza@topl.me

James Aman
Houston, TX
j.aman@topl.me

November 5, 2020

ABSTRACT

This work proposes an extension of the staking procedure of Ouroboros Genesis [1]. This modification, as we observe, can enhance the security properties of the underlying protocol. A novel mechanism called *Local Dynamic Difficulty* (LDD) has been established which has no Proof-of-Work analogue. LDD supplants the active slots coefficient with a function that depends on the *slot interval*, the discrete time interval between blocks on a tine. This function, the *LDD curve*, is a global setup and all parties may assess leadership eligibility by evaluating the LDD staking procedure. We postulate an LDD curve with a simple analytic form on distinct slot interval domains. The domains consist of a *forging window* and a *recovery phase*. We provide empirical results showing *regularization*, a non-uniform bias in slot intervals on tines. A regression of forging power with respect to controlled resources is performed. Simulation output shows a non-linear dependence of chain-quality as a function of active stake that favors the honest majority. This effect in turn weakens nothing-at-stake and covert adversaries in control of a given portion of stake. We assess the security bounds in the context of GUC in a new parameter space including LDD curve constants.

1 Introduction

Composable Proof-of-Stake (PoS) blockchain protocols encompass a new design paradigm that overcomes many of the scaling limitations of Proof-of-Work systems (PoW). The Global Universal Composability (GUC) framework is a formalism that enables subtle refinement of these protocols in a self-consistent manner. The security bounds of protocol parameters can be concretely analyzed using the GUC treatment. New protocols are born out of compositional extensions of underlying GUC protocols.

Recently the first composable PoS system has been specified, Ouroboros Genesis [1], that realizes a secure permissionless blockchain tolerant to dynamic availability. This protocol divides time into intervals called *slots* that are synchronized among all parties. The mechanism used to assess leadership eligibility is the *staking procedure*, a repeating trial that occurs once per slot. The mathematical form of the staking procedure has been carefully chosen to induce block dynamics similar to that of PoW systems, while constraining the probability of forging to be independent of the division of resources among participating parties. Central to this mechanism is the *active slots coefficient*, a parameter directly analogous to mining difficulty in PoW systems.

PoS consensus systems aim to overcome the problems of PoW consensus while providing a live and robust immutable ledger. Many of the issues with PoW stem from resource scaling limitations, cost of operation, environmental impact, and low throughput. PoS systems provide the same functionality at practically no energy cost. Nodes are only responsible for testing and validating blocks, and forging consists of a pseudo-random deterministic trial that executes once per round. A common mechanism among PoS and PoW is a leadership election process that samples unbiased randomness. The PoW mechanism uses a Global Random Oracle (GRO) to produce unpredictable pseudo-random proofs that are assessed by validating parties for leadership eligibility. PoS protocols are designed to reproduce this leadership election mechanism so that an honest majority of participants reaches consensus. In the case of PoS systems, leadership eligibility is predictable, so the randomness that identifies eligibility must be constructed in a way that cannot be biased by any one party.

A convenient way to produce tamper free entropy is to sample verifiable information from a sufficient depth of the blockchain. Genesis solves this by having forgers and validators commit to a nonce generated with verifiable randomness that is concretely associated with the forging party. This constraint is enabled by a Verifiable Random Function (VRF), a cryptographic functionality similar to that of digital signatures. VRF output consists of a pseudorandom byte-string y of length ℓ_{VRF} and a proof π. Each pair (y_p, π_p) is verifiable with an associated public key corresponding to the forging party p. The VRF randomness y_p is evaluated by forging parties and validators to elect slot leaders in the staking procedure.

The staking procedure and leadership validation are predicated on a synchronization mechanism. Since all nodes must agree on the current slot, Genesis has all nodes query a trusted authority on time. Recently a compositional extention of Genesis was introduced, Ouroboros Chronos [2], that enables permissionless synchronization of the global clock that nodes use to tell what slot is currently transpiring. This reaffirms the distributed nature of the protocol and prevents adversarial manipulation of the current slot. This *global slot* is a new constraint that previously didn't exist in PoW systems.

Genesis [1] has been carefully constructed to reproduce the uniformly random leader election that PoW systems use. This produces some effects that are undesirable for network performance that have plagued PoW networks for over a decade. Block time intervals have to be on the order of minutes to prevent forks. Simultaneously, throughput can stall when no leaders are elected for long intervals. The uniform randomness that is used for leader election is *too* random. It induces a time interval between blocks that can either be too short or too long.

Using the newly established timing constraint introduced in [1] and [2] we can have nodes stall forging for a short period and then reengage the staking procedure. We can also temporarily boost the probability of a block to be forged by having all nodes agree on a time dependent forging difficulty. The interplay between these two effects may be finely tuned to estabilish new block dynamics, with completely different block time distributions and security properties.

We have developed a simulation test-bed that implements Genesis [1], called *Prosomo* [4], that includes time dependent difficulty thresholds. This new functionality is called Local Dynamic Difficulty (LDD). All ideal functionalities specified in [1] have been implemented and are carried out in the simulation. The client can be executed as local simulations or node-to-node testnet configurations seamlessly, using the same codebase.

In this paper we present analysis and results of simulation output [4] affirming that security bounds, for the first time in the PoS setting, can be improved beyond the limit of traditional PoW consensus.

2 The Staking Procedure

The staking procedure is a repeating trial that occurs once per slot. Slot leaders are elected by comparing the VRF output y_p to a threshold that is a function of the relative stake controlled by that party α_p. For party p the probability of being elected slot leader, given an active slots coefficient f is

$$\phi(\alpha_p) = 1 - (1 - f)^{\alpha_p}. \tag{1}$$

This threshold has been chosen to satisfy independent aggregation. This ensures that the probability of leader election is the same no matter what the distribution of α_p is. With p as an index across all forging parties, independent aggregation is enforced by

$$1 - \phi\Big(\sum_p \alpha_p\Big) = \prod_p \big(1 - \phi(\alpha_p)\big). \tag{2}$$

In each independent trial, the VRF test value included in block headers is compared with the threshold of forging party p. In slot s with epoch nonce $\eta^{\mathtt{ep}}$ [1], if the following inequality is true:

$$y_p^{\mathtt{ep}}(s) < \phi(\alpha_p^{\mathtt{ep}}) \tag{3}$$

where

$$y_p^{\mathtt{ep}}(s) = \frac{y_p[\eta^{\mathtt{ep}}||s||\texttt{"TEST"}]}{2^{\ell_{\text{VRF}}}}$$

then the block header is considered valid. The randomness y_p is unique among all parties indexed by p and is independently verifiable with the corresponding VRF proof π_p.

The repeating tests of (3) constitutes a series of independent trials. Sampling independent trials makes analysis of protocol dynamics more straightforward. In [1] an analysis of the distribution of *tines*, i.e forks, shows that it is dominated by tines produced by the honest majority. The independent trials makes the derivation of this dominant distribution easier to handle. It also provides the benefit that nothing-at-stake adversaries have no advantage since leadership eligibility cannot be biased whatsoever.

The dominant distribution will have certain features induced by the staking procedure. The probabilities of empty slots $P_\perp$, unique honest leader elections P_0, and adversarial leader elections P_1 play a role in the probability distribution function of time intervals on each tine. Recently it has been shown [3] that PoS consensus is robust in the presence of honest ties, where more than one leader is elected in a given slot. These honest ties don't necessarily contribute to the density of malicious trials P_1. Security bounds may be assessed in terms of the probability of at least one leader being elected $P_\parallel$ across distributions of honest stake and adversarial stake.

The staking procedure induces a distribution of time intervals on the honest majority tine. The time intervals are discrete numbers since the slot corresponds to the time that the block was forged. The distribution of time intervals correlates with the probability of leader election in each independent trial. We define the probability density funciton as

$$\langle n(\delta) \rangle = P_\parallel \prod_i^\delta P_\perp.$$

The discrete slot intervals δ correspond to the time since the last block was observed. The probability of no slot leader being elected $P_\perp$ in each slot interval δ is:

$$P_\perp(\delta) = \prod_i^\delta (1 - f) = (1 - f)^\delta.$$

The probability of at least one slot leader being elected $P_{\parallel}$ is:

$$P_{\parallel}(\delta) = f.$$

The expectation value that at least one slot leader had been elected in a given slot interval where $\delta > 0$ is then:

$$\langle n(\delta) \rangle = P_{\parallel}(\delta) P_{\perp}(\delta - 1) = f(1-f)^{\delta - 1}. \tag{4}$$

The expectation value of block time $\langle t \rangle$ can be calculated from this distribution:

$$\langle t \rangle = \sum_{\delta=0}^{\infty} \delta \langle n(\delta) \rangle \approx \int_0^{\infty} \delta \langle n(\delta) \rangle d\delta = \frac{f}{(1-f)\log^2(1-f)}.$$

3 Local Dynamic Difficulty

We introduce a new staking procedure that induces a dominant distribution with different properties. The slot interval is defined over the domain $\delta > 0$ with the slot of a header s_i and the slot of the parent header s_{i-1}. Note that $s_0 = 0$ is the genesis slot. The slot interval on each block header is then:

$$\delta_i = s_i - s_{i-1}.$$

In the LDD staking procedure, we use a threshold modeled after (1) with the slot interval as an argument:

$$\phi(\alpha, \delta) = 1 - \big(1 - f(\delta)\big)^{\alpha} \tag{5}$$

where $f(\delta)$ is a function called the LDD curve, or difficulty curve. With (5), independent aggregation holds across each discrete slot interval:

$$1 - \phi\bigg(\sum_p \alpha_p, \delta\bigg) = \prod_p \big(1 - \phi(\alpha_p, \delta)\big). \tag{6}$$

A VRF test procedure analogous to (3) is carried out in each slot. For header validation in slot s_i with forging party p on a parent block with slot s_{i-1}, if the following inequality is true:

$$y_p^{\mathtt{ep}}(s_i) < \phi(\alpha_p^{\mathtt{ep}}, \delta_i) \tag{7}$$

then the header is valid.

A dynamic difficulty curve was introduced to find a mechanism to space out blocks on tines. Uniform random leader election induced by (3) allows for back-to-back leader election in the smallest possible time interval of $\delta = 1$. The probability density function (4) peaks at the slot interval $\delta = 1$. This means that the most frequent time interval between blocks is always $\delta = 1$, no matter the value of f. On the other hand, slot intervals can be much greater than the expectation value of block time, causing long stalls in leadership election. These two situations can be taken advantage of by covert adversaries that selectively withhold and broadcast blocks in an effort to bias the network.

We have no specific constraints on the form of $f(\delta)$ besides the range $0 \leq f(\delta) < 1$. All parties must agree on an explicit difficulty curve so we consider $f(\delta)$ as a global setup. Intuitively, the lower $f(\delta)$ is, the harder it is to forge blocks. The higher $f(\delta)$ is, the easier it is to forge. If we enforce that $f(\delta) = 0$ over some domain of δ, it would space out blocks on tines. This would stop the covert adversary's advantage of back-to-back leadership. Localization is too beneficial in the default staking procedure of (3) because of successive leader election. There will inevitably be *slot battles*, where many leaders are elected within the time span of the network delay Δ, for any constant f. If we introduce a *slot gap*, ψ over which $f(\delta < \psi) = 0$, then there will be no blocks closer than ψ slots apart on any valid tine. The value of ψ should be less than the network delay Δ to ensure network consistency.

If we allow for stalls in forging with $\psi > 0$ we might wish to temporarily increase the capacity of the network to forge blocks after the slot gap. This would compensate for the loss of chain growth induced by $\psi > 0$ that reduces the average forging power. The slope of the difficulty curve $\frac{df(\delta)}{d\delta}$ plays an important role when variation is on the order of the network delay, as a changing difficulty can introduce adversarial advantage by allowing nothing-at-stake forging to bias leadership eligibility by manipulating the forging threshold of (7). The LDD staking procedure is locally predictable, but leader election is no longer static. Even thought the values of y_p can be predicted, the threshold in (5) cannot be uniquely predicted since different tines will give varying values of δ.

The mechanism of LDD effectively changes the probability of blocks to be forged by the network based on how recently a block has been observed. When we have a difficulty curve with a temporary increase in $f(\delta)$ then the presence of blocks on the network will temporarily cause more blocks to be forged. If we specify a *forging window* of γ slots where $f(\delta \leq \gamma)$ temporarily increases, blocks will trigger more blocks to be forged on timescales of γ. To retain the security properties of the underlying protocol, the difficulty curve should fall to some constant baseline difficulty f_B that's independent of the slot interval δ. The domain of $\delta > \gamma$ corresponds to the *recovery phase* of the LDD staking procedure. We postulate a curve including these parameters, with $0 \leq \psi < \gamma$, called the Taktikos curve:

$$f(\delta) = \begin{cases} 0 & \delta < \psi \\ f_A \cdot \left(\frac{\delta - \psi}{\gamma - \psi}\right) & \psi \leq \delta \leq \gamma \\ f_B & \gamma < \delta \end{cases} \tag{8}$$

where ψ is the slot gap, γ is the forging window cutoff, f_A is the amplitude, f_B is the baseline difficulty. Refer to Figure 1 for a schematic of (8). The two difficulty parameters are constrained by $0 < f_B < f_A < 1$. The Taktikos curve is ideal for the desired effect of an LDD curve. Leadership eligibility may only be biased in the forging window $\delta \leq \gamma$. First order dependence on δ is crucial for the prospect of security analysis since it represents the simplest form of variation over time for $\frac{df(\delta)}{d\delta} \neq 0$. Let's simplify the form of $f(\delta)$ by setting $\psi = 0$ and $\gamma \gg \delta$. We have a constant slope $f(\delta) = c\delta$ where

$$c = \frac{df(\delta)}{d\delta} = \frac{f_A}{\gamma} \tag{9}$$

and $0 < c \ll 1$. The probability of no leader being elected in δ slots is then

$$P_{\perp}(\delta) = \prod_{i}^{\delta}(1 - f(i)) = \prod_{i}^{\delta}(1 - ci)$$

where we are taking the product over successive conditioned trials. The probability of at least one slot leader being elected is:

$$P_{\parallel}(\delta) = f(\delta)$$

The distribution of slot intervals on tines $\langle n(\delta) \rangle$ is the expectation value of the number of discrete slot intervals $n(\delta)$ over a set of trials:

$$\langle n(\delta) \rangle = P_{\parallel}(\delta) P_{\perp}(\delta - 1) = f(\delta) \prod_{i}^{\delta - 1}(1 - f(i)).$$

Substituting $f(\delta) = c\delta$ we have a closed form for the probability density function over slot intervals:

$$\langle n(\delta) \rangle = c\delta \prod_{i}^{\delta - 1}(1 - ci). \tag{10}$$

Figure 3 shows (10) for two values of the slope c. The expectation value of the block time can be directly calculated from the probability density function (10):

$$\langle t \rangle = \sum_{\delta=0}^{\infty} \delta \langle n(\delta) \rangle. \tag{11}$$

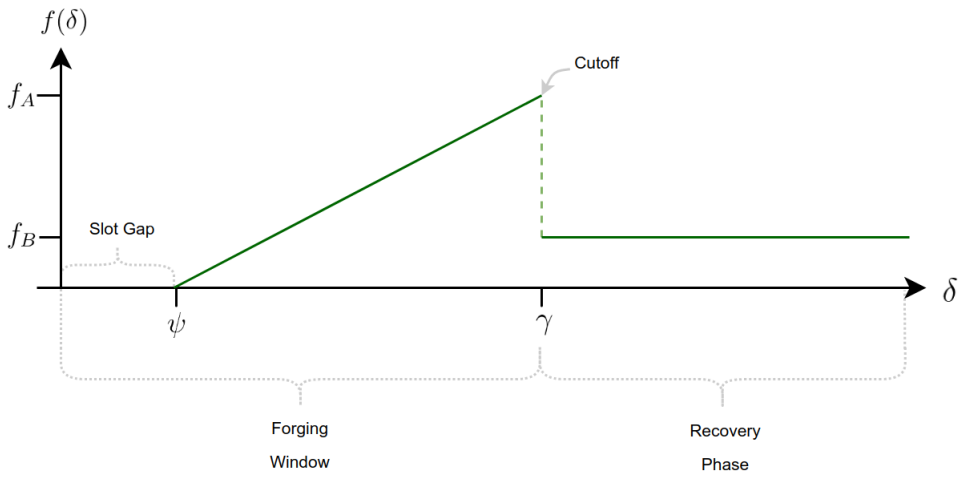

Figure 1: A cartoon of the Taktikos curve given in (8).

Figure 2 shows the expectation of block time for different slopes c.

We wish to find an analytic form for the distribution $\langle n(\delta) \rangle$. Using the definition of the geometric integral in the continuum limit of $\delta \Rightarrow d\delta$:

$$\langle n(\delta) \rangle \approx c\delta \exp\left[\int_0^{\delta-1} \log(1-cx)dx\right].$$

This simplifies to the Taktikos distribution function:

$$\langle n(\delta) \rangle = c\,\delta\, e^{1-\delta}(1-c\delta+c)^{\delta-c^{-1}-1}. \tag{12}$$

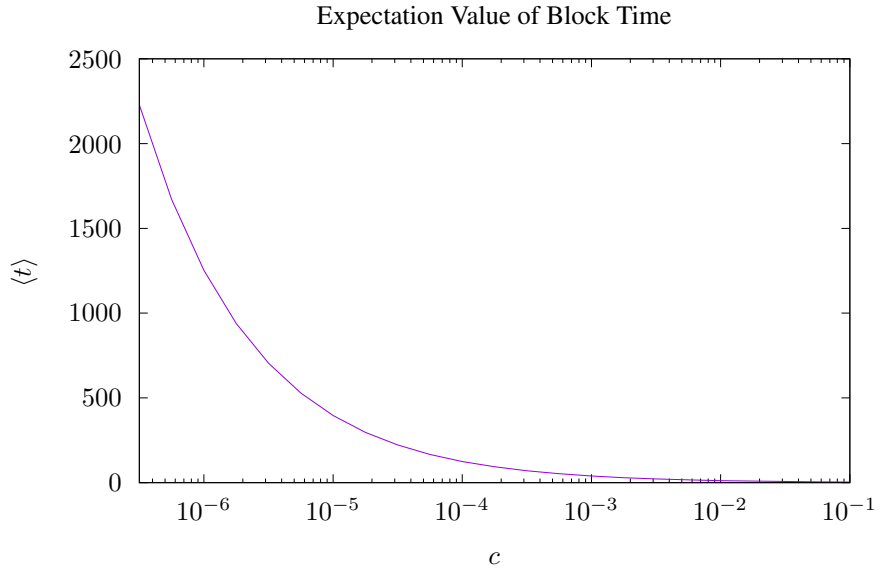

Figure 2: A plot of (11) in units of slots with varying c on a log scale.

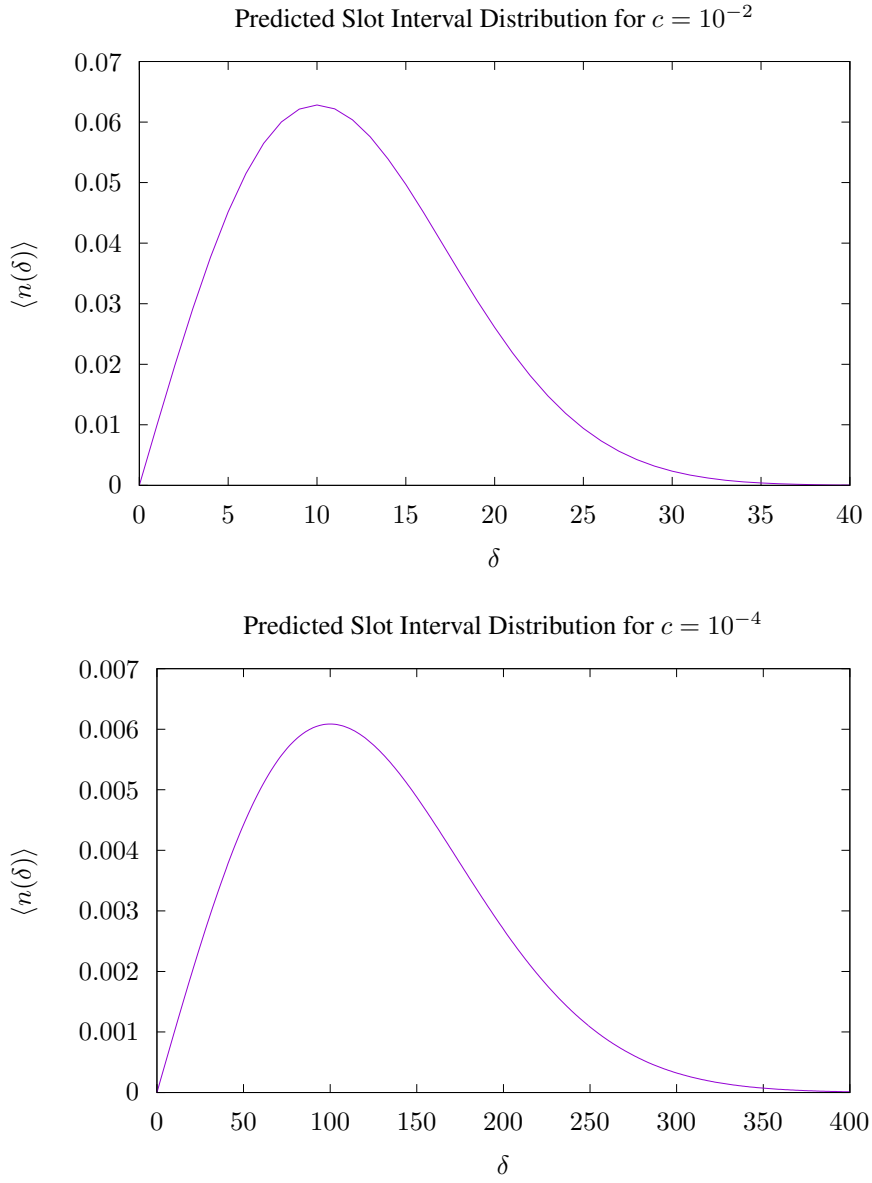

Figure 3: A plot of (10) with $c = 10^{-2}$ (top) and $c = 10^{-4}$ (bottom). This slot interval distribution peaks at $\delta_{\text{peak}} \approx c^{-\frac{1}{2}}$.

4 Resource Scaling

The Taktikos distribution scales in a way that is non-linear with respect to the amount of active stake contributing to leader election on a given tine. We may explore the resource scaling properties by introducing a scaling factor r. We consider uniform stakers $\alpha_p = \alpha_{p'}\ \forall p \neq p'$. The net stake is divided between active and inactive pools. $\Sigma_{\mathcal{H}}$ is active stake, $\Sigma_{\mathcal{I}}$ is inactive stake, $\Sigma = \Sigma_{\mathcal{H}} + \Sigma_{\mathcal{I}}$ is the net stake, and the party number is P. The relative stake is

$$\alpha_p = \frac{1}{P}\frac{\Sigma_{\mathcal{H}}}{\Sigma}$$

and resource scale factor r is

$$r = \frac{\Sigma_{\mathcal{H}}}{\Sigma_{\mathcal{H}} + \Sigma_{\mathcal{I}}}. \tag{13}$$

The system may be setup with proportion of inactive stake

$$\Sigma_{\mathcal{I}} = (r^{-1} - 1)\Sigma_{\mathcal{H}}$$

to see how forging power changes with respect the amount of active resources in terms of (13). In each trial the value of $\Sigma_{\mathcal{I}}$ is allotted to an inactive account that will not be used to forge by any participant on the network. The forging power is the reciprocal block time $\langle f \rangle$ and we expect the amount of controlled resources to affect the forging power of active stake. The absolute forging power of the active stake is $f(r) = \langle f_{\mathcal{H}} \rangle|_r$ and the relative staking power is:

$$F(r) = \frac{f(r)}{f(r) + \langle f_{\mathcal{I}} \rangle|_r}. \tag{14}$$

This gives a normalized metric of relative influence when stake is split among two populations. We expect $F(r = \frac{1}{2}) = \frac{1}{2}$ and that the scaling of the function over the domain $0 \leq r \leq 1$ will indicate the relative influence between two distributions of resources that are not interacting on the network. To evaluate $\langle f_{\mathcal{I}} \rangle|_r$ we use a parity constraint such that the staking power of inactive resources is equal to the staking power of the active resources:

$$r' = \frac{\Sigma_{\mathcal{I}}}{\Sigma_{\mathcal{H}} + \Sigma_{\mathcal{I}}}.$$

The two resource indices are unitary:

$$r + r' = 1.$$

With this we have

$$\langle f_{\mathcal{I}} \rangle|_r \equiv \langle f_{\mathcal{H}} \rangle|_{r'} = f(1 - r)$$

and evaluating into (14) the relative staking power is:

$$F(r) = \frac{f(r)}{f(r) + f(1 - r)}. \tag{15}$$

To evaluate the probability density function for different values of r, the independent aggregation constraint in (6) is used to introduce (13) to the expression for the probability of empty slots:

$$1 - \phi(r, \delta) = \prod_{p}^{P} (1 - \phi(\alpha_p, \delta))$$

and we have

$$\phi(r, \delta) = 1 - (1 - f(\delta))^r.$$

We now have for $P_{\perp}(\delta)$:

$$P_{\perp}(r, \delta) = \prod_{i}^{\delta} (1 - f(i))^r.$$

The probability density function as a function of r can now be constructed:

$$\langle n(\delta, r) \rangle = \phi(r, \delta) P_{\perp}(r, \delta - 1) = \big(1 - (1 - f(\delta))^r\big) \prod_{i}^{\delta - 1} (1 - f(i))^r.$$

Substituting for $f(\delta) = c\delta$ and taking the continuum limit gives

$$\langle n(r, \delta) \rangle \approx \big(1 - (1 - c\delta)^r\big) \exp\left[\int_0^{\delta - 1} \log\left[(1 - cx)^r\right] dx \right]$$

$$\langle n(r, \delta) \rangle = \big(1 - (1 - c\delta)^r\big) e^{r - r\delta} (1 - c\delta + c)^{r\delta - \frac{r}{c} - r}.$$

This gives the block time in terms of r:

$$\langle t(r) \rangle = \int_0^{\infty} \delta \langle n(\delta, r) \rangle d\delta.$$

We may evaluate this numerically to get a relationship between absolute forging power, controlled resources r, and the slope c:

$$f(r, c) = \langle t(r) \rangle^{-1}|_c. \tag{16}$$

This absolute forging power (16) has undesirable scaling properties with respect to r. To improve the security scaling properties while keeping the Taktikos distribution profile at short slot intervals, we reintroduce the forging window. To account for the cutoff γ and amplitude f_A we have the probability of empty slots in terms of the Taktikos curve parameters:

$$P_{\perp}(\delta, \gamma) = \begin{cases} \prod_i^{\delta} (1 - ci)^r & \delta \leq \gamma \\ (1 - f_B)^{(\delta - \gamma)r} \prod_i^{\gamma} (1 - ci)^r & \delta > \gamma \end{cases}$$

and the distribution function in terms of r, γ, $c = \frac{f_A}{\gamma}$, and f_B is

$$\langle n(r, \delta, \gamma, c) \rangle = \begin{cases} \left(1 - (1 - c\delta)^r\right) \prod_i^{\delta} (1 - ci)^r & \delta \leq \gamma \\ (1 - f_B)^{(\delta - \gamma)r} \left(1 - (1 - c\gamma)^r\right) \prod_i^{\gamma} (1 - ci)^r & \delta > \gamma. \end{cases} \tag{17}$$

The absolute forging power is evaluated as

$$f(r) = \langle t(r) \rangle^{-1}|_{c,\gamma} = \left[\sum_{\delta=0}^{\infty} \delta \langle n(r, \delta, \gamma, c) \rangle \right]^{-1}. \tag{18}$$

With these expressions we have a direct path to security bound analysis.

5 Simulation Results

Here we show results exhibiting regularization, a non-uniform distribution of slot intervals on tines induced by the LDD staking procedure. The setup constitutes a series of trials with epoch parameters bounded by Genesis estimates. For a detailed set of Genesis parameters used in the setup refer to [4]. LDD parameters have been chosen that exhibit noticeable effects on chain quality while adhering to the security bounds of the underlying protocol. Refer to Figure 4 for simulation output of the measured probability density function with LDD turned on. As a control, the same setup was run with LDD turned off, also shown in Figure 4.

6 Security Bounds

To gauge the security of the modified protocol with the LDD staking procedure, bounds must be placed on the quantities ψ, γ, f_A, and f_B. Firgure 5 displays simulation output along with theoretical profiles of resource scaling that clearly shows improved security properties. Security bounds may be estimated by considering the LDD mechanism as a compositional extension of Genesis expressed in the language of GUC. A full security analysis of this protocol using the Taktikos difficulty curve (8) is forthcoming. LDD represents a new class of protocols with different difficulty curves that will have unique distributions and scaling properties. The optimal curve has not been found and future research will uncover new difficulty curves with better properties. The methods used in this paper provide very good estimates of the effect of LDD parameters on chain quality. With appropriate treatment of security bounds, the LDD staking procedure constitutes a new protocol, *Ouroboros Taktikos*, with security properties that can exceed the bounds set by Genesis [1].

References

[1] C. Badertscher, P. Gaži, A. Kiayias, A. Russel, V. Zikas "Ouroboros Genesis: Composable Proof-of-Stake Blockchains with Dynamic Availability" *CCS '18: Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, January 2018* Pages 913–930, https://doi.org/10.1145/3243734.3243848

[2] C. Badertscher, P. Gaži, A. Kiayias, A. Russel, V. Zikas "Ouroboros Chronos: Permissionless Clock Synchronization via Proof-of-Stake", July 2019, ia.cr/2019/838

[3] A. Kiayias, S. Quader, A. Russell "Consistency of Proof-of-Stake Blockchains with Concurrent Honest Slot Leaders" *ArXiv Preprint: arXiv:2001.06403*, 2020 – arxiv.org

[4] https://github.com/Topl/Prosomo

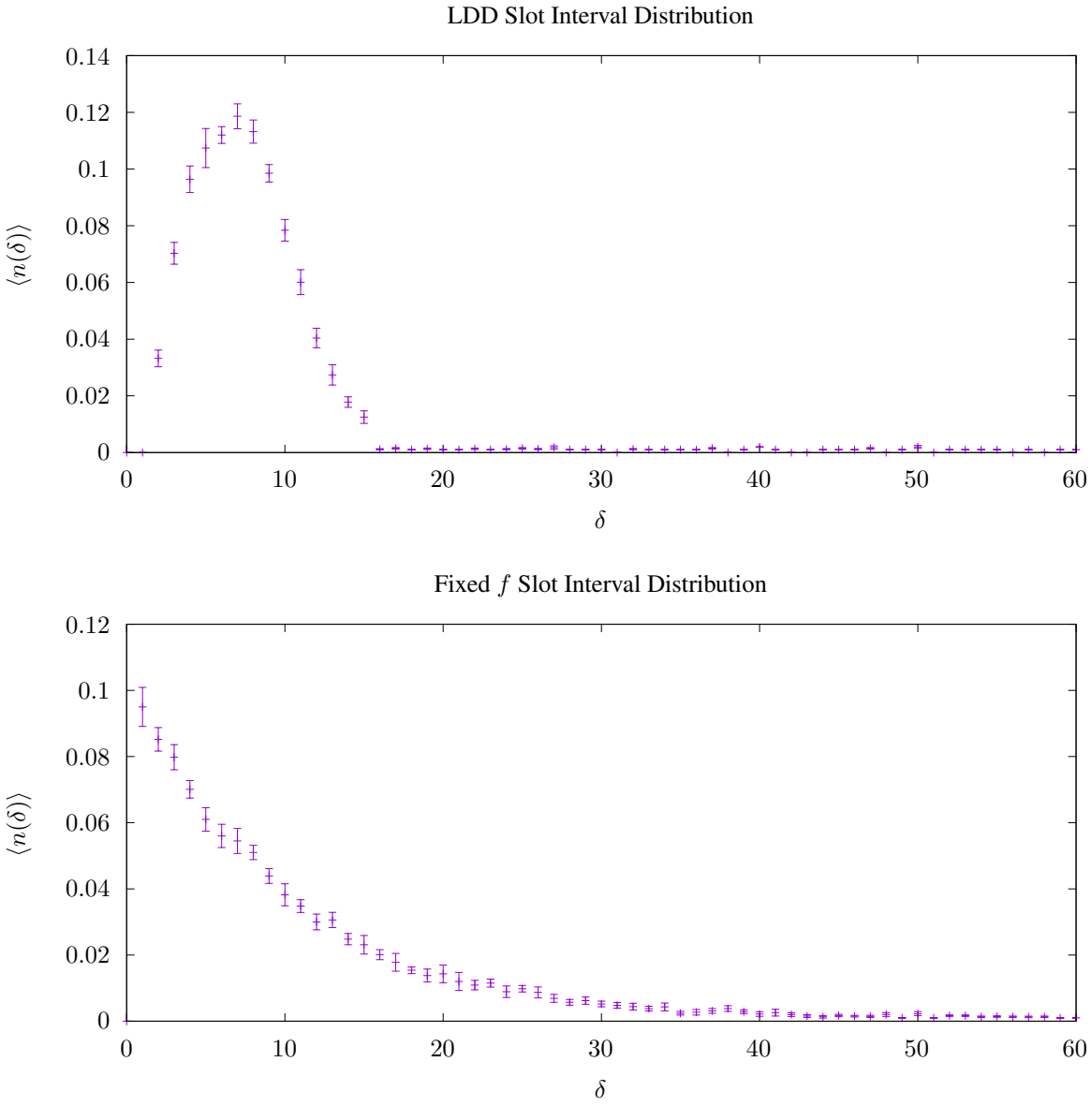

Figure 4: (Top) The measured probability density function with LDD turned on. 10 repeated trials were executed with 64 uniform Stakeholders. Each bin represents the normalized number of slot intervals recorded on the honest majority tine after 1000 blocks. Unique input entropy was used across each trial. This corresponds to random global positions in the simplified global delay model [4] and different VRF nonce output in each trial. A slot gap of $\psi = 1$ was used with $\gamma = 15$, $f_A = 0.5$, $f_B = 0.05$, and 1 second slot times ($\Delta \approx 10$). The confidence intervals correspond to the standard deviation across each independent trial. This represents a very crude preliminary data set, but the trend is clearly visible. The profile is quite similar to the theory prediction plotted in Figure 3. (Bottom) The same setup with static $f = 0.1$. The slot interval distribution matches the expected profile for leadership sampling from uniform randomness. The profile has an exponential drop-off similar to that of PoW block time intervals. This profile matches the theoretical prediction of (4) using the standard staking procedure (3).

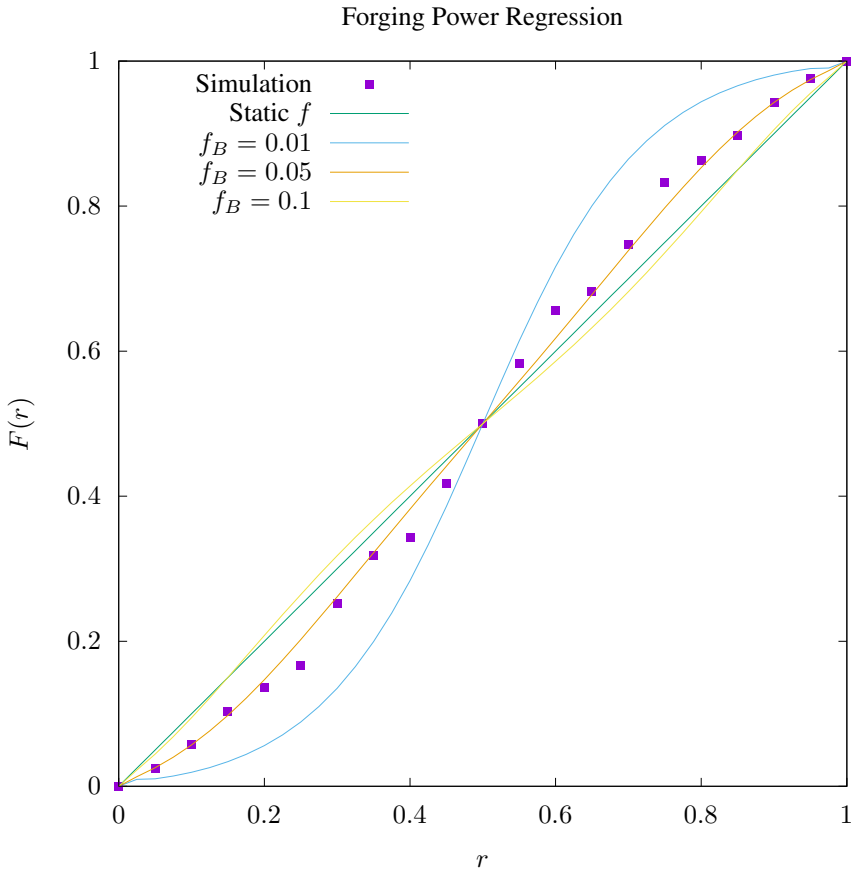

Figure 5: The relative staking power $F(r)$ evaluated using (15) with $\gamma = 15$, $\psi = 0$, and $f_A = 0.5$. The static f curve corresponds to the default staking procedure of (1). Other curves calculated with (18) and (17) are shown with varying values of f_B. These are provided to show how the relative staking power scales with different baseline difficulties. The dots are simulation trials corresponding to a regression over r with a value of $f_B = 0.05$, 1 second slot time. The simulation points are calculated from the absolute forging power $f(r)$ recorded after 100 blocks with the same input entropy and global position across 16 uniform stakeholders per trial. On the domain $0 < r < \frac{1}{2}$, sub-linear $F(r)$ corresponds to resource scaling that is better than the linear PoW trend where r corresponds to the rate of queries to the GRO and $F_{\text{PoW}}(r) = r$. On the domain $\frac{1}{2} < r < 1$, super-linear $F(r)$ corresponds to resource scaling that boosts the resiliance of the honest majortiy of staking power, exceeding the PoW trend. Taktikos is the first consensus protocol that exceeds the security scaling properties of PoW.