Heavy-coin Committee Selection for the Decentralized Era *(Technical Report)*

Yucheng Sun
Department of Computer Science
National University of Singapore
Singapore, Singapore
sunyuch@comp.nus.edu.sg

Haifeng Yu
Department of Computer Science
National University of Singapore
Singapore, Singapore
haifeng@comp.nus.edu.sg

Abstract

Committee selection is a fundamental building block for enabling blockchains to scale. Most blockchains use the *canonical design* for committee selection, where committee members are chosen uniformly randomly, weighted by the stake of each node. Several recent breakthrough works [24, 45] interestingly show that the canonical design is not the only way to go. These works show that by deterministically including certain nodes in the committee, we can significantly reduce the committee size needed.

These recent designs [24, 45] however are meaningful only under *highly-centralized* stake distributions. If the stake distribution is more *decentralized* (i.e., with less wealth concentration), then they quickly degrade to the canonical design. As the central contribution of this work, we propose a novel *heavy-coin committee selection scheme* or *HC scheme*. For more decentralized distributions, our HC scheme is the *first* and *only* design that can do better than the classic canonical design. For other (highly-centralized) distributions, our HC scheme retains the benefits of the best existing design.

1 Introduction

Background. Committee selection is a fundamental building block for enabling blockchains to scale [15–17, 26, 31, 32, 37, 42]. Here a *committee* is essentially a representative subset of nodes selected from the system to handle critical operations. For example, Algorand [26] uses committees to run its byzantine agreement protocol. In Ouroboros [37] and Ouroboros Praos [16], an (implicit) committee proposes blocks in their longest-chain protocol.

To ensure the blockchain's security, the committee must properly "represent" the whole system, in terms of the corruption ratio. Namely, if at most f fraction of the stake in the system is corrupted (i.e., held by corrupted nodes), then the corruption ratio in the committee should not be much larger than f. For example in Algorand [26] with $f = 0.2$, if the corruption ratio in the committee reaches $1/3$, then its byzantine agreement protocol will fail, and the system is no longer secure. A larger committee can always better "represent" the system, but at the cost of more nodes in the committee (and hence larger overheads).

Canonical design. To properly "represent" the system, intuitively, the committee members need to be chosen "uniformly randomly" — any deviation can imply "misrepresentation". Such uniformly random selection is precisely what most blockchains [15–17, 26, 31, 32, 37, 42] do, and we call this as the *canonical design*. Specifically, the canonical design has two versions, depending on whether the committee size is fixed or a random variable:

- Fixed-size [17, 31, 32, 37, 42]: The system uses some global public randomness to repeatedly select committee members, until the target committee size is reached. Every time a committee member is to be selected, it is chosen from all nodes uniformly randomly weighted by their respective stake.
- Variable-size [15, 16, 26]: Let τ be the smallest unit of cryptocurrency that can be held by a node. A node with stake s will act as $\frac{s}{\tau}$ *virtual nodes*. Each virtual node gets into the committee with the same probability p, independently. The final committee size is a random variable, with a fixed expectation.

As expected, in both versions, a node with more stake has a higher probability of getting into the committee.

Recent breakthrough [24, 45]. Committee selection is a rather basic problem with a simple formulation. Hence on the surface, there is not much we can do, beyond the canonical design. In the past few years, however, several breakthrough works [24, 45] interestingly show that the canonical design is not the only way to go. We call these (i.e., the FA scheme [24] and the TPL scheme [45]) as *semi-deterministic schemes*, since they all use the following paradigm.

These schemes first find a set $\mathbb{C}$ of *critical nodes*, all of which will be deterministically included in the committee. Each node in $\mathbb{C}$ is considered to have "sufficiently large" stake, to warrant such inclusion. Here the threshold of "sufficiently large" may be determined by a carefully designed algorithm (rather than being a fixed value), which takes into account the entire stake distribution. Next, additional committee members are selected from the remaining non-critical nodes, by following the canonical design.

It has been shown [24, 45] that compared to the canonical design, these semi-deterministic schemes can offer much better security for the same committee size, or need much smaller committee size for achieving the same security level. Such improvements have been demonstrated under the real-world stake distributions of a number of popular cryptocurrencies (e.g., Ethereum [20] and Bitcoin [5]).

Semi-deterministic schemes and highly-centralized stake distributions. For the FA scheme, it is known [25] that the more centralized the stake distribution is, the more improvement it can bring. This is simply because more centralization will usually lead to more critical nodes, which the FA scheme can effectively exploit. Since the TPL scheme [45] also exploits critical nodes, one would expect a similar trend. Gazi et al. [24] also observe, in one of their experiments, that when the stake distribution is not sufficiently centralized to give critical nodes, the FA scheme does not improve upon the canonical design.

In their experiments [24, 45], most of the stake distributions used have Gini coefficients [14] above **0.7**. As a reference point, countries in today's world have Gini coefficients ranging from **0.238 to 0.591** [46]. Note that even a difference of 0.1 in Gini coefficient is considered significant [29].

Decentralized stake distributions. The fact that these semi-deterministic designs [24, 45] rely on highly-centralized distributions is at odds with the hallmark decentralization goal of blockchains. Blockchains today are actively trying to reduce their centralization: For example, various blockchains (Ethereum [19], Cardano [30], and Cosmos Hub [28]) are starting to impose various hard or soft caps on a node's stake, to make the stake distribution less centralized.

In this work, we first obtain a more generalized set of results on how semi-deterministic schemes [24, 45] perform under decentralized distributions:

- We consider a spectrum of decentralized stake distributions, with Gini coefficients ranging from **0.2 to 0.6**. This roughly corresponds to the range of Gini coefficient of all countries/economies (**0.238 to 0.591**) in the world [46].
- We evaluate both the TPL scheme [45] and the FA scheme [24] under such distributions.

Our results show that in many cases, semi-deterministic schemes [24, 45] degrade to the canonical design, when the Gini coefficient is about 0.5 or smaller. This is despite that a Gini coefficient of 0.5 is actually still considered as *severe inequality* [29], if it were for a country.

Our work: Heavy-coin committee selection. As the central contribution of this work, we propose a novel committee selection scheme called *heavy-coin committee selection scheme* or *HC scheme*. For more decentralized distributions, our HC scheme is the *first* and *only* design that can outperform the classic canonical design. (It is the "only" design, since all existing designs degrade to the canonical design under such distributions.) For other (highly-centralized) distributions, our HC scheme retains the benefits of the best existing design.

Note that this is *not* just about better end results for committee selection. More importantly, our design reveals that even for decentralized distributions, there exist meaningful alternatives to the classic canonical design. This implies that this basic problem of committee selection actually has a richer structure than previously expected.

Our approach: Heavy-coin effect. Our novel HC scheme is driven by clear mathematical insights. Specifically, we identify and exploit a simple, yet somewhat counter-intuitive, mathematical property, which we call as the *heavy-coin effect*. The heavy-coin effect roughly says that, to reduce uncertainty, it is better to place a small number of bets each with large winning probability, rather than a large number of bets each with small winning probability. Despite this property being rather basic, it does not have a name — hence we use our own term here.

Note that this heavy-coin effect can be subtle, because it appears to contradict the well-known Law of Large Numbers (LLN) [18]: By LLN, more trials make the probability distribution more concentrated, which seems to imply that making a large number of bets would help to reduce uncertainty.

Our approach: Exploiting heavy-coin effect. We observe that this heavy-coin effect can be useful in committee selection. Roughly speaking, we will try to toss fewer number of coins (each with "heavy" winning probability), during committee selection. Note that our design will not involve any crypto construction.

Naively exploiting heavy-coin effect in committee selection would result in two problems:

- First, a naive design would prevent nodes with small stake from participating, since their stake is not sufficient to qualify as even a single "heavy" coin. We overcome this by allowing different nodes to toss coins with different winning probability, and then relying on the additivity of variance to preserve the heavy-coin effect.
- Second, with different nodes tossing different kinds of coins, it becomes challenging to find the worst-case adversary and to determine the needed committee size. To deal with this, we use the idea of currency *denominations*, and let each node break its stake into (a few) denominations.

Our experimental results. We have fully implemented our HC scheme (see Appendix A). For head-on comparison, we use the implementation of the FA scheme and TPL scheme from [45], as well as the real-world stake distributions of a number of popular cryptocurrencies (e.g., Ethereum and Bitcoin) from [45].

Our evaluation results show that under decentralized distributions (e.g., Gini coefficient **0.6 or less**) and realistic settings, our HC scheme can significantly outperform all existing designs: Under the same committee size, our HC scheme can achieve up to about 20 orders of magnitude smaller error. For achieving a given target error, the FA/TPL scheme [24, 45] in many cases gives no benefit over the canonical design, while our HC scheme achieves up to 30% reduction in committee size. It is important to note that such improvements of our HC scheme do not come at the cost of any other aspects of committee selection.

Such (up to 30%) reduction in committee size is non-trivial:

- In terms of economic efficiency, smaller committees can help lower on-chain expenses. For example in Ethereum, transactions are validated by a committee, and are charged some *gas fee*. The gas fees are partly meant for repaying the efforts of the committee members who validate the transactions. A 20% smaller committee will lead to 20% less work from the committee, which helps pave the path to lower the gas fees. The total annual gas fees charged from Ethereum transactions is on the order of 100 million USD [13, 22, 23]. Thus, reduction in committee size may potentially bring substantial economic benefits.
- In terms of performance, committees in blockchains are typically used to run byzantine agreement protocols [3, 4, 10, 27, 40, 47]. Such protocols can be communication-intensive, with message complexities often growing quadratically with the committee size. Such characteristics also show up in real-world experiments in prior works [2]. As a result, just 20% reduction in committee size could potentially lead to performance improvement of 50%.

Finally, our evaluation also shows that under other (highly-centralized) distributions (e.g., with Gini coefficients **above 0.6**), our HC scheme retains the benefits of the best existing design.

Notation	Description
$\mathbb{U}$	all nodes in the system
s_u	stake of node u
$\mathbb{S}$	stake distribution of all nodes
$\|\mathbb{S}\|$	total stake in $\mathbb{S}$
τ	minimum possible stake for a node
$\mathcal{A}$	adversary
t	security threshold
f	maximum amount of stake that the adversary can corrupt, where $f = t - \epsilon$
Π	committee selection protocol
k	committee size
$\mathbb{C}$	critical nodes
H	total honest weight in the committee
M	total corrupted weight in the committee
r	number of denominations in our HC scheme
$(d_1, \ldots, d_r)$	the denominations

Table 1: Key Notations

Roadmap. Section 2 formulates the committee selection problem. Section 3 reviews existing designs. Section 4 presents our finding that semi-deterministic schemes are meaningful only under highly-centralized distributions. Sections 5 through 7 elaborates our novel HC committee selection scheme. Section 8 presents our experimental results. Finally, Section 9 discusses related works, and Section 10 draws conclusions.

2 System Model and Attack Model

Table 1 summarizes our key notations.

Nodes and stake. We inherit the standard system/attack model in prior works [15, 16, 24, 26, 31, 32, 37, 42, 45]. Let $\mathbb{U}$ be the set of all parties (also called *nodes*) in the blockchain. Each node $u \in \mathbb{U}$ has some stake s_u, which is the amount of cryptocurrency it holds. Without loss of generality and unless otherwise mentioned, we assume that the stake distribution is normalized (i.e., $\sum_{u \in \mathbb{U}} s_u = 1.0$). The *stake distribution* $\mathbb{S}$ describes the stake held by each node $u \in \mathbb{U}$. Note that $\mathbb{S}$ is part of the blockchain state, and hence is publicly visible to all nodes as well as to the adversary. We use $|\mathbb{S}|$ to denote the total stake in $\mathbb{S}$, which is 1.0 if $\mathbb{S}$ is normalized.

Adversary. There is some adversary $\mathcal{A}$ who can corrupt an arbitrary subset of nodes in $\mathbb{U}$, so long as the total stake held by those *corrupted nodes* (also called *malicious nodes*) does not exceed a certain threshold f (e.g., $f = 0.2$). If a node is not corrupted, we say that the node is *honest*.

Committee selection. A *committee* is simply a subset of $\mathbb{U}$. In certain designs, each node in the committee further has a *weight*. The weight can be different from the node's stake. A *committee selection protocol* Π is a randomized algorithm that selects a committee from $\mathbb{U}$, and assigns a weight to each node in the committee. If the algorithm does not assign weights, we will view each node in the committee as having the default weight of 1. We call the total weight of the corrupted nodes (honest nodes) in the committee as *corrupted weight* (*honest weight*) in the committee. We use M (H) to denote the total malicious (honest) weight. We use k to denote the committee size. (For variable-size committee selection, k will be the expectation of the committee size.)

Security requirement. In blockchains, the selected committee is typically used [17, 26, 31, 33] to run byzantine agreement protocols. These protocols can only tolerate less than a certain fraction t of corrupted weight in the committee. For example, synchronous byzantine agreement protocols [1, 3] usually have $t = 1/2$, while asynchronous ones [26, 43, 47] have $t = 1/3$. Order-fair byzantine agreement protocols [36, 41] need $t = 1/4$.

Define $\epsilon = t - f$ to be the gap between f and t. For committee selection to work, ϵ must be positive. Usually blockchains use a small ϵ (e.g., $\epsilon = 0.1$ as in [24]), so that f can be close to t.

Error of committee selection. Following prior works [15–17, 24, 26, 31, 32, 37, 42, 45], the security (or more precisely, *error*) of a committee selection protocol Π is defined via the process below:

Step 1: $\mathcal{A}$ chooses nodes to corrupt, while knowing $\mathbb{U}$, $\mathbb{S}$, τ, f, t, k, and Π. Following prior works [15–17, 24, 26, 31, 32, 37, 42, 45], here it is assumed that $\mathcal{A}$ does not know the random coin flip outcomes in Π (despite that $\mathcal{A}$ knows Π). This assumption is fundamentally needed — otherwise $\mathcal{A}$ can just cherry-pick the committee members to corrupt. In practice, this is usually satisfied for example, by using verifiable random functions (VRFs) [39] as in [16, 26] or by considering slowly-adaptive adversary [17, 31, 32, 37, 42].

Step 2: The randomized algorithm Π selects a committee, while knowing $\mathbb{U}$, $\mathbb{S}$, τ, f, t, and k. Here Π knows neither $\mathcal{A}$ nor which nodes are corrupted by $\mathcal{A}$.

Step 3: The adversary wins iff $\frac{M}{M+H} \geq t$. Let $\mathbb{A}_{\mathbb{S}}^{f}$ denote the set of all possible adversaries for the stake distribution $\mathbb{S}$ where the adversary can corrupt up to f stake. Then the *error* of Π is the worst-case adversary's winning probability, namely:

$$\max_{\mathcal{A} \in \mathbb{A}_{\mathbb{S}}^{f}} \left\{ \Pr\left[\frac{M}{M+H} \geq t \right] \right\} \tag{1}$$

As committee selection is often periodically invoked, the error typically needs to be negligible (e.g., 10^{-40}).

There is usually a trade-off between error and committee size: Larger committees give smaller error, but at the cost of larger size.

3 Review of Existing Designs

A committee selection scheme usually has two versions, depending on whether the selected committee has a *fixed-size* or *variable-size*. Selecting a fixed-size committee usually needs some public randomness, which can be obtained from the blockchain itself [31, 32, 37, 42]. For variable-size committee selection, it suffices to use individual nodes' local randomness. The size of the resulting committee, however, will be a random variable with some fixed expectation [15, 16, 26]. Our work will consider both versions.

Canonical design. Section 1 informally mentioned the canonical design [15–17, 26, 31, 32, 37, 42]. Now we give a precise description:

- Variable-size version [15, 16, 26]: Each node u acts as s_u/τ *virtual nodes*. Each virtual node gets into the committee with the same probability $p = \tau k$, independently. Each

virtual node in the committee has the same weight 1. (If a node has $x > 1$ of its virtual nodes in the committee, then the node will have weight x.)

- Fixed-size version [17, 31, 32, 37, 42]: To select a committee member, each node u is chosen with probability s_u. This is repeated independently for k times, giving k committee members each with a weight of 1. (If a node is selected $x > 1$ times, its weight is x.)

Semi-deterministic schemes. We call the recent FA scheme [24] and TPL scheme [45] as *semi-deterministic schemes*, since they follow the same paradigm. These committee selection schemes first find a set $\mathbb{C}$ of nodes, which we call as *critical nodes*. Each node in $\mathbb{C}$ is considered to have "sufficiently large" stake, to warrant its deterministic inclusion into the committee. Each $u \in \mathbb{C}$ is then added to the committee, with its weight being equal to its stake.

Here the threshold of "sufficiently large" is often not based on some formula. Instead, this may be determined via some complex process, and may depend on $\mathbb{U}$, $\mathbb{S}$, f, and t. In fact, finding a good threshold is a key technical focus in [24], and they have designed various algorithms for that.

FA scheme [24]. The FA scheme has several different ways of finding $\mathbb{C}$. In our later comparison, we will actually consider an enhanced FA scheme: We will exhaustively consider all possibilities for $\mathbb{C}$, and use the best one. Obviously doing so will only make their results better.

After finding $\mathbb{C}$ and adding those critical nodes into the committee, the FA scheme selects the remaining committee members by following the canonical design. Specifically, let k_1 be the number of nodes in $\mathbb{C}$, and S_1 be their total stake. Let $k_2 = k - k_1$ and $S_2 = 1 - S_1$:

- Variable-size version: Each node $u \notin \mathbb{C}$ acts as s_u/τ *virtual nodes*. Each virtual node independently gets into the committee with the same probability p, where $\frac{S_2}{\tau} \cdot p = k_2$. Namely, doing so will in expectation select k_2 committee members from non-critical nodes. Each such committee member will have weight $\frac{S_2}{y}$, where y is the total number of such committee members.
- Fixed-size version: To select a committee member, each node $u \notin \mathbb{C}$ is chosen with probability $\frac{s_u}{S_2}$. This is repeated independently for k_2 times, giving k_2 committee members each with weight $\frac{S_2}{k_2}$. (If a node is selected $x > 1$ times, its weight is $x \cdot \frac{S_2}{k_2}$).

TPL scheme [45]. The TPL scheme follows the enhanced FA scheme described above, except for the following. The TPL scheme finds the best $\mathbb{C}$ and two multiplier α and β (e.g., $\alpha = 1.1$ and $\beta = 0.9$). It then constructs the committee in the same way as the FA scheme. Next, the weight of each critical (non-critical) node in the committee is multiplied by α (β). Doing so essentially makes the weights *non-proportional* [45]. Their work [45] shows that surprisingly, with properly chosen α and β, the error of the committee can be significantly decreased.

4 Semi-deterministic Schemes under Decentralized Stake Distributions

4.1 Overview

It is obvious that when there are no critical nodes, semi-deterministic schemes are just the same as the canonical design. In fact, Gazi et al. [24] already observe this in one of their experiments, for the FA scheme. Furthermore, they explicitly point out [25] that the more centralized the stake distribution is, the more improvement the FA scheme can achieve. They have also quantified the Gini coefficients for all the stake distributions they use, which range from 0.76 to 0.95.

The main focus of [24], however, is not on the effects of decentralization on semi-deterministic schemes. As the first step in this work, we aim to build upon the observations in [24], and obtain a more generalized set of results:

- We will consider a spectrum of decentralized stake distributions, with Gini coefficients ranging from **0.2 to 0.6**. This roughly corresponds to the range of Gini coefficient of all countries/economies (**0.238 to 0.591**) in the world [46].
- We will consider both the TPL scheme [45] and the FA scheme [24].

Our results will show that in many cases, semi-deterministic schemes [24, 45] degrade to the canonical design, when the Gini coefficient is about 0.5 or smaller.

4.2 Reproducing Existing Results: Semi-deterministic Schemes under Highly-centralized Stake Distributions

Our exploration starts by reproducing the improvements achieved by the FA scheme and the TPL scheme as shown in [24, 45]. To do so, we use the publicly-available implementation from [45], for both the FA scheme and the TPL scheme.[1,2] We also use 4 stake distributions that are publicly-available from [45].[3] These 4 stake distributions are real-world stake distributions of 4 major cryptocurrencies[4]: Ethereum [21], Bitcoin [6], BitcoinCash [7], and Dogecoin [8].

Figure 1(a) plots the error achieved by different committee sizes[5] in the FA/TPL scheme, as compared to the canonical design, under

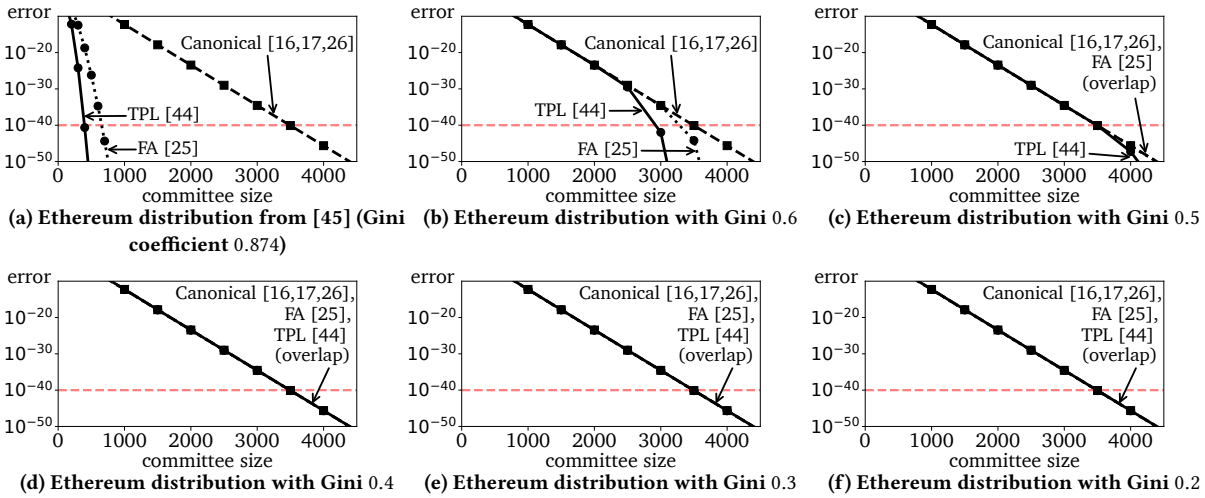
[1] Their implementation of the FA/TPL scheme implicitly assumes that no two nodes have exactly the same stake. For running our experiments, we fixed this issue, and also made the code faster in computing the tail of probability distributions. All these only make their results better.

[2] Their implementation of the TPL scheme has a technicality: For achieving a certain target error under $f = a$ (e.g., $a = 0.2$), the implementation will actually find a committee that works for $f = b$, where b is some value in $[a, a + 0.001]$ (e.g., $b \in [0.2, 0.201]$). Hence the actual committee obtained might be slightly larger than necessary. But we expect the effect to be negligible, since 0.001 is rather small compared to the value of f.

[3] They also make available 2 additional stake distributions, which are however too small for our investigation — the number of nodes in these distributions is sometimes smaller than the committee size, making committee selection irrelevant.

[4] Some of these cryptocurrencies (e.g., Bitcoin) are Proof-of-Work blockchains, instead of Proof-of-Stake. Nevertheless, Bitcoin still has a valid stake distribution (which describes the amount of cryptocurrency held by each node). Hence following [45], our evaluation also considers such stake distributions, which only makes our results more extensive.

[5] Occasionally a node might be selected more than once into the committee, resulting in multiple occurrences of the node in the committee. To allow for a direct comparison between our results and the results in [24, 45], in this paper we always follow the existing practice in [24, 45], and count each and every such occurrence toward the committee size.



(a) Ethereum distribution from [45] (Gini coefficient 0.874**)** **(b) Ethereum distribution with Gini** 0.6 **(c) Ethereum distribution with Gini** 0.5

(d) Ethereum distribution with Gini 0.4 **(e) Ethereum distribution with Gini** 0.3 **(f) Ethereum distribution with Gini** 0.2

Figure 1: FA/TPL scheme vs. the canonical design, under $t = 1/3$, $\epsilon = 0.1$, and various Gini coefficients.

the Ethereum stake distribution and $t = 1/3$. The figure shows that the FA/TPL scheme significantly outperforms the canonical design, which is consistent with the results in [45]. We have also obtained similar results, under the other 3 stake distributions and other t values.

We now quantify how centralized these stake distributions are, using the standard metric of *Gini coefficient* [14]. *Gini coefficient* is widely used to quantify wealth inequality, and ranges from 0 to 1.0. For example, World Bank gives the Gini coefficients for more than 150 countries/economies in the world [46], which range from 0.238 to 0.591. Gini coefficient above 0.4 (respectively, 0.5) is generally viewed as *high inequality* (respectively, *severe inequality*) [29]. Hence, even a small 0.1 difference in Gini coefficient is substantial.

The Gini coefficients of the 4 stake distributions (Ethereum [21], Bitcoin [6], BitcoinCash [7], Dogecoin [8]), are 0.874, 0.656, 0.798, and 0.922 respectively. These are much larger than even the largest country Gini coefficient (i.e., 0.591) in the world, indicating highly-centralized stake distributions.

4.3 Semi-deterministic Schemes under Decentralized Stake Distributions

We next intend to extensively examine how the FA/TPL scheme [24, 45] perform if the stake distributions are less centralized. For systematic evaluation under a spectrum of decentralized stake distributions, we obtain additional stake distributions, by applying a simple capping mechanism to the above 4 stake distributions from [45]. Such capping is motivated by how various blockchains today are trying to make their stake distribution more decentralized: Ethereum's recent Pectra update [19] imposes a hard cap on the stake of a node in its consensus protocol, Cosmos Hub has a similar on-going proposal for imposing such a hard cap [28], while Cardano imposes soft caps that determines the maximum stake for which a node can get reward [30].

Specifically, our capping mechanism imposes a cap c on the stake of a node. A node with stake $x > c$ will split into $\lfloor x/c \rfloor$ nodes each with stake c, and one additional node with stake $x - c \cdot \lfloor x/c \rfloor$. A smaller c makes the distribution more decentralized, with a smaller Gini coefficient. For our experiments, we use various c values to get a spectrum of distributions, with different Gini coefficients.

FA and TPL under decentralized distributions. We now evaluate the FA/TPL scheme under such decentralized distributions. Figure 1 presents the results under the Ethereum stake distribution with Gini coefficients from 0.2 to 0.6, as well as the Ethereum distribution from [45] whose Gini coefficient is 0.874. We consider Gini coefficients from **0.2 to 0.6**, because this is roughly the range of Gini coefficient of all countries/economies (**0.238 to 0.591**) in the world [46], as mentioned earlier. We view these real countries/economies as examples with a more reasonable level of centralization.

In Figure 1, the curve for the canonical design does not change under different Gini coefficients. This is expected, since the error of the canonical design is a function of k, t, and ϵ, and does not depend on the stake distribution.

For the FA/TPL scheme, however, Figure 1 shows that their advantages over the canonical design quickly vanish as Gini coefficient decreases from 0.874. For example, for achieving a target error of 10^{-40}, the FA/TPL schemes need as large committee size as the canonical design, once the Gini coefficient is 0.5 or smaller.

Figure 1 is for the Ethereum distribution, under $t = 1/3$. Figure 6 in Section 8 later will *exhaustively* present results under all 4 stake distributions, and under various t values. The results there are quite consistent with Figure 1. Namely, the gap between the FA/TPL scheme and the canonical design quickly vanishes, as the stake distribution becomes more decentralized. In many cases, when the Gini coefficient is about 0.5 or smaller, the FA/TPL schemes degrade to the canonical design. Note that if it were for a country, Gini coefficient of 0.5 is actually still viewed as *severe inequality* [29]. This means that the degradation already happens, when the distribution is still quite centralized.

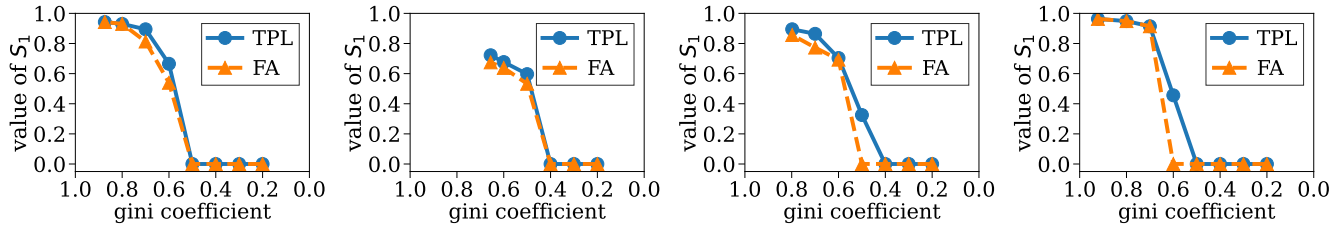


Figure 2: Value of S_1, under various stake distributions and different Gini coefficients, for $t = 1/3$. The 4 sub-figures start with different Gini coefficients (i.e., 0.656 for Bitcoin), since these are the Gini coefficients of the distributions in [45].

4.4 Critical Nodes and Stake

We next aim to better understand why the degradation starts to happen, when the Gini coefficient is still as large as 0.5. To this end, we consider an example committee size of about 3500. This is the size needed by the canonical design to achieve a target error of 10^{-40}, under $t = 1/3$ and the Ethereum stake distribution. We then examine the set $\mathbb{C}$ of critical nodes, when the FA/TPL scheme selects a committee of this size. Let S_1 be the total stake held by these critical nodes.

Figure 2 plots S_1, under various stake distributions and different Gini coefficients. As expected, S_1 drops as the Gini coefficient decreases. But perhaps surprisingly, such drop is far from linear: There is a *sharp drop around Gini coefficient of 0.5*, in all cases. This explains why the semi-deterministic schemes degrade to the canonical design around Gini coefficient of 0.5. Obviously, the advantages of these schemes come from their better handling of the critical nodes, whose total stake is S_1. When S_1 is small or close to zero, their advantages naturally disappear.

Figure 2 is not artifact of capping. Recall that we apply a capping mechanism, in order to obtain stake distributions with various Gini coefficients. One may wonder whether the sharp drops Figure 2 (around Gini coefficient of 0.5) are simply artifacts of such capping. For example, imagine that a node is a critical node iff its stake reaches some threshold γ. Let c be the cap. If $c \geq \gamma$, then a critical node will remain critical after capping. But if $c < \gamma$, then after capping and splitting, no node will be critical node. These seem to imply a sharp transition when c is near γ, which appears to correspond to the sharp drops in Figure 2.

But the above reasoning (incorrectly) assumes that γ is fixed. The value of γ is determined by algorithms in [24, 45], based on the entire stake distribution. When c changes, which affects the stake distribution, γ will also change. In Figure 2(a), the values of γ under different Gini coefficients can differ by as much as about an order of magnitude. Now because γ changes with c, the sharp transition in Figure 2 cannot be simply explained by capping.

5 Our Heavy-coin Committee Selection Scheme (Variable-size Version)

This section proposes our novel *heavy-coin committee selection scheme* (*HC scheme*). Our HC scheme can significantly outperform the canonical design even when the distribution is decentralized. It is driven by clear insights, which lead to a simple design. Also, no crypto construction is needed for our design. This section will present the variable-size version of our HC scheme, while Section 6 will describe the fixed-size version.

Algorithm 1: HeavyCoinCommitteeSel($\mathbb{S}, f, t, k, r$)

1: $(d_1, d_2, \ldots, d_r) \leftarrow$ FindDenominations($\mathbb{S}, f, t, k, r$);
2: each node breaks down its stake into denominations $(d_1, d_2, \ldots, d_r)$, greedily favoring larger denominations;
3: // each node in the system now has stake d_i for some i
4: a node with stake d_i gets into the committee with probability $\frac{d_i}{|\mathbb{S}|}k$, where $|\mathbb{S}|$ is the total stake in $\mathbb{S}$;
5: every node in committee has the same default weight 1;

5.1 Overview of Our Design

Overview. Algorithm 1 gives an overview of our HC scheme. The scheme first finds an appropriate set of *denominations* d_1 through d_r (where $d_1 > d_2 > \cdots > d_r = \tau$ and every d_i is a multiple of τ). Each node then breaks down its stake s into these denominations, where $s = x_1 d_1 + x_2 d_2 + \ldots + x_r d_r$, while greedily favoring larger denominations. The node will then act as x_1 nodes each with stake d_1, x_2 nodes each with stake d_2, and so on.

After this step, each node in the system must have stake d_i for some i. Each node with stake d_i will then get into the committee with probability $\frac{d_i}{|\mathbb{S}|}k$ independently. (We will make sure $\frac{d_i}{|\mathbb{S}|}k \leq 1$, by limiting d_i.) Each node in the committee has weight of 1.

To better understand, note that if we were instead using the canonical design, then such a node with stake d_i would act as $\frac{d_i}{\tau}$ virtual nodes. Each virtual node would get into the committee with probability p, where $\frac{|\mathbb{S}|}{\tau}p = k$. Hence, the key difference here is:

- In our HC scheme: 1 node getting into the committee with probability $\frac{d_i}{|\mathbb{S}|}k$.
- In the canonical design: $\frac{d_i}{\tau}$ *nodes each getting into the committee with probability* $p = \frac{\tau}{|\mathbb{S}|}k$.

Our technical contributions. On the surface, the high-level techniques in our HC scheme are quite simple, with only "minor" differences from the canonical design. Yet, our results later will show significant final improvements. One may wonder why our design has not been proposed before, despite that committee selection has been extensively studied.

This precisely leads to our contributions at the technical level: There are numerous similar ways of doing committee selection,

which differ from each other only in some minor/subtle aspects. Usually these minor differences have no real impact on the final results. But certain subtle differences (as in our HC scheme), interestingly, turn out to help substantially. Hence our contribution is less about these techniques themselves, but more about discovering that certain subtle differences, surprisingly, can bring considerable benefits. We manage to obtain this discovery, by being guided by clear mathematical insights.

Remainder of Section 5. The remainder of Section 5 elaborates our design in a *bottom-up* fashion. Section 5.2 and 5.3 describe our design while assuming that there is a denomination for every possible value, so that a node never needs to break down its stake. Section 5.4 then discusses denominations.

5.2 Our Insight: Heavy-coin Effect

Adversary wins only at the tail of the distribution. We start with some basic facts. Recall that the error of a committee selection scheme is the probability of the adversary winning, and the adversary wins if the fraction of malicious weight in the committee reaches t (e.g., $t = 1/3$). Also recall that $f = t - \epsilon < t$.

Roughly speaking, in expectation, the adversary does *not* win: This is because as long as the committee represents the system without bias (e.g., in the canonical scheme), the *expected* fraction of malicious weight in the committee will be $f < t$, and the adversary does not win. Hence, the adversary can win only at the tail of the distribution.

Now consider the canonical design, and imagine that all the malicious nodes combined correspond to total 2000 virtual nodes, with each virtual node getting into the committee with probability 0.05. Let $\mathbf{Bin}(n, p)$ denote the binomial distribution with n independent trials, where each trial has p success probability. Then the total number (and hence also weight) of malicious nodes in the committee will be $X \sim \mathbf{Bin}(2000, 0.05)$, with $\mathrm{E}[X] = 100$. Assume that the adversary wins iff $X \geq 200$. Figure 3 plots the PDF for X, highlighting the region where the adversary wins.

Heavy-coin effect. We want to reduce the tail region. To this end, we identify a basic mathematical property that will be useful. This is illustrated in Figure 3, which shows that $\mathbf{Bin}(200, 0.5)$ has a smaller tail than $\mathbf{Bin}(2000, 0.05)$. Note that these two distributions have the same mean. Now if the malicious weight in the committee followed $\mathbf{Bin}(200, 0.5)$ instead $\mathbf{Bin}(2000, 0.05)$, then the adversary would have smaller chance of winning.

This illustrates what we call as the *heavy-coin effect*. Intuitively, this effect says:

> To reduce uncertainty, it is better to place a small number of bets each with large winning probability (i.e., coins with "heavy" winning probability) , rather than a large number of bets each with small winning probability.

This effect can be simply shown via the variance of $\mathbf{Bin}(n, p)$: If the mean is fixed at μ, then smaller n will result in smaller variance, since $\mathrm{Var}[\mathbf{Bin}(n, p)] = np(1-p) = \mu(1 - \frac{\mu}{n})$.

Subtlety of the heavy-coin effect. The heavy-coin effect can be somewhat counter-intuitive, because it appears to contradict with the well-known Law of Large Numbers (LLN) [18]: According to

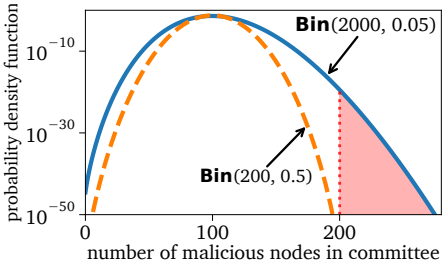


Figure 3: Illustrating the heavy-coin effect.

LLN, $\mathbf{Bin}(n, p)$ should concentrate better when n is larger, instead of when n is smaller. This paradox arises because the contexts are different:

- LLN is about how $\frac{\mathbf{Bin}(n,p)}{n}$ concentrates around p, for fixed p. In such context, LLN tells us that **larger** n leads to better concentration.
- Heavy-coin effect is about how $\mathbf{Bin}(n, p)$ concentrates around $\mu = np$, for fixed μ. In such context, heavy-coin effect tells us that **smaller** n leads to better concentration.

To further illustrate, consider $X \sim \mathbf{Bin}(2000, 0.05)$, $Y \sim \mathbf{Bin}(200, 0.5)$, and $Z \sim \mathbf{Bin}(200, 0.05)$, then:

- LLN says that $\frac{X}{2000}$ concentrates around 0.05 better than $\frac{Z}{200}$.
- Heavy-coin effect says that Y concentrates around 100 better than X.

5.3 Leverage Heavy-coin Effect

Leveraging heavy-coin effect in committee selection is not trivial, and we need to deal with two challenges. We deal with these in this section and Section 5.4.

Naive design does not work. Imagine that the adversary controls total 0.1 malicious stake, which is spread across 3 malicious nodes, each with 0.05, 0.03, and 0.02 stake, respectively. Recall that in the canonical design, each virtual node holds τ stake and gets into the committee with p probability. Then the malicious weight in the committee follows $\mathbf{Bin}(\frac{0.05}{\tau}, p) + \mathbf{Bin}(\frac{0.03}{\tau}, p) + \mathbf{Bin}(\frac{0.02}{\tau}, p)$.

To exploit the heavy-coin effect, one could let each virtual node hold 1000τ stake, instead of τ stake. This makes a virtual node "heavier": There will be fewer virtual nodes, and the "winning" probability for each virtual node becomes larger. But this unfortunately would prevent nodes with less than 1000τ stake from participating.

Allowing different winning probabilities. We deal with this by allowing different nodes to toss coins with different winning probabilities. Specifically in our HC design, a node with s stake conceptually just corresponds to a *single* virtual node, and this single virtual node gets into the committee with probability $\frac{s}{\tau}p$.[6] The malicious weight in the committee (in our earlier example) then follows the distribution $\mathbf{Bin}(1, \frac{0.05}{\tau}p) + \mathbf{Bin}(1, \frac{0.03}{\tau}p) + \mathbf{Bin}(1, \frac{0.02}{\tau}p)$.

Now since this is a sum of 3 binomial distributions, instead of a single binomial distribution, it is not immediately obvious whether the heavy-coin effect still applies. Fortunately, the additivity of variance (for independent random variables) enables us to reason

[6] The $\frac{s}{\tau}p$ term here is the same as the $\frac{d_i}{|\mathbb{S}|}k$ term at Line 4 of Algorithm 1, since here $\frac{|\mathbb{S}|}{\tau}p = k$ and $d_i = s$.

Algorithm 2: FindDenominations($\mathbb{S}, f, t, k, r$)

$(d_1, d_2, \ldots, d_r) \leftarrow (\tau, \tau, \ldots, \tau)$;
for $i = 1, 2, \ldots, r-1$: // search for value of d_i's
  let $\mathcal{D}_i$ be the set of candidate values for d_i in its *valid range*;
  // d_i's *valid range* is $[\tau, \min(\frac{|\mathbb{S}|}{k}, 1)]$ for $i = 1$, and $[\tau, d_{i-1})$ for $2 \le i \le r-1$;
  $d_i \leftarrow \arg\min_{d_i \in \mathcal{D}_i} \text{VarZ}\big((d_1, d_2, \ldots, d_r), \mathbb{S}, f, t, k\big)$
return $(d_1, d_2, \ldots, d_r)$;

Procedure VarZ($(d_1, d_2, \ldots, d_r), \mathbb{S}, f, t, k$):
// all nodes should have already broken down their stake into denominations, so that each node now has stake d_i for some i
consider the adversary that corrupts nodes one-by-one in increasing order of stake, until its budget f is used up;
$\text{Var}_{\text{total}} \leftarrow \sum_{u \text{ is corrupted}} \frac{s_u}{|\mathbb{S}|} k \cdot (1 - \frac{s_u}{|\mathbb{S}|} k) + \sum_{u \text{ is honest}} \frac{s_u}{|\mathbb{S}|} k \cdot (1 - \frac{s_u}{|\mathbb{S}|} k)$; // this is $\text{Var}[M] + \text{Var}[H]$
$\text{Var}_M \leftarrow \sum_{u \text{ is corrupted}} \frac{s_u}{|\mathbb{S}|} k \cdot (1 - \frac{s_u}{|\mathbb{S}|} k)$; // this is $\text{Var}[M]$
return $(1-2t) \cdot \text{Var}_M + t^2 \cdot \text{Var}_{\text{total}}$; // by Equation (2)

about each of the 3 terms separately, while applying the heavy-coin effect to each term individually. This eventually makes the variance in our design smaller, as compared to the canonical design:

$$
\begin{aligned}
& \text{Var}[\mathbf{Bin}(1, \frac{0.05}{\tau}p) + \mathbf{Bin}(1, \frac{0.03}{\tau}p) + \mathbf{Bin}(1, \frac{0.02}{\tau}p)] \\
= \; & \text{Var}[\mathbf{Bin}(1, \frac{0.05}{\tau}p)] + \text{Var}[\mathbf{Bin}(1, \frac{0.03}{\tau}p)] + \text{Var}[\mathbf{Bin}(1, \frac{0.02}{\tau}p)] \\
& \text{(by additivity of variance)} \\
< \; & \text{Var}[\mathbf{Bin}(\frac{0.05}{\tau}, p)] + \text{Var}[\mathbf{Bin}(\frac{0.03}{\tau}, p)] + \text{Var}[\mathbf{Bin}(\frac{0.02}{\tau}, p)] \\
& \text{(by heavy-coin effect)} \\
= \; & \text{Var}[\mathbf{Bin}(\frac{0.05}{\tau}, p) + \mathbf{Bin}(\frac{0.03}{\tau}, p) + \mathbf{Bin}(\frac{0.02}{\tau}, p)] \\
& \text{(by additivity of variance)}
\end{aligned}
$$

5.4 Use Denominations to Facilitate Finding Worst-Case Adversary

Heavy-coin effect makes it hard to find worst-case adversary. In the canonical design, all virtual nodes are *symmetric*. Hence it is trivial to find the worst-case adversary: Regardless of the adversary's choice on which nodes to corrupt, the adversary always eventually controls at most $\frac{f}{\tau}$ malicious virtual nodes.

Now in our HC scheme, a node u will get into the committee with $\frac{s_u}{\tau}p$ probability. This causes different nodes (with different s_u) to be *non-symmetric*, making it challenging to find the worst-case adversary. For example, should the adversary compromise three nodes with 0.01, 0.03, and 0.06 stake each, or spend the same budget to compromise two nodes with 0.02 and 0.08 stake each?

Not knowing the worst-case adversary for our HC scheme would prevent us from reasoning about error, or determining the committee size needed to achieve a certain target error.

Using denominations. To overcome this, we adopt the idea of currency denominations. We will find a small set of stake values $(d_1, d_2, \ldots, d_r)$ as *denominations*, where $d_1 > d_2 > \ldots > d_r = \tau$ and each d_i is a multiple of τ. A node breaks down its stake s into these denominations, greedily favoring larger ones. Imagine that $s = x_1 d_1 + x_2 d_2 + \ldots + x_r d_r$. The node will then act as x_1 nodes with stake d_1, x_2 nodes with stake d_2, and so on, in committee selection.

With the denomination idea, nodes in the systems will now only have r possible stake values. For any given $i \in [1, r]$, obviously all nodes with stake d_i are *symmetric*: If the adversary corrupts y_i of them, it does not matter which subset of them are corrupted. Hence the adversary can be uniquely specified by the vector $\langle y_1, y_2, \ldots, y_r \rangle$. As long as r is small, it is feasible to exhaustively enumerate all adversaries. Appendix B explains in detail how this is done in our implementation.

Design of denominations. We have not yet explained how to determine the denominations:

- What should be the value of r?
- Given r, what should be the values for d_1 through d_r?

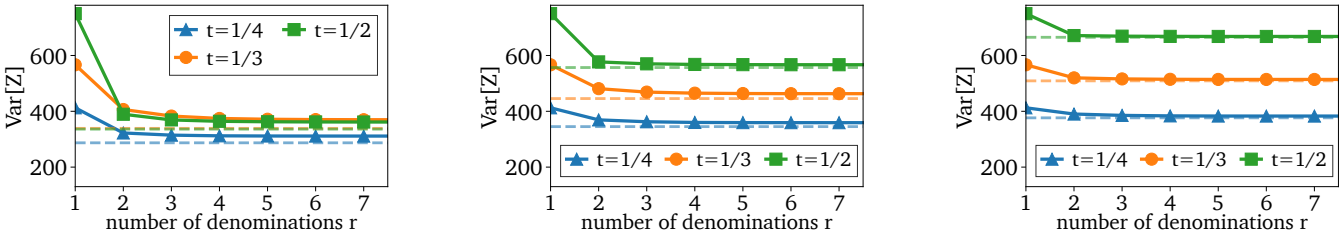
The value r corresponds to a trade-off between computational overhead and effectiveness of our approach: A larger r increases computational overhead, for searching for the worst-case adversary. But it also helps to maximize the benefits of our design, because breaking down a node's stake weakens the heavy-coin effect. A larger r gives a node more options, and hence avoids excessive break down. Our later experiments will use a rather small r value (e.g., 3), which is already sufficient to achieve our improvements over existing designs.

For values of the d_i's, there is also a trade-off. To understand, imagine $r = 2$ and $d_2 = \tau$, so that we only need to determine d_1. A larger d_1 makes the heavy-coin effect stronger, since it conceptually makes the coin "heavier". At the same time, a larger d_1 implies that fewer nodes can benefit from that heavy-coin effect, since a node needs to have at least d_1 stake to benefit.

In the next, we first explain how to determine d_1 through d_r for given r, and then discuss how to choose r.

Determine values for the d_i's. We use a simple greedy algorithm (see Algorithm 2) for determining d_1 through d_r. Note that the values of d_i's have no effect on the correctness of our committee design. They only affect committee size: If the denominations are not chosen in the best way, then our committee may be larger than necessary.

Algorithm 2 first finds d_1, while assuming $d_2 = d_3 = \ldots d_r = \tau$. It searches for d_1 by enumerating all candidate d_1 values (with a



(a) Gini coefficient 0.6 **(b) Gini coefficient** 0.4 **(c) Gini coefficient** 0.2

Figure 4: Final Var$[Z]$, as a function of r. This is under the Ethereum stake distribution, with $t = 1/3$, $\epsilon = 0.1$, and an example committee size of 3000**. The dashed horizontal lines correspond to lower bounds on Var$[Z]$.**

certain granularity[7]), as in Line 13 of Algorithm 2. For each candidate value, the algorithm calculates the resulting variance of a certain quantity Z (see later), under the worst-case adversary. It then chooses the d_1 value that minimizes Var$[Z]$. After that, it proceeds to $d_2, d_3, \ldots$, until d_{r-1}. Finally, d_r is always τ.

Why minimizing Var$[Z]$**.** Ideally, we hope to choose denominations that can minimize the final error. Recall from Equation 1 that the error is $\Pr[\frac{M}{M+H} \geq t]$, or equivalently, $\Pr[Z \geq 0]$ where $Z = (1-t)M - tH$. Recall that M and H are the malicious and honest weight in the committee, respectively.

Directly calculating $\Pr[Z \geq 0]$ for all possible denominations is neither efficient nor needed: After all, even if we do not choose the best denominations, it will only affect performance and not security. Hence instead we use Var$[Z]$ to guide our heuristic search: The intuition is that, as explained earlier, the adversary only wins at the tail, and smaller Var$[Z]$ tends to correspond to smaller tail probability for Z.

We will need to calculate Var$[Z]$ under the adversary that maximizes Var$[Z]$.[8] We claim that this adversary must be the one that corrupts the nodes one-by-one in increasing order of node stake, until it uses up its budget of f.

To see why our claim holds, note that:

$$\begin{aligned} \text{Var}[Z] &= \text{Var}[(1-t)M - tH] \\ &= (1-t)^2\text{Var}[M] + t^2\text{Var}[H] \\ &= (1-2t)\text{Var}[M] + t^2(\text{Var}[M] + \text{Var}[H]) \end{aligned} \quad (2)$$

Lemma 2 in Appendix C will prove that $(\text{Var}[M] + \text{Var}[H])$ is independent of the adversary. Together with the fact that $(1-2t)$ is non-negative,[9] this implies that Var$[Z]$ is maximized iff Var$[M]$ is maximized. Next, Lemma 3 in Appendix C will prove the earlier adversary maximizes Var$[M]$. Hence, that adversary also maximizes Var$[Z]$.

How many denominations? Our experiments in Section 8 later will use rather small r value (e.g., 3), and the results there show that doing so already enables our design to outperform existing ones.

To better understand why a small r is sufficient, we now examine how r affects Var$[Z]$. Intuitively, the smaller Var$[Z]$ is, the more benefit our design will be able to bring. In addition, we will compare against a *lower bound* on Var$[Z]$. This lower bound conceptually corresponds to $r = \infty$, where there is a denomination for every possible value, so that a node never needs to break down its stake.[10]

Figure 4 plots the final Var$[Z]$ under different r values (and the corresponding d_1 through d_r values). The figure shows that most reduction for Var$[Z]$ occurs when r increases to 2 or 3. Beyond that, the benefit is limited. This explains why using a rather small r value will already lead to good final results.

Figure 4 is not artifact of capping. Recall that we apply a capping mechanism, for obtaining stake distributions with various Gini coefficients. One may wonder whether the small number of denominations needed in Figure 4 is simply an artifact of such capping. Namely, the capping mechanism (with a cap of c) may result in many nodes with stake c. Using just a single denomination of c can already perfectly capture all those nodes.

It turns out that Figure 4 is not an artifact of such capping. Consider Figure 4(a) as an example. Here in the original distribution (before capping), only about 5% of the nodes have stake at least c, where c is the capping threshold. Even after capping/splitting, those nodes comprise only 25% of the distribution. This means that the vast majority (specifically 75%) of the nodes in Figure 4(a) were not affected by capping at all. Capping hence cannot explain why a small number of denominations suffices for those 75% of the nodes, as we discover in Figure 4(a).

6 Our Heavy-coin Committee Selection Scheme (Fixed-size Version)

The previous section has presented the variable-size version of our HC scheme. This section explains how to extend to the fixed-size version.

From variable-size to fixed-size. Recall that in the variable-size version, after applying the denominations, each node will have stake d_i for some $i \in [1, r]$. Let k_i be the expected number of committee members selected from nodes with stake d_i. Obviously, the expected size of the final committee will be $\sum k_i$.

Now to make the size of committee *fixed* (instead of a random variable), we will select exactly k_i committee members[11] from all

[7] Our implementation enumerates 1000 equally-spaced candidate d_i values in the possible range for d_i, for each i.

[8] One may wonder why we do not consider the adversary that maximizes the error. The reason is that, as explained earlier, we are using Var$[Z]$ to guide our search for the denominations. After the denominations are determined, for computing the final error of the committee, we will consider the adversary that maximizes the error.

[9] This is true for all t values (i.e., 1/2, 1/3, 1/4) that we care about.

[10] Our design always requires $\frac{d_1}{|\mathbb{S}|}k \leq 1$. Hence if a node has stake larger than this thresold of $|\mathbb{S}|/k$, it will still need to break down its stake.

[11] In case k_i is not an integer, one can simply round k_i to the nearest integer. Such rounding has no material effect in practice, given that committee sizes are on the order of hundreds or more.

nodes with stake d_i. To do so, conceptually, we follow our variable-size design, while *conditioning upon the number of committee members selected being exactly k_i*. Namely, we first let each node get into the committee with a certain probability. We then check the number of committee members elected. If it is exactly k_i, we are done. Otherwise we annul the election, and repeat.

Implementation issue. The above conceptual process is not efficient to implement, due to the retrying needed. But interestingly, the result of this process is exactly the same as sampling k_i nodes *without replacement* (from all nodes with stake d_i). The reason is that the above process always eventually returns a set (of committee members) of size exactly k_i. Since all nodes with stake d_i are symmetric with each other, all subsets of size k_i are equally likely. Hence the distribution for the final outcome is the same as sampling without replacement. This means for implementation, we just need to do sample without replacement.

7 Further Optimizations for Our HC Scheme

This section presents two further optimizations for our HC scheme. Note that these optimizations are not the central contribution of our work: Our primary contribution is about exploiting the heavy-coin effect, as elaborated in the previous two sections. The heavy-coin effect, as well as how we exploit it, is orthogonal to the techniques in the optimizations in this section.

7.1 Biased Weights

Review of biased weights in [45]. Recall from Section 3 that the TPL scheme [45] finds two multipliers α and β, for example $\alpha = 1.1$ and $\beta = 0.9$, such that the weight of each critical (non-critical) node in the committee is multiplied by α (β). Doing so essentially makes the weights *non-proportional* [45] and *biased*.

Such biased weights help, because if the adversary corrupts a critical node by spending a certain budget, then the reward it gets in terms of corrupted weight in the committee is *fixed*. Such fixed reward is bad for the adversary, since the adversary only wins at the tail of the distribution, and a fixed reward essentially makes the tail smaller. Because of this, the TPL scheme exploits such *aversion* of the adversary [45], by giving the critical nodes disproportionally larger weight.

Note that such biased weights do not result in unfairness, when it comes to incentives: As shown in [45], incentive and security can be decoupled. Namely, the biased weights are used only when the parties, for example, cast votes in a byzantine agreement protocol. The system can easily use another different set of unbiased weights, for giving out incentives.

Biased weight in our design: Intuitions. The biased weights in [45] no longer apply when there are no critical nodes. We will, however, generalize this high-level idea to our design, even when there are no critical nodes.

To see why this is possible, recall that in our design, there are denominations $d_1 > d_2 > \ldots > d_r$. Assume that $d_1 = 0.1$ and $d_2 = 0.02$, and that there are no critical nodes. Imagine two options for the adversary: i) corrupting 2 nodes each with stake 0.1, or ii) corrupting 10 nodes each with stake 0.02. By the heavy-coin effect, under the first option, the malicious weight in the committee has smaller variance. For the same reason as earlier, the adversary will want to avert this first option. In turn, we can exploit such aversion, by giving larger weight to the nodes with 0.1 stake.

Algorithm 3: FindWeights($\mathbb{S}, f, t, k, r$)

let $(d_1, d_2, \ldots, d_r) \leftarrow$ FindDenominations($\mathbb{S}, f, t, k, r$);
let $(w_1, w_2, \ldots, w_r) \leftarrow (1, 1, \ldots, 1)$;
for $i = 1, 2, \ldots, r$: // determining w_i
    let $\mathcal{W}_i$ be the set of candidate values for w_i;
    let error(x) be the error of our HC scheme with weights $(w_1, w_2, \ldots, w_r)$ where $w_i = x$;
    $w_i \leftarrow \arg\min_{x \in \mathcal{W}_i} \text{error}(x)$;
return $(w_1, w_2, \ldots, w_r)$;

Biased weight in our design: Details. Driven by such intuition, in our design, if a node in the committee has stake d_i, then it will have a weight of w_i, where $w_1 \geq w_2 \geq \ldots \geq w_r$. We use a simple greedy algorithm (see Algorithm 3) to search for the w_i's.[12]

Algorithm 3 is rather basic. Better (and more sophisticated) approaches may exist. But we choose our simple design, which already suffices for obtaining our final improvements.

7.2 Exploiting Critical Nodes when They Exist

Overview. While our focus is on decentralized distributions, we want our design to also work well for highly-centralized distributions that have critical nodes. We can easily optimize to do so, since our design is largely orthogonal to how the critical nodes are exploited in [24, 45]. Intuitively, when there are critical nodes, we can let those schemes [24, 45] handle the critical nodes, and then apply our design on the remaining (non-critical) nodes.

Details. In this optimization, we will first use the TPL scheme [45] to select a committee $\mathbb{T}$. We then keep the critical nodes (and their weights) in $\mathbb{T}$ unchanged. Let x (y) be the total number (weight) of the non-critical nodes in $\mathbb{T}$. We will remove all these x nodes from $\mathbb{T}$, and then use our HC scheme to re-elect x non-critical nodes (with the same total weight y) to join $\mathbb{T}$.

To re-elect x committee members from non-critical nodes, we invoke our Algorithm 1 through Algorithm 3 with $k = x$ and with $\mathbb{S}$ being the stake distribution of all the non-critical nodes. Let y' be the total weight, as determined by our algorithms, of these newly-elected non-critical nodes. We then scale each of their weights by a factor of $\frac{y}{y'}$, so that the total weight of these new committee members becomes y.

Finally, we observe that occasionally, even when there are critical nodes, using the TPL scheme to exploit the critical nodes still gives inferior results than using our HC scheme directly on all (critical and non-critical) nodes. Hence when applying this optimization, we will actually find two committees (one using the optimization, and another one not using), and then adopt the better one.

[12] In our experiments, following [45], we search in the range of $[0.5, 2]$ for each w_i. We use a granularity of 0.1 in the search.

8 Experimental Evaluation

We have fully implemented our HC committee selection scheme, with publicly-available source code (see Appendix A). We use this implementation for our experimental evaluation.

8.1 Experimental Methodology

In order to do a head-on comparison with the most relevant prior works [24, 45], we use the same experimental methodology as previously elaborated in Section 4: As a brief reminder, we use the publicly-available implementation from [45] for the FA scheme and TPL scheme. We also use 4 publicly-available stake distributions from [45], which are the real-world stake distributions of Ethereum [21], Bitcoin [6], BitcoinCash [7], and Dogecoin [8]. Based on these, we obtain (see Section 4.3) a spectrum of stake distributions, with different Gini coefficients.

Our experiments will consider various commonly-used t values: $t = 1/2$ as required by synchronous byzantine agreement protocols [1, 3], $t = 1/3$ as required by asynchronous ones [26, 43, 47], and $t = 1/4$ as required by order-fair byzantine agreement protocols [36, 41]. Recall that $f = t - \epsilon$, and we use $\epsilon = 0.1$ as in prior work [24].

Finally, recall that a committee selection scheme has two versions: variable-size and fixed-size. Our results here will be for the variable-size version. The results for the fixed-size version are rather similar, which we defer to Appendix D.

8.2 Experimental Results

Comparing our HC scheme against other designs. Figure 5 provides a head-on comparison of 4 committee selection schemes: the canonical design [15, 16, 26], the FA scheme [24], the TPL scheme [45], and our HC scheme. These results are under the Ethereum stake distribution and $t = 1/3$. As in Section 4, we consider Gini coefficients from 0.2 to 0.6 — this is roughly the range of Gini coefficients of countries/economies (0.238 to 0.591) in the world [46].

Figure 5 shows that our HC scheme always outperforms all the existing schemes: With the same committee size, our HC scheme can achieve up to about 20 orders of magnitude smaller error than the FA/TPL scheme.

Reduction in committee size. To further analyze our results, we consider an example target error of 10^{-40}. Such 10^{-40} error is roughly what is achieved by the committee selection mechanism in the production deployment [9] of Algorand. The same work [9] also explains that such a small error is necessary, because committee selection is done repeatedly and hence error accumulation needs to be taken into account.

We ask what committee size is needed for achieving this target error of 10^{-40}, under various schemes. We then quantify the reduction in the committee size achieved by the FA/TPL/HC scheme, as compared to the canonical design. We will exhaustively consider all 4 stake distributions, as well as all values for t. Figure 6 summarizes such reduction in committee size, for all the settings.

Figure 6(a)[13] shows that under $t = 1/3$ and under the Ethereum distribution with Gini coefficient 0.6, our HC scheme can reduce committee size by about 35% as compared to the canonical scheme, while the FA/TPL scheme can only achieve a reduction of about 15% or less. Perhaps more importantly, for Gini coefficient 0.5, the FA/TPL scheme gives almost zero benefit over the canonical design, while our HC scheme still achieves about 25% reduction. Finally, the reduction achieved by our HC scheme becomes smaller as Gini coefficient decreases from 0.6 to 0.2. This is expected: as the stake distribution gets closer to uniform, the heavy-coin effect becomes less significant. Nevertheless, our HC scheme still always outperforms the FA/TPL scheme — the FA/TPL scheme degrade to the canonical design for all Gini coefficients 0.5 or below.

Figure 6(b)-(l) present the results under other stake distributions as well as other t values. The same general observations can be made from these results, as in Figure 6(a). In particular, in many cases, the FA/TPL scheme gives minimal/zero benefits over the canonical design, while our HC scheme can still achieve to up to about 30% reduction in committee size.

Highly-centralized stake distributions. We have shown that our HC scheme provides significant benefits under decentralized stake distributions, which are the focus of this work. We next aim to show that when the stake distribution is not decentralized (i.e., highly-centralized), our scheme still retains the benefits of the FA/TPL scheme.

To this end, we evaluate our HC scheme under stake distributions with large Gini coefficients. We defer these results to Appendix E. The results there show that our HC scheme always retains the benefits of the FA/TPL scheme. This is expected, since by Section 7.2, our HC scheme is able to properly leverage critical nodes, as the FA/TPL scheme does.

9 Additional Related Works

The FA scheme [24] and the TPL scheme [45] are the most closely related works to this paper. We have already reviewed them in Section 3, and provided a thorough comparison in Section 8. In the following, we discuss additional related works beyond [24, 45].

Preliminary design. Kniep et al. [38] describe a preliminary design for committee selection — it partitions the stake in the system into equal-sized bins, and then selects one committee member from each bin. Their work is preliminary, because it does not provide a complete solution on how to select a committee: Specifically, it is unclear how to choose the committee size needed for achieving a certain security level. Their results are only for a few specific adversaries, rather than for the worst-case adversary. It is unclear how to reason about the worst-case adversary, given their use of bins.

Reputation-based design. Jameel et al. [34] present a committee selection scheme based on nodes' *reputation*, where a node's reputation is computed from its past behavior. Due to the heuristic nature of such reputation, it is unclear how to reason about the security of their design.

[13] Figure 6(a) is based on the same results as in Figure 5, but presented from a different perspective.

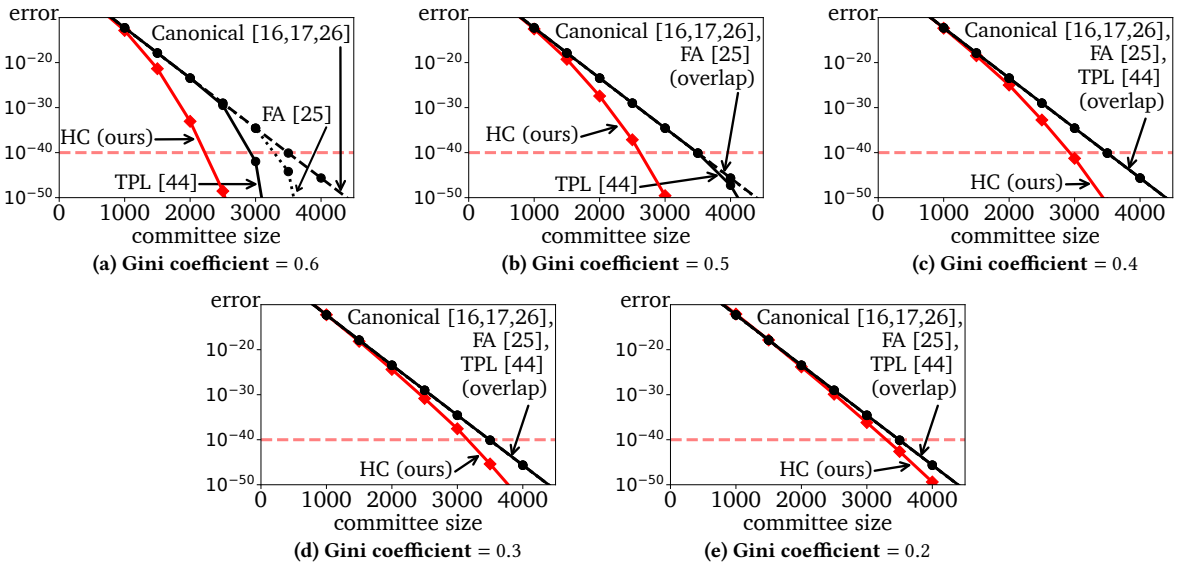


Figure 5: Comparing our HC scheme against existing designs, for Ethereum distribution and $t = 1/3$. We consider Gini coefficients from 0.2 to 0.6 — this is roughly the range of Gini coefficients of countries/economies (0.238 to 0.591) in the world [46].

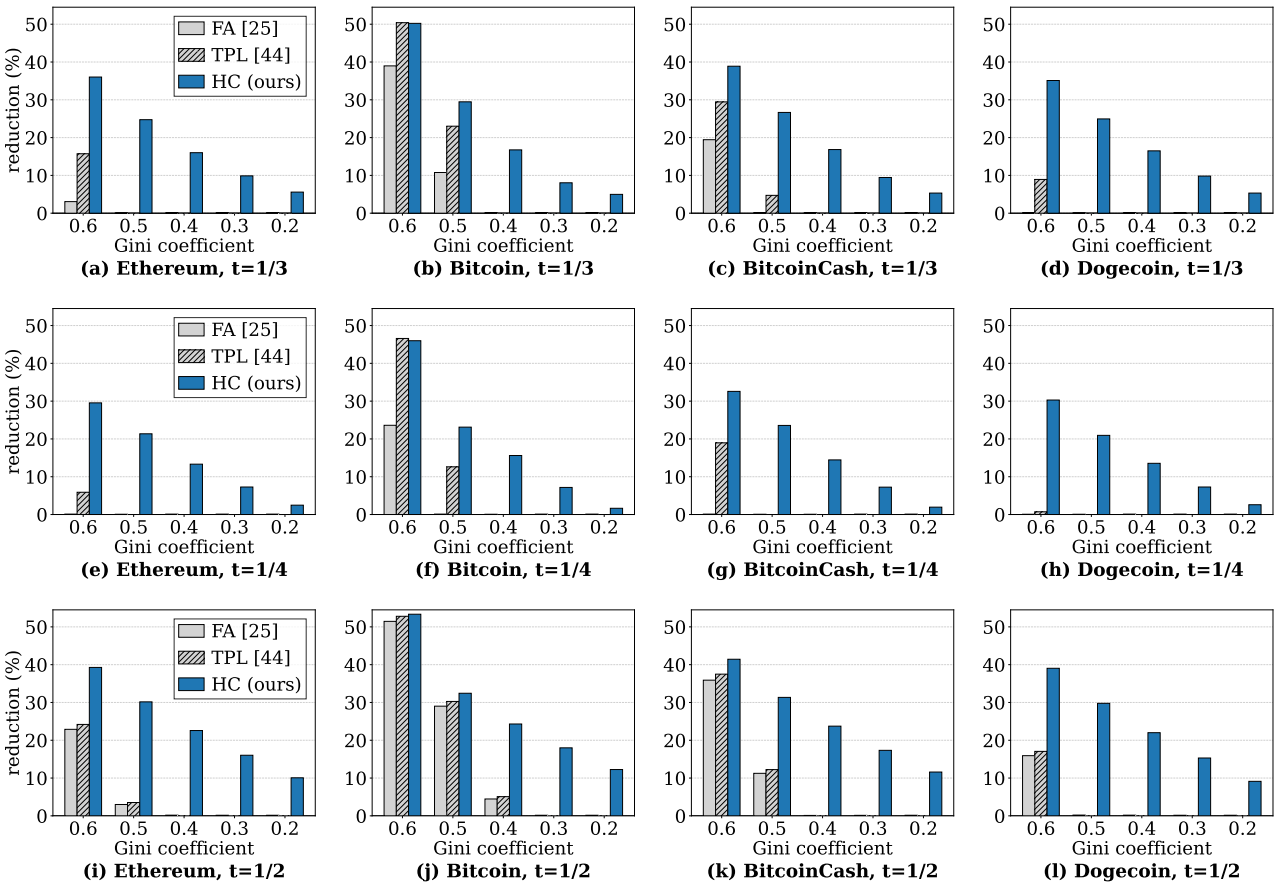


Figure 6: Reduction in committee size achieved by the FA/TPL/HC scheme, as compared to the canonical design. The reduction is presented in terms of percentage. We consider Gini coefficients from 0.2 to 0.6, for the same reason as in Figure 5.

Anonymity in committee selection. In some cases, we want the committee members to remain anonymous, so that they cannot be easily targeted by, for example, DoS attacks. There are various ways [16, 26, 48] to achieve this. One example approach is to use Verifiable Random Functions (VRFs) [16, 26]. Another approach is to use zero-knowledge proof and threshold signatures [48]. Achieving such anonymity is orthogonal to our work.

Committee selection in broader context. Committee selection is also relevant in contexts beyond blockchains. For example, in social choice theory, committee selection focuses on aggregating voter preferences to select a representative subset of "candidates". In such context, [12, 35] have investigated various fairness issues via a game-theoretic framework. As another example, committees are also used in distributed learning [11], to mask the disruption from malicious participants. These works have no direct relation to our work, since their contexts are quite different from ours.

10 Conclusion

This work focuses on the fundamental problem of committee selection, in the context of blockchains. The state-of-the-art FA/TPL designs [24, 45] for committee selection are meaningful only under highly-centralized stake distributions. If the stake distribution is more decentralized, then they quickly degrade to the canonical design. We hence propose our novel heavy-coin (HC) committee selection scheme. Our implementation and experimental results show that for more decentralized distributions, our HC scheme is the first and only design that can beat the classic canonical design. For other distributions, our HC scheme retains the benefits of the best state-of-the-art design.

References

[1] Ittai Abraham, Srinivas Devadas, Danny Dolev, Kartik Nayak, and Ling Ren. 2019. Synchronous Byzantine Agreement with Expected $O(1)$ Rounds, Expected $O(n^2)$ Communication, and Optimal Resilience. In *Financial Cryptography and Data Security*.

[2] Ittai Abraham, Dahlia Malkhi, Kartik Nayak, Ling Ren, and Alexander Spiegelman. 2017. Solida: A Blockchain Protocol Based on Reconfigurable Byzantine Consensus. In *International Conference on Principles of Distributed Systems*.

[3] Ittai Abraham, Dahlia Malkhi, Kartik Nayak, Ling Ren, and Maofan Yin. 2020. Sync HotStuff: Simple and Practical Synchronous State Machine Replication. In *IEEE S&P*.

[4] Pierre-Louis Aublin, Sonia Ben Mokhtar, and Vivien Quéma. 2013. RBFT: Redundant Byzantine Fault Tolerance. In *ICDCS*.

[5] Bitcoin.org. 2026. Bitcoin - Open source P2P money. https://bitcoin.org/.

[6] BitInfoCharts. 2024. *Bitcoin stake distributions.* Data is from the following webpages: https://bitinfocharts.com/top-100-richest-bitcoin-addresses-1.html, https://bitinfocharts.com/top-100-richest-bitcoin-addresses-2.html, . . ., https://bitinfocharts.com/top-100-richest-bitcoin-addresses-100.html.

[7] BitInfoCharts. 2024. *BitcoinCash stake distributions.* Data is from the following webpages: https://bitinfocharts.com/top-100-richest-bitcoin%20cash-addresses-1.html, https://bitinfocharts.com/top-100-richest-bitcoin%20cash-addresses-2.html, . . ., https://bitinfocharts.com/top-100-richest-bitcoin%20cash-addresses-100.html.

[8] BitInfoCharts. 2024. *Dogecoin stake distributions.* Data is from the following webpages: https://bitinfocharts.com/top-100-richest-dogecoin-addresses-1.html, https://bitinfocharts.com/top-100-richest-dogecoin-addresses-2.html, . . ., https://bitinfocharts.com/top-100-richest-dogecoin-addresses-100.html.

[9] Erica Blum, Derek Leung, Julian Loss, Jonathan Katz, and Tal Rabin. 2023. Analyzing the Real-World Security of the Algorand Blockchain. In *CCS*.

[10] Miguel Castro and Barbara Liskov. 1999. Practical Byzantine Fault Tolerance. In *Symposium on Operating Systems Design and Implementation*.

[11] Chunjiang Che, Xiaoli Li, Chuan Chen, Xiaoyu He, and Zibin Zheng. 2022. A Decentralized Federated Learning Framework via Committee Mechanism With Convergence Guarantee. *IEEE Transactions on Parallel and Distributed Systems* (2022), 4783–4800.

[12] Yu Cheng, Zhihao Jiang, Kamesh Munagala, and Kangning Wang. 2020. Group Fairness in Committee Selection. *ACM Transactions on Economics and Computation (TEAC)* (2020), 1–18.

[13] Coinbase. 2026. *Ethereum USD conversion.* https://www.coinbase.com/en-sg/converter/eth/usd.

[14] Frank Cowell. 2011. *Measuring Inequality.* Oxford University Press.

[15] Phil Daian, Rafael Pass, and Elaine Shi. 2019. Snow white: Robustly reconfigurable consensus and applications to provably secure proof of stake. In *FC*.

[16] Bernardo David, Peter Gaži, Aggelos Kiayias, and Alexander Russell. 2018. Ouroboros praos: An adaptively-secure, semi-synchronous proof-of-stake blockchain. In *EUROCRYPT*.

[17] Bernardo David, Bernardo Magri, Christian Matt, Jesper Buus Nielsen, and Daniel Tschudi. 2022. GearBox: Optimal-size Shard Committees by Leveraging the Safety-Liveness Dichotomy. In *CCS*.

[18] Rick Durrett. 2019. *Probability: Theory and Examples.* Cambridge University Press.

[19] Ethereum. 2025. Pectra. https://ethereum.org/roadmap/pectra/.

[20] Ethereum.org. 2026. Ethereum. https://ethereum.org/.

[21] Etherscan. 2024. *Ethereum stake distribution.* https://etherscan.io/accounts.

[22] Etherscan. 2026. *Ethereum Average Gas Price Chart.* https://etherscan.io/chart/gasprice.

[23] Etherscan. 2026. *Ethereum Daily Gas Used Chart.* https://etherscan.io/chart/gasused.

[24] Peter Gazi, Aggelos Kiayias, and Alexander Russell. 2023. Fait Accompli Committee Selection: Improving the Size-Security Tradeoff of Stake-Based Committees. In *CCS*.

[25] Peter Gazi, Aggelos Kiayias, and Alexander Russell. 2023. *Fait Accompli Committee Selection: Improving the Size-Security Tradeoff of Stake-Based Committees.* Technical Report. https://eprint.iacr.org/2023/1273

[26] Yossi Gilad, Rotem Hemo, Silvio Micali, Georgios Vlachos, and Nickolai Zeldovich. 2017. Algorand: Scaling byzantine agreements for cryptocurrencies. In *SOSP*.

[27] Guy Golan Gueta, Ittai Abraham, Shelly Grossman, Dahlia Malkhi, Benny Pinkas, Michael Reiter, Dragos-Adrian Seredinschi, Orr Tamir, and Alin Tomescu. 2019. SBFT: A Scalable and Decentralized Trust Infrastructure. In *DSN*.

[28] Govmos. 2025. [LAST CALL] CHIPs signaling phase: Validator Vote Power Cap. https://forum.cosmos.network/t/last-call-chips-signaling-phase-validator-vote-power-cap/16204.

[29] Cameron Nadim Haddad, Daniel Gerszon Mahler, Carolina Diaz-Bonilla, Ruth Hill, Christoph Lakner, and Gabriel Lara Ibarra. 2024. *The World Bank's New Inequality Indicator: The Number of Countries with High Inequality.* Technical Report. https://documents1.worldbank.org/curated/en/099549506102441825/pdf/IDU1bd155bac16d78143af188331f87564a9d6c8.pdf

[30] Rebecca Hopwood. 2022. Staking parameters and network optimization – where next for k and min fee? https://www.iog.io/news/staking-parameters-and-network-optimization-where-next-for-k-and-min-fee.

[31] Ruomu Hou and Haifeng Yu. 2023. Optimistic Fast Confirmation While Tolerating Malicious Majority in Blockchains. In *IEEE S&P*.

[32] Ruomu Hou, Haifeng Yu, and Prateek Saxena. 2022. Using throughput-centric byzantine broadcast to tolerate malicious majority in blockchains. In *IEEE S&P*.

[33] Ruomu Hou, Haifeng Yu, and Yucheng Sun. 2025. Selfied: Sybil defense in permissionless blockchains via in-protocol bandwidth consumption. *Computer Networks* (2025), 110890.

[34] Marwan Jameel and Oğuz Yayla. 2021. PSO based Blockchain Committee Member Selection. In *6th International Conference on Computer Science and Engineering*.

[35] Zhihao Jiang, Kamesh Munagala, and Kangning Wang. 2020. Approximately stable committee selection. In *STOC*.

[36] Mahimna Kelkar, Soubhik Deb, Sishan Long, Ari Juels, and Sreeram Kannan. 2023. Themis: Fast, Strong Order-Fairness in Byzantine Consensus. In *CCS*.

[37] Aggelos Kiayias, Alexander Russell, Bernardo David, and Roman Oliynykov. 2017. Ouroboros: A provably secure proof-of-stake blockchain protocol. In *CRYPTO*.

[38] Quentin Kniep, Jakub Sliwinski, and Roger Wattenhofer. 2025. *Solana Alpenglow Consensus.* Technical Report. https://github.com/rogerANZA/Alpenglow-White-Paper/blob/main/Alpenglow-v1.1.pdf

[39] Silvio Micali, Salil Vadhan, and Michael Rabin. 1999. Verifiable Random Functions. In *FOCS*.

[40] Andrew Miller, Yu Xia, Kyle Croman, Elaine Shi, and Dawn Song. 2016. The Honey Badger of BFT Protocols. In *CCS*.

[41] Ke Mu, Bo Yin, Alia Asheralieva, and Xuetao Wei. 2024. Separation is Good: A Faster Order-Fairness Byzantine Consensus. In *NDSS*.

[42] Daniel Reijsbergen, Pawel Szalachowski, Junming Ke, Zengpeng Li, and Jianying Zhou. 2021. LaKSA: A Probabilistic Proof-of-Stake Protocol. In *NDSS*.

[43] Alexander Spiegelman, Neil Giridharan, Alberto Sonnino, and Lefteris Kokoris-Kogias. 2022. Bullshark: DAG BFT Protocols Made Practical. In *CCS*.

[44] Yucheng Sun. 2026. Source code for experiments in this paper. https://www.comp.nus.edu.sg/~sunyuch/projects/ccs26/ccs26.html.

[45] Yucheng Sun, Haifeng Yu, and Ruomu Hou. 2025. Committee Selection with Non-Proportional Weights. In *CCS*.

[46] World Bank Group. 2026. Gini index. https://data.worldbank.org/indicator/SI.POV.GINI.
[47] Maofan Yin, Dahlia Malkhi, Michael K. Reiter, Guy Golan Gueta, and Ittai Abraham. 2019. HotStuff: BFT Consensus with Linearity and Responsiveness. In *PODC*.
[48] Mingzhe Zhai, Yizhong Liu, Qianhong Wu, Bo Qin, Haibin Zheng, Xiaopeng Dai, Zhenyang Ding, and Willy Susilo. 2024. Accountable Secret Committee Election and Anonymous Sharding Blockchain Consensus. *IEEE Transactions on Information Forensics and Security* (2024), 9158–9172.

A Open Science

The complete source code for our HC committee selection scheme is publicly available at [44].

B Exhaustive Enumeration of Adversaries

Section 5.4 explained that using denominations, it becomes feasible to search for the worst-case adversary via exhaustive enumeration. The section explains exactly how our implementation does such enumeration.

Let y_i denote the total number of corrupted nodes each with stake d_i. Let f_i denote the total stake of these y_i nodes. Because all nodes with stake d_i are symmetric, the values of all the y_i's (or all the f_i's) uniquely describe the adversary.

Conceptually, we will enumerate the f_i values — doing so will be more efficient than enumerating the y_i values. We use Algorithm 4 to do so. The algorithm can be viewed as a generalization of how the TPL scheme [45] enumerates their adversaries: In [45], their adversary has 2 degrees of freedom (namely, for critical nodes and non-critical nodes). While in our HC scheme, the adversary has r degrees of freedom (i.e., for the r denominations).

Specifically, for each f_i ($1 \le i \le r-1$), Algorithm 4 enumerates $f_i = 0, \Delta, 2\Delta, 3\Delta, \ldots$, where Δ is some small constant (i.e., 0.001). Next for given f_i's ($1 \le i \le r-1$), Algorithm 4 considers $f_r = f + (r-1)\Delta - \sum_{i=1}^{r-1} f_i$, subject to some boundary conditions. Note that here:

$$f_1 + f_2 + \ldots + f_r = f + (r-1)\Delta > f \tag{3}$$

It is obvious that Algorithm 4 skips certain adversaries. For example, the algorithm does not enumerate the adversary with $f_1 = 1.5\Delta$. The extra "$(r-1)\Delta$" term in Equation 3 exactly serves to ensure that such skipping does not cause any problem. Roughly speaking, this extra term gives the adversary $(r-1)\Delta$ extra budget, beyond f. This extra budget ensures that every "skipped" adversary will be "dominated" by some adversary that is indeed enumerated by Algorithm 4. Hence evaluating our HC scheme against the adversaries enumerated by Algorithm 4 will give a pessimistic *upper bound* on our error probability. This also explains the role of Δ: Smaller Δ makes the upper bound tighter, but at the cost of Algorithm 4 enumerating more adversaries (and hence larger computational overhead).

Formally, Lemma 1 below proves that for every possible adversary $\mathcal{A}^*$, there must exist some adversary $\mathcal{A}$ in the set returned by Algorithm 4, such that $\mathcal{A}$ *dominates* $\mathcal{A}^*$. Here "dominates" means that $\mathcal{A}$ compromises a superset of nodes as compared to $\mathcal{A}^*$.

Lemma 1. *Consider any given adversary $\mathcal{A}^*$ whose total budget is f, and let f_i^* be the total stake of those nodes that are corrupted by $\mathcal{A}^*$ and hold d_i stake each, where $\sum_i f_i^* \le f$. Then the set $\mathcal{F}$ returned by Algorithm 4 must contain some element $(f_1, f_2, \ldots, f_r)$ such that $f_i \ge f_i^*$ for all $1 \le i \le r$.*

Proof. We prove the lemma constructively, by finding the element $(f_1, f_2, \ldots, f_r)$ with the desired property. Consider x_i and $\mathcal{F}_i$ in Algorithm 4.

For $1 \le i \le r-1$, we let f_i be the smallest element in $\mathcal{F}_i$ that is larger than or equal to f_i^*. Such f_i must exist, because $x_i \in \mathcal{F}_i$ and $x_i \ge f_i^*$. We let $f_r = \min\left\{x_r, f + (r-1)\Delta - \sum_{i=1}^{r-1} f_i\right\}$. We will later prove that $f_r \ge f_r^*$. Then we immediately have that $f_i \ge f_i^* \ge 0$ for $1 \le i \le r$, and that $(f_1, f_2, \ldots, f_r)$ is in the set $\mathcal{F}$ returned by Algorithm 4. This completes the proof.

Now we just need to prove $f_r \ge f_r^*$. If $f_r = x_r$, then $f_r \ge f_r^*$ trivially holds. Hence we only prove for $f_r = f + (r-1)\Delta - \sum_{i=1}^{r-1} f_i$. For $1 \le i \le r-1$, because two adjacent elements in $\mathcal{F}_i$ differ by at most Δ, we must have $f_i - f_i^* < \Delta$. Then:

$$\begin{aligned} f_r &= f + (r-1)\Delta - \sum_{i=1}^{r-1} f_i > f + (r-1)\Delta - \sum_{i=1}^{r-1}(f_i^* + \Delta) \\ &= f - \sum_{i=1}^{r-1} f_i^* \ge f_r^* \end{aligned}$$

□

C Technical Lemmas

Lemma 2. $\mathrm{Var}[M] + \mathrm{Var}[H]$ *is a fixed quantity that does not depend on the adversaries.*

Proof. Recall that after applying the denominations, each node u will have $s_u = d_i$ for some i. Furthermore, we have ensured $d_i k \le 1$. Node u will hence get into the committee with $s_u k$ probability. By the additivity of variance for independent random variables, we have:

$$\begin{aligned} & \mathrm{Var}[M] + \mathrm{Var}[H] \\ = & \sum_{u \text{ is corrupted}} s_u k(1 - s_u k) + \sum_{u \text{ is honest}} s_u k(1 - s_u k) \\ = & \sum_u s_u k(1 - s_u k) \end{aligned} \tag{4}$$

The term $\sum_u s_u k(1 - s_u k)$ clearly does not depend on the adversary. □

Lemma 3. $\mathrm{Var}[M]$ *is maximized when the adversary corrupts the nodes one-by-one in increasing order of node stake, until it uses up its budget of f.*

Proof. Recall that after applying the denominations, each node u will have $s_u = d_i$ for some i. Furthermore, we have ensured $d_i k \le 1$. Node u will hence get into the committee with $s_u k$ probability. Define $x_u = s_u$ if node u is corrupted, and $x_u = 0$ otherwise. By the

Algorithm 4: EnumerateAdversaries()

for $i = 1, 2, \ldots, r-1$:
  $x_i \leftarrow$ total stake of those nodes with stake d_i;
  $\mathcal{F}_i \leftarrow \{0,\ \Delta,\ 2\Delta,\ \ldots,\ \lfloor \frac{x_i}{\Delta} \rfloor \Delta,\ x_i\}$, where Δ is some small positive constant (e.g., 0.001)
endfor
$x_r \leftarrow$ total stake of those nodes with stake d_r;
$\mathcal{F} \leftarrow \left\{ (f_1, f_2, \ldots, f_r) \;\middle|\; f_i \in \mathcal{F}_i \text{ for } 1 \le i \le r-1 \textbf{ and } f_r = \min\{x_r,\ f + (r-1)\Delta - \sum_{i=1}^{r-1} f_i\} \textbf{ and } f_r \ge 0 \right\}$
return $\mathcal{F}$;

additivity of variance for independent random variables, we have:

$$\begin{aligned} \mathrm{Var}[M] &= \sum_{u \text{ is corrupted}} s_u k(1 - s_u k) \\ &= \sum_{u \text{ is corrupted}} (k - s_u k^2) \cdot x_u \\ &= \sum_{u} (k - s_u k^2) \cdot x_u \\ &= \sum_{u} a_u \cdot x_u \qquad (\text{where } a_u = k - s_u k^2) \end{aligned} \tag{5}$$

We now view Equation 5 as a linear combination of x_u's, and view a_u's as coefficients. Note that a_u is larger when s_u is smaller. The adversary has the constraint of $\sum_u x_u \le f$. Subject to such a constraint, to maximize this linear combination, one should maximize x_u's for larger a_u's. This then corresponds to the adversary that corrupts the nodes one-by-one in increasing order of node stake, until it uses up its budget of f. □

D Experimental Results for Fixed-size Version

Our results in Section 8 are for the variable-size version of various committee selection schemes. In this section, we present the corresponding results for the fixed-size version, in Figure 7 and 8. The settings of these results are the same as in Figure 5 and 6, except that the results here are for the fixed-size version. As one can see, these results are rather similar to those in Section 8.

E Experimental Results under Highly-centralized Stake Distributions

Section 8 mentioned that when the stake distribution is not decentralized (i.e., it is highly-centralized), our HC scheme still always retains the benefits of the FA/TPL scheme. We now give the results under these highly-centralized stake distributions in Figure 9. As one can see, our HC scheme either roughly matches, or outperforms, the FA/TPL scheme under these stake distributions.

Finally, the results in Figure 9 are for variable-size committee selection under $t = 1/3$. We have further verified that the results for fixed-size committee selection as well as for other t values (i.e., $t = 1/2$ and $t = 1/4$) similarly confirm the above conclusion from Figure 9.

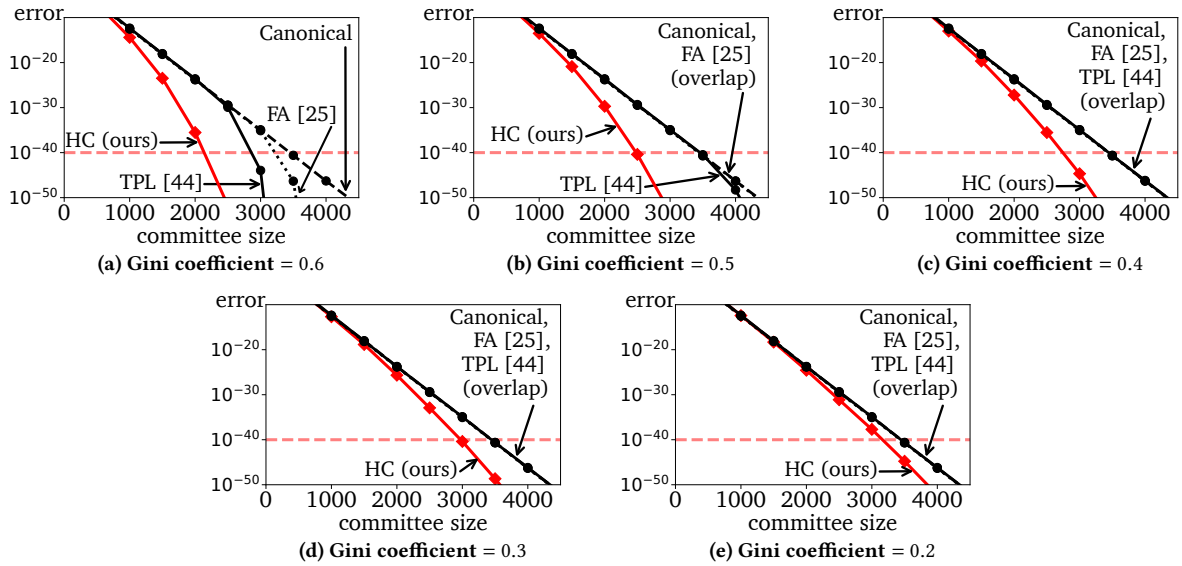


Figure 7: Comparing (fixed-size versions of) our HC scheme against existing designs, for Ethereum distribution and $t = 1/3$. We consider Gini coefficients from 0.2 to 0.6, for the same reason as in Figure 5.

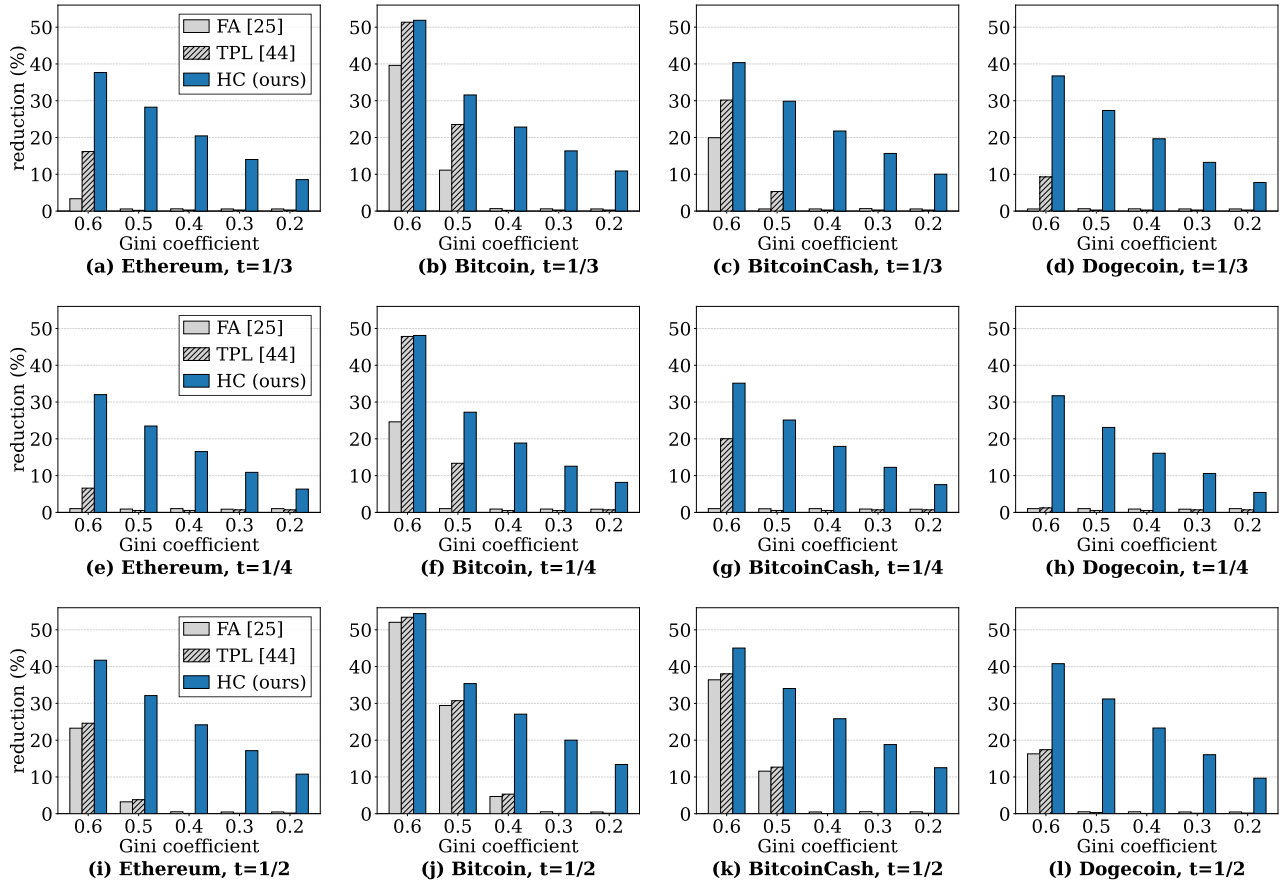
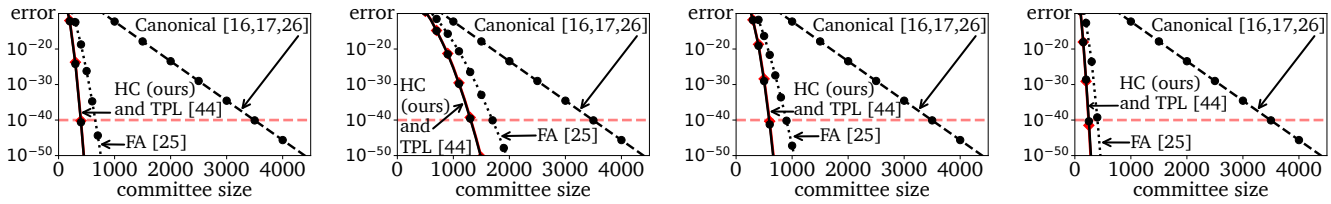


Figure 8: Reduction in committee size achieved by the FA/TPL/HC scheme (fixed-size version), as compared to the canonical design (fixed-size version). We consider Gini coefficients from 0.2 to 0.6, for the same reason as in Figure 5.



(a) Ethereum **(Gini coefficient** 0.874**)**

(b) Bitcoin **(Gini coefficient** 0.656**)**

(c) BitcoinCash **(Gini coefficient** 0.798**)**

(d) Dogecoin **(Gini coefficient** 0.922**)**

Figure 9: Comparing our HC scheme against existing designs, under $t = 1/3$. The 4 stake distributions (Ethereum, Bitcoin, BitcoinCash, Dogecoin) are from [45]. As explained in Section 4, these stake distributions are highly-centralized. Note that in all 4 sub-figures, the curve for our HC scheme overlaps with the curve for the TPL scheme.