DAO Decentralization: Voting-Bloc Entropy, Bribery, and Dark DAOs

James Austgen* Andrés Fábrega* Sarah Allen Kushal Babel
Mahimna Kelkar Ari Juels

Cornell Tech, IC3

1 November 2023 (v1.0)

Abstract

Decentralized Autonomous Organizations (DAOs) use smart contracts to foster communities working toward common goals. Existing definitions of decentralization, however—the 'D' in DAO—fall short of capturing key properties characteristic of diverse and equitable participation.

We propose a new metric called **V**oting-**B**loc **E**ntropy (VBE, pronounced "vibe") that formalizes a broad notion of decentralization in voting on DAO proposals. VBE measures the similarity of participants' utility functions across a set of proposals. We use VBE to prove a number of results about the decentralizing effects of vote delegation, proposal bundling, bribery, and quadratic voting. Our results lead to practical suggestions for enhancing DAO decentralization.

One of our results highlights the risk of systemic bribery with increasing DAO decentralization. To show that this threat is realistic, we present the first practical realization of a *Dark DAO*, a proposed mechanism for privacy-preserving corruption of identity systems, including those used in DAO voting. Our Dark-DAO prototype uses trusted execution environments (TEEs) in the Oasis Sapphire blockchain for attacks on Ethereum DAOs. It demonstrates that Dark DAOs constitute a realistic future concern for DAO governance.

1 Introduction

A Decentralized Autonomous Organization (DAO) is an entity or community that operates based on rules encoded and executed on a public blockchain [22, 46]. As the name suggests, a DAO's governance is decentralized, meaning that it does not rely on a single individual or highly concentrated authority—in contrast to, e.g., a corporation, where a CEO and board of directors make major decisions. Instead, decisions in a DAO are typically made through community votes on proposals. A DAO's treasury, consisting of crypto assets, also generally resides in its smart contract. The contract enforces adherence to community decisions regarding use of its treasury and also offers operational transparency.

DAOs can serve many goals, including investment (e.g., The DAO [48, 64], Mantle Network [58]), grant distribution (e.g., MolochDAO [6], ResearchDAO [9]), gaming-guild organization (e.g., AvocadoDAO [2], GuildFi [5]) and—as is the case for DAOs with the largest treasuries—ecosystem governance (e.g., Uniswap [11], Lido [53], Arbitrum [15], Optimism Collective [8], MakerDAO [56]).

[*] These authors contributed equally to this work.

DAOs of all types are rising rapidly in popularity. At the time of writing (Nov. 2023), the aggregate value across all DAO treasuries exceeds \$17 billion [3], almost double the amount just a year ago.

DAOs today vary considerably in their *true* degree of decentralization. Most have their own associated crypto assets (or "tokens") and weigh voting power by token holdings. It is common for vote outcomes to be determined by a small set of "whales"—a colloquial term used to denote the largest token holders. Such centralization, as well as low voting participation, are a pervasive source of concern in DAO communities. Vulnerability to centralization has even led to plundering of DAO treasuries [57].

A number of works have sought to recommend ways to improve DAO decentralization. But first it's necessary to be able to *measure* it in a way that is reflective of a broad set of real-world concerns. That requirement is the starting point for our work in this paper.

1.1 Measuring DAO Decentralization

A common basis for evaluating decentralization in DAOs and other blockchain settings is *token ownership*, specifically the distribution of assets and consequently voting rights among participants [72, 44]. Informally, concentration of a large fraction of tokens in a small number of hands—and thus the ability of a small group to determine voting outcomes—is indicative of strong centralization. More widespread distribution, conversely, suggests decentralization.

Entropy is one popular metric for measuring decentralization in the distribution of token ownership in a DAO.[1] For a set of addresses $A = \{a_1, \ldots, a_n\}$, where address a_i holds t_i tokens and $T = \sum_{i=1}^{n} t_i$:

$$\mathsf{entropy}(A) \triangleq -\sum_{i=1}^{n} \frac{t_i}{T} \log\left(\frac{t_i}{T}\right).$$

Low entropy corresponds to a high degree of asset concentration and thus strong centralization. High entropy implies the opposite. Other popular decentralization metrics, e.g., the Gini coefficient [41] and the Nakamoto coefficient [7, 74], are related to various notions of entropy.

Token ownership distribution alone, however, has serious shortcomings as a decentralization metric. To begin with, it is visible on chain only in terms of per-address holdings, not control by real-world individuals. Thus, for instance, an individual who holds 51% of tokens in a DAO, but spreads them among a large number of addresses could create an appearance of decentralization while having majority control.

Even if tokens are held by distinct entities, a notion put forward in, e.g., [50], those entities may have aligned interests and act in concert—a form of centralization. The following examples illustrate cases in which a DAO may be strongly centralized, *even if token ownership appears to imply strong decentralization.*

Example 1 (Low participation / apathy)**.** Lack of participation in DAO governance votes is widespread in practice [31] and induces a form of centralization. Consider, for example, a DAO that requires a quorum of 50% participation for a vote to be ratified. Suppose 50% of voters do not

[1] Entropy is typically defined over a random variable. A token ownership distribution may be viewed as a random variable for an experiment where a token is selected uniformly at random and its owner is output.

cast votes and voters other than whales vote 2:1 in favor of the proposal. Whales with just 12.6% of all tokens can swing the vote and cause the proposal to be rejected.

Example 2 (Herding)**.** Interviews with DAO participants have revealed a tendency to vote in alignment with influential community members to preserve reputation [72], as individual votes are today usually publicly observable. This effect—often called *herding* [13, 17]—has a centralizing effect. It aligns votes around the choices of a small set of participants. (This problem is similar to the notion of "herding" in classical voting theory [45, 13].)

Example 3 (Bribery / vote-buying)**.** Bribery—specifically, *vote-buying*—has been a longstanding concern of DAO organizers [25]. It has a centralizing effect, as it aligns voters around a choice dictated by the briber.

Recognizing that token-ownership alone doesn't give a full picture of decentralization, researchers have explored broader notions. Most notably, Sharma et al. [72] have considered entropy measures limited to those voters who participate in votes and have also explored graph-based representations of voting patterns (degree centralization, degree assortativity, etc.). Token-ownership distribution among voting participants fails to capture important issues, such as those in Examples 2 and 3, however, and it's unclear how to interpret graph-based metrics.

With no consensus in the community about how to measure DAO decentralization today, there is a lack of principled guidance on ways to improve DAO decentralization and to combat threats to decentralization, such as vote-buying.

1.2 Voting-Bloc Entropy (VBE)

We introduce a decentralization metric tailored to DAO governance called ***V****oting-****B****loc* ***E****ntropy* (VBE, pronounced "vibe"). VBE is based on a foundational principle, that voters with closely aligned interests across elections are a centralizing force, in contrast to the notion of "credible neutrality" in voting, which is characterized by "positive ratings from people across a diverse range of perspectives" [21]. Expressed differently, the key idea in VBE is to *define centralization as the existence of large voting blocs.*

Formally, we express this principle in terms of the *utility functions* of DAO participants, i.e., quantification of the gain or loss associated with voting outcomes. For a given set of elections, a voting bloc is a cluster of voters whose utility functions are similar over outcomes. Utility functions are *latent variables*—conceptually important, but not always directly measurable—and consequently VBE is as well [42].

VBE then, measures entropy over voting blocs based on utility functions—rather than over individual token holdings. The result is a broad concept that captures the centralization embodied in all of our examples above. VBE is in fact a framework: It allows different notions of clustering and entropy to be plugged in.

We stress that VBE is a theoretical metric: It cannot be measured *directly*, since users do not typically express (or often even know) their utility functions explicitly. But VBE provides an important basis for *reasoning about the directional influence of policy choices on decentralization*, and does lend itself to *indirect* measurement.

VBE Implications: We use VBE to prove a number of theoretical results showing how various practices tend to increase or decrease DAO decentralization. (We prove these results relative to particular notions of clustering and entropy.)

Our main results are as follows:

- **Apathy / inactivity whale:** A large population of apathetic, i.e., non-voting DAO participants, is a centralizing force (Theorem 3.3).
- **Delegation:** Given an inactivity whale of large size relative to delegatees, delegation tends (perhaps counterintuitively) to increase decentralization (Theorem 3.4).
- **Bribery:** Bribery and decentralization are closely related in the context of DAO governance. The act of bribery decreases decentralization (Theorem 3.7). Additionally, as the decentralization of a DAO rises, so does the risk of systemic bribery—and vice versa (Theorems 3.8 and 3.9).

We additionally prove results relating to herding and privacy (Theorem 3.5), quadratic voting (Theorem 3.10) and owning multiple accounts (Theorem 3.2).

Looking ahead, our theorem statements and proofs are simple, and some just show how VBE confirms a known pattern (for example, that quadratic voting is susceptible to sybil attacks). However, our goal is to show the flexibility of VBE, and how, unlike prior metrics, it is able to capture the subtle impacts the certain mechanisms have on decentralization. Thus, the main contribution of VBE is to put forth a new way to *think* about DAO decentralization. That is, VBE introduces a paradigm shift, namely, framing elections in terms of abstract voting entities—instead of individual accounts—as defined by their aligned incentives and interests.

1.3 Dark DAOs

Our results on bribery and decentralization (Theorems 3.7, 3.8, and 3.9) show that as decentralization increases, bribery can only be successful if it operates on a large scale. This observation raises a critical followup question: Is large-scale bribery a realistic future threat to DAOs?

One mechanism postulated for systemic DAO-voting bribery is a *Dark DAO*. A Dark DAO was originally defined as "a decentralized cartel that buys on-chain votes opaquely." [33]. Here, "opaquely" means that participation in the Dark DAO is confidential. A Dark DAO must also ensure correct execution of a bribery scheme, i.e., bribees are paid if and only if they vote as directed. We define a Dark DAO more broadly as a DAO designed to subvert credentials in an identity system. The goal may be to attack a voting scheme, but could be to attack another system, e.g., we also consider attacks against *privacy pools* [27], which are effectively DAOs to enhance cryptocurrency privacy.

To date, the feasibility of a fully functional Dark DAO has yet to be demonstrated. In this work, we present a Dark DAO prototype to facilitate vote-buying in DAOs on Ethereum—the most popular blockchain for DAOs today. In its back end, it leverages the confidentiality enforced by trusted hardware (specifically Intel SGX) in the Oasis Sapphire blockchain[2]. We also present a "Dark DAO Lite" prototype variant that offers greater ease of usability than our basic prototype, but at the cost of weaker confidentiality.

[2] `https://github.com/oasisprotocol/sapphire-paratime`

We underscore our belief that Dark DAOs do not pose a current threat, given the limited decentralization of DAOs today, and briefly review ethical considerations in this paper. Our work demonstrates that Dark DAOs are an eminently realistic future threat, however. We discuss possible Dark-DAO mitigations as a first step toward community development of countermeasures.

Our techniques for Dark-DAO construction are of independent interest, as they point the way toward general techniques for the construction of new financial instruments.

1.4 Contributions

In summary, our contributions in this work are:

- **Voting-Bloc Entropy (VBE):** We introduce VBE, a new metric for DAO decentralization that generalizes prior metrics and addresses a number of their shortcomings (Section 2).
- **Theoretical results:** Using VBE, we prove a range of results about how various DAO practices and design choices impact decentralization (Section 3).
- **Dark DAOs:** Our theoretical results highlight risks of systemic bribery in attacking DAOs—via Dark DAOs—against highly decentralized target DAOs. To show that these risks are a realistic long-term concern, we implement two end-to-end Dark DAO prototypes with different confidentiality / ease-of-use trade-offs (Sections 5 to 7). Our techniques are of general interest as they include innovations in the construction of decentralized-finance instruments.
- **Practical guidance:** Based on our theoretical and experimental results, we present concrete points of practical guidance for DAO design and deployment around issues including delegation, voting privacy, voting-slate composition, decentralized identity, and more (Section 8). We summarize this guidance in Table 3.

We review related work in Section 9 and conclude with some open research questions in Section 10.

2 Voting-Bloc Entropy (VBE)

In this section, we define *Voting-Bloc Entropy* (VBE), which sidesteps the aforementioned limitations of prior metrics. It does so by normalizing token holdings based on voters' utility functions.

Intuition. The key idea behind our definition is to reason about centralization with respect to the tokens held by *groups of DAO members with aligned interests*, instead of with respect to individual members. That is, instead of measuring the distribution of tokens across individual addresses, we focus instead on how tokens are distributed across *blocs* of voters with the same incentives, which are functionally acting as a single entity. Looking ahead, we formalize the notion of "aligned interests" by considering the DAO members' *utility functions* across elections.

Aggregating voters based on utility functions allows us to capture the rich interactions and relationships between players in the system, all of which play a role in understanding the true degree of decentralization of a DAO, as discussed in Section 1.1. Indeed, the limitations highlighted there are captured by considering voters with similar utility functions as a single entity; we discuss this extensively in Section 3.

DAO abstraction. We now introduce the notation and formalism that our definition and theorems rely on.

Let $\mathcal{P} = \{P_1, \ldots, P_n\}$ be the set of token holders in a system, and $\mathsf{tokens}\colon \mathcal{P} \to \mathbb{R}^+$ a mapping specifying the number of tokens held by each $P \in \mathcal{P}$. (We will often overload this notation, and input a *set* of accounts to tokens instead, by which we mean the total tokens held across all accounts in the set). These token holders participate in a set of (binary) elections $E = \{e_1, e_2, \ldots, e_m\}$, where we denote by $\mathsf{vote}_P\colon E \to \{\mathtt{true}, \mathtt{false}, \bot\}$ player P's vote in election e; $\bot$ indicates that P abstained from voting in e. We define $\mathsf{util}_P\colon E \times \{\mathtt{true}, \mathtt{false}\} \to \mathbb{R}$ to be the monetary utility of an outcome of $\mathtt{true}$ or $\mathtt{false}$ in e to player P, where we make the simplifying assumption that $\mathsf{util}_P(e, \mathtt{true}) = -\mathsf{util}_P(e, \mathtt{false})$. Player P's total utility across all elections E is represented by a vector $U_{E,P} \coloneqq (\mathsf{util}_P(e_i, \mathtt{true}))_{i \in [m]} \in \mathbb{R}^m$; we denote by $U_{E,\mathcal{P}}$ all players' utilities, i.e., $U_{E,\mathcal{P}} \coloneqq (U_{E,P})_{P \in \mathcal{P}}$.

Token holders often have low stakes in the elections, resulting in lack of interest or abstaining from voting altogether. More formally, we say that player P is *ϵ-apathetic* in election e if and only if $|\mathsf{util}_P(e, \mathtt{true})| \leq \epsilon$. We denote this set of apathetic voters by $\mathcal{A}$. If the system supports vote delegation (for example, as a means to combat apathetic voters), players may delegate their tokens to others, who cast a single vote on behalf of all the tokens they now hold.

Lastly, we define $\mathsf{bribe}_P\colon E \times \{\mathtt{true}, \mathtt{false}\} \to \mathbb{R}$ to be such that it is possible for player P to achieve an outcome of $\mathtt{true}$ (resp., $\mathtt{false}$) in a particular election e via bribery for any expenditure greater than $\mathsf{bribe}_P(e, \mathtt{true})$ (resp., $\mathsf{bribe}_P(e, \mathtt{false})$). Note that we make the simplifying assumption that bribery costs are independent across elections. We assume that bribing a given P to flip its vote from $\mathtt{true}$ to $\mathtt{false}$ (respectively, $\mathtt{false}$ to $\mathtt{true}$) costs $\max(2 \cdot \mathsf{util}_P(e, \mathtt{true}) + \epsilon, 0)$ (respectively, $\max(2 \cdot \mathsf{util}_P(e, \mathtt{false}) + \epsilon, 0)$), for some constant ϵ. Successful bribery to achieve an outcome of $\mathtt{true}$ in e means flipping enough votes to cross a certain threshold q of votes for $\mathtt{true}$ in e (and vice versa for $\mathtt{false}$), i.e., ensuring

$$\sum_{P \in \mathcal{P}} \Big(\mathsf{tokens}(P) \,\Big|\, \mathsf{vote}_P(e) = \mathtt{true} \Big) > q \cdot \sum_{P \in \mathcal{P}} \mathsf{tokens}(P).$$

For example, a typical value for q may be $q = 0.5$, which corresponds to an absolute majority. We stress that the equation above represents the threshold to *ensure* the desired outcome in an election, and not just to win it. Albeit related, these notions are not equivalent; the former implies the latter, but not vice-versa. In particular, winning an election is a function of the number of votes cast for the undesired option, whereas ensuring an outcome is agnostic to this.

2.1 Framework for VBE

We present VBE in this section. To do so, we introduce an abstract framework that is parameterized by: (1) a clustering metric, and (2) an entropy notion, which are the two key ingredients that underpin our definition.

Clustering. We let $C\colon U_{E,\mathcal{P}} \times U_{E,\mathcal{P}} \to \{0, 1\}$ be a clustering function that outputs 1 if the utilities of two players are "aligned" across all elections E, and 0 otherwise. Our definition of VBE is agnostic to a specific clustering algorithm, and instead only assumes that C specifies an equivalence relation $\sim_C$ on the set $\mathcal{P}$, such that $P_i \sim_C P_j$ if and only if $C(U_{E,P_i}, U_{E,P_j}) = 1$. That is, C partitions $\mathcal{P}$ into classes of players with aligned utility functions across elections. Following standard notation, we denote the set of all classes by $\mathcal{P}/\sim_C$, and the class P belongs to by $[P]$.

Entropy. We denote by F a function from the distribution of tokens across *sets* of accounts to real numbers. The purpose of F is to measure, in some sense, how "evenly distributed" tokens are across voting blocs. Thus, in practice, F will generally consist of some notion of entropy,[3] such as one of the many variants of Rényi entropy, e.g., min-entropy, Shannon entropy, or max entropy. (Note that if the blocs are comprised of single entities, this is equivalent to entropy over individual accounts, as defined in prior work [72].) We stress, however, that in principle F can be any function, and our definition makes no assumptions about its structure. As one example, given a clustering with single entities, we can use any distance metric $d(\cdot,\cdot)$ on $U_{E,\mathcal{P}}$ and define F as the negative of the sum of squares distance between all pairs of player utilities, i.e., $F = -\sum_{P_1,P_2\in\mathcal{P}} d(U_{E,P_1}, U_{E,P_2})^2$

We are now ready to define VBE. Intuitively, our definition says that a DAO is more decentralized if the distribution of tokens across the abstract voting entities specified by $\sim_C$ has high entropy according to F. More concretely:

Definition 1 (Voting-Bloc Entropy)**.** For a set of elections E, a set of players $\mathcal{P}$ with corresponding utilities $U_{E,\mathcal{P}}$, a mapping specifying the distribution of token ownership tokens, a clustering metric C, and an entropy function F, we define *Voting-Bloc Entropy (VBE)* to be:

$$\mathsf{VBE}_{C,F}(E,\mathcal{P},U_{E,\mathcal{P}},\mathsf{tokens}) \coloneqq F(\mathcal{P}/\sim_C,\mathsf{tokens}).$$

2.2 Instantiation of VBE

There are various concrete algorithms with which one can instantiate our VBE framework. We propose one such example in this section, which we use throughout the rest of the paper. The advantages and disadvantages of this variant, and VBE in general, are discussed in Section 4.

Clustering. We define *ϵ-threshold ordinal clustering (ϵ-TOC)* as follows:

$$C_\epsilon(U_{E,P_i}, U_{E,P_j}) \coloneqq \begin{cases} 1 & \text{if } \forall k \in [m],\ \Big(\mathrm{sgn}(U_{E,P_i}[k]) = \mathrm{sgn}(U_{E,P_j}[k])\Big) \vee \Big(|U_{E,P_i}[k]|, |U_{E,P_j}[k]| \le \epsilon\Big) \\ 0 & \text{otherwise} \end{cases}.$$

More simply, ϵ-TOC clusters together token holders who have the same preferred outcome across all elections, regardless of how strong this preference is. That is, clusters correspond to token holders whose utility functions are ordinally equivalent. Even though a more granular metric could create clusters based on cardinal utility, we regard ordinal equivalence to be indicative of aligned preferences. Further, as we discuss in Section 4, ϵ-TOC has the benefit of being computable based on historical voting data, whereas more complex clustering metrics may be more difficult (or impossible) to estimate.

In addition to these blocs, ϵ-TOC also creates an additional cluster corresponding to all apathetic voters $\mathcal{A}$, i.e., those whose utilities are close to 0. These voters have aligned preferences, namely, little to no interest in election outcomes.

[3] Entropy is formally defined over a random variable, but we are overloading notation to think of the mapping between sets of accounts and their respective cumulative token balances as the probability mass function of a random variable.

Entropy. In this work, we use *min-entropy* as our entropy notion. That is, for a set of sets of addresses A with a total of T tokens held across all individual accounts,

$$F_{\min}(A, \mathsf{tokens}) \coloneqq \log_2 \left(\frac{\max\limits_{A' \in A} \mathsf{tokens}(A')}{T} \right).$$

Our entropy notion thus measures the amount of "information" in the largest voting bloc by token holdings. As we discuss later on, more granular entropy notions result in more detailed analysis (at the cost of being more difficult to estimate in practice), as these may capture the information in other voting blocs beyond the largest.

Putting the two together, we thus get a concrete instantiation of the framework:

$$\mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U_{E,\mathcal{P}}, \mathsf{tokens}) \coloneqq F_{\min}(\mathcal{P}/\sim_{C_\epsilon}, \mathsf{tokens}) = -\log_2 \left(\frac{\max\limits_{\mathcal{P}' \in \mathcal{P}/\sim_{C_\epsilon}} \mathsf{tokens}(\mathcal{P}')}{\sum\limits_{P \in \mathcal{P}} \mathsf{tokens}(P)} \right).$$

Given that utility functions cannot be measured directly, we cannot compute $\mathsf{VBE}_{C_\epsilon,\min}$ (or any other variant of VBE) directly. However, as we show in the subsequent section, VBE can be used as a conceptual tool to reason about the high-level impact that changes in the system (such as the implementation of policy choices) have on the decentralization of a DAO. Namely, one can reason about the *directional* influence of said changes on the utility functions of the players, and thus derive conclusions about whether VBE broadly increased or decreased.

Further, utility functions (and, thus, VBE) can be estimated via *observable variables*, which can be measured directly. In this case, one can explicitly compute VBE, and derive concrete metrics. The degree to which observable variables accurately estimate utility functions (for the purposes of VBE) will depend, to a great extent, on the specific clustering algorithm that is used. We discuss this in more detail in Section 4.

3 Implications of VBE: Theoretical Results

We now present a variety of theoretical results implied by VBE. These results show how, unlike prior notions, VBE reflects many of the subtle issues that impact decentralization in a DAO—such as those described in Section 1.1—and thus can serve as a springboard for more accurate understanding of the goals of a DAO.

We first note that, for most "reasonable" instantiations of F (such as any Shannon or min-entropy), computing an analogous metric over account balances alone, instead of over voting blocs, gives an upper bound on VBE. (Note that the former includes the entropy-based metrics of prior work such as [72].) Concretely, this fact holds for any F that increases whenever the tokens held by any players's voting bloc increase. More formally:

Lemma 1. Let $C_{\mathtt{solo}}$ be the clustering metric that partitions $\mathcal{P}$ into singleton sets, i.e.,

$$C_{\mathtt{solo}}(U_{E,P_i}, U_{E,P_j}) \coloneqq 1 \iff i = j,$$

and let F be any function that is monotonically increasing with respect to $\mathsf{tokens}([P])$ for every $P \in \mathcal{P}$. Then, for any clustering metric C, it follows that

$$\mathsf{VBE}_{C_{\mathtt{solo}},F}(E, \mathcal{P}, U_{E,\mathcal{P}}, \mathsf{tokens}) \geq \mathsf{VBE}_{C,F}(E, \mathcal{P}, U_{E,\mathcal{P}}, \mathsf{tokens}).$$

Proof. This simply follows from the fact that, for all players P, the number of tokens held by their bloc according to C is necessarily greater than or equal to the number of tokens held by their bloc according to $C_{\texttt{solo}}$: in the former, either P got grouped in a bloc with more players (and thus holds more total tokens), or she stayed alone in her bloc. As such, by definition, F will increase correspondingly. □

We stress that this lemma holds for most F of practical interest, such as Shannon entropy and min-entropy. As such, VBE is, at worst, equivalent to the entropy-based notions introduced by prior work, which focus on account balances. In the subsections that follow, we will show how, in fact, VBE reveals more information, as it is able to capture how decentralization is affected by various mechanisms, irrespective of (lack of) fluctuations in account balances. Towards this, we will first present the general recipe of our theorems, before moving on to concrete results.

We note that, for clarity of presentation, our theorems focus exclusively on one instantiation of VBE, namely, using ϵ-TOC and min-entropy as the clustering metric and entropy function, respectively. However, even though the theorem statements and proof details would differ for other variants of VBE, the conceptual takeaways are general (and, in fact, can be made more specific with more granular instantiations of VBE).

3.1 VBE Master Theorem

The theorems in the subsections that follow all aim to show the impact of policy choices or system changes on DAO decentralization, in terms of VBE. They all have a similar structure: (1) we consider two systems such that the only difference between them is some "transformation" of interest, e.g., a portion of the voters become apathetic, elections are instead private, etc; (2) we reason about the impact of this transformation on the largest voting bloc of both systems; (3) based on this, we compute and compare the VBE of both systems.

We now define a "master" theorem for $\mathsf{VBE}_{C_\epsilon,\min}$ which captures this template, and thus serves as a proof framework that can be instantiated with concrete transformations of interest. Indeed, our theoretical results that follow are examples of this, as they all invoke this master theorem. (We note that the master theorem can be easily tweaked to accommodate different instantiations of VBE, but here we focus exclusively on ϵ-TOC and min-entropy for clarity of presentation.)

In all theorems that follow, we denote by E a set of binary elections, $\mathcal{P}$ a set of players that participate in such elections, tokens a mapping specifying the number of tokens owned by each player, and $U_{E,\mathcal{P}}$ the players's utilities across the elections. The master theorem then proceeds as follows:

Theorem 3.1 (Voting-Bloc Entropy Master Theorem)**.** We define T to be a function that represents a *system transformation*, i.e., a change in the players, elections, utilities of the players, and/or the distribution of tokens, which we denote by $(\mathcal{P}', E', U'_{E',\mathcal{P}}, \mathsf{tokens}') := T(\mathcal{P}, E, U_{E,\mathcal{P}}, \mathsf{tokens})$. The total number of tokens in the system stays constant, however. Let B and B' be the (not necessarily unique) largest ϵ-TOC clusters by token holdings according to $(E, U_{E,\mathcal{P}}, \mathsf{tokens})$ and $(E', U_{E',\mathcal{P}}, \mathsf{tokens}')$, respectively. Then, it follows that

$$\mathsf{tokens}'(B') \geq \mathsf{tokens}(B) \iff \mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U_{E,\mathcal{P}}, \mathsf{tokens}) \geq \mathsf{VBE}_{C_\epsilon,\min}(E', \mathcal{P}', U'_{E',\mathcal{P}}, \mathsf{tokens}').$$

Proof. This follows directly from the definition of $\mathsf{VBE}_{C_\epsilon,\min}$:

$$\begin{aligned}
& \mathsf{tokens}'(B') \geq \mathsf{tokens}(B) \\
\iff \quad & \frac{\mathsf{tokens}'(B')}{\sum\limits_{P\in\mathcal{P}'} \mathsf{tokens}'(P)} \geq \frac{\mathsf{tokens}(B)}{\sum\limits_{P\in\mathcal{P}} \mathsf{tokens}(P)} \\
\iff \quad & -\log_2\Big(\frac{\mathsf{tokens}'(B')}{\sum\limits_{P\in\mathcal{P}'} \mathsf{tokens}'(P)}\Big) \leq -\log_2\Big(\frac{\mathsf{tokens}(B)}{\sum\limits_{P\in\mathcal{P}} \mathsf{tokens}(P)}\Big) \\
\iff \quad & \mathsf{VBE}_{C_\epsilon,\min}(E',\mathcal{P}',U'_{E',\mathcal{P}'},\mathsf{tokens}') \leq \mathsf{VBE}_{C_\epsilon,\min}(E,\mathcal{P},U_{E,\mathcal{P}},\mathsf{tokens})
\end{aligned}$$

□

Note that, if B' represents a (new) majority by token holdings, then VBE strictly increases; equality follows when $\mathsf{tokens}'(B') = \mathsf{tokens}(B)$.

This master theorem thus serves as a template that individual theorems can bootstrap off of: simply specify a transformation T, explain how this modifies the largest voting bloc (if at all), and invoke Theorem 3.1. Armed with this formula, we now move on to concrete theoretical insights implied by VBE. Our theorem statements and proofs are simple, and often just show how VBE confirms a known pattern (for example, that quadratic voting is susceptible to sybil attacks). However, our goal is to show the flexibility of (a limited instantiation of) VBE, and how, unlike prior metrics, it is able to capture the subtle impacts the certain mechanisms have on decentralization.

3.2 Owning Multiple Accounts

As explained in Section 1, previous notions of entropy fail to capture the centralization that is present (but hidden) when a whale distributes tokens across multiple accounts / addresses. In such cases, it may appear that tokens are well diversified across accounts, while a large fraction are in fact under the control of one entity. Unlike prior notions, VBE captures this nuance, since these accounts would indeed be considered a single voting bloc (we make the simplifying assumption that an individual's utility function is the same across all her accounts; this need not always be true). We formalize this below.

Theorem 3.2 (Sybil Attacks and VBE)**.** Let $(\mathcal{P}', E, U'_{E,\mathcal{P}'}, \mathsf{tokens}') = T_{\mathtt{mult}}(\mathcal{P}, E, U_{E,\mathcal{P}}, \mathsf{tokens})$ be the transformation where some player $P \in \mathcal{P}$ divides her tokens across a new set of accounts $\hat{\mathcal{P}}$, i.e., $\mathcal{P}' = \mathcal{P} \cup \hat{\mathcal{P}}$, $\mathsf{tokens}'(\hat{\mathcal{P}}) = \mathsf{tokens}(P)$, and $\forall \hat{P} \in \hat{\mathcal{P}}$, $U'_{E,\hat{P}} = U_{E,P}$. The rest of the system remains unchanged. Then, it follows that

$$\mathsf{VBE}_{C_\epsilon,\min}(E,\mathcal{P},U_{E,\mathcal{P}},\mathsf{tokens}) = \mathsf{VBE}_{C_\epsilon,\min}(E,\mathcal{P}',U'_{E,\mathcal{P}'},\mathsf{tokens}').$$

Proof. Let B be the largest voting bloc by token holdings before $T_{\mathtt{mult}}$, which may or may not include P. By assumption, all $\hat{P} \in \hat{\mathcal{P}}$ are such that $U'_{E,\hat{P}} = U_{E,P}$. Thus, all new accounts will be in the same voting bloc B' after $T_{\mathtt{mult}}$, namely, $B' = [P]$.

It follows then that, even though P's tokens are distributed between all individual accounts in $\hat{P}$, they are in fact still under the control of the same block, i.e., B'. As such, $\mathsf{tokens}'(B') = \mathsf{tokens}(B')$.

So, since no blocs acquire any new tokens, B is still the largest voting bloc by token holdings after $T_{\texttt{mult}}$. Then, from Theorem 3.1 it follows that

$$\mathsf{VBE}_{C_\epsilon,\min}(E,\mathcal{P},U_{E,\mathcal{P}},\mathsf{tokens}) = \mathsf{VBE}_{C_\epsilon,\min}(E,\mathcal{P},U'_{E,\mathcal{P}},\mathsf{tokens})$$

as desired. □

This result shows that, according to VBE, the "true" decentralization of the system does not change when a whale splits her tokens into multiple accounts, as they are all still under the control of the same voting entity. Conversely, metrics that focus on account balances alone would mistakenly conclude that the decentralization of the system strictly increased, since a set of tokens is diversified across more accounts.

3.3 Apathy

A system where voters are apathetic, i.e., not interested in the direction of the community, is not aligned with the goals of a DAO: distribution of tokens is irrelevant if individuals abstain from voting, as elections are narrowed squarely to the set of more invested stakeholders. Our definition captures this fact. Intuitively, apathetic voters all have similar utility functions, which reflect their lack of stake in the elections. VBE groups all of these players within the same voting bloc, due to their aligned utilities. (Recall that we use $\mathcal{A}$ to denote this set of apathetic voters.)

As we formalize below, if the disinterested players are small stakeholders to begin with, apathy has a centralizing effect, as they now belong to a larger bloc of aligned voters. Indeed, in practice, it is common for the set of apathetic voters to represent a majority of token holdings [31, 44]. We note, however, that interestingly apathy can potentially also have a decentralizing effect, in the (rare) case where it helps diversify a larger coalition of voters.

Theorem 3.3 (Apathy and VBE)**.** Let $(E,U'_{E,\mathcal{P}},\mathsf{tokens}) = T_{\texttt{apath}}(\mathcal{P},E,U_{E,\mathcal{P}},\mathsf{tokens})$ be the transformation where players $\hat{\mathcal{P}} \subseteq \mathcal{P}$ become ϵ-apathetic, i.e., $\forall P \in \hat{\mathcal{P}}\ \forall e \in E$, $|\mathsf{util}'_P(e,\texttt{true})| \leq \epsilon$. The rest of the system remains unchanged. Then, if $\forall P \in \hat{\mathcal{P}}$, $\mathsf{tokens}(\mathcal{A}) \geq \mathsf{tokens}([P])$, it follows that

$$\mathsf{VBE}_{C_\epsilon,\min}(E,\mathcal{P},U_{E,\mathcal{P}},\mathsf{tokens}) \geq \mathsf{VBE}_{C_\epsilon,\min}(E,\mathcal{P},U'_{E,\mathcal{P}},\mathsf{tokens}).$$

Proof. Let B be the largest voting bloc by token holdings before $T_{\texttt{apath}}$. We first note that all apathetic voters belong to the same voting bloc B', according to ϵ-TOC: by the definition of ϵ-apathetic, it follows that, for all $P_i, P_j \in \hat{\mathcal{P}}$ and $e \in E$,

$$|\mathsf{util}'_{P_i}(e,\texttt{true})|, |\mathsf{util}'_{P_j}(e,\texttt{true})| \leq \epsilon,$$

which corresponds precisely to the bloc of apathetic voters in ϵ-TOC, containing all players in $\mathcal{A}$. Then, by assumption, $\mathsf{tokens}(B') = \mathsf{tokens}(\mathcal{A}) \geq \mathsf{tokens}([P])$, $\forall P \in \hat{\mathcal{P}}$. So, since no other blocs decrease in size, it follows that $\mathsf{tokens}(B') \geq \mathsf{tokens}(B)$: either the bloc that aggregates all apathetic voters is now the largest bloc, or the same bloc is the largest in both instances. Thus, from Theorem 3.1, it follows that

$$\mathsf{VBE}_{C_\epsilon,\min}(E,\mathcal{P},U_{E,\mathcal{P}},\mathsf{tokens}) \geq \mathsf{VBE}_{C_\epsilon,\min}(E,\mathcal{P},U'_{E,\mathcal{P}},\mathsf{tokens})$$

as desired. □

This result shows that VBE captures the intuition that large-scale apathy, which is common in practice, has a centralizing effect. We refer to the bloc of apathetic voters in a DAO, i.e., non-voting token holders, as the *inactivity whale*. This term reflects the collective and potentially systemically important inactive behavior of this group.

3.4 Delegation

Intuition would suggest that delegation leads to a more centralized system: tokens that were originally held by a large set of players, are instead under the control of the (smaller) set of delegates. As we prove formally below, however, VBE shows how this situation is more nuanced, as delegation actually tends to make a DAO *more* decentralized: before delegation, the tokens are all held by a *single* voting bloc, namely, the inactivity whale. Delegation then diversifies the tokens held by this "whale", and distributes them amongst a set of voting blocs (the delegates). Assuming that the size of the inactivity whale is larger than each delegate's total tokens—which tends to be true in practice [31, 44]—the system is now more decentralized.

Theorem 3.4 (Delegation and VBE)**.** Let $(E, U'_{E,\mathcal{P}}, \mathsf{tokens}') = T_{\mathtt{deleg}}(\mathcal{P}, E, U_{E,\mathcal{P}}, \mathsf{tokens})$ be the transformation where players $\hat{\mathcal{P}} \subseteq \mathcal{P}$, who are ϵ-apathetic, instead delegate their votes to a set of delegates $D \subset \mathcal{P}$, i.e., $\mathsf{tokens}'(D) = \mathsf{tokens}(\hat{P})$ and $\mathsf{tokens}'(\hat{\mathcal{P}}) = 0$. The rest of the system remains unchanged. Then, if $\forall d \in D$, $\mathsf{tokens}(\mathcal{A}) \geq \mathsf{tokens}'([d])$, it follows that,

$$\mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U'_{E,\mathcal{P}}, \mathsf{tokens}') \geq \mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U_{E,\mathcal{P}}, \mathsf{tokens}).$$

Proof. Let B by the largest voting bloc by token holdings before $T_{\mathtt{deleg}}$. As discussed in the proof of Theorem 3.3, all players in $\hat{P}$ belong to the same voting bloc for all elections in E—the inactivity whale—since they are all part of the set of apathetic voters $\mathcal{A}$. Let B' be the largest voting bloc by token holdings after $T_{\mathtt{deleg}}$; note that it may be the case that $B' = [d]$ for some $d \in D$.

We first note that B' is equal to either (1) B itself, (2) the second largest voting bloc after B before delegation, or (3) $[d]$, for some $d \in D$. That is, since the only blocs that change after $T_{\mathtt{deleg}}$ are all the $[d]$ and the inactivity whale (which lost $\mathsf{tokens}(\hat{P})$ tokens), it must the be case that the new largest voting bloc is either the same one as before delegation, the second largest voting bloc before delegation (i.e., B was the inactivity whale, which got fractionated by delegation), or one of the $[d]$ which increased in size.

For (1) and (2), it is clearly the case that $\mathsf{tokens}(B) \geq \mathsf{tokens}'(B')$. Then, for (3), note that, by assumption, $\mathsf{tokens}(\mathcal{A}) \geq \mathsf{tokens}'([d])$, for all $d \in D$. So, $\mathsf{tokens}(B) \geq \mathsf{tokens}(\mathcal{A}) \implies \mathsf{tokens}(B) \geq \mathsf{tokens}'([d]) = \mathsf{tokens}'(B')$.

It follows then that, in all cases, $\mathsf{tokens}(B) \geq \mathsf{tokens}'(B')$. Thus, from Theorem 3.1, we get that

$$\mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U'_{E,\mathcal{P}}, \mathsf{tokens}') \geq \mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U_{E,\mathcal{P}}, \mathsf{tokens})$$

as desired. □

The intuition behind this result is that, as long as the delegates are not "too big", delegation actually has a decentralizing effect. Conversely, if some delegate is a whale, or gets delegated an overwhelming majority of tokens, then the system may become more centralized. Thus, delegation is most useful in cases where apathy is high. This idea is captured by the following corollary:

Corollary 1. If, in Theorem 3.4, there exists some delegate $d \in D$ such that $\mathsf{tokens}'([d]) \geq \mathsf{tokens}(\hat{P})$, then $\mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U'_{E,\mathcal{P}}, \mathsf{tokens}') \leq \mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U_{E,\mathcal{P}}, \mathsf{tokens})$.

In practice, it is common for delegates to be small relative to the inactivity whale [31], but this, of course, need not always be true.

3.5 Herding

A core goal of DAOs—and any democratic system more broadly—is for token holders to vote according to their true preferences. In practice, however, many DAOs fail to meet this goal and instead exhibit *herding* behavior. Specifically, when votes are publicly observable, social dynamics lead to the formation of "coalitions" of voters. For example, token holders have reported feeling influenced to vote a certain way, often in alignment with influential community members, in order to thwart the reputational risks associated with opposing popular viewpoints [72]. Similarly, it has been observed and measured that token holders often vote in alignment with their peers [63], who now operate as a single, large entity. In both cases, the monetary utility derived from the social impact of a player's vote skews the monetary utility of her desired outcome in a vacuum.

Herding leads to more centralization, as votes artificially converge on one outcome. Token distribution alone, however, does not show this. Indeed, a system where tokens are distributed evenly, but all players vote for the same outcome due to herding, would be deemed optimally decentralized according to such metrics. Conversely, VBE does conclude that reputational risks lead to more centralization, as it aligns the utilities of the players towards the socially preferred outcome.

Theorem 3.5 (Herding and VBE)**.** Let $(E, U'_{E,\mathcal{P}}, \mathsf{tokens}) = T_{\mathtt{herd}}(\mathcal{P}, E, U_{E,\mathcal{P}}, \mathsf{tokens})$ be the transformation where players $\hat{\mathcal{P}} \subseteq \mathcal{P}$ exhibit herding toward, without loss of generality, $\mathtt{true}$. That is, for all $P \in \hat{P}$ and $e \in E$, the monetary reputational cost of voting for $\mathtt{false}$ is greater than or equal to $\max(2 \cdot \mathsf{util}_P(e, \mathtt{false}]) + \epsilon, 0)$ for some constant ϵ. The rest of the system remains unchanged. Then, it follows that

$$\mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U_{E,\mathcal{P}}, \mathsf{tokens}) \geq \mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U'_{E,\mathcal{P}}, \mathsf{tokens}).$$

Proof. Let B be the largest voting bloc by token holdings before $T_{\mathtt{herd}}$. Note that, after $T_{\mathtt{herd}}$, all voters in $\hat{P}$ belong to the same voting bloc B': for every $P \in \hat{\mathcal{P}}$, $U'_{E,P}$ will consist of only positive values: either P preferred an outcome of $\mathtt{true}$ in e to begin with, or their monetary utility of $\mathtt{true}$ is now $|\mathsf{util}_P(e, \mathtt{false})| + \epsilon$. Thus, since $\operatorname{sgn}(\mathsf{util}_P(e, \mathtt{true})) = 1$ for all $e \in E$, all of $\hat{P}$ consists of a single voting bloc B' according to ϵ-TOC.

It follows then that $\mathsf{tokens}(B') \geq \mathsf{tokens}(B)$, as either the "new" voting bloc B' is now the largest bloc, or the same bloc is the largest before and after $T_{\mathtt{mirr}}$. Then, from Theorem 3.1, it follows that

$$\mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U_{E,\mathcal{P}}, \mathsf{tokens}) \geq \mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U'_{E,\mathcal{P}}, \mathsf{tokens})$$

as desired. □

An important conclusion of this theorem is that privacy instead *increases* the decentralization of a system, as it serves as a "mitigation" to herding. That is, if votes are private, token holders can vote for their true preferences, instead of being influenced by, e.g., social optics or the votes of their peers. (We omit a formal proof of this corollary, as it follows directly via a proof by contradiction of Theorem 3.5).

3.6 Voting slates

Grouping together various elections into a lesser number of (more general) elections—so-called "voting slates"—is in opposition with decentralization: decision-making is more diluted, thus decreasing the relative impact of each voter in the underlying proposals. That is, voting slates "factor out" differences in the viewpoints of individuals, yielding more homogeneous utilities. For example, two players may disagree in many of the individual proposals, but agree on a few of the more important ones, resulting in them casting the same overall vote.

We model a player's utility for a slate of elections simply by adding the utilities of the underlying proposals. That is, for all $P \in \mathcal{P}$ and some election $\mathcal{E}$ comprised of some subset of elections of E, the utility of P in $\mathcal{E}$ is:

$$\mathsf{util}_P(\mathcal{E}, \texttt{true}) = \sum_{e \in \mathcal{E}} \mathsf{util}_P(e, \texttt{true}).$$

Voting slates are generally used to "hide" unpopular or proposals among a larger set of benign, popular proposals, and thus increase their chances of passing. We model this by saying that if two P_i, P_j have aligned utilities (according to ϵ-TOC) on all proposals underlying $\mathcal{E}$, then they will agree on $\mathcal{E}$ itself, i.e.,

$$C_\epsilon(U_{E,P_i}, U_{E,P_j}) = 1 \implies \operatorname{sgn}(\sum_{e \in \mathcal{E}} \mathsf{util}_{P_i}(e, \texttt{true})) = \operatorname{sgn}(\sum_{e \in \mathcal{E}} \mathsf{util}_{P_j}(e, \texttt{true}))$$

As we show below, VBE reflects the fact that bundling proposals indeed decreased decentralization: by considering a narrower set of elections, which smoothens utility functions, different voting blocs are combined to form larger ones.

Theorem 3.6 (Voting Slates and VBE)**.** Let $(E', U'_{E',\mathcal{P}}, \mathsf{tokens}) = T_{\texttt{slates}}(\mathcal{P}, E, U_{E,\mathcal{P}}, \mathsf{tokens})$ be the transformation where all elections E are bundled together into slates to form a smaller set of elections E'. The rest of the system remains unchanged. Then, it follows that

$$\mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U_{E,\mathcal{P}}, \mathsf{tokens}) \geq \mathsf{VBE}_{C_\epsilon,\min}(E', \mathcal{P}, U'_{E',\mathcal{P}}, \mathsf{tokens}). \tag{1}$$

Proof. Let B be the largest voting bloc by token holdings before $T_{\texttt{slates}}$. Then, note that all players in B are still in the same voting bloc B' after $T_{\texttt{slates}}$: since $C_\epsilon(U_{E,P_i}, U_{E,P_j}) = 1$ for every pair of players in B, by assumption, it follows that

$$\forall \mathcal{E} \in E',\ \operatorname{sgn}(\sum_{e \in \mathcal{E}} \mathsf{util}_{P_i}(e, \texttt{true})) = \operatorname{sgn}(\sum_{e \in \mathcal{E}} \mathsf{util}_{P_j}(e, \texttt{true})).$$

Conversely, players who did not belong to B may, in fact, join B' after $T_{\texttt{slates}}$: even if the players disagree in some of the underlying proposals for a particular slate $\mathcal{E}$, they may cast the same overall vote for the entire slate. As such, B' contains strictly more players than B, which implies that $\mathsf{tokens}(B') \geq \mathsf{tokens}(B)$. Then, from Theorem 3.1, it follows that

$$\mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U_{E,\mathcal{P}}, \mathsf{tokens}) \geq \mathsf{VBE}_{C_\epsilon,\min}(E', \mathcal{P}, U'_{E',\mathcal{P}}, \mathsf{tokens})$$

as desired.

□

3.7 Bribery

There is an intuitive relationship between decentralization and bribery, namely, that successful bribery poses a threat to decentralization: in such a case, the entity that acquires the votes of the other players now controls a higher proportion of the total tokens than before. However, traditional decentralization metrics, i.e., based on token distribution across accounts, fail to capture this fact: bribed voters, albeit casting votes as instructed by the briber, still technically hold their tokens. Conversely, VBE groups all bribed voters in the briber's bloc, as all bribee's now have aligned utility functions, in line with the bribers desired outcome.

We note that, interestingly, similar to our result from Section 3.3, bribery can have the surprising consequence of leading to a more *decentralized* system, in the case where it fragments a larger bloc (say, the inactivity whale, or some large coalition of voters). However, we ignore this edge case and assume instead that the bloc of bribed voters represents a majority by token holdings. (In particular, for the inactivity whale, it would be rational for all apathetic voters to accept a bribe, in which case the entire inactivity whale is absorbed.) As such, even though bribery need not, unconditionally, increase centralization, it poses a practical *threat* to decentralization.

Theorem 3.7 (Bribery and VBE)**.** Let $(E, U'_{E,\mathcal{P}}, \mathsf{tokens}) = T_{\texttt{bribe}}(\mathcal{P}, E, U_{E,\mathcal{P}}, \mathsf{tokens})$ be the transformation where an entity successfully bribes players $\hat{\mathcal{P}} \subseteq \mathcal{P}$ in elections E to achieve an outcome of, without loss of generality, `true`. The rest of the system remains unchanged. Then, it follows that

$$\mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U_{E,\mathcal{P}}, \mathsf{tokens}) > \mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U'_{E,\mathcal{P}}, \mathsf{tokens}). \tag{2}$$

Proof. Let B be the largest voting bloc by token holdings before $T_{\texttt{bribe}}$. First, note that, after $T_{\texttt{bribe}}$, all voters in $\hat{\mathcal{P}}$ belong to the same voting bloc B'. Recall that, in our DAO abstraction, bribing a player P to flip its vote in election e from `false` to `true` costs $\max(2\cdot \mathsf{util}_P(e, \texttt{false})+\epsilon, 0)$. So, for every $P \in \hat{\mathcal{P}}$ and $e \in E$, either $\mathsf{util}_P(e, \texttt{true})$ was already positive to begin with, or it is now $|\mathsf{util}_P(e, \texttt{false})| + \epsilon$. Then, since $\operatorname{sgn}(\mathsf{util}_P(e, \texttt{true})) = 1$ for all $e \in E$, all of $\hat{P}$ consists of a single voting bloc B' according to ϵ-TOC.

It follows then that $\mathsf{tokens}(B') \geq \mathsf{tokens}(B)$, as either the "new" voting bloc B' is now the largest bloc, or the same bloc is the largest before and after $T_{\texttt{bribe}}$. Then, from Theorem 3.1, it follows that

$$\mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U_{E,\mathcal{P}}, \mathsf{tokens}) \geq \mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U'_{E,\mathcal{P}}, \mathsf{tokens})$$

as desired. □

Scale of bribery and decentralization. A second, more nuanced observation is that successful bribery must be systemic, i.e., must involve a large number of tokens, if (and only if) a system is highly decentralized. Intuitively, if a DAO is highly centralized, a briber can directly coordinate with a few large players to guarantee an election outcome; or, if the briber is a whale herself, she only needs to bribe a few of the smaller players to accumulate enough tokens to mount a successful attack. Instead, in a more decentralized system, players are smaller, so a briber needs to widen the scale of their attack if they want to win an election. That is, in this case, successful bribery requires large-scale coordination among various smallholders.

Theorem 3.8 (Internal Bribery and VBE)**.** Let $(E, U'_{E,\mathcal{P}}, \mathsf{tokens}) = T_{\mathtt{bribe}}(\mathcal{P}, E, U_{E,\mathcal{P}}, \mathsf{tokens})$ be the transformation where $U'_{E,\mathcal{P}}$ is some arbitrary change in the utilities of the players. The rest of the system remains unchanged. Assume that an entity in $\mathcal{P}$ needs to bribe other players holding a total of at least n_1 and n_2 tokens to guarantee an outcome of `true` in elections E before and after $T_{\mathtt{bribe}}$, respectively. Then, it follows that

$$n_1 > n_2 \iff \mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U'_{E,\mathcal{P}}, \mathsf{tokens}) < \mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U_{E,\mathcal{P}}, \mathsf{tokens}). \tag{3}$$

Proof. We first make the trivial observation that the minimum number of tokens that must be bought to guarantee an election outcome occurs when the bribing entity belongs to the largest voting bloc by token holdings. Let B_1 and B_2 be such blocs before and after $T_{\mathtt{bribe}}$, respectively. By Theorem 3.1, we get that

$$\mathsf{tokens}(B_2) > \mathsf{tokens}(B_1) \iff \mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U'_{E,\mathcal{P}}, \mathsf{tokens}) < \mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U_{E,\mathcal{P}}, \mathsf{tokens}).$$

Then, note that, for $i \in \{1, 2\}$,

$$n_i = q \cdot \sum_{P \in \mathcal{P}} \mathsf{tokens}(P) - \mathsf{tokens}(B_i)$$

It thus follows that $n_1 > n_2 \iff \mathsf{tokens}(B_2) > \mathsf{tokens}(B_1)$, i.e.,

$$n_1 > n_2 \iff \mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U'_{E,\mathcal{P}}, \mathsf{tokens}) < \mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U_{E,\mathcal{P}}, \mathsf{tokens})$$

as desired.

□

The theorem above shows how, as a DAO becomes more decentralized, a higher number of tokens need to be corrupted to guarantee an election outcome, since all players are small to begin with. Conversely, in a more centralized DAO, large whales only need to corrupt a few tokens to guarantee their desired election outcome.

This result sheds light on the scale of bribery in the case where the briber is a malicious tokenholder a priori. Conversely, the briber may instead be some external entity. In this case, decentralization also raises the risk of systemic bribery: if there are large players in the system, the briber can directly coordinate with whales to achieve their desired election outcome. If, however, the DAO is highly decentralized, the outcome of the election depends on many stakeholders, which thus requires large-scale coordination among these. More formally:

Theorem 3.9 (External Bribery and VBE)**.** Let $(E, U'_{E,\mathcal{P}}, \mathsf{tokens}) = T_{\mathtt{bribe}}(\mathcal{P}, E, U_{E,\mathcal{P}}, \mathsf{tokens})$ be the transformation where $U'_{E,\mathcal{P}}$ is some arbitrary change in the utilities of the players. The rest of the system remains unchanged. Let n_1 and n_2 be the minimum number of players that an external entity needs to corrupt to guarantee an outcome of `true` in elections E before and after $T_{\mathtt{bribe}}$, respectively. Then, it follows that

$$n_1 > n_2 \iff \mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U'_{E,\mathcal{P}}, \mathsf{tokens}) < \mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U_{E,\mathcal{P}}, \mathsf{tokens}). \tag{4}$$

Proof. This proof is very similar to that of Theorem 3.8. Let B_1 and B_2 be the largest blocs by token holdings before and after $T_{\texttt{bribe}}$, respectively. By Theorem 3.1, we get that

$$\mathsf{tokens}(B_2) > \mathsf{tokens}(B_1) \iff \mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U'_{E,\mathcal{P}}, \mathsf{tokens}) < \mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U_{E,\mathcal{P}}, \mathsf{tokens}).$$

Then, note that, for $i \in \{1, 2\}$:

$$n_i = \frac{q \cdot \sum\limits_{P \in \mathcal{P}} \mathsf{tokens}(P)}{\mathsf{tokens}(B_i)}$$

It thus follows that $n_1 > n_2 \iff \mathsf{tokens}(B_2) > \mathsf{tokens}(B_1)$, i.e.,

$$n_1 > n_2 \iff \mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U'_{E,\mathcal{P}}, \mathsf{tokens}) < \mathsf{VBE}_{C_\epsilon,\min}(E, \mathcal{P}, U_{E,\mathcal{P}}, \mathsf{tokens})$$

as desired.

□

We make the important note that, to acquire a fixed number of target tokens (i.e., in the case where the briber is an external actor), bribing a larger set of smaller players is, in fact, *cheaper* than bribing a smaller set of whales to acquire the same number of tokens. Intuitively, larger players are more "pivotal" [80], i.e., have a greater influence on election outcomes, and thus are more expensive to bribe. As such, decentralization *decreases* the cost to mount a bribery attack on a DAO. We discuss this in detail in Section 5.3).

Systemic bribery has long been recognized as one of the main threats to traditional elections, and we have now shown that this is also the case for DAOs. However, bribery is not considered a realistic concern in secret ballot elections, due to the fact that such large scale vote buying would be logistically and economically infeasible to coordinate, and would be traceable. Further, rational vote sellers would simply take the bribe but still vote according to their preferences, instead of following the briber's demands. Looking ahead, we will show in Section 5 that, conversely, bribery in DAOs can be done via cost-free, untraceable mechanisms, which guarantee fair exchange. Thus, bribery is a realistic and practical threat for DAO elections.

3.8 Quadratic Voting

Quadratic voting [52] is a voting mechanism that attempts to dilute the influence of whales on election outcomes. To do so, a vote from a player that owns n tokens will only have an impact of $\sqrt{n}$ in the outcome election. At face value, quadratic voting seems to make a system more decentralized: the quadratic "tax" is directly proportional to the number of tokens a player owns, which thus shrinks the gap between smaller players and whales. However, quadratic voting is known to be vulnerable to sybil attacks and other forms of malicious coordination [80, 68], and thus may have a *centralizing* effect: players that are in large voting blocs implicitly subvert the quadratic tax due to the fact that their true token count is split amongst all bloc members. As a concrete example, consider a quadratic voting system with no verification of real-world identities. In this case, a whale can divide her tokens amongst multiple accounts, which increases the impact that her votes have on the election outcome. (In fact, as we show in Section 5.3, a whale can subvert quadratic voting even in the presence of robust identity mechanisms.)

Traditional DAO decentralization metrics fail to capture this attack on quadratic voting, since they do not reason about the relationship between the individual accounts. Conversely, VBE would group together all the accounts under the control of the same entity as part of the same voting bloc, and thus concluding that decentralization has decreased. This is analogous to our result from Theorem 3.2, which shows that, in general, splitting tokens across multiple accounts does not increases decentralization. In the case of quadratic voting, this mechanisms actually strictly decreases decentralization, since the votes of this smaller accounts have a higher impact on the election outcome.

Quadratic voting and bribery. Similar to a whale splitting off her tokens into multiple accounts, a set of colluding players can have a greater impact in the election outcome if quadratic voting is employed. Namely, quadratic voting may have the surprising consequence of *decreasing* the cost of bribery. The high-level idea is that quadratic voting "amplifies" the power of small accounts, which may be cheaper to bribe. Thus, for the same cost, a briber is able to have a greater impact on the outcome of an election.

Prior work has informally identified these issues in the context of traditional elections [80, 68], and DAOs specifically [32]. For the former, as discussed at the end of Section 3.7, large-scale collusion is not considered a realistic threat, and thus the fact that bribery can have a bigger impact on an election outcome if quadratic voting is employed is not seen as a practical limitation. However, since our Dark DAO prototype from Section 5 makes bribery inexpensive and guarantees fair exchange, bribery poses a realistic threat to quadratic voting (and any blockchain-based voting scheme for that matter).

Our formalism captures this relationship between quadratic voting and bribery. We define "small" accounts to be, concretely, those whose fraction of the total tokens increases with quadratic voting in place, and thus have their impact amplified. More formally, we denote that a player $P \in \mathcal{P}$ benefits from quadratic voting by $\mathsf{quad}(P, \mathsf{tokens}) = 1$, where

$$\mathsf{quad}(P, \mathsf{tokens}) = 1 \iff \frac{\mathsf{tokens}(P)}{\sum\limits_{p \in \mathcal{P}} \mathsf{tokens}(p)} < \frac{\sqrt{\mathsf{tokens}(P)}}{\sum\limits_{p \in \mathcal{P}} \sqrt{\mathsf{tokens}(p)}}.$$

The relationship between quadratic voting and bribery hinges on whether the cost to bribe a player is the same with or without quadratic voting. That is, does the monetary utility of a player in an election change if the impact of their vote changes? If not, then the cost to bribe a small player is the same, but the impact is greater. Otherwise, the cost to bribe the player increases as their relative power increases, which offset each other.

Whether quadratic voting changes a player's utility or not will vary across systems. Broadly speaking, if DAO members take governance seriously and are invested in election outcomes, quadratic voting indeed changes utilities: since smaller accounts become more "pivotal" as a result of quadratic voting, their utilities increase correspondingly. Conversely, if members have little interest in governance, the fact that their vote can now have a greater impact in the election will not change their utilities. As such, the nature of a community must be taken into account when deciding to use quadratic voting.

Theorem 3.10 (Quadratic Voting and Bribery)**.** Let $(E', U_{E',\mathcal{P}}, \mathsf{tokens}) = T_{\mathsf{quad}}(\mathcal{P}, E, U_{E,\mathcal{P}}, \mathsf{tokens})$ be the transformation where all elections E employ quadratic voting. We denote the election corresponding to $e \in E$ by $e' \in E'$. Let f and f' be the fraction of total votes that a bribing entity is able to control for some fixed expenditure t in elections E and E', respectively. Then, it follows that

$$f < f' \iff \exists \hat{\mathcal{P}} \subseteq \mathcal{P} \mid \forall P \in \hat{\mathcal{P}},\ \Big(\mathsf{quad}(P, \mathsf{tokens}) = 1 \land U_{E,P} = U_{E',P}\Big)$$

Proof. Recall from Section 2 that the cost of bribing all players in $\hat{P}$ to vote for, without loss of generality, `false` in all elections E is

$$t = \sum_{p \in \mathcal{P}} \sum_{e \in E'} \max(2 \cdot \mathsf{util}_P(e, \texttt{false}) + \epsilon, 0)$$

Since, by assumption, $\forall P \in \hat{\mathcal{P}}\ \forall e \in E$, $\mathsf{util}_P(e, \texttt{false}) = \mathsf{util}_P(e', \texttt{false})$, it follows that the cost to bribe all players in $\hat{P}$ across all elections E' is also t. Then, since all players in $\hat{P}$ benefit from quadratic voting, the bribing entity has thus managed to control a larger fraction of the total votes for this same expenditure: by definition, for all $P \in \mathcal{P}$,

$$\begin{aligned} & \frac{\mathsf{tokens}(P)}{\sum\limits_{p \in \mathcal{P}} \mathsf{tokens}(p)} < \frac{\sqrt{\mathsf{tokens}(P)}}{\sum\limits_{p \in \mathcal{P}} \sqrt{\mathsf{tokens}(p)}} \\ \iff \quad & \sum_{P \in \hat{\mathcal{P}}} \Big(\frac{\mathsf{tokens}(P)}{\sum\limits_{p \in \mathcal{P}} \mathsf{tokens}(p)} \Big) < \sum_{P \in \hat{\mathcal{P}}} \Big(\frac{\sqrt{\mathsf{tokens}(P)}}{\sum\limits_{p \in \mathcal{P}} \sqrt{\mathsf{tokens}(p)}} \Big) \\ \iff \quad & f < f' \end{aligned}$$

as desired. □

This result thus shows that quadratic voting may be favorable for a bribing entity. In particular, if there are enough small voters whose utilities are unchanged, the cost to guarantee successful bribery decreases:

Corollary 2. Assume that, for $\hat{P}$ as defined in Theorem 3.10, $\mathsf{tokens}(\hat{P}) > q \cdot \sum_{P \in \mathcal{P}} \mathsf{tokens}(P)$. Let t and t'' be the expenditure required to guarantee an outcome of `true` in elections E and E', respectively. Then, it follows that $t' < t$.

This corollary simply follows from the fact that, as proved in Theorem 3.10, the expenditure t' required to control a fraction of q votes in E', and thus guarantee successful bribery in E', would only be enough to acquire a fraction of $q - \epsilon$ votes in E. As such, some additional expenditure is required to cross the threshold of q votes.

4 Practical Considerations of VBE

Since VBE is a theoretical metric, it serves primarily as a conceptual tool to reason about how certain decisions or events may impact the decentralization of DAOs. However, VBE can also be estimated by using real-world data alongside a particular instantiation of our framework. In this section, we discuss some directions towards this, and the limitations of this approach (and our model more broadly). We note, however, that an empirical study of DAOs, such as concretely computing VBE for popular DAOs based on on-chain data, is left as future work.

Limitations of our formal model. Our DAO abstraction from Section 2, which underpins VBE, makes several assumptions, which need not be true in practice. For example, we only consider binary elections, whereas DAO proposals can involve multiple options, e.g., Optimism's process for (retroactively) funding public goods, where votes explicitly expressed the allocation of funds to different organizations [67]. We note, however, that our model can be reframed to consider arbitrary elections, as this would simply involve using a clustering metric for VBE that takes into account multiple potential election outcomes when partitioning $\mathcal{P}$.

A more important limitation is the fact that we assume token holdings remain constant across elections. For simplicity and clarity of presentation, we deem this to be sufficient due to the conceptual nature of our theoretical results. Further, our model can be modified to assume variable token holdings. For example, $\mathsf{tokens}(P)$ can be defined to be the maximum number of tokens held by P at any point across all elections E. We leave such extensions as future work.

Measuring VBE. As we have emphasized throughout this work, VBE is a theoretical notion and cannot be measured directly. This is due to the fact that utility functions are *latent variables* [42], which are not directly measurable. This is an inherent limitation of any metric that depends on utility functions, including important results and models from voting theory, e.g., [80, 35].

Due to this limitation, VBE is most useful as a theoretical tool to estimate the directional impact of policy choices in decentralization. However, VBE does lend itself to indirect measurement: latent variables, such as utility functions, can be estimated via *observable variables*, which are indeed measurable. In our context, observable variables may be gathered from on-chain data (such as past voting history), low-cost straw polls, social dynamics, etc.

The accuracy of estimating VBE via observable variables will depend, to a large extent, on the specific clustering algorithm with which VBE is instantiated, and how much "information" is required from the utility funtions in order to partition $\mathcal{P}$. For example, a trivial partition of $C(U_{E,P_i}, U_{E,P_j}) = 1 \iff U_{E,P_i} = U_{E,P_j}$ will yield a less accurate estimate than some other function which only takes into account the voting history of the players.

Practitioners can thus use VBE to derive concrete metrics, and analyze the real-world behavior of a system, by initialize our framework with a particular clustering metric and entropy function, and using observable variables to estimate utility functions of players. Deciding which clustering and entropy functions to use requires careful consideration of what data is available. In general, more granular notions of VBE are, of course, more informative. For example, Shannon entropy yields more nuanced results than min-entropy, and a clustering metric based on cardinal utilities will result in blocs of players that are more closely aligned. However, such functions may require data that is not easy to gather, or may not even be tractable at all. As such, there is a trade-off between how informative VBE is, and how easy it is to compute.

In the particular case of $\mathsf{VBE}_{C_e,\min}$, for example, historical voting data is sufficient for ϵ-TOC, since clusters are assigned based on ordinal utility. If we assume voters are rational, we can extrapolate this from the casted votes: for any player P and election e, it follows that

$$\begin{aligned}
\mathsf{vote}_P(e) = \mathtt{true} &\iff \mathsf{util}_P(e, \mathtt{true}) > \epsilon \\
\mathsf{vote}_P(e) = \mathtt{false} &\iff \mathsf{util}_P(e, \mathtt{false}) > \epsilon \\
\mathsf{vote}_P(e) = \bot &\iff |\mathsf{util}_P(e, \mathtt{true})| < \epsilon.
\end{aligned}$$

Even though we cannot extrapolate the exact value of $\mathsf{util}_P(e, \mathtt{true})$ based on election outcomes, the equations above are sufficient to use C_ϵ to cluster players. We stress, however, that different instantiations of the VBE framework may require different measurement techniques.

Another natural limitation of VBE is that, given that it is a framework, two instances of VBE are not directly comparable. That is, in order to reason about the relative level of decentralization between two DAOs, or how decentralization has fluctuated over time for a single DAO, the same variant of VBE must be used. In practice, however, we expect that broad VBE adoption would involve a handful of standard parameters agreed upon by the DAO community.

Limitations of $\mathsf{VBE}_{C_\epsilon,\min}$. In addition to the general limitations described above, each variant of VBE may pose additional constraints. In the case of $\mathsf{VBE}_{C_\epsilon,\min}$, we lose most of the information provided by all voting blocs except the largest one, since min-entropy is only a function of the latter. This does not imply that the analysis is not accurate, as all subsequent blocs are strictly smaller than the one our definition focuses on, but rather that other entropy notions may yield additional insights; indeed, min-entropy is always less than or equal to Shannon entropy and max-entropy [81].

Our clustering metric, ϵ-TOC, is also quite strict, as it does not reason about voters who are aligned in most (but not all) elections. One could instead consider, for example, a generalization ϵ-threshold ordinal clustering that is parametrized by the fraction of elections two players must agree on to be considered part of the same cluster. We opted for the simpler variant in this work, as it serves as a proof-of-concept for our theoretical results; more general clustering metrics would yield the same conceptual conclusions, while making the theorem statements and proofs more opaque with orthogonal mathematical details.

Data Collection. Even though blockchain-based elections are public, extrapolating relevant observable data for analyzing VBE (and other decentralization metrics) is surprisingly difficult. Indeed, as prior work has also pointed out, "in practice it is not trivial to acquire all governance related information from raw blockchain data" [39]. To aid the analysis and computation of VBE, we thus propose that DAOs publish relevant statistics in a way that is easy to understand and use.

We propose that DAOs choose and specify a variant of VBE to support, which then guides how to present voter data. For ϵ-TOC, DAOs can keep a log mapping all token holders to a list of the elections they were eligible for, denoting their vote (if any). Other variants of VBE may require more detailed information. Feichtinger et al. [39] successfully extrapolated a vast amount of governance-related data from 21 DAOs along multiple axes (albeit noting that it was surprisingly challenging), and open-sourced their data set in the form of "subgraphs" from The Graph protocol [4]. Their work may serve as inspiration for user-friendly ways to present voter data: each DAO could implement a publicly-accessible subgraph of governance data, maintained either by a dedicated set of curators or by the community more broadly.

5 Dark DAOs: Overview

A *Dark DAO* is itself a DAO, but one whose objective is to subvert a system of decentralized credentials and thereby to target, e.g., voting in another DAO or DAOs. Dark DAOs were first introduced in a 2018 blog post [33]. We have shown in Section 3 that as decentralization increases, the cost of a bribery attack rises. Consequently, the need arises for a briber to perform broad coordination, as there is a need to target more users. Thus the threat of Dark DAO deployment increases.

In this section, we briefly explain what Dark DAOs are, giving an informal definition in Section 5.1. We outline their main design principle, *key encumbrance*, in Section 5.2. We explain the various ways in which they can disrupt votes in target DAO Section 5.3.

5.1 Dark DAO Definition

A *Dark DAO* is defined specifically in [33] as a "decentralized cartel that buys on-chain votes opaquely." We believe that a broader definition is more informative—one that encompasses any corruption of any system of credentials, whether used for voting or other purposes. Like an ordinary DAO, a Dark DAO designed to be trust-minimized: it ensures that a bribe is "fair," i.e., a bribee receives money from a briber iff the briber gains agreed-upon access to the bribee's credential(s). Additionally, a Dark DAO is "opaque" in the sense of ensuring that participation is *private*.

Informally, then, a Dark DAO has the following three key properties:

1. **Opacity:** Participants in a Dark DAO are indistinguishable on chain from other credential holders. (Consequently, statistics like the number of participants in a Dark DAO are also hidden.)

2. **Fair exchange:** Once a bribee commits to accepting a briber's offered bribe, the briber obtains access to the bribee's credential and the bribee is paid the bribe.

3. **Bounded scope:** A bribee who participates in a Dark DAO contributes no resource to the Dark DAO beyond a committed credential and pre-agreed-upon costs. (E.g., the bribee may also pay normal transaction fees.)

Example (voting): A Dark DAO that aims to corrupt voting in a target DAO would involve voters (bribees) selling their votes to a vote buyer (briber). *Opacity* would mean that bribees are indistinguishable from other voters in the target DAO. *Fair exchange* would mean that the briber pays a pre-agreed-upon amount to a bribee iff the bribee's vote is cast as the briber prescribes in a particular election. Finally, *bounded scope* means in this context that the bribee can use her voting credential in an unrestricted way outside of the election in question.

Remark: Fair exchange requires not just that a briber gain access to a user's credential, but that the credential be usable in a pre-agreed upon way. For example, if the briber gains access to the bribee's credential, and the bribee is paid, but bribee can revoke the credential before use by the briber, then the exchange is not fair. To capture such subtleties, a more formal definition of fair exchange may be couched in terms of a universe of possible target-system states $\mathcal{S}$ and a transcript $T = \{S_1, S_2, \ldots\}$ for $S_i \in \mathcal{S}$ of state transitions. Fair exchange means that for $\mathcal{S}$ and a set of transcripts $\mathcal{T}$—both agreed upon by the briber and bribee—$T \in \mathcal{T}$ for the transcript T of the target system's history. We defer the development of such formalism to future work.

5.2 Main tool: Key encumbrance

The main mechanism by which a Dark DAO achieves its properties is *key encumbrance* [51]. Key encumbrance is a form of life-cycle management for keys that facilitates selective delegation. In the case of voting, it enables a Dark DAO to ensure that delegated keys are used to cast the votes to which bribees have committed, but gives the Dark DAO no further control over encumbered keys.

There are two main tools that can enforce such delegation in principle in a way that does not require use of a trusted third party: secure multiparty computation (MPC) [66] and trusted execution environments (TEEs) [62, 61].

TEEs are the more practical, particularly as we demonstrate below that existing hardware-based TEE systems are sufficient to realize Dark DAOs. Such TEEs enable applications to run in an integrity- and confidentiality-protected environment.

Dark DAO running in a TEE creates an encumbered key, or imports an already encumbered key sk. The briber and the bribee can request use of sk from the dark dao, which ensures compliance with the Dark DAO policy, i.e., enforces the properties enumerated in Section 5.1. In the particular case of voting, the briber can use sk as a voting credential, while the bribee can use sk to manage their cryptocurrency and interact with smart contracts in a way that does not violate the fair exchange property.

5.3 Dark DAO goals

Globally, the goal of a Dark DAO is to subvert voting in a target DAO. There are a number of ways in which it can do this, of which we enumerate several here.

Vote buying: A briber desiring a given election outcome (e.g., a "yes" vote) can simply offer payment for votes toward this outcome—where payments are scaled to the weight associated with a given bribee's vote (e.g., proportional to her DAO token holdings). Various forms of conditional payment are also possible, e.g., paying bribes only if the desired outcome is achieved or offering a fixed payment for distribution across the total population of bribees.

We note too that vote buying works not just for systems in which votes are weighted by token holdings, but also "one-vote-per-person" systems, e.g., [65, 82]. In such cases, an encumbered key sk might be a user's credential in a decentralized identity system, e.g., in Gitcoin Passport or Worldcoin.

A Dark DAO can further increase the threat of so-called cost-less bribery. For instance, Bó [19] introduces "pivotal" bribes as a way to bribe voters at virtually no cost. Consider a binary vote where the final result is the option chosen by the majority of voters (for simplicity, assume an odd number of users n) and suppose the utility of a user for a "yes" vote is U (and for a "no" vote is $-U$). A briber, wishing to flip votes to "no" bribes the user as follows: If the user votes "no" and there are exactly $(n + 1)/2$ "no" votes (i.e., the voter is "pivotal" in the sense that the outcome would have changed if the voter chose "yes" instead), then the briber pays $P + \epsilon$ (for any $\epsilon > 0$). Otherwise, if the user votes "no" (regardless of the result), a bribe of ϵ is still paid. No bribe is paid if the user votes "yes." It is easy to see now that it is always individually rational for a user to take such a bribe—regardless of the result, the user's utility will be ϵ larger than if the bribe was not taken. But this means that if all users are rational, and the bribe is offered to everyone, then all users will vote "no" resulting in no pivotal voters and the cost of bribing being only $n\epsilon$ (which can be arbitrarily small). A major practical hurdle in deploying such a bribe is conducting is coordinating enough voters, which is made significantly easier by a Dark DAO. In essence, a Dark DAO can make such a "pivotal"-bribe attack extremely cheap.

Coordinated price manipulation: As noted in [33], it is possible for a Dark DAO to operate without an explicit party distributing bribes. A Dark DAO can instead orchestrate collective action that rewards participants indirectly.

For example, a Dark DAO can orchestrate the following steps among a cabal: (1) Purchase a collective short position in a target asset X; (2) Vote collectively toward an outcome that causes the price of X to drop; (3) Close the short position at a profit; and (4) Distribute profits among Dark DAO participants. Dark DAO goals can in principle extend beyond voting to other actions as well, such as coordinated attacks on consensus protocols [33] or—if the Dark DAO ingests assets from participants—market manipulation.

Undermining perceived election integrity: The mere presence of a Dark DAO may be enough to cast doubt on a DAO election and call into question whether it is being attacked. DAO opacity conceals the size of a Dark DAO such that even with limited participation, a Dark DAO could impact community trust in an election.

Alternatively, a Dark DAO could selectively reveal (and prove) statistics—e.g., participation of at least 10% of token holdings—that would substantiate the threat it poses.

Exploiting quadratic voting and quadratic funding: Quadratic voting [52] is a mechanism that seeks to limit the influence of whales in determining election outcomes. It weights a given voter's vote as the square root of her token balance.

Quadratic voting is only enforceable if tokens are assignable to real-world identities. For instance, if votes are weighted as the square-root of token holdings by address, a whale can boost her voting weight by dividing her tokens among multiple accounts.

A Dark DAO can subvert quadratic voting *even when vote is conducted using a secure decentralized identity system.* That is because a Dark DAO can encumber keys not just in a way that enforces voting choices, but *also* use of digital assets.

A whale can thus subvert a quadratic voting scheme as follows. The whale does not just bribe voters to vote for a particular outcome, but also *temporarily deposits some of the whale's funds with them.* As bribees' keys are encumbered, the Dark DAO can ensure not just that they vote as directed, but that they will return the whale's funds.

For example, a whale with 256 tokens can deposit 4 tokens with each of 63 distinct bribees. The result would be an increase in the whale's voting weight of a factor of $(\sqrt{4} \times 64)/\sqrt{256} = 8$.

A similar attack is possible against quadratic funding [24].

Subverting privacy pools: *Privacy pools* [27] aim to strike a balance between privacy and accountability in privacy coins and privacy services for cryptocurrencies. A privacy pool is an association of users of a certain class that acts as an anonymity set for members' transactions.

For example, a pool might require members to prove that they are not on a sanctions list (e.g., from the U.S. Office for Foreign Asset Control (OFAC), a requirement for most banks [43]). Pool membership then implies sanctions compliance, enabling a user to provide assurance that she is not sanctioned, while still preserving her privacy.

Any set of users may choose to create a pool. Membership requirements for a pool are determined by the community making up the pool. In this sense, a pool is like a DAO. It is also subject to attack by a Dark DAO.

A Dark DAO can target a privacy pool by facilitating *identity-selling* and thus selling of access to a pool. A privacy-pool member ("lender") can sell temporary access to her pool-compliant address to an adversary ("borrower") who isn't eligible for pool membership. For example, a user in a sanctions-compliant pool can sell pool access to another user who is in fact on a sanctions list. To do so, the seller encumbers her address so that it is subject to limited control by the buyer.

Example: Alice holds a Dark DAO address a that is a member of sanctions-compliant privacy pool P. Mallory will be receiving money from an address z. Alice agrees to help Mallory launder the money through P.

Alice sets the Dark DAO policy for address a so that when funds are received from z: (1) 99% of funds are subject to control by Mallory, i.e., Mallory can send those funds from a to any other desired address and (2) 1% of funds are subject to control by Alice—as payment for Mallory's borrowing of a.

Interestingly, a Dark DAO can conversely *reinforce* the security of a privacy pool by enforcing a policy in which members' addresses must maintain a minimum balance for some period of time. Such a policy would limit weakening or dissolution of the pool.

6 Basic Dark DAO

To illustrate that Dark DAOs are practical, we have implemented a Dark DAO prototype. Our implementation is written in what is currently the most popular smart-contract programming language, Solidity. Furthermore, while it uses TEEs, it demonstrates that developing a Dark DAO need not require any special knowledge of TEEs, thanks to the abstractions provided by the TEE-based Oasis Sapphire blockchain.

Our prototype demonstrates in particular how Dark DAOs might coordinate bribery on a popular off-chain voting platform, Snapshot [10]. To the best of our knowledge, however, all current DAO voting platforms are susceptible to Dark DAO interference. Our open-source implementation can be found at `https://github.com/DAO-Decentralization/dark-dao`.

In what follows, we first give a background on Oasis Sapphire and Snapshot (Section 6.1) followed by the details of our Dark DAO design (Section 6.2) and possible future design enhancements. We then discuss the cost of participation (Section 6.3), security (Section 6.4), deployment considerations (Section 6.5), mitigations against negative impacts of Dark DAOs (Section 6.6), and ethical the considerations of building a Dark DAO prototype (Section 6.7).

6.1 Background

Oasis Sapphire. Oasis Sapphire is an implementation of the Ethereum Virtual Machine that runs entirely inside a TEE. Assuming the TEE is not broken, Sapphire is able to execute smart contracts whose state is kept private both during execution (in memory) and after execution (in storage). In addition to matching the base implementation of the EVM, Sapphire also includes several built-in precompiled contracts that make it both easy and cheap to perform cryptographic operations pertinent to Dark DAOs: generating entropy and signing messages. These methods are not available in blockchains that do not use TEEs or encrypted computation, since the private key material or entropy would necessarily be leaked to all blockchain nodes.

In its current state, Sapphire does not provide confidentiality for all transaction metadata: senders and recipients of every transaction are public. Additionally, its persistent storage lies outside the TEE, making contract storage access patterns a vector for information leakage [47]. In this paper, we do not address side channel attacks such as these.

Sapphire is compatible with many cryptocurrency wallets today, making it a good candidate for hosting a key encumbrance system.

Snapshot. Snapshot is an open source, centralized, off-chain voting platform for DAOs. Rather than requiring DAO users to pay the costs of making on-chain voting transactions, it accepts votes submitted as signed messages to the Snapshot website. The website is organized into "spaces," typically one per DAO, each of which is moderated and/or controlled by a permissioned hierarchy of administrators and moderators of the DAO. At the time of writing, there exist over 28,000 Snapshot spaces [10].

Individual DAOs are free to adjust the algorithm used to calculate the weight of an individual vote, termed its *voting power*. How much voting power a particular user gets often is determined by how many DAO tokens he or she is holding on a blockchain at the moment a proposal is created, and thus a "snapshot" of voting power is taken at the corresponding block. For a given DAO proposal, the signed voting messages are collected and, once the voting period is over, are published as receipts to IPFS [1], a distributed file sharing network. Voters can verify that their votes were included in the proposal's outcome by checking their voting receipts.

Snapshot also provides a means for *delegating* one's voting power to another, presumably more active voter. A delegator can override his or her delegate's vote, but would generally choose a delegate based on the delegate's public reputation and likely voting profile.

6.2 A Key-Encumbrance Dark DAO Design

Recall the Dark DAO "guaranteed vote delivery" property (Section 5.1): a Dark DAO must guarantee that a bribed voter will cast a vote as specified by the briber. Many DAO voting systems, however, including Snapshot, allow voters to change their votes before a proposal has passed. Thus the only way to ensure guaranteed vote delivery is for the Dark DAO to control the voter's voting credential. At the same time, the "bounded scope" property of Dark DAOs (Section 5.1) means that the Dark DAO should have *limited access* to the voter's credential and be able to use it exclusively to cast the vote for which the voter has committed to receiving a bribe.

We resolve this tension by designing a *key-encumbered wallet*, which stores and manages private keys in smart contracts and enforces access-control policies that we refer to as *encumbrance policies*. A key-encumbered wallet simultaneously allows: (1) use of a key by a Dark DAO specifically for voting in response to a bribe and (2) unrestricted use of the key by its owner for any other purpose.

Key-encumbered wallet. Our key-encumbered wallet application is powered by a smart contract that runs on Oasis Sapphire. The smart contract generates private-public key pairs within a TEE using Sapphire's built-in entropy generation methods. Only the smart contract itself can use the keys it has generated to sign messages. To create a key-encumbered wallet, one can invoke a "create wallet" function in the smart contract using an external account, typically one that is not encumbered. We emphasize that while the aforementioned external account is the owner of the wallet, it does not have unfettered access to the private keys created and managed by the wallet smart contract.

Owners of key-encumbered wallets can request signatures for messages by issuing read-only calls to the wallet smart contract. These calls are signed by the owner's external account for authentication.

Dark-DAO encumbrance policy. To create our own Dark DAO based on key-encumbered wallets, we first designed a *key encumbrance policy* contract that regulates all Snapshot-related messages signed by an enrolled wallet, including votes for DAO proposals. The policy will not allow a key owner to sign a vote directly; instead, the owner must unlock the ability to do so for a particular proposal, after which it can sign any voting message related to that proposal. But rather than unlocking a proposal to sign a vote, an owner may delegate its right to vote to a sub-policy: this is the Dark DAO contract. If the vote is given to a sub-policy, the owner forgoes the ability to sign messages relating to that proposal. This mechanism guarantees to the Dark DAO that a user will not change a vote that it signs on the user's behalf. (A user could try to pre-sign a vote, but that would be impractical for, e.g., Snapshot, where ballots incorporate proposal hashes whose inputs include a timestamp, exact proposal title and body, and proposal-text wording. See Section 6.5 for more on pre-signing.) The hierarchical design of key encumbrance enables users to participate in several Dark DAOs at once.

We summarize the components of our basic Dark DAO prototype in Figure 1, in the form of pseudocode for each of the main functionalities.

6.3 Dark DAO execution costs

Transaction	Gas Usage	ROSE Cost	USD Cost
Create encumbered account	237,640	0.0237640	$0.00123
Enroll in Snapshot encumbrance policy	144,981	0.0144981	$0.00075
Enter Dark DAO	167,299	0.0167299	$0.00086
Claim bribe payment	85,064	0.0085064	$0.00044
Deploy Snapshot encumbrance policy	2,543,239	0.2543239	$0.01314
Deploy Dark DAO contract	1,690,955	0.1690955	$0.00873

Table 1: Costs of Dark DAO transactions.
1 ROSE = $0.05165, as of October 27, 2023. Transactions are priced at 100 Gwei, the Sapphire default.

Table 1 describes the costs of the various Oasis Sapphire transactions that are necessary to participate in a Dark DAO. A bribee would perform the first four transactions; among those, the first two are the one-time costs of setting up an encumbered account, and the second two occur whenever a bribe is taken. The Snapshot encumbrance policy deployment is a one-time cost, and Dark DAO contracts would presumably all reference the same Snapshot encumbrance policy until API changes require an upgrade. The Dark DAO contract as written needs to be deployed for every DAO proposal a briber wishes to participate in, but it is straightforward to make the contract reusable.

```
Initialization: Set $\texttt{accounts} := \{\}, \texttt{bribes} := []$.

On receive keygen() from party $\mathcal{P}$:
  $(\mathsf{sk}, \mathsf{pk}) \leftarrow_{\$} S.\mathsf{keygen}()$
  $\texttt{accounts}[\mathsf{pk}] = (\texttt{sk: } \mathsf{sk}, \mathsf{party}\texttt{: } \mathcal{P}, \texttt{bribeId: } \bot, \texttt{signed: } \emptyset)$
  Send $\mathsf{pk}$ to $\mathcal{P}$.

On receive sign(pk,$m$) from party $\mathcal{P}$:
  $\texttt{ret} = (\mathsf{sk}, \mathcal{P}^*, \texttt{bribeId}, \texttt{signed}) \leftarrow \texttt{accounts}[\mathsf{pk}]$
  $\mathsf{assert}\ (\texttt{ret} \neq \bot) \wedge (\mathcal{P}^* = \mathcal{P})$
  $\mathsf{assert}\ \texttt{bribeId} = \bot \vee m \notin \texttt{bribes[bribeId]}.\mathcal{M}$
  $\sigma = S.\mathsf{sign}(\mathsf{sk}, m)$
  $\texttt{accounts}[\mathsf{pk}]\texttt{.signed.add}(m)$
  Send $\sigma$ to $\mathcal{P}$.

On receive registerBribe(bribeAmount, $\mathcal{M}$) from party $\mathcal{B}$ along with $T$ tokens:
  $\texttt{bribeId} \leftarrow \mathsf{len}(\texttt{bribes}) + 1$
  $\texttt{bribes}[\texttt{bribeId}] = \texttt{(bribeAmount}, T, \mathcal{M}, \mathcal{B}\texttt{)}$.
  Send bribeId to $\mathcal{B}$.

On receive takeBribe(pk, bribeID) from party $\mathcal{P}$:
  $\mathsf{assert}\ \texttt{accounts}[\mathsf{pk}]\texttt{.party} = \mathcal{P}$
  $(\texttt{bribeAmount}, T, \mathcal{M}, \mathcal{B}) \leftarrow \texttt{bribes}[\texttt{bribeId}]$.
  $\mathsf{assert}\ \texttt{accounts}[\mathsf{pk}]\texttt{.bribeId} = \bot \wedge \texttt{accounts}[\mathsf{pk}]\texttt{.signed} \cap \mathcal{M} = \emptyset \wedge T \geq \texttt{bribeAmount}$
  $\texttt{accounts}[\mathsf{pk}]\texttt{.bribeId} = \texttt{bribeId}$
  $\texttt{bribes}[\texttt{bribeId}].T \mathrel{-}= \texttt{bribeAmount}$
  Send $T$ tokens to $\mathcal{P}$.

On receive signViaEncumberedKey(pk, $m$, bribeID) from party $\mathcal{B}$:
  $\texttt{bribe} = (\texttt{bribeAmount}, \mathcal{M}, \mathcal{B}^*) \leftarrow \texttt{bribes[accounts[pk].bribeId]}$
  $\mathsf{assert}\ (\texttt{bribe} \neq \bot) \wedge (\mathcal{B}^* = \mathcal{B}) \wedge m \in \mathcal{M}$
  $\sigma = S.\mathsf{sign}(\mathsf{sk}, m)$
  $\texttt{accounts}[\mathsf{pk}]\texttt{.signed.add}(m)$
  Send $\sigma$ to $\mathcal{B}$.
```

Figure 1: Key encumbrance and Dark DAO pseudocode

6.4 Security

We consider an idealized model of Oasis Sapphire's trusted execution environment [69], treating side-channel issues and platform-level deployment mistakes (see, e.g., [29, 78, 47]) as out of scope for our exploration in this paper. We also assume the integrity, i.e., correct execution, and liveness of Sapphire. Communications with Oasis Sapphire can in principle be observed by a network adversary. The system supports secure channels to application instances, however, and we exclude consideration of side channels resulting from, e.g., analysis correlating Oasis Sapphire traffic with on-chain behavior. (Such side channels can be mitigated through injection of noise, e.g., randomized delays.)

Informally, in this model, the Basic Dark DAO we have described achieves confidentiality (i.e., "opacity") as follows. An adversary—an entity seeking to probe the Dark DAO, e.g., on behalf of the target DAO—can mount an active attack against the Dark DAO, posing as a briber and as a set of vote-sellers. Such an adversary can learn two forms of information.

First, to the extent that it registers bribes, the adversary learns information about voters that accept these bribes. The adversary learns two things about these voters: (1) The number of votes they are selling and (2) Their on-chain addresses. We stress that the adversary learns (1) only for votes it purchases, but those votes no longer then pose a threat to the target DAO. Here, (2) arises in a model where the adversary submits the votes it has obtained via bribery. There are three reasons why (2) is probably of limited practical concern:

1. *Token fungibility:* If an adversary buys votes from some set of addresses, the adversary controls those votes for a given election. Those addresses might be blacklisted from future participation in the target DAO. But since tokens are fungible, they could simply be sent to new addresses, greatly complicating potential blacklisting policies.

2. *Cost of acquisition:* Buying votes to learn associated addresses—and of course control their votes—is a costly strategy. It also creates a perverse incentive. It actually encourages the creation of Dark DAOs, as the adversary is subsidizing bribes.

3. *Private voting:* Votes could in principle retain confidentiality during submission, rather than be obtained in cleartext by a briber. A TEE could, for instance, perform submission to a website (e.g., Snapshot). Although Oasis Sapphire does not directly support TLS traffic and thus would not enable straightforward implementation of this functionality, other TEE-based systems can in principle play this role, e.g., [84].

The second form of information available to the adversary is the size of bribes registered by (other) bribers—which are published to signal the opportunity to vote sellers. Published bribes only represent an upper bound on Dark DAO activity, however.

In summary, at the application layer, the only feasible way for an adversary to impact the behavior of the Dark DAO through active attack is by buying votes. Additionally, the Dark DAO to the best of our knowledge presents no application-layer denial-of-service attack vectors.

6.5 Deployment Considerations

Pre-signing attacks. Key-encumbered wallet owners can sign an unlimited number of messages prior to enrolling in an encumbrance policy. This creates an opportunity for wallet owners to pre-sign messages they predict will be encumbered by a policy in the future and defeat the encumbrance scheme. For example, if a wallet owner is about to enroll in an encumbrance policy that restricts its ability to sign a message saying "vote for Alice," the owner could just pre-sign this message before enrolling in the policy. Therefore, encumbrance policies must either be enforced from the moment the wallet is created or work with messages the owner could not have possibly predicted prior to encumbrance, such as those containing a pseudorandom value or the hash of high-entropy inputs, e.g., Snapshot proposal hashes.

An alternative key-encumbered wallet design is to record every message that has ever been signed with an encumbered wallet; on enrollment, an encumbrance policy could then check that no previous message breached the encumbrance assumptions of the policy. However, to assess whether a type of message has ever been signed requires each message to be recorded on the same blockchain as the wallet contract, incurring a cost in transaction fees to the wallet owner whenever he or she wishes to sign a message.

Often, these schemes can be combined to produce workable encumbrance policies that would otherwise require enrollment from the moment of wallet creation. If an encumbrance policy can anchor relevant message characteristics to a specific timestamp (e.g., a DAO proposal's hash to its creation timestamp), it can distinguish whether a particular message could have been signed before the encumbered wallet was enrolled in the policy. All the messages whose timestamps are earlier than the time of enrollment are unrestricted; all those after are eligible for restriction by the policy.

Campaigns. Rather than selling votes for a given election, a Dark DAO can orchestrate a bribery *campaign*, dispensing bribes that are contingent upon a successful outcome. This might be an election outcome, or the outcome of multiple elections. But a campaign may target any of a range of outcomes reflected in blockchain state—e.g., successful installation of a particular user in a privileged role (e.g., membership in a DAO committee responsible for disbursing funds). Any of a range of bribery policies are also possible, e.g., rewards for recruiting fresh Dark DAO participants.

6.6 Mitigation: Complete Knowledge

An application can prevent access to accounts created by a key encumbrance system by requiring a *proof of complete knowledge* to be demonstrated for each public key requesting access [51]. Such a proof demonstrates that the associated private key either has been shown or could have been shown, in totality, to an eavesdropper. A key-encumbered wallet can no longer make its encumbrance guarantees if its private key is ever leaked, so correctly implemented key-encumbered wallets are naturally forbidden from creating valid proofs of complete knowledge. Lightweight proofs of complete knowledge via mobile device TEEs may soon be practical if signature verification of the relevant curves becomes cheap, such as through implementation of EIP-7212 [37].

6.7 Ethical Considerations

We have open-sourced the code of our key encumbered wallet and message-based Dark DAO contracts. We have chosen to do this because, given the current risks of participation in Dark DAOs, the short-term threat is limited. DAOs are currently not highly decentralized, as shown in, e.g., [39]—a precondition for Dark DAOs to be effective. We feel it is important to have a clear demonstration of the practicality of Dark DAOs so that the community can understand the contours of the risk in the long term and consider effective countermeasures.

Additionally, our code has beneficial use cases, which we will show in future work. For example, a confidential DAO whose treasury funds are themselves encumbered inside the wallets of its participants would have sidestepped some of the shortcomings of the Constitution DAO [77], whose public fundraising appears to have facilitated it being outbid in a silent auction.

7 Dark DAO Lite

Although our Dark DAO prototype demonstrates that Dark DAOs are practical to build, participating in the Dark DAO as a bribee is not straightforward. The requirements of setting up an encumbered account and managing funds through it, discovering bribes, and enrolling in encumbrance policies create friction for ordinary users.

To alleviate these usability issues and further emphasize the versatility of key encumbrance, we have created a second Dark DAO system that we call a *Dark DAO "Lite"*. Our Dark DAO Lite scheme involves a trade-off: It achieves greater usability than our basic Dark DAO, but weaker confidentiality. (Thus our use of the term "lite.")

The key idea behind the Dark DAO Lite is its use of a DAO-token derivative to hide the complexity of participation. We call this derivative, which is itself a token, a *Dark-DAO token* or *DD token* for short.

DD tokens in a Dark DAO Lite are derived from ordinary tokens in a target DAO through a conversion process. The key steps of this process, which we explain in detail further below, are summarized in Figure 2. A Dark DAO Lite itself, like a basic Dark DAO, is a smart contract running on Oasis Sapphire and benefits from that chain's confidentiality properties. Its functionality is described in Figure 4. DD tokens, however, are ordinary ERC-20 tokens held on Ethereum that can be traded on existing token markets, such as Uniswap. Converting between the target DAO token and the non-voting derivative DD tokens requires technical knowledge, but need only be done once by a small set of actors—who can obtain remuneration through DD token markets. Once DD tokens are created, no sophistication is required to manage them.

DD tokens realize a concept that we refer to as *DAO-token fractionalization*.

DAO-token fractionalization. DAO tokens grant two capabilities to their owner: (1) the ability to sign votes on proposals and (2) ownership rights in the DAO, expressible as the ability to send the tokens to a different address. Key-encumbered wallets and encumbrance policies enable a separation of these two capabilities by creating two paths of access control to a single private key controlling the tokens of a target DAO. This fractionalization yields two distinct resources:

- *Voting rights* corresponding to converted target-DAO tokens. A pool of these voting rights may be purchased by a vote-buyer / briber through an auction mechanism we describe below. We refer to the pool of voting rights as *self-auctioning*, since the auction process is automated and requires no intervention by DD-token holders.
- *DD tokens*, which may be individually owned and correspond to ownership rights in the target DAO plus the right to receive revenue from the auctioning of the pool of fractionalized voting rights.

In the remainder of this section, we explain how the various parts of a Dark DAO Lite work (Section 7.1) and its security properties (Section 7.2). In what follows, unless otherwise specified, we use the term Dark DAO to refer to a Dark DAO Lite.

7.1 Dark-DAO Lite functionality

Converting target-DAO tokens to DD tokens. Before authorizing the creation of new DD tokens, the Dark DAO needs to gain control over target-DAO tokens. This is accomplished by having the target-DAO tokens sent to a freshly generated Ethereum account under the Dark DAO's control.

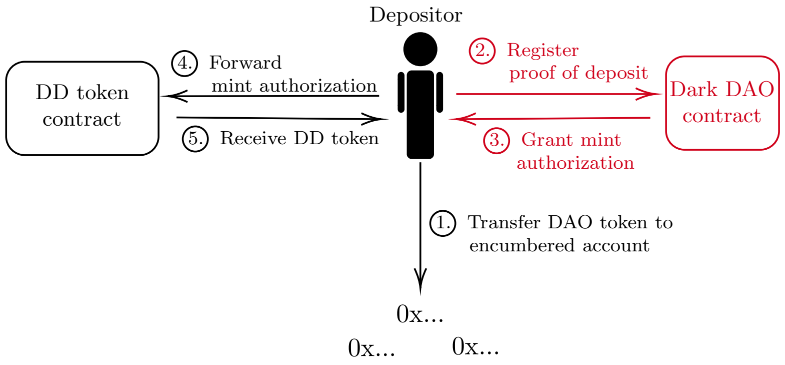
Depositor
DD token contract
4. Forward mint authorization
5. Receive DD token
2. Register proof of deposit
3. Grant mint authorization
Dark DAO contract
1. Transfer DAO token to encumbered account
0x...
0x...
0x...

Figure 2: Transaction flow for converting DAO tokens to DD tokens. The Dark DAO contract, denoted in red, is on Oasis, while the rest is on Ethereum. The accounts at the bottom are encumbered, under the control of the Dark DAO contract; each deposit is sent to a fresh encumbered account.

```
DD Token, extending ERC-20

Initialize(pk): DDpk := pk, supply := 0, authNonces := {}

On receive mint($m = (T, \mathsf{nonce}), \sigma$) from party $P$:
    assert $S.\mathsf{ver}(\mathsf{DDpk}, m, \sigma)$
    assert $\mathsf{nonce} \notin \mathsf{authNonces}[P]$
    $\mathsf{supply} \leftarrow \mathsf{supply} + T$
    $\mathsf{authNonces}[P].\mathsf{add}(\mathsf{nonce})$
    Send $T$ tokens to $P$

On receive burn() from party $P$ along with $T$ DD-tokens:
    $\mathsf{supply} \leftarrow \mathsf{supply} - T$
```

Figure 3: DD token pseudocode

A user wishing to receive DD tokens queries the Dark DAO in an off-chain query for a newly generated such address into which a batch of target-DAO tokens may be deposited. The Dark DAO contract responds with the deposit address A and a ciphertext C on deposit data: (sk_A, R), where sk_A is the private key for A and R is the address for receiving DD tokens minted as a result of a deposit to A.

The reason for creating a fresh address for each deposit is *confidentiality* of these addresses. Because the generation process happens off chain, there is no public indication that A is controlled by the Dark DAO. The deposit is indistinguishable from a simple transfer of target-DAO tokens to a new EOA (externally owned account).

After the user transfers target-DAO tokens to A, the user submits a state proof of A's target-DAO token balance to the Dark DAO contract, along with the ciphertext C. The Dark DAO contract checks the proof which begins an optional *lockup period* on the tokens. (See Section 7.2 for details.) After the lockup period is complete, the user can query it to receive a signed message authorizing the minting of an equivalent amount of DD tokens. A user with can then send this signed message to the DD token contract to mint the DD tokens, as shown in Figure 3. The mint

```
Dark DAO Lite

function initialize(header) :
    eth_block_header := header        // updated by an oracle or by piggybacking on proofs
    encumbered_accounts := {}
    dark_dao_sk, dark_dao_pk $\xleftarrow{\$}$ S.keygen()
    balances := {}
    registered_proofs := {}
    return dark_dao_pk

function get_deposit_address() :
    sk, pk $\xleftarrow{\$}$ S.keygen()
    encrypted_key $\xleftarrow{\$}$ dark_dao_sk.encrypt(sk)
    return pk, encrypted_key

function deposit_and_mint($\pi$, recipient) :
    assert $\pi \notin$ registered_proofs
    assert verify_deposit_proof(eth_block_header, $\pi$)
    encumbered_accounts.insert($\pi$.pk, dark_dao_sk.decrypt($\pi$.encrypted_key))
    balances[$\pi$.pk] := balances[$\pi$.pk] + $\pi$.amount
    registered_proofs.insert($\pi$)
    message.amount := $\pi$.amount
    message.recipient := $\pi$.recipient
    return message, dark_dao_sk.sign(message)

function redeem_and_withdraw($\pi$, recipient) :
    assert $\pi \notin$ registered_proofs
    assert verify_burn_proof(eth_block_header, $\pi$)
    registered_proofs.insert($\pi$)
    accounts, amounts := select_withdrawal_accounts(encumbered_accounts, balances, $\pi$.amount)
    signed_transactions := {}
    for account, amount $\in$ accounts, amounts
        signed_transactions.insert(account.sk.sign(transfer_from(account.pk, amount, recipient)))
        balances[account.pk] := balances[account.pk] $-$ amount
    return signed_transactions
```

Figure 4: Pseudocode for Dark DAO Lite smart contract on Oasis

operation need not happen immediately; the user could presumably wait until the DD tokens need to be transferred or sold.

The use of state proofs serves as a bridge between Ethereum and Oasis. We assume a trusted source of block hashes for this purpose. The "Oasis Privacy Layer," which uses the Celer Network as a bridge system under the hood, is the existing bridge from Oasis Sapphire to any supported EVM network.

Redeeming DD tokens for target-DAO tokens. The conversion process may be reversed: DD tokens can be redeemed for their underlying target-DAO tokens. A user holding DD tokens first issues a burn transaction of n tokens to the DD token contract, which removes the n DD tokens from circulation and records a receipt of the burn to persistent storage. The user then submits a state proof of the burn receipt to the Dark DAO contract on Oasis, which in return sends back a proportional amount of bribe money and authorizes the user to submit off-chain withdrawal requests to the Dark DAO contract. The Dark DAO responds to these requests with a signed Ethereum transaction which transfers up to n target-DAO tokens from a Dark DAO controlled account to the user. It is the user's responsibility to include this transaction on the Ethereum mainnet. Note that DD tokens must be fungible and liquid, or else they would not be easily tradable; therefore, the Dark DAO contract must be able to handle partial withdrawals from its accounts. A withdrawal that is greater than the current withdrawal account's balance would require multiple withdrawal transactions.

On Ethereum, transactions are ordered by sender according to increasing transaction nonce: the first transaction by a particular sender must be signed with nonce 0, the second with nonce 1, and so on. Target-DAO token transfers out of Dark DAO accounts are also transactions and must be included in increasing nonce order. To prevent users who fail to include their transactions on the Ethereum mainnet from blocking other target-DAO token withdrawals, everyone who is ready to withdraw is issued a signed transaction from the same Dark DAO account and with the same nonce. The first withdrawal transaction to be included in an Ethereum block "wins," and the other competing transactions with the same nonce and sender are automatically invalidated at no cost, per Ethereum's rules. To allow the next withdrawal to process, a user can show a Merkle proof of transaction inclusion in an Ethereum block, which simultaneously increments the nonce of the Dark DAO account (or chooses a new withdrawal account) and marks the included withdrawal as completed.

Ethereum transactions need to be funded before they are included, so to pay for the target-DAO token transfer, some ETH must be sent to the Dark DAO account in an earlier transaction. We expect withdrawers will use Flashbots bundles to execute the funding and token transfer transactions atomically and to prevent other withdrawals from backrunning their funding transactions.

DD tokens. As we have explained, DD tokens are the primary financial instrument of a Dark DAO Lite, issued when deposits of target DAO tokens are made to Dark DAO accounts. They can later be redeemed for the underlying target DAO tokens plus any accumulated bribes on the voting rights to the encumbered target DAO tokens. The Dark DAO smart contract on Oasis acts as the primary controller of all participating encumbered Ethereum accounts and is itself an encumbrance policy.

We emphasize that users with DD tokens cannot vote in the target DAO with them; this voting ability is in the Dark DAO's self-auctioning pool. Rather, users with DD tokens hold a claim to ownership in the target DAO plus a proportional fraction of the bribe revenue the Dark DAO creates from selling its votes. In short, 1 DD token is equivalent to the ownership rights of 1 target DAO token plus fractional bribe revenue.

Voting-rights auctioning. We assume that the target DAO utilizes a message-based, off-chain voting system with voting power assigned to accounts based on their DAO token balances, though the Dark DAO contract could be adapted for other voting schemes.

When a target-DAO proposal is published, the proposal hash is made public. Bribers who wish to purchase the Dark DAO's voting power for the proposal can start an auction for that proposal from the Dark DAO contract and bid on the right to sign votes from all Dark DAO accounts[4]. We implemented a first-price auction in our implementation, but other auction types could be substituted. All auctions have a fixed duration and must end before the proposal expires, or else the purchased votes cannot be used.

A briber who wins an auction can ask the Dark DAO contract to sign votes for the proposal from all of the Dark DAO accounts. Bidders could bid on a hash that does not correspond to a proposal, so as implemented, any auction winner can enumerate Dark DAO accounts by reading the vote signatures. In Section 7.2, we briefly discuss an alternative, privacy-preserving approach.

How the DD-token market works. As conversion of target-DAO tokens to DD tokens requires a (small) degree of technical knowledge, including interaction with the Oasis Sapphire chain, our expectation is that arbitrageurs will perform the conversion and sell DD tokens in exchanges for Ethereum. As ordinary ERC-20 tokens, DD tokens may be sold in either decentralized or centralized exchanges. The value of generating DD tokens—and thus revenue for arbitrageurs—may be priced into the market value of DD tokens.

Whether a given user P prefers to hold target-DAO tokens or DD tokens depends upon the value to the user of voting, which is related $\mathsf{util}_P(E, \mathtt{true})$ for a set of elections E over which the user intends to hold DD tokens. Given high utility, i.e., a particular desired outcome, a user may prefer to vote and thus hold target-DAO tokens. Many users, however, are apathetic (as discussed in Section 3.3) and would derive higher utility instead from holding DD tokens. The technical requirements and user experience for holding the two types of token are identical.

We emphasize that DD tokens may be redeemed for target-DAO tokens. Hence the fair market price of DD tokens should be at least that of target-DAO tokens minus the transaction cost for redemption.

Aside from making Dark DAOs more practical, our DD-token scheme demonstrates a concept of broad interest: key encumbrance enables new financial assets with sophisticated policies to be created from the restructuring of existing ones. While use of TEEs to realize this concept has been previously explored [59, 70], our work is the first instance of which we're aware in which such assets are realized as decentralized-finance tokens.

Execution costs. Table 2 outlines the transaction costs of creating and participating in a Dark DAO Lite.

7.2 Security

We assume the same security model as in Section 6.4. A Dark DAO Lite, as noted above, achieves *weaker confidentiality* than a basic Dark DAO. (Although its integrity and DoS properties are the same.)

The main reason the Dark DAO Lite does not achieve the same strong confidentiality as the basic Dark DAO is the *liquidity* of DD tokens. Recall that when *deposited*, target-DAO tokens are transferred to a Dark-DAO-generated address that is indistinguishable on chain from an ordinary

[4] In our implementation, bids are made in Oasis's native token, ROSE. Over time, the DD token will have increased exposure to this other asset. To remedy this, the Dark DAO contract can sell its proceeds periodically for the target-DAO token.

Ethereum Transaction	**Gas Usage**	**ETH Cost**	**USD Cost**
Deploy DD token contract (*one-time cost*)	1,552,447	0.0240629	\$42.88543
Transfer DAO token to Dark DAO account	51,438	0.0007973	\$1.42096
Mint DD tokens	99,050	0.0015353	\$2.73624
Burn DD tokens	58,665	0.0009093	\$1.62057
Fund DAO token transfer from Dark DAO account	21,000	0.0003255	\$0.58011
DAO token transfer from Dark DAO account	51,438	0.0007973	\$1.42096
Oasis Transaction	**Gas Usage**	**ROSE Cost**	**USD Cost**
Deploy Dark DAO contracts (*one-time cost*)	6,801,449	0.6801449	\$0.03513
Prove DAO token deposit to Dark DAO contract	863,885	0.0863885	\$0.00446
Prove DD token burn to Dark DAO contract	501,138	0.0501138	\$0.00259
Prove DD withdrawal inclusion	332,495	0.0332495	\$0.00171
Create DAO proposal voting rights auction	122,912	0.0122912	\$0.00063
Bid on voting rights for a proposal	53,017	0.0053017	\$0.00027

Table 2: Costs of Dark DAO Lite transactions.
1 ETH = \$1,782.22, as of October 27, 2023. Ethereum transactions are priced at 13.5 Gwei, the 60-day average ending on October 26, 2023.
1 ROSE = \$0.05165, as of October 27, 2023. Oasis transactions are priced at 100 Gwei, the Sapphire default.

EOA. When DD tokens are *redeemed* for target-DAO tokens, however, the address in which they are held is revealed to the redeeming player to be part of the Dark DAO. Furthermore, the liquidity of tokens means that one user can redeem tokens deposited by another user. An adversary can perform redemptions strategically in an attempt to enumerate Dark-DAO deposit addresses.

For example, suppose that players only deposit one token at a time and that redemptions are LIFO (last in, first out). An adversary $\mathcal{A}$ can deposit a token and wait until another player P_1 deposits a token. $\mathcal{A}$ then withdraws its token, revealing P_1's deposit address, and then redeposits its token. $\mathcal{A}$ can do the same when P_2 deposits a token, and so forth. In principle, by withdrawing and depositing t tokens over intervals of length Δ, where t is at least as large as other players' deposits over any interval of length Δ, $\mathcal{A}$ can identify all deposit addresses.

Such discovery of deposit addresses through strategic redemption is somewhat costly in practice, because it incurs transaction costs. Our current Dark DAO Lite implementation in fact uses FIFO scheduling. It is possible to impede adversarial address-discovery strategies against FIFO by imposing a lockup period on deposited tokens. The effect of this practice is to raise the adversary's capital requirements, i.e., require the adversary to control a large number tokens. Other approaches might be more effective and their exploration constitutes an interesting research challenge.

An additional form of information leakage in a Dark DAO Lite arises because the circulating supply of DD tokens is publicly visible. The quantity of these tokens specifies corresponds to the size of the available pool of votes available for purchase in the Dark DAO. (It is possible in principle to enhance a Dark DAO Lite to mint fake DD tokens, a potential future enhancement.) Furthermore, on-chain transaction analysis may leak futher information. For example, the timing of target-DAO token deposits and DD token issuance can help an adversary infer target-DAO token addresses.

8 Summary Guidance for DAOs

Our VBE framework and the implications we show in Section 3 suggest a number of forms of concrete guidance for DAOs seeking to enforce or improve meaningful decentralization. We discuss them in this section. We summarize our guidance for practitioners in Table 3.

Topic	*General Guidance*	*Reason*	*Relevant result*
1. **Vote delegation**	Given a large inactivity whale, vote delegation tends to increase decentralization.	Delegation (counterintuitively) increases decentralization by diversifying tokens away from a big inactivity whale.	Thm. 3.4
2. **Voting privacy**	Voting privacy increases decentralization.	Private voting eases herding, whose effects are centralizing.	Thm. 3.5
3. **Voter bribery**	The scale of bribery increases with decentralization.	Low alignment of utility functions means systemic coordination is required to impose alignment.	Thms. 3.7, 3.8, and 3.9
4. **Dark DAO risks**	Dark DAO risks are likely to increase with decentralization.	As bribery coordination costs grow, Dark DAOs become a more compelling approach to influencing vote outcomes.	Inference from Thm. 3.8 and 3.9
5. **Dark DAO feasibility**	Dark DAOs are feasible today.	We have shown that existing tools enable effective Dark-DAO deployment. Technical feasibility is unlikely to prove a barrier to their use by adversaries. Complete Knowledge (CK) for voter keys may be a useful countermeasure.	Sections 6 and 7
6. **Identity verification**	Weak identity verification increases centralization in quadratic voting.	A whale that can spread tokens across identities amplifies its voting power.	Analysis in Sections 3.8 and 5.3
7. **Voting slates / proposal bundling**	Bundling choices into slates (like protocol upgrades that include many voting issues in one package) decreases decentralization.	Bundled choices artificially align otherwise heterogeneous utility functions and/or induce apathy by smoothing out utility functions.	Thm. 3.6
8. **Data collection**	Careful voting-statistic collection facilitates decentralization measurement.	Lack of systematic collection and publication of detailed voting statistics makes decentralization measurement challenging today.	Discussion in Section 4.

Table 3: Guidance implied by this paper's results regarding DAO decentralization.

Apathy / inactivity whale and delegation: As we show in Section 3.3, token holders who do not vote—those, in a rational model, with near-zero utility functions—have a centralizing effect.

Recall that our term for this group is the *inactivity whale*.

One way to diminish the size of the inactivity whale is through delegation. Intuitively, if tokens associated with the inactivity whale are distributed between at least two delegatees in distinct clusters, then they come to represent distinct utility functions——and thus contribute to decentralization.

We show in Section 3.4 that when the inactivity whale is large—with respect to delegatees—delegation increases decentralization. (Otherwise, delegation may or may not have this effect.)

Herding / voting privacy: There is anecdotal evidence suggesting that social pressure causes herding—specifically that voters align themselves with whales or voting blocs [72]. We may view the effect as a shift in utility function. As we show in Section 3.5, this shift has a centralizing effect.

Herding arises because votes are publicly visible. Voting privacy in principle alleviates such pressure and therefore has a decentralizing effect.

Snapshot, a popular platform for DAO voting, has recently implemented a form of privacy called *shielded voting* [73].This form of privacy, however, is only ephemeral: Votes are private when submitted, but revealed at the end of the vote-casting period. So it is unclear that it can fully address the centralizing effects of herding.

End-to-end verifiable voting systems have been proposed in the literature for decades that achieve both voting integrity and confidentiality [12]. How to implemented them with token-based weighting is, to the best of our knowledge, though, an open problem.

Voter bribery: Our work shows a relationship between centralization and bribery. In general, bribery causes an increase in centralization, as it has the effect of aligning the utility functions of other players with those of the briber, as we show in Section 3.7.

We also show that as decentralization increases, bribery cost increases. Roughly speaking, increasing diversity in utility functions means increasing cost to align them.

DAOs today are largely centralized [72, 39, 25, 26, 76]. Bribery may not be especially useful, as whales generally exert strong control and require relatively little coordination to align utility functions into a favorable voting bloc. Voter bribery, however, is a problem in many settings, both in political voting [60] and in corporate governance (see, e.g., [71]).

One implication of our results is that as DAO decentralization increases, in order for bribery to succeed, it will need to be systemic. DAO designers should therefore recognize large-scale bribery as a future risk.

Dark DAO risks: We hypothesize that the most technically feasible way to implement large-scale bribery is through a Dark DAO.

We have presented in Section 6 the first fully functional private Dark DAO capable of subverting votes on Ethereum. Our architecture leverages the privacy assurances of TEEs in Oasis, but bridges to Ethereum, where most DAOs operate. Our results show that Dark DAOs are technically feasible (and incur low transaction costs) and thus represent a viable future threat.

Dark DAOs pose not just a technical threat, but also a psychological one. The mere existence of a Dark DAO may create a perception of vote-manipulation even if the Dark DAO has minimal impact. Moreover, Dark DAOs can be used not just for direct bribery but also for more subtle attacks. They can, for instance, subvert quadratic voting schemes even when such schemes rely on well-functioning decentralized identity systems.

One possible countermeasure DAO designers may ultimately wish to consider is requiring voting participants to execute complete-knowledge (CK) proofs on their keys [51].

Voting slates / bundling proposals: A common trick for passing pieces of legislation that are unpopular or have a narrow base of support is to bundle them together in large, bills. Earmarks are a prime example [14].

This practice may be regarded as a form of utility-function "smoothing": The utility function of the bill as a whole (for the legislators voting on it) differs from that of its components.

As the practice of bundling proposals / measures has the goal of aligning utility functions, from the standpoint of VBE, it generally has a centralizing effect, as we show in Section 3.6. DAOs may therefore wish to consider limiting the practice and instead explore way to unbundle multi-component proposals.

Data collection: There is no practical way to compute VBE directly—since players' typically do not express their utility functions. As we discuss in Section 4, however, there are ways to estimate it for a DAO based on voting history. We have found it challenging to collect full voting histories for even popular DAOs. A recommendation for the community is to establish and adhere to standards for archival preservation of DAO voting data. A few

9 Related Work

DAOs: Research literature on DAOs has been limited to date, but fairly broad. It has included measurement studies [39, 72], retrospectives on the failure of The DAO (e.g., [36]) and ways of addressing related technical flaws in smart contracts such as dangerous reentrancy (e.g., [55, 28]), DAO mechanism design (e.g., [16]), and exploration of DAOs from the standpoint of legal theory (e.g., [46, 79]) and economics and governance (e.g., [18]).

Works exploring measurement of DAOs' degree of decentralization most notably include Feichtinger et al. [39],who explore Gini and Nakamoto indices, as well as participation rates and the monetary cost of governance, Sharma et al. [72], who consider various notions of entropy, as well as participation rates and graph-based measures of decentralization, and [83], which taxonomizes DAOs by comparison with other autonomous systems. Sun et al. use clustering to identify voting blocs in a study of MakerDAO [75]. Also of note is the informal notion of "credible neutrality," a community standard articulated in, e.g., [23, 21].

Social choice and voting theory: A long line of work on social choice and voting theory investigates how best to aggregate preferences of individual voters—the same functionality that DAOs seek to provide in the decentralized setting. There are some major differences in the DAO setting however, which may reduce how effective existing techniques will be. For instance, the permissionless nature of DAOs allows for the presence of Sybils which is not typically accounted for in existing voting theory literature. Further, while the threat of large-scale voter bribery is typically safe to ignore in classical voting, both due to the high likelihood of detecting such an attack, as well as the the challenge in coordinating the attack itself, as shown in our paper, Dark DAOs invalidate these prior assumptions in the DAO setting.

Still, we believe that DAOs can provide an excellent practical battleground for experimenting with different social choice and voting techniques in the real world.

Vote-buying / coercion: There is a considerable literature on the notion of *coercion-resistance* in end-to-end verifiable voting [49, 34, 54]. Broadly speaking, coercion-resistance means that a voter cannot convince a would-be briber or coercer of how she voted. Influential proposed coercion-resistant voting systems include notably Civitas [30] and, more recently, MACI [20]. None of these definitions or system designs contemplate the risk of key encumbrance. Dark DAOs effectively break all of them.

Unauthorized credential delegation: Daian et al. put forth the notion of a Dark DAO—a DAO that aims to subvert voting in other DAOs—in [33]. In related work, Matetic et al. propose use of TEEs as a tool for secure credential delegation—which may be unauthorized [59], and Puddu et al. explore malicious uses, including subversion of e-voting [70].

10 Conclusion and Open Research Questions

We have proposed Voting-Bloc Entropy (VBE) as a new metric for DAO decentralization. VBE measures the entropy of voting blocs. It is in fact a framework into which it is possible to plug any desired method of clustering to identify blocs and any notion of entropy.

Evaluating VBE—instantiated with ϵ-threshold ordinal clustering and min-entropy—we have proven a number of results that held shed light how a number of practices may impact DAO decentralization. We have also shown both in theory and through implementation of a practical system how Dark DAOs pose a potential long-term threat.

Our work gives rise to a number of open research questions. A few deserve particular mention:

- **Privacy:** Our results suggest the potential decentralizing effects of ballot secrecy, i.e., private voting. Existing verifiable end-to-end voting systems implement a one-vote-per-person policy [12]. One open research question is whether token-weighted variants are possible. Additionally, we emphasize in our work the importance of collecting voting data to facilitate VBE estimates. How to harmonize these opposing goals represents a second research challenge.
- **Forking and escape hatches:** DAOs may suffer catastrophic failures, as was famously the case with The DAO [40]. Proposed remedies including forking / splitting, in which a new, quasi-independent or independent DAO is created and escape hatches, which are committee-controlled shutdowns [38]. How their existence and use impact decentralization are unclear and deserves study.
- **VBE impact:** VBE is designed to formalize a view of decentralization in DAOs reflected in the literature and in the views of practitioners. A natural question is what impact high VBE has on decision making in DAOs. Does it correlate with community growth, participation, and financial outcomes in DAOs? How does it relate to notions of democratic participation in non-blockchain settings?

Acknowledgements

This work was funded by NSF grant CNS-2112751 and by generous support from IC3 industry partners. Andrés Fábrega is funded in part by a Uniswap Foundation TLDR Fellowship.

Thanks to Oasis Labs for answering technical questions, Phil Daian for extensive discussions about Dark DAOs, and Sylvain Bellemare for suggesting the pronunciation "vibe" for VBE.

References

[1] Snapshot - the technical architecture. https://snapshot.mirror.xyz/-C3bd5TI3XEbPRt_FIBB97fkhwXk-81
2023.

[2] AvocadoDAO. https://www.avocadodao.io, Referenced Oct. 2023.

[3] DeepDAO. https://deepdao.io/, Referenced Oct. 2023.

[4] The graph. https://thegraph.com/, Referenced Oct. 2023.

[5] GuildFi. https://guildfi.com, Referenced Oct. 2023.

[6] MolochDAO: The original grant giving DAO. https://molochdao.com, Referenced Oct. 2023.

[7] Nakamoto coefficient: An accurate indicator for blockchain decentralization? Bybit Learn, Referenced Oct. 2023.

[8] Optimism collective. https://app.optimism.io/announcement, Referenced Oct. 2023.

[9] Research DAO. https://researchdao.io/, Referenced Oct. 2023.

[10] Snapshot. https://snapshot.org, Referenced Oct. 2023.

[11] Uniswap: Governance. https://uniswap.org/governance, Referenced Oct. 2023.

[12] Syed Taha Ali and Judy Murray. An overview of end-to-end verifiable voting systems. *Real-World Electronic Voting*, pages 189–234, 2016.

[13] Noga Alon, Moshe Babaioff, Ron Karidi, Ron Lavi, and Moshe Tennenholtz. Sequential voting with externalities: herding in social networks. In *EC*, page 36, 2012.

[14] Chris Cassella E.J. Fagan Sean Theriault. Earmarks are back: How Democrats and Republicans differ. Brooking Institution Commentary, https://www.brookings.edu/articles/earmarks-are-back-how-democrats-and-republicans-differ, 12 Jan. 2023.

[15] Arbitrum DAO: A conceptual overview. https://docs.arbitrum.foundation/concepts/arbitrum-dao, Referenced Oct. 2023.

[16] Maryam Bahrani, Pranav Garimidi, and Tim Roughgarden. When bidders are DAOs. *arXiv preprint arXiv:2306.17099*, 2023.

[17] Abhijit V Banerjee. A simple model of herd behavior. *The quarterly journal of economics*, 107(3):797–817, 1992.

[18] Roman Beck, Christoph Müller-Bloch, and John Leslie King. Governance in the blockchain economy: A framework and research agenda. *Journal of the association for information systems*, 19(10):1, 2018.

[19] Ernesto Dal Bó. Bribing voters. *American Journal of Political Science*, 2007.

[20] V. Buterin. Minimal anti-collusion infrastructure (MACI). Ethereum Research blog post at `https://ethresear.ch/t/minimal-anti-collusion-infrastructure/5413`, 2 May 2019.

[21] Vitalik Buterin. What do I think about Community Notes? `https://vitalik.ca/general/2023/08/16/communitynotes.html`, 16 Oct. 2023.

[22] Vitalik Buterin. Bootstrapping a decentralized autonomous corporation: part i. *Bitcoin Magazine*, 19, 2013.

[23] Vitalik Buterin. Credible neutrality as a guiding principle. `https://nakamoto.com/credible-neutrality/`, 3 Jan. 2020.

[24] Vitalik Buterin. Quadratic payments: A primer. `https://vitalik.ca/general/2019/12/07/quadratic.htm` 7 Dec. 2019.

[25] Vitalik Buterin. Moving beyond coin voting governance, Aug. 2016.

[26] Vitalik Buterin. Notes on blockchain governance, Dec. 2017.

[27] Vitalik Buterin, Jacob Illum, Matthias Nadler, Fabian Schär, and Ameen Soleimani. Blockchain privacy and regulatory compliance: Towards a practical equilibrium. *Available at SSRN*, 2023.

[28] Ethan Cecchetti, Siqiu Yao, Haobin Ni, and Andrew C Myers. Compositional security for reentrant applications. In *2021 IEEE Symposium on Security and Privacy (SP)*, pages 1249–1267. IEEE, 2021.

[29] Guoxing Chen, Sanchuan Chen, Yuan Xiao, Yinqian Zhang, Zhiqiang Lin, and Ten H Lai. Sgxpectre: Stealing intel secrets from sgx enclaves via speculative execution. In *2019 IEEE European Symposium on Security and Privacy (EuroS&P)*, pages 142–157. IEEE, 2019.

[30] Michael R Clarkson, Stephen Chong, and Andrew C Myers. Civitas: Toward a secure voting system. In *IEEE S&P*, pages 354–368, 2008.

[31] Saudu Clement. A list of possible solutions to DAO voter apathy. DAO Times, Dec. 2022.

[32] Phil Daian. Vote buying, on-chain governance, and quadratic plutocracy. `https://web.archive.org/web/20210122070951/https://pdaian.com/blog/vote-buying-on-chain-go` June 2018.

[33] Philip Daian, Tyler Kell, Ian Miers, and Ari Juels. On-chain vote buying and the rise of dark DAOs. `https://hackingdistributed.com/2018/07/02/on-chain-vote-buying/`, 2018.

[34] Stéphanie Delaune, Steve Kremer, and Mark Ryan. Coercion-resistance and receipt-freeness in electronic voting. In *19th IEEE Computer Security Foundations Workshop (CSFW'06)*, pages 12–pp. IEEE, 2006.

[35] John Duffy and Margit Tavits. Beliefs and voting decisions: A test of the pivotal voter model. *American Journal of Political Science*, 2008.

[36] Quinn DuPont. Experiments in algorithmic governance: A history and ethnography of "the DAO," a failed decentralized autonomous organization. In *Bitcoin and beyond*, pages 157–177. Routledge, 2017.

[37] Ulaş Erdoğan and Doğan Alpaslan. Eip-7212: Precompiled for secp256r1 curve support [draft]. `https://eips.ethereum.org/EIPS/eip-7212`, retrieved Sep. 2023.

[38] Ittay Eyal and Emin Gün Sirer. A decentralized escape hatch for daos. `https://hackingdistributed.com/2016/07/11/decentralized-escape-hatches-for-smart-contracts` 2016.

[39] Rainer Feichtinger, Robin Fritsch, Yann Vonlanthen, and Roger Wattenhofer. The hidden shortcomings of (D)AOs–an empirical study of on-chain governance. *arXiv preprint arXiv:2302.12125*, 2023.

[40] Wikimedia Foundation. The DAO (organization). `https://en.wikipedia.org/wiki/The_DAO_(organizati` Referenced Oct. 2023.

[41] Wikimedia Foundation. Gini coefficient. `https://en.wikipedia.org/wiki/Gini_coefficient`, Referenced Oct. 2023.

[42] Wikimedia Foundation. Latent and observable variables. `https://en.wikipedia.org/wiki/Latent_and_observable_variables`, Referenced Oct. 2023.

[43] BD Frey. Sanctions compliance pitfalls for banks. *ABA Banking Journal*, 24, 2019.

[44] Robin Fritsch, Marino Müller, and Roger Wattenhofer. Analyzing voting power in decentralized governance: Who controls DAOs? *arXiv preprint arXiv:2204.01176*, 2022.

[45] Maximiliano González, Renato Modernell, and Elisa París. Herding behaviour inside the board: An experimental approach. *Corporate Governance: An International Review*, 2006.

[46] Samer Hassan and Primavera De Filippi. Decentralized autonomous organization. *Internet Policy Review*, 10(2):1–10, 2021.

[47] Nerla Jean-Louis, Yunqi Li, Yan Ji, Harjasleen Malvai, Thomas Yurek, Sylvain Bellemare, and Andrew Miller. Sgxonerated: Finding (and partially fixing) privacy flaws in tee-based smart contract platforms without breaking the tee. Cryptology ePrint Archive, Paper 2023/378, 2023. `https://eprint.iacr.org/2023/378`.

[48] Christoph Jentzsch. Decentralized autonomous organization to automate governance. *White paper, November*, 2016.

[49] Ari Juels, Dario Catalano, and Markus Jakobsson. Coercion-resistant electronic elections. In *WPES*, pages 61–70, 2005.

[50] Dimitris Karakostas, Aggelos Kiayias, and Christina Ovezik. Sok: A stratified approach to blockchain decentralization. *arXiv preprint arXiv:2211.01291*, 2022.

[51] Mahimna Kelkar, Kushal Babel, Philip Daian, James Austgen, Vitalik Buterin, and Ari Juels. Complete knowledge: Preventing encumbrance of cryptographic secrets. *Cryptology ePrint Archive*, 2023.

[52] Steven P Lalley and E Glen Weyl. Quadratic voting: How mechanism design can radicalize democracy. In *AEA Papers and Proceedings*, volume 108, pages 33–37. American Economic Association 2014 Broadway, Suite 305, Nashville, TN 37203, 2018.

[53] Lido DAO. `https://docs.lido.fi/lido-dao`, Referenced Oct. 2023.

[54] Wouter Lueks, Iñigo Querejeta-Azurmendi, and Carmela Troncoso. VoteAgain: A scalable coercion-resistant voting system. In *USENIX Security*, pages 1553–1570, 2020.

[55] Loi Luu, Duc-Hiep Chu, Hrishi Olickel, Prateek Saxena, and Aquinas Hobor. Making smart contracts smarter. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pages 254–269, 2016.

[56] MakerDAO. The maker protocol: MakerDAO's multi-collateral Dai (MCD) system, 2020. `https://makerdao.com/en/whitepaper/`.

[57] Shaurya Malwa. Attacker takes over Tornado Cash DAO with vote fraud, token slumps 40%. *CoinDesk*, 21 May 2023.

[58] Mantle network. `https://www.mantle.xyz`, Referenced Oct. 2023.

[59] Sinisa Matetic, Moritz Schneider, Andrew Miller, Ari Juels, and Srdjan Capkun. Delegatee: Brokered delegation using trusted execution environments. In *27th USENIX Security Symposium (USENIX Security)*, pages 1387–1403, 2018.

[60] Robert W McGee and Yanira Petrides. How often are voters bribed? a ranking of 82 countries. In *The Ethics of Bribery: Theoretical and Empirical Studies*, pages 367–384. Springer, 2023.

[61] Frank McKeen, Ilya Alexandrovich, Ittai Anati, Dror Caspi, Simon Johnson, Rebekah Leslie-Hurd, and Carlos Rozas. Intel® software guard extensions (Intel® SGX) support for dynamic memory management inside an enclave. In *HASP*, pages 1–9. 2016.

[62] Frank McKeen, Ilya Alexandrovich, Alex Berenzon, Carlos V Rozas, Hisham Shafi, Vedvyas Shanbhogue, and Uday R Savagaonkar. Innovative instructions and software model for isolated execution. In *HASP*, page 10, 2013.

[63] Johnnatan Messias, Vabuk Pahari, Balakrishnan Chandrasekaran, Krishna P Gummadi, and Patrick Loiseau. Understanding blockchain governance: Analyzing decentralized voting to amend defi smart contracts. *arXiv preprint arXiv:2305.17655*, 2023.

[64] David Z. Morris. Coindesk turns 10: 2016 - how the DAO hack changed Ethereum and crypto. *CoinDesk*, 9 May 2023.

[65] Elizabeth Napolitano. Decentralized app sweat economy introduces 1-person, 1-vote governance system. *CoinDesk*, 18 Apr. 2023.

[66] Goldreich Oded. *Foundations of Cryptography: Volume 2, Basic Applications.* Cambridge University Press, USA, 1st edition, 2009.

[67] Optimism. Retroactive public goods funcding. `https://app.optimism.io/retropgf`, Referenced Oct. 2023.

[68] Sunoo Park and Ronald L Rivest. Towards secure quadratic voting. *Public Choice*, 172(1-2):151–175, 2017.

[69] Rafael Pass, Elaine Shi, and Florian Tramèr. Formal abstractions for attested execution secure processors. In *EUROCRYPT*, pages 260–289, 2017.

[70] Ivan Puddu, Daniele Lain, Moritz Schneider, Elizaveta Tretiakova, Sinisa Matetic, and Srdjan Capkun. Teevil: Identity lease via trusted execution environments. *arXiv preprint arXiv:1903.00449*, 2019.

[71] Jack Schickler. Creditors accuse genesis of ballot-stuffing over $175m FTX deal. *CoinDesk*, 1 Sept. 2023.

[72] Tanusree Sharma, Yujin Kwon, Kornrapat Pongmala, Henry Wang, Andrew Miller, Dawn Song, and Yang Wang. Unpacking how Decentralized Autonomous Organizations (DAOs) work in practice. *arXiv preprint arXiv:2304.09822*, 2023.

[73] Snapshot Labs. Shielded voting is live! `https://snapshot.mirror.xyz/yGz91njKbw-sXsnAT6RkoMzPwvuddZ` 13 Oct. 2022.

[74] Balaji Srinivasan and Leland Lee. Quantifying decentralization. `news.earn.com`, 28 Jul. 2017.

[75] Xiaotong Sun, Xi Chen, Charalampos Stasinakis, and Georgios Sermpinis. Multiparty democracy in decentralized autonomous organization (DAO): Evidence from MakerDAO. *arXiv preprint arXiv:2210.11203*, 2022.

[76] Kevin Tai. DAOs are meant to be completely autonomous and decentralized, but are they?, Feb. 2022.

[77] Eli Tan. “Do you believe in second chances?”: Another DAO is raising funds to buy a copy of the US Constitution. *CoinDesk*, 7 Dec. 2022.

[78] Stephan van Schaik, Andrew Kwong, Daniel Genkin, and Yuval Yarom. SGAxe: How SGX fails in practice, 2020. `https://sgaxe.com/files/SGAxe.pdf`.

[79] Kevin Werbach. Trust, but verify: Why the blockchain needs the law. *Berkeley Technology Law Journal*, 33(2):487–550, 2018.

[80] E Glen Weyl. The robustness of quadratic voting. *Public choice*, 2017.

[81] Wikipedia contributors. Min-entropy — Wikipedia, the free encyclopedia, Referenced Oct. 2023.

[82] Worldcoin. The protocol to bring global proof of personhood to the internet. Wordcoin Docs, `https://docs.worldcoin.org/use-cases`, Referenced Oct. 2023.

[83] Steven A Wright. Measuring DAO autonomy: Lessons from other autonomous systems. *IEEE Transactions on Technology and Society*, 2(1):43–53, 2021.

[84] Fan Zhang, Ethan Cecchetti, Kyle Croman, Ari Juels, and Elaine Shi. Town crier: An authenticated data feed for smart contracts. In *Proceedings of the 2016 aCM sIGSAC conference on computer and communications security*, pages 270–282, 2016.