How Hard is Takeover in DPoS Blockchains? Understanding the Security of Coin-based Voting Governance

Chao Li
Beijing Key Laboratory of Security
and Privacy in Intelligent
Transportation
Beijing Jiaotong University
Beijing, China
li.chao@bjtu.edu.cn

Balaji Palanisamy*
Department of Informatics and
Networked Systems
School of Computing and Information
University of Pittsburgh
Pittsburgh, USA
bpalan@pitt.edu

Runhua Xu
School of Computer Science and
Engineering
Beihang University
Beijing, China
runhua@buaa.edu.cn

Li Duan
Beijing Key Laboratory of Security
and Privacy in Intelligent
Transportation
Beijing Jiaotong University
Beijing, China
duanli@bjtu.edu.cn

Jiqiang Liu*
Beijing Key Laboratory of Security
and Privacy in Intelligent
Transportation
Beijing Jiaotong University
Beijing, China
jqliu@bjtu.edu.cn

Wei Wang*
Beijing Key Laboratory of Security
and Privacy in Intelligent
Transportation
Beijing Jiaotong University
Beijing, China
wangwei1@bjtu.edu.cn

ABSTRACT

Delegated-Proof-of-Stake (DPoS) blockchains, such as EOSIO, Steem and TRON, are governed by a committee of block producers elected via a coin-based voting system. We recently witnessed the first de facto blockchain takeover that happened between Steem and TRON. Within one hour of this incident, TRON founder took over the entire Steem committee, forcing the original Steem community to leave the blockchain that they maintained for years. This is a historical event in the evolution of blockchains and Web 3.0. Despite its significant disruptive impact, little is known about how vulnerable DPoS blockchains are in general to takeovers and the ways in which we can improve their resistance to takeovers.

In this paper, we demonstrate that the resistance of a DPoS blockchain to takeovers is governed by both the theoretical design and the actual use of its underlying coin-based voting governance system. When voters actively cooperate to resist potential takeovers, our theoretical analysis reveals that the current active resistance of DPoS blockchains is far below the theoretical upper bound. However in practice, voter preferences could be significantly different. This paper presents the first large-scale empirical study of the passive takeover resistance of EOSIO, Steem and TRON. Our study identifies the diversity in voter preferences and characterizes the impact of this diversity on takeover resistance. Through both theoretical and empirical analyses, our study provides novel insights into the security of coin-based voting governance and suggests potential ways to improve the takeover resistance of any blockchain that implements this governance model.

CCS CONCEPTS

• **Security and privacy → Distributed systems security**.

KEYWORDS

Blockchain; Decentralized Governance; Governance Security; Voting Governance; Delegated Proof of Stake; Web 3.0

1 INTRODUCTION

Blockchain technologies are fueling the emergence of decentralized applications and Web 3.0, where authority and power are spread across the network without interference from any single entity. Traditional Proof-of-Work (PoW) consensus protocols, used by Bitcoin [44] and Ethereum 1.0 [5], require the decentralized consensus to be made throughout the entire network. As a result, the throughput of transactions in these networks is limited by the network scale (e.g., Bitcoin has a maximum throughput of 7 transactions/sec [7]), making it practically difficult to satisfy the needs of many applications. On the other hand, traditional Proof-of-Stake (PoS) consensus protocols, as adopted by blockchains such as Ethereum 2.0 [33], require coin holders to make substantial collateral deposits (e.g., 32 ETH in Ethereum) to participate in governance, preventing the involvement of numerous coin holders with insufficient funds. To address scalability and participation concerns, the Delegated Proof-of-Stake (DPoS) consensus protocol [37] has recently gained popularity and has given rise to a series of successful blockchains, such

as EOSIO [22], Steem [39] and TRON [52][1]. In DPoS, the consensus is only reached among a small set of block producers (BPs) (e.g., 21 BPs in EOSIO and Steem and 27 BPs in TRON). Furthermore, any coin holder can participate in BP elections, making DPoS a very promising technical choice for applications that require high transaction throughput and inclusive governance participation.

Coin-based voting governance. DPoS blockchains are governed by a committee of block producers (BPs) who are periodically elected by coin holders[2] via a coin-based voting system. Coin holders (or voters) are encouraged to stake (i.e., freeze) their coins to gain voting power and cast votes that are weighted by their voting power. The top candidates ranked by the received voting power then become BPs. In DPoS, BPs are essentially the rule makers of the blockchain. BPs can update a wide range of rules in the blockchain by executing a proposal, sometimes called a *fork*, ranging from changing system parameters to blacklisting certain accounts, or even reversing confirmed transactions, as long as the supermajority of BPs (15 out of 21 BPs in EOSIO, 17 out of 21 BPs in Steem and 19 out of 27 BPs in TRON) agree on the proposal. For instance, in TRON, BPs can propose to modify the amount of block generation reward [30], which allows them to determine their own salaries. A more interesting incident occurred in Nov. 2018 when 16 out of 21 BPs of EOSIO approved the first-ever proposal of changing the private key of an EOS account to resolve a dispute on the account's ownership [47]. This marks a significant event in the history of blockchains as an account's private key, used for signing transactions issued by the account is generally considered to be immutable.

Takeover. A takeover in DPoS blockchains refers to an attacker controlling the supermajority of BPs and as a result, gaining immense control of the blockchain including the ability to reverse confirmed transactions and change the private keys of accounts. In contrast, the most significant attack in PoW blockchains is the double-spending attack [32, 44], which occurs when an attacker with the majority of the mining power reverses confirmed transactions to spend a coin twice. The first de facto takeover attack, known as TRON's takeover of Steem, has occurred recently. In early 2020, TRON founder purchased pre-mined coins[3] from Steemit Inc. [29], the company that launched the Steem blockchain. Although Steemit Inc. promised to never use these coins in BP election, TRON did not make such a commitment. Therefore, the top BPs in Steem (those not belonging to Steemit Inc.) prohibited the use of pre-mined coins in BP election via *fork* 0.22.2 [24]. However, on Mar. 2, 2020, within one hour, all the BPs in Steem were quickly replaced by accounts controlled by TRON founder, who then immediately revoked *fork* 0.22.2 via *fork* 0.22.5 [25], forcing the original BPs and the Steem community to leave the blockchain they maintained for years.

TRON's takeover of Steem is not the only attempt of takeovers in DPoS blockchains. In Dec. 2021, Block.one, the company that launched the EOSIO blockchain, announced its plan of transferring pre-mined coins (about 6% of EOS total supply) to another company. At that moment, the top BPs in EOSIO are members of an organization named EOS Network Foundation [23]. Therefore, they deployed a proposal which basically stopped Block.one from controlling pre-mined coins [46]. As a result, takeover did not happen in this instance. The reason is directly attributed to the higher takeover resistance in EOSIO, a key topic of focus in this paper.

This paper. Despite its significant disruptive impact, little is known about how vulnerable DPoS blockchains are in general to takeovers and the ways in which we can improve their resistance to takeovers. In this paper, we demonstrate that the resistance of a DPoS blockchain to takeovers is governed by both the theoretical design and the actual use of its underlying coin-based voting governance system. We formally describe a three-phase model for coin-based voting governance and formalize the takeover attack and resistance model based on our analysis of TRON's takeover of Steem. We formally model the *takeover game* between an attacker and the cooperative resisters and prove the existence of a Nash equilibrium. When voters actively cooperate to resist potential takeovers, our theoretical analysis of the impact of the design of the underlying voting system on takeover resistance demonstrates that the current active resistance is far below the theoretical upper bound. However in practice, voter preferences could be significantly different. We present the first large-scale empirical study of the passive takeover resistance of EOSIO, Steem and TRON. Our study identifies the diversity in voter preferences and characterizes the impact of this diversity on takeover resistance. Through both theoretical and empirical analyses, our study provides novel insights into the security of coin-based voting governance and suggests potential ways to improve the takeover resistance of any blockchain that implements this governance model.

Organization. We start by introducing the background in Section 2. We model the coin-based voting governance in Section 3 and formalize the takeover attack and resistance model in Section 4. In Section 5, we investigate the *takeover game* and demonstrate the existence of an upper bound for the active takeover resistance. In Section 6, we study the passive takeover resistance of EOSIO, Steem and TRON. We suggest potential ways to improve the takeover resistance and discuss the generalization of our analysis in Section 7. We discuss related work in Section 8 and conclude in Section 9.

2 BACKGROUND

In this section, we introduce various DPoS blockchains, primarily from the perspective of governance. We focus our discussion specifically around EOSIO, Steem and TRON blockchains. These three blockchains were involved in events related to takeovers recently. Also, these blockchains are among the top cryptocurrency projects that have attracted millions of users and collected billions of transactions[4] from users [36]. Their rich data helps validate our results. Furthermore, the design of coin-based voting governance in these blockchains is consistent and shares several common aspects which help generalize our results.

---

[1]We chose EOSIO, TRON, and Steem for our study due to their representativeness within the DPoS ecosystem [36], their rich data relevant to takeovers, as well as the widespread attention they have received from researchers [21, 22, 39, 42, 45].

[2]Blockchains usually issue tradable cryptocurrencies as coins (e.g., EOS for EOSIO, TRX for TRON and STEEM for Steem).

[3]The amount of coins issued to founders as rewards, which is about 20% of STEEM total supply in this case.

[4]A basic record of user behavior, such as casting a vote or transferring a coin, is named an action/operation/transaction in EOSIO/Steem/TRON, respectively. In the rest of this paper, we refer to them collectively as transactions.

2.1 EOSIO

EOSIO is a successful DPoS blockchain and its market capitalization was consistently among the top 10 blockchain projects [36]. Similar to Ethereum [5], EOSIO supports smart contracts [53] with its underlying virtual machine, enabling developers to quickly build decentralized applications (dapps)[5] on the EOSIO platform. Rapid developments in EOSIO have attracted researchers to study various aspects of EOSIO including smart contract security [22], dapps [8] and decentralization [42].

Governance in EOSIO. The design of the governance system here is primarily based on a combination of two voting rules, liquid democracy [56] and multi-winner approval voting [49].

- **Liquid democracy:** This voting rule allows a voter to choose between two options: (1) cast her votes directly for BP candidates by herself; (2) delegate her voting power to a proxy, who may in turn choose between the two options. With the first option, a voter's votes would be weighted by her own voting power. However, with the second option, multiple voters may form a delegation chain (i.e., everybody except the end voter in the chain chose option two) or a tree (i.e., everybody except the root voter in the tree chose option two), and voting power of the chain (tree) would be aggregated at the end (root) voter, whose votes would be weighted by the aggregated voting power.

- **Multi-winner approval voting:** In this voting rule, a voter is allowed to cast multiple votes (30 votes in EOSIO) with each vote going to a distinct BP candidate. Here, each vote of a voter would be weighted by the voter's entire voting power, including her own voting power and any voting power concentrated from delegations. By the end of the election cycles, BP candidates are ranked by the voting power they received and a set of top candidates (top 21 in EOSIO) win the election and form a committee. From then on, any proposal issued to the committee needs to be approved by at least 15 BPs to get adopted.

2.2 Steem

Steem is another prominent DPoS blockchain that supports numerous social applications. There have been over 324 Steem-based decentralized applications [2], many of which are designed to serve social users. *Steemit* [28] is one of the first and the most prominent application in Steem. It represents a decentralized version of Reddit, where users can create and share content as blog posts to receive replies, reposts, upvotes or downvotes. The platform periodically allocates a number of coins called STEEM to reward authors of top-ranked posts. Steem has received extensive attention from both the blockchain community [39, 40] as well as the social network community [21, 45] in the recent years.

Governance in Steem. The governance system in Steem is very similar to that of EOSIO. Steem also employs both liquid democracy and multi-winner approval voting and allows each voter to cast at most 30 votes. Steem is governed by a committee of 21 members. However, there are two main differences between EOSIO and Steem. In Steem, only 20 out of 21 BPs are determined by the election, while the last BP in the committee is rotated among candidates outside

Chain	Voting Rule	MaxVote (v)	CmteSize (n)	MinApprov (t)
EOSIO	AV(+LD)	30	21	15
Steem	AV(+LD)	30	20(+1)	17
TRON	CV	30	27	19

Table 1: Summary of key design choices made by EOSIO, Steem and TRON. Here, LD/AV/CV refer to liquid democracy, approval voting and cumulative voting, respectively.

the top 20. Also, a proposal in Steem needs to receive 17 approvals to get implemented.

2.3 TRON

TRON is one of the youngest blockchains employing proof-of-stake principles as its consensus algorithm. Its market capitalization was also among the top-20 blockchain projects [36, 51]. Similar to EOSIO, through its support for smart contracts, the ecosystem of TRON has quickly spread across various areas including Non-Fungible Token (NFT), stable coins and decentralized exchanges.

Governance in TRON. The governance system in TRON is quite different from those in EOSIO and Steem. TRON replaces liquid democracy and approval voting with cumulative voting [3], another well-studied voting system. We briefly introduce its concept here and we formally model it in Section 3.

- **Multi-winner cumulative voting:** Similar to approval voting, multi-winner cumulative voting allows a voter to cast multiple votes (30 votes in TRON) with each vote going to a distinct BP candidate. However, unlike approval voting, here, if a voter decides to cast multiple votes, she must divide her entire voting power into different votes so that the sum of voting power allocated to all votes is no more than her voting power in total. Similar to approval voting, BP candidates are then ranked by the voting power they received and multiple top candidates (top 27 in TRON) win the election (i.e., become BPs) and form a committee. In TRON, however, a proposal needs to be approved by at least 19 BPs to get adopted.

In Table 1, we summarize the key design choices made by EOSIO, Steem and TRON. Please note that, in the rest of this paper, we denote the max votes per voter parameter by MaxVote v, the committee size parameter by CmteSize n and the min approvals per proposal parameter by MinApprov t.

3 COIN-BASED VOTING GOVERNANCE

In this section, we distill the governance systems introduced in Section 2 into three distinct phases and provide a formal description of a three-phase model for coin-based voting governance.

3.1 Phases of Coin-based Voting Governance

In coin-based voting governance, the process of gradually transforming individual coins into governance decision-making power takes place through three distinct phases. This process enables all coin holders to participate in the process while maintaining a unique balance between scalability and decentralization.

- **Phase 1: staking.** During the first phase, individual coins are converted into individual voting power. Coin holders lock or *stake* their coins to obtain voting power proportional to the amount

---

[5]The back-end of dapps runs by BPs of DPoS blockchains in the form of codes named smart contracts.
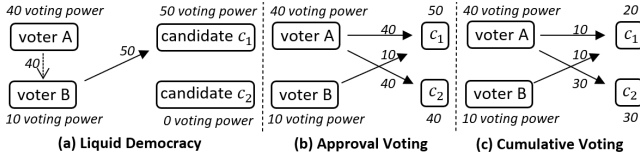
Figure 1: Phase 2 with distinct voting systems.

Figure 2: A model of coin-based voting governance

of coins staked. This encourages active participation from all coin holders, fostering inclusiveness and decentralization in the decision-making process.

- **Phase 2: voting.** In the voting phase, individual voting power is aggregated. Coin holders cast their votes, weighted by their voting power, in support of their preferred block producers (BPs). This pooling of voting power enables the community to collectively determine the most suitable BPs for governance.
- **Phase 3: governing.** In the governing phase, the pooled voting power is converted into governance decision-making power. The top candidates ranked by the received voting power become BPs. These BPs form a smaller consensus group that is responsible for making decisions on behalf of the entire network.

Among them, Phase 2 may utilize distinct voting systems for aggregating individual voting power, resulting in diverse outcomes.

Example. In the example illustrated by Figure 1, voters A and B obtain 40 and 10 voting power in Phase 1, and vote for candidates c_1 and c_2 in Phase 2. With liquid democracy, voter A delegates her 40 voting power to voter B, increasing B's weight to 50. In approval voting, both of A's votes receive a full 40 voting power. In cumulative voting, A's two votes share her 40 voting power.

3.2 Modeling Coin-based Voting Governance

We present a high-level overview of the coin-based voting governance model in Figure 2. We consider a setting (M, C) for coin-based voting governance based on voting rules [49], where $M = \{1, 2, ..., m\}$ represents the set of voters and $C = \{c_1, ..., c_k\}$ represents the set of candidates. Based on this notion, we model the three phases of coin-based voting governance described in Section 3.1.

3.2.1 **Phase 1: (un)staking.** To make any contribution to the governance, a voter i needs to *stake* (i.e., freeze/lock) her coins to earn some voting power via a staking function $p_i = S(coin_i, \lambda)$, where $coin_i$ represents the coins of the voter and the parameter λ governs how much voting power is earned by each coin. The staking function returns the amount of voting power p_i for the voter. The configuration of the parameter λ typically varies across different DPoS blockchains. In TRON, $\lambda = 1$ and it indicates that one coin simply corresponds to one unit of voting power. In Steem, λ is approximately 2000 and therefore, each coin could be converted into 2000 units of voting power. On the other hand, EOSIO adopts a more sophisticated approach[6] where λ is set to be the timestamp of the latest voting transaction performed by the voter. This encourages voters to frequently cast votes to increase the value of λ so that they could receive a higher amount of voting power with the same number of staked coins.

In almost all DPoS blockchains, staked coins are not allowed to be withdrawn for a certain period of time, ranging from a few days

to several weeks. After this time, a voter i may choose to unstake her coins using an unstaking function $coin_i = S^{-1}(p_i, \lambda)$. Naturally, unstaking the voter's staked coins will result in a decrease of her voting power. Overall, the staking and the unstaking processes results in a dynamically changing voting power profile for the blockchain denoted as $\mathbf{p} = (p_1, ..., p_m)$, which captures a snapshot of the voting power of the set of voters M at any given time point.

3.2.2 **Phase 2: (v, n)-voting.** Given the set of voters and candidates (M, C) and the voting power profile $\mathbf{p}$, the goal of phase 2 is to determine a winning set $W \subseteq C$, also referred to as a committee, which maximizes a community choice score τ. Phase 2 is referred to as the (v, n)-voting phase. Here, v and n refer to the max votes per voter parameter MaxVote and the committee size parameter CmteSize, respectively, as shown in Table 1. The score τ represents the sum of voting power received by the top-n candidates.

Liquid democracy. As discussed in Section 2, liquid democracy and approval voting are two primary mechanisms used in the DPoS governance structure. Specifically, liquid democracy is used in combination with approval voting in both EOSIO and Steem. On the other hand, a governance system may also choose to not use liquid democracy, as in the case of TRON. Based on recent work in liquid democracy [56], we define the delegation profile of the blockchain as $\mathbf{d} = (d_1, ..., d_m)$, where $d_i = j$ indicates that voter $i \in M$ is delegating her entire voting power to voter $j \in M$ [7]. As we discussed in Section 2.1, voters may form delegation chains (trees), and all their voting power will be aggregated at the end (root) voters, who are commonly referred to as gurus [56]. Note that, if voter i did not delegate her voting power, we have $d_i = i$ which indicates that voter i is her own guru. We define liquid democracy as follows:

DEFINITION 1 (**LIQUID DEMOCRACY**). *Given a triple $\langle M, \mathbf{p}, \mathbf{d} \rangle$ as input, Liquid Democracy determines a couple $\langle M^*, \mathbf{p}^* \rangle$, where M is the set of voters, $\mathbf{p}$ is the voting power profile of voters, $\mathbf{d}$ is the delegation profile of voters, $M^* \subseteq M$ is the set of gurus, $\mathbf{p}^*$ is the (aggregated) voting power profile of gurus.*

In other words, based on $\langle M, \mathbf{p}, \mathbf{d} \rangle$, liquid democracy would be able to determine a subset M^* of the set of voters M who have either been end (root) voters of delegation chains (trees) or voted by themselves. This subset M^* and its corresponding (aggregated) voting power profile $\mathbf{p}^*$ would then be delivered to the next subphase, namely to the approval voting or cumulative voting subphase. Please note that, in the rest of this paper, to avoid any confusion between the two terms 'voters' and 'gurus', we will ignore their differences and use 'voters' consistently to refer to the subset M^* because our work is more focused on approval/cumulative voting.

Approval voting. We now proceed to modeling approval voting and cumulative voting. In multi-winner approval voting, multiple winners are determined via approval voting. Here, the MaxVote

---

[6]We refer the interested readers to [42] for more details on EOSIO staking function.

[7]In EOSIO and Steem, a voter is only allowed to delegate her voting power to a single voter and she must delegate her entire voting power.

parameter v denotes that a voter i is allowed to cast at most v votes to v distinct BP candidates and each vote is equally weighted by voter i's entire voting power p_i^*. Therefore, we define a voting profile for all voters in M^* as $V = \{V_i | i \in M^*\}$, where voter i selects a subset V_i of the set of candidates C to vote such that $|V_i| \le v$. We could then define multi-winner approval voting as follows:

DEFINITION 2 (**MULTI-WINNER APPROVAL VOTING**). *Given a tuple $\langle M^*, \mathbf{p}^*, V, C, n \rangle$ as input, Multi-Winner Approval Voting determines a committee $W \subseteq C$, such that $|W| = n$ and the community choice score $\tau = \sum_{i \in M^*} |W \cap V_i| p_i^*$ is maximized.*

This definition indicates that after ranking all candidates (C) based on the voting power that they have received from all voters (M^*), the top n (i.e., CmteSize) candidates form a committee W, which is then provided to phase 3. The voting power received by the committee W represents the maximized community choice score τ.
Cumulative voting. In contrast to approval voting, multi-winner cumulative voting adopts a different approach to determine multiple winners. Instead of equally weighting each vote by voter i's entire voting power p_i^*, a voter i in cumulative voting has to distribute her voting power p_i^* across all selected candidates, namely V_i. Thus, we could define a power distribution profile for all voters in M^* as $P = \{P_i | i \in M^*\}$, where $P_i = \{p_{i,j}^* | c_j \in V_i, \sum_j p_{i,j}^* \le p_i^*\}$ so that different candidates c_j selected by voter i may receive different amounts of voting power $p_{i,j}^*$ from voter i. We could then define multi-winner cumulative voting as follows:

DEFINITION 3 (**MULTI-WINNER CUMULATIVE VOTING**). *Given a tuple $\langle M^*, \mathbf{p}^*, V, P, C, n \rangle$ as input, Multi-Winner Cumulative Voting determines a committee $W \subseteq C$, such that $|W| = n$ and the community choice score $\tau = \sum_{i \in M^*} \sum_{c_j \in W \cap V_i} p_{i,j}^*$ is maximized.*

In summary, multi-winner approval voting allows each unit of voting power to be used for up to v times (i.e., MaxVote), while multi-winner cumulative voting allows each unit of voting power to be used only once. The output of both voting systems is a committee W. However, the ranking of BP candidates may change whenever new delegating/voting/(un)staking transactions arrive.

3.2.3 **Phase 3: (t, n)-governing.** Given a committee W, in phase 3, every proposal issued to the committee W must receive a minimum of t distinct approvals (i.e., MinApprov in Table 1) to get adopted.

Together, the three phases gradually convert coins owned by holders into voting power to a committee and finally into governance decision-making power. However, from the point of view of security, little is known about how vulnerable coin-based voting governance is in general to takeover and the ways in which we can improve their resistance to takeovers. In the next section, we will start answering questions along these aspects.

4 TAKEOVER ATTACK/RESISTANCE: LESSONS FROM TRON'S TAKEOVER OF STEEM

In this section, by reviewing the intricacies of TRON's takeover of Steem and Steem's resistance against TRON's takeover, we introduce and formalize the takeover attack and resistance model. We also present the two key research questions that drive our study in the next two sections.
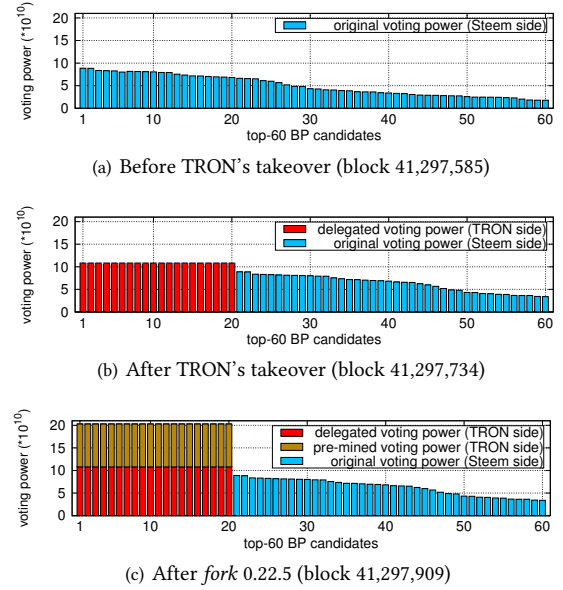

(a) Before TRON's takeover (block 41,297,585)

(b) After TRON's takeover (block 41,297,734)

(c) After *fork* 0.22.5 (block 41,297,909)

Figure 3: The shift in rankings of the top BP candidates in Steem before and after TRON's takeover.
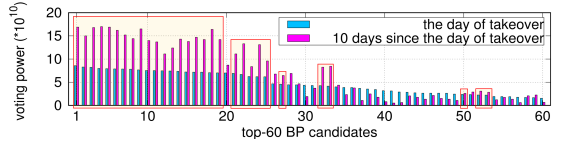

Figure 4: The variation in voting power of top BP candidates before takeover versus 10 days after takeover. The boxed BPs represent those suggested by the call-to-action.

4.1 TRON's Takeover of Steem

We carefully review the blockchain data of Steem that was generated on the takeover day (March 2, 2020). Based on our investigation, it is interesting to find out that the takeover was implemented within 44 minutes, from block 41,297,060 to 41,297,909 (3 seconds per block), in three phases that closely follow the model presented in Section 3 well. It is important to note that we only present the objective information that we obtained from the blockchain data unless explicitly stated.

- **Phase 1: staking.** During the first 27 minutes of the 44 minutes, we find that three accounts converted $7,469,573 worth of coins to voting power. Meanwhile, based on the Liquid Democracy rule, these three accounts, along with six other accounts, delegated their voting power to the same proxy, which means that any vote cast by this proxy would be weighted by the huge amount of voting power delegated to it. As illustrated in Figure 3(a), by the end of this phase, the proxy had not yet cast any vote and therefore, all the top-60 BP candidates were still controlled by the original voting power owned by the Steem community.
- **Phase 2: $(30, 21)$-voting.** During the next 8 minutes after the first 27 minutes, the proxy cast votes to 20 distinct BP candidates one by one. These BP candidates had received nearly no voting power previously. As illustrated in Figure 3(b), by the end of this phase, all the top-20 seats were occupied by the BP candidates that were supported by the proxy. We can also observe that all the

original top BP candidates in Figure 3(a), whose voting power did not significantly change during the 8 minutes, dropped exactly 20 places in the ranking. It is worth noting that, as we have introduced in Section 2.2, one may at most take over 20 out of all the 21 seats in the committee of Steem because the last seat rotates among candidates outside the top 20.

- **Phase 3:** $(17, 21)$-**governing.** During the last 9 minutes, the top-20 BPs had the ability to pass any proposal they wanted. Recall that the use of pre-mined coins (i.e., the coins that TRON founder purchased from Steemit Inc.) in BP election had been prohibited at an earlier time by a proposal passed by the original committee, namely *fork* 0.22.2. The new committee then revoked the prohibition of the use of pre-mined coins in BP election by implementing a new proposal, namely *fork* 0.22.5. The pre-mined coins were then immediately used to support the top-20 BPs. By the end of this phase (Figure 3(c)), fuelled by the power of pre-mined coins, all the top-20 BPs gained significant advantages, rendering them nearly undefeated.

4.2 Steem's Resistance Against TRON's takeover

We identified two resistance patterns against TRON's takeover.
Passive resistance. In the takeover process outlined in Section 4.1, the original voting power acquired by BP candidates from Steem community members forms a passive resistance against the takeover, compelling the attacker to amass substantial voting power. For example, during Phase 1, TRON founder accumulated $7,469,573 worth of coins as voting power.
Active resistance. Subsequently, we investigated the blockchain data of Steem within ten days after the takeover and discovered an active resistance against the takeover. The active resistance consists of two crucial stages: a leader initiates a call-to-action [6], followed by the collaborative response of community members. More concretely, amidst a hostile takeover, a well-respected community member leads the resistance by issuing a call-to-action, which functions as a rallying cry that inspires the community to protect their shared interests against the takeover attempt. In response, some community members pool their resources and form a cohesive voting front to counter the takeover. Together, they create a formidable voting power dedicated to supporting a list of candidates suggested by the call-to-action. We formalize the active resistance in Section 4.3 and provide more details in Appendix A.

Figure 4 illustrates the practical impact of the active resistance in the case of TRON's takeover. On the day of the takeover, a renowned Steem community member posted a call-to-action [6] on the Steemit platform, which garnered the highest number of comments within ten days. Ten days later, all the BPs suggested by the call-to-action witnessed positive growth in voting power and occupied the top 25 rankings, emerging as the core BPs countering the takeover. In contrast, the majority of BPs not endorsed in the call-to-action experienced a decrease in their voting power.

4.3 Modeling Takeover Attack and Resistance

As illustrated by TRON's takeover of Steem, in a takeover event, an attacker attempts to take over a blockchain, while some community members of the target blockchain strive to resist this takeover. The takeover attack and resistance model is depicted in Figure 5.
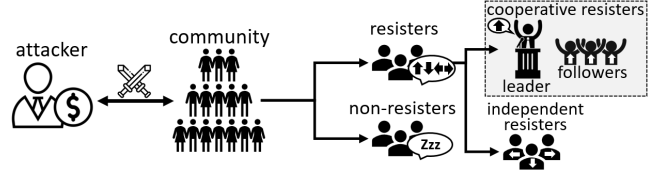

Figure 5: The takeover attack and resistance model.

Attack. In a takeover event, the goal of an attacker $\mathcal{A}$ is to be able to launch *forks* (i.e., passing proposals to change the blockchain rule), which is equivalent to occupying at least t seats of the committee. To do that, an attacker $\mathcal{A}$ needs to explicitly or implicitly control three types of resources, namely a subset C_a of the set of candidates C where $|C_a| \geq t$ (i.e., MinApprov), a subset M_a of the set of voters M, as well as an amount of voting power p_a. Then, $\mathcal{A}$ needs to follow a strategy s_a, which is simply defined as the way of distributing $\mathcal{A}$'s voting power p_a across $\mathcal{A}$'s candidates C_a via votes cast by $\mathcal{A}$'s voters M_a. Here, it's worth noting that some voting systems (e.g., approval voting) allow each unit of voting power to be used multiple times. Therefore, we introduce a power amplification coefficient ζ_a to capture the amplification effect of the voting system settings to $\mathcal{A}$'s voting power p_a. The value of ζ_a is only related to parameters (v, t, n). We present more details of the amplification effect in Section 5.

We formalize takeover attacks as follows:

DEFINITION 4 (**TAKEOVER ATTACK**). *An attacker $\mathcal{A}$, who controls $\langle M_a, C_a, p_a \rangle$, implements a strategy $s_a = \{p_{a,i} | c_i \in C_a, p_{a,i} \leq p_a, \sum_i p_{a,i} \leq \zeta_a p_a\}$ of distributing $\zeta_a p_a$ across C_a, such that the committee W output from the (v, n)-voting phase satisfies $|W \cap C_a| \geq t$, where ζ_a is the power amplification coefficient of $\mathcal{A}$.*

Resistance. It is important to note that in a takeover event, the behaviors of the target blockchain community members may not be monolithic. Intuitively, some community members may engage in active resistance, while others may remain indifferent and abstain from taking any action. These two types of community members are referred to as *resisters* and *non-resisters*, respectively. Moreover, the behaviors of resisters may exhibit variations. Some resisters might follow a call-to-action and concentrate their voting power on a few suggested candidates, while others might disregard any suggested candidates. We denote these two type of resisters as *cooperative resisters* and *independent resisters*, respectively.

More formally, we categorize community members who modify their selected candidate set by executing delegating/voting transactions within a short period (e.g., 1 day) after the takeover as *resisters*, and those who retain their selected candidate set as *non-resisters*. Expanding upon this classification, we identify the author of the most popular call-to-action post (e.g., [6]) as the leader and denote the leader's chosen candidate set as C_l. We then classify *resisters* with a chosen candidate set C_r satisfying $|C_r \cap C_l| \geq 1$ as *cooperative resisters (co-resisters)*, who follow the active resistance pattern introduced in Section 4.2, and those with a C_r satisfying $|C_r \cap C_l| = 0$ as *independent resisters (ind-resisters)*.

In practice, we observe that co-resisters in the Steem community, who adopt the active resistance pattern, serve as the primary force in countering TRON's takeover. As depicted in Figure 6, within one day after the takeover, voters generated nearly 40,000 voting/delegating transactions, which is a hundred times the daily
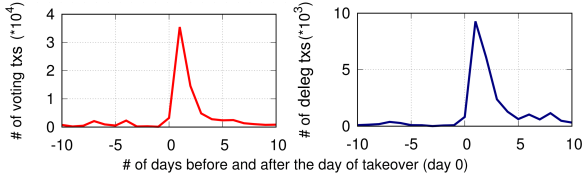
Figure 6: Variations in the number of voting transactions (voting txs) and delegating transactions (deleg txs).
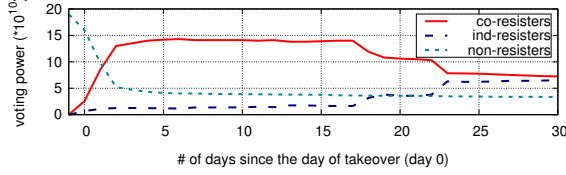

Figure 7: Variations in the total voting power of different types of community members.

average before the takeover. This indicates that a substantial number of community members began to take action. As demonstrated in Figure 7, after the takeover, the total voting power of co-resisters grew rapidly, exceeding ten times the total voting power of ind-resisters on the second day. It then remained stable and surpassed the total voting power of non-resisters by three times on the fifth day, only to decrease when the Steem community began migrating to the new blockchain, Hive [45]. This demonstrates that co-resisters played a pivotal role in resisting the takeover.

Based on the aforementioned analysis, we formalize the active resistance led by co-resisters as follows:

DEFINITION 5 (**ACTIVE RESISTANCE**). *A group of co-resisters $\mathcal{R}$, who controls an amount of voting power p_r, implements a strategy $s_r = \{p_{r,i}|c_i \in C_l, p_{r,i} \le p_r, \sum_i p_{r,i} \le \zeta_r p_r\}$ of distributing $\zeta_r p_r$ across the leader's chosen candidate set C_l, such that the committee W output from the (v, n)-voting phase satisfies $|W \cap C_a| < t$, where ζ_r is the power amplification coefficient of $\mathcal{R}$. The active takeover resistance, denoted as R_A, is quantified as the minimum amount of voting power p_a an attacker $\mathcal{A}$ needs to defeat the co-resisters and successfully take over the target blockchain.*

In this definition, ζ_r is used to capture the amplification effect (i.e., reuse each unit of voting power multiple times in approval voting) of the voting system settings to $\mathcal{R}$'s voting power p_r, which is solely related to parameters (v, t, n). We elaborate more on both ζ_r and ζ_a in Section 5. Also, we provide the quantification for the active takeover resistance R_A here and will discuss its theoretical upper bound in Section 5 in more detail.

Similarly, when co-resisters are either absent or their power is insignificant, we define the passive resistance as follows:

DEFINITION 6 (**PASSIVE RESISTANCE**). *The target blockchain community members distribute their voting power across the candidate set C. The passive takeover resistance, denoted as R_P, is quantified as the minimum amount of voting power p_a an attacker $\mathcal{A}$ needs to defeat the target blockchain community members and successfully take over the target blockchain.*

4.4 Discussion

From the takeover event between TRON and Steem, we observe two key factors that may influence the active and/or passive resistance.

Design of the voting system: The first potential factor involves choosing between approval and cumulative voting, as well as selecting parameters (v, t, n). For instance, intuitively, TRON's takeover would have been more difficult if the Steem blockchain had adopted a smaller v (i.e., MaxVote). Currently, in EOSIO, Steem and TRON, the MaxVote parameter v is larger than the MinApprov parameter t, enabling an attacker $\mathcal{A}$ to reuse $\mathcal{A}$'s voting power p_a to contest each top-20 committee seat, as illustrated by TRON's takeover of Steem. However, it may be non-trivial to draw conclusions because a smaller v would also constrain the power of both sides.

Actual voter preferences: The second potential factor pertains to the characteristics of voter preferences, including the number of votes cast and the priorities assigned to selected candidates. As shown in Figure 3(a), during TRON's takeover, a significant portion of the original voting power was allocated to low-ranking BP candidates due to the diversity of voter preferences. The phenomenon may be desirable from the perspective of community choice. However, such a dispersion of defensive voting power may make a DPoS blockchain more vulnerable to takeovers because an attacker's voting power is presumed to be always highly concentrated.

To further analyze active and passive takeover resistance, we pose the following two key research questions. We address each in the subsequent sections.

> **RQ 1 [Active Resistance]:** When resistance is led by co-resisters (e.g., Section 4.2), how can the voting system be designed to maximize the effectiveness of active resistance?
>
> **RQ 2 [Passive Resistance]:** When resistance is passive (e.g., Section 4.1) or the power of co-resisters is much lower than that of non-resisters, how can we understand actual voter preferences and based on them, how can we design a voting system to enhance the effectiveness of passive resistance?

5 ACTIVE RESISTANCE: TAKEOVER GAME

In this section, we address the first research question by modeling a *takeover game* between two players, namely an attacker and the co-operative resisters (co-resisters). We show the strategies of the two players in a Nash equilibrium and demonstrate the existence of an upper bound for the active takeover resistance of DPoS blockchains for both approval voting and cumulative voting.

The game model. A takeover attack in DPoS blockchains can be modeled as a perfect-information extensive-form game [38], which reflects the real-world event of TRON's takeover of Steem. The game involves two players $(\mathcal{A}, \mathcal{R})$. The first player is an attacker $\mathcal{A}$ who controls an *alterable* amount of voting power p_a and a set of candidates C_a, where $|C_a| = n$ (i.e., CmteSize). The second player is the co-resisters $\mathcal{R}$ who controls a *fixed* amount of voting power p_r and a set of leader's chosen candidates C_l, where $|C_l| = n$ and $|C_l \cap C_a| = 0$. The attacker and co-resisters play sequentially in the game, observing all prior steps from the blockchain. This aligns with the perfect-information extensive-form game model. The game concludes in one round because the attacker, controlling an alterable amount of voting power p_a, can secure over t seats in the committee (Definition 4). The game consists of two stages. In the first stage, $\mathcal{R}$ needs to determine the strategy s_r of distributing $\mathcal{R}$'s (amplified) voting power $\zeta_r p_r$ across $\mathcal{R}$'s candidates C_l. In the second stage, after learning the distribution of $\mathcal{R}$'s voting power
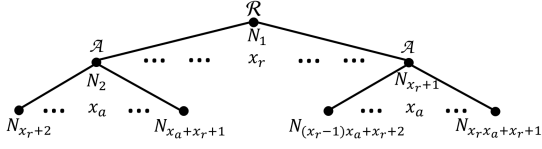
Figure 8: The game tree in active resistance.

$\zeta_r p_r$ from the blockchain data (i.e, perfect information), $\mathcal{A}$ needs to determine both the required amount of p_a and the strategy s_a of distributing $\mathcal{A}$'s (amplified) voting power $\zeta_a p_a$ across $\mathcal{A}$'s candidates C_a. Recall that both ζ_r and ζ_a capture the amplification effect to p_r and p_a respectively, and solely depend on (v, t, n), which are constants in this game. We skip their explanation here as they have no influence on the game. Next, we express the smallest unit of voting power as δ. We can then express the number of strategies of distributing $\frac{\zeta_r p_r}{\delta}$ ($\frac{\zeta_a p_a}{\delta}$) pieces of δ across candidates in C_l (C_a) as a finite positive integer x_r (x_a). Consequently, in this game, $\mathcal{R}$ has a number of x_r (pure) strategies and $\mathcal{A}$ has a number of $x_a{}^{x_r}$ (pure) strategies, as shown in a game tree in Figure 8. In this game, the goal of $\mathcal{A}$ is to take over the blockchain successfully while minimizing the required voting power p_a and the goal of $\mathcal{R}$ is thus to maximize p_a. In other words, a higher value of p_a indicates that an attacker $\mathcal{A}$ needs to invest more voting power to defeat the co-resisters $\mathcal{R}$, which suggests higher attack cost and thus higher resistance to takeovers. Therefore, by denoting the payoffs of $\mathcal{R}$ and $\mathcal{A}$ as u_r and u_a respectively, we simply define $u_r = \zeta_a p_a, u_a = -\zeta_a p_a$ for a successful takeover while $u_r = \infty, u_a = -\infty$ denotes a failed one.

The equilibrium. Next, we compute the subgame-perfect Nash equilibrium of the game via Backward Induction [38], which proves that the game can rapidly reach equilibrium within a single round. The game has $x_r + 1$ subgames rooted at the non-leaf nodes, namely N_1 to N_{x_r+1} as shown in Figure 8.

Before further analysis, we introduce p^r_{n-t+1}, which represents the voting power of the $(n-t+1)^{th}$ candidate in the sorted vector $\mathbf{c^r} = (c^r_1, ..., c^r_n)$ of $\mathcal{R}$'s candidates, sorted by their voting power assigned by $\mathcal{R}$ from high to low. Then, based on the introduced p^r_{n-t+1}, we define two key strategies:

- $\widehat{s_a}$: A strategy of $\mathcal{A}$ in which $\mathcal{A}$ evenly distributes $\zeta_a p_a$ across t candidates in C_a. For any s_r, in the subgame induced by s_r, $\zeta_a p_a$ is set to $t p^r_{n-t+1}$ as per the proof of Lemma 1 for that subgame.
- $\widehat{s_r}$: A strategy of $\mathcal{R}$ in which $\mathcal{R}$ evenly distributes $\zeta_r p_r$ across $(n-t+1)$ candidates in C_l.

Intuitively, $\widehat{s_a}$ suggests that an attacker $\mathcal{A}$ should always choose to invest all the voting power p_a just to a minimum number of candidates required by t (i.e., MinApprov) and make these t candidates equally strong so that none of them may be easily defeated by the co-resisters $\mathcal{R}$ as a breakthrough. Similarly, $\widehat{s_r}$ suggests that the co-resisters $\mathcal{R}$ should always choose to invest all the voting power p_r just to a number of $(n-t+1)$ candidates and make these candidates equally strong so that an attacker $\mathcal{A}$, after easily controlling $t-1$ seats where no defensive power exists, feels difficult to defeat any of the $\mathcal{R}$'s candidates to control the last seat required by t.

Next, we prove Theorem 1 through two lemmas.

THEOREM 1. $(\widehat{s_a}, \widehat{s_r})$ is the subgame-perfect Nash equilibrium.

LEMMA 1. $\widehat{s_a}$ is the unique best response in any subgame rooted among nodes N_2 to N_{x_r+1}.

PROOF. In any of these subgames, $\mathcal{A}$ is the sole player in the one-shot game and is given the sorted vector $\mathbf{c^r}$ of $\mathcal{R}$'s candidates. The best response for $\mathcal{A}$ to maximize its payoff u_a is to set $\zeta_a p_a = t p^r_{n-t+1}$ and assign an amount of p^r_{n-t+1} voting power to exactly t (i.e., MinApprov) of $\mathcal{A}$'s candidates[8], where p^r_{n-t+1} stands for the voting power of c^r_{n-t+1}, the $(n-t+1)^{th}$ element in the vector $\mathbf{c^r}$. To illustrate this point, if $\mathcal{A}$ removes an amount of δ voting power from any of the t candidates, the corresponding candidate will be defeated by c^r_{n-t+1}, resulting in $u_a = -\infty$. In contrast, if $\mathcal{A}$ assigns an additional amount of δ voting power to any of $\mathcal{A}$'s candidates, u_a will be decreased by δ. In either case, u_a would become smaller than the payoff of $\mathcal{A}$ by taking the strategy $\widehat{s_a}$. □

LEMMA 2. $\widehat{s_r}$ is the best response of $\mathcal{R}$ to $\widehat{s_a}$.

PROOF. Given $\widehat{s_a}$, to maximize u_r, $\mathcal{R}$ needs to assign an amount of $\frac{\zeta_r p_r}{n-t+1}$ voting power to exactly $(n-t+1)$ candidates in C_l, which means that $p^r_{n-t+1} = \frac{\zeta_r p_r}{n-t+1}$. To prove it, if $\mathcal{R}$ moves an amount of δ voting power from any of the $(n-t+1)$ candidates to another candidate in C_l, p^r_{n-t+1} would be decreased by δ, which decreases u_r by $t\delta$. □

The amplification effect. Let us now discuss ζ_a and ζ_r, the two power amplification coefficients. Intuitively, depending on whether a voting system allows voters to weight multiple votes using the same coins, an approval voting system tends to amplify voters' power by v (i.e., MaxVote) while a cumulative voting system does not. In a cumulative voting system, for both $\mathcal{A}$ and $\mathcal{R}$, every single unit of voting power can only be assigned to a single candidate and therefore, we can simply set both ζ_a and ζ_r as 1. In an approval voting system, for both $\mathcal{A}$ and $\mathcal{R}$, every single unit of voting power can be assigned to up to v distinct candidates and therefore, both ζ_a and ζ_r are upper-bounded by v. However, the strategy $\widehat{s_a}$ suggests $\mathcal{A}$ to pick exactly t candidates, which actually bounds ζ_a by t. Similarly, $\widehat{s_r}$ bounds ζ_r by $n-t+1$. To sum up, we have:

$$\zeta_a = \begin{cases} min\{v, t\} & (approval\ voting) \\ 1 & (cumulative\ voting) \end{cases} \quad (1)$$

$$\zeta_r = \begin{cases} min\{v, n-t+1\} & (approval\ voting) \\ 1 & (cumulative\ voting) \end{cases} \quad (2)$$

The quantification. Based on the above discussion, we can now quantify the *active takeover resistance*, R_A introduced in Definition 5, as the value of p_a in the equilibrium. Next, in Lemma 3, we quantify R_A by combining both the two strategies $\widehat{s_a}$ and $\widehat{s_r}$ in the equilibrium. Furthermore, we demonstrate the upper bound of R_A.

LEMMA 3. On the equilibrium path induced by $\widehat{s_r}$ and $\widehat{s_a}$ together, the active takeover resistance $R_A = \frac{\zeta_r t p_r}{\zeta_a(n-t+1)}$, which is upper-bounded by $\frac{t p_r}{n-t+1}$ for a supermajority governance system where $\frac{2}{3}n < t < n$.

PROOF. Based on Definition 5, Lemma 1 and Lemma 2, we have $\zeta_a R_A = \zeta_a p_a = t p^r_{n-t+1} = t\frac{\zeta_r p_r}{n-t+1}$, so $R_A = \frac{\zeta_r t p_r}{\zeta_a(n-t+1)}$. Next, based on Equation 1 and Equation 2, given the cumulative voting rule, we have $\zeta_a = \zeta_r = 1$, which makes $R_A = \frac{t p_r}{n-t+1}$. However, given the approval voting rule, R_A is a piecewise function and is maximized

---

[8]We assume that $\mathcal{A}$ wins in a tie vote.

Chain	R_A (current)	R_A (upper)
EOSIO	p_r	$2.14p_r$
Steem	p_r	$4.25p_r$
TRON	p_r	$2.11p_r$

Table 2: The current active resistance R_A (left column) and the theoretical upper bound of R_A by setting $v = n - t + 1$.

when $v \leq n - t + 1$ and $\frac{2}{3}n < t < n$, which makes $\zeta_a = \zeta_r = v$ and $R_A = \frac{tp_r}{n-t+1}$. □

Finally, based on the proof for Lemma 3, we can easily prove Lemma 4.

LEMMA 4. *Given a pair of parameters (t, n) such that $\frac{2}{3}n < t < n$, by setting the MaxVote parameter $v \leq n - t + 1$, the active takeover resistance R_A can achieve the upper bound, regardless of whether approval voting or cumulative voting is employed.*

EOS, Steem and TRON. We now revisit[9] the resistance R_A deduced from the parameters of EOS, Steem and TRON shown in Table 1. The results shown in Table 2 demonstrate that the current resistance of DPoS blockchains is far below the theoretical upper bound.

While we now have the answer to our first research question, which is to maximize active resistance by setting $v \leq n - t + 1$, we discuss a more complex scenario, namely community-to-community takeover, in Appendix B. Next, we answer our second research question via an empirical analysis.

6 PASSIVE RESISTANCE: AN EMPIRICAL ANALYSIS

In this section, we answer the second research question by performing the first large-scale empirical study of the passive takeover resistance of EOSIO, Steem and TRON. We first describe the collected dataset and investigate the actual voter preferences, including the number of votes cast and the priorities assigned to chosen candidates. Then, based on the observed voter preferences, we simulate the distribution of voting power under diverse voting system design choices and quantitatively evaluate the passive takeover resistance across different design choices using two metrics.

6.1 Voter Preferences

We collected and parsed real data from the EOSIO, Steem, and TRON blockchains. Based on this dataset, we measure and analyze voters' number of votes cast and voting priorities.

Dataset. We collect the Steem blockchain data and TRON blockchain data using their official APIs [26, 27] and obtain the EOSIO blockchain data from the dataset released by a recent work [57]. The basic information and statistics of our dataset are shown in Table 3. Based on this dataset, we construct per day power snapshots and also per day voting snapshots for all three blockchains. Specifically, a power snapshot refers to a collection of *<voter, voting power>* pairs by the end of a certain day, where a voter's voting power consists of her own voting power and voting power delegated to her, if any. Similarly, a voting snapshot refers to a collection of *<voter, candidates>* pairs by the end of a certain day. Based on these snapshots, we are capable of capturing daily changes in the blockchains to perform fine-grained empirical analysis. Our empirical study presented in

---

	EOSIO	Steem	TRON
Start date	2016-03-24	2018-06-18	2018-06-25
End date	2020-07-31	2020-07-31	2020-07-31
End block	134,193,882	45,568,376	21,980,572
Voters	56,119	67,605	115,508
Candidates	596	890	268

Table 3: Basic information and statistics of the dataset.

this section focuses on a period of two years from July 2018 to July 2020 so that we can compare the three blockchains after both EOS and TRON have been created in June 2018.

No. of votes. We present the results of daily changes in the size of different voter categories, based on the number of votes cast (ranging from 1 to 30) as a stacked line chart in Figure 9. Surprisingly, even though voters in all three blockchains can cast up to 30 votes, many choose to cast only a few or, in some cases, a single vote. It may be easier to understand the phenomenon in TRON because TRON adopts the cumulative voting rule so that voters can not amplify their power by casting more votes. Nevertheless, we find that nearly half of EOSIO voters choose to cast fewer than 5 votes, and more than half of Steem voters consistently cast fewer than 3 votes. There are many possible reasons that can drive voters to cast a few votes. For instance, a voter may be recommended only a few candidates by a friend or an online article, may find it tedious to repeatedly click the vote buttons, and may belong to or be compromised or bribed by a single candidate. The phenomenon may be desirable from perspectives such as diversity, but clearly not desirable from a perspective of protecting DPoS blockchains against takeovers. The fact that voters do not fully utilize the amplification effect potentially makes takeovers easier for attackers who understand and exploit the rule. More concretely, the value of $(n - t + 1)$ is 7 and 4 in current EOSIO and Steem, respectively. However, over half of the voters in both blockchains cast fewer votes than the two equilibrium-suggested thresholds, implying that most voters may not consider takeover risks in practice.

Voting priority. We have seen that voters often cast fewer votes than expected and it is actually quite common in practice. Intuitively, DPoS blockchains should reduce the MaxVote parameter v to minimize the gap between voters and attackers, due to their different preferences regarding the number of votes to cast. However, it is then important to estimate the priorities that voters would assign to candidates. For instance, in a voting system where $v = 2$, voters A and B are voting for two sets of candidates (C,D) and (C,E), respectively, where candidate C is their shared choice. Now, if we want to study the passive resistance of the system to takeovers in case of a smaller v and thus reduce v from 2 to 1, each voter will need to withdraw one vote and decide which one to remove. The withdrawal order matters because if both voters A and B retain candidate C, the shared choice, after their withdrawal, their voting power will still be aggregated at candidate C. This potentially makes takeovers more difficult even if the aggregation occurs unintentionally. We understand the difficulty in accurately estimating voters' behaviors due to the complexity of their motivations and the lack of ground truth data after altering system parameters. Similar to recent works on approval voting [49, 50], we propose a simple but reasonable heuristic and assume all voters follow it. Specifically, we assume that a voter would assign the lowest priority to the newly
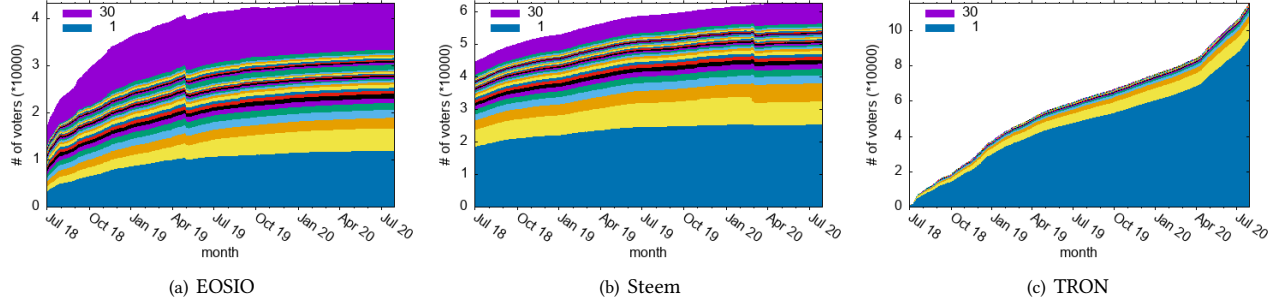
Figure 9: The daily variations in the size of different voter categories, based on the number of votes cast.
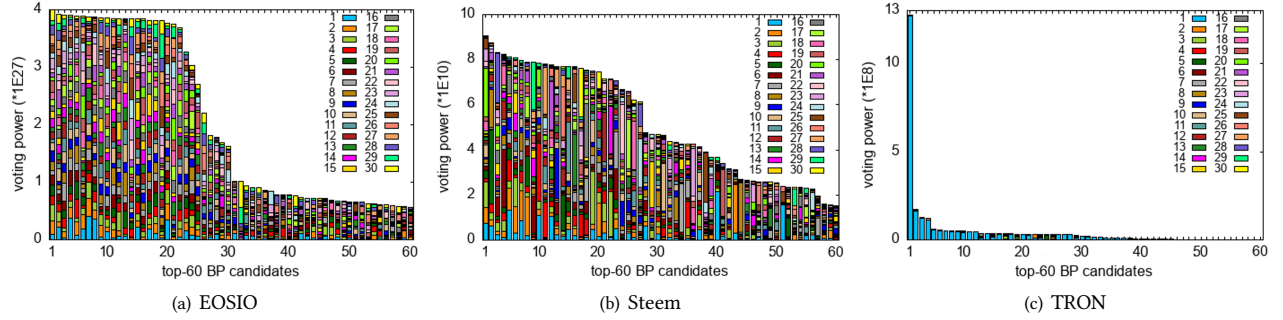

Figure 10: The snapshot of the top-60 BP candidates on Feb. 14, 2020, where each candidate's voting power is divided into up to 30 segments based on the priorities assigned to them by voters.

selected candidate while the highest priority to the candidate that the voter has voted for the longest time. In other words, we regard the candidates chosen by a voter as a vector, sorted by the duration they have remained in the vector, and assume that voters will remove the last candidate from the vector first.

In Figure 10, we present a stacked bar chart illustrating the snapshot of the top 60 BP candidates as of Feb. 14, 2020, which is half a month before TRON's takeover. Each candidate's voting power is divided into up to 30 segments, with segment i representing the voting power contributed by voters who assigned the ith priority to the candidate. The results reveal several interesting characteristics. From a macro perspective, we observe that the voting power in EOSIO is more concentrated among the top 22 candidates and declines rapidly beyond the 31st candidate. In contrast, voting power in Steem decreases more smoothly. In TRON, however, we find that the first BP receives an overwhelming amount of voting power, over 7 times that of the second BP. From a micro perspective, we note that in EOSIO, voters tended to be highly inconsistent with the priorities assigned to candidates, indicating that voters do not generally assign their top-k priorities to the same candidates. Again, the relatively even distribution of priorities may not be desirable for resisting takeovers, as it suggests that voting power may not become more concentrated when the MaxVote parameter v is reduced. In contrast, priorities assigned to the top 12 candidates in TRON are dominated by the first priority, which is not surprising given the large proportion of voters casting a single vote.

In summary, we observe that EOSIO voters exhibit the most diverse preferences from a micro perspective, whereas TRON voters demonstrate the highest consistency in their preferences.

6.2 Passive Takeover Resistance

Next, based on the actual dataset and voter preferences, we simulate the voting power distribution for EOSIO, Steem, and TRON when adopting different voting system design choices. We then quantitatively evaluate the passive takeover resistance of these blockchains under various voting system design choices, using two metrics.

Simulation. For EOSIO, Steem, and TRON, we simulate scenarios where the blockchain employs an approval voting system with a fixed pair of (t, n), as displayed in Table 1, and a MaxVote v varying from 30 to 1. Additionally, we simulate situations where the blockchain utilizes a cumulative voting system with $v = 30$, as in TRON. As previously observed in Section 6.1, the diversity in voter preferences naturally leads to a phenomenon we term *voting power decay*, which refers to the decrease in voting power capable of passively resisting takeovers as the MaxVote parameter v is reduced. However, voter behavior may exhibit a certain level of uncertainty after modifying the voting system. Consequently, we made several assumptions during the simulation process. Specifically, to simulate an approval voting system, we assumed that voters would withdraw their votes based on their priorities as v is reduced. Moreover, we assumed that a voter in TRON, upon adopting the approval voting rule, would give all her votes the weight of her full voting power. To simulate a cumulative voting system, we assumed that voters who cast multiple votes in EOSIO and Steem would evenly distribute their voting power among all the candidates they select, which is the most commonly observed heuristic in TRON.

Metrics. We propose two metrics to evaluate the passive resistance to takeover attacks. The first metric quantifies the *passive takeover resistance*, R_P, in the way described in Definition 6. Specifically, R_P
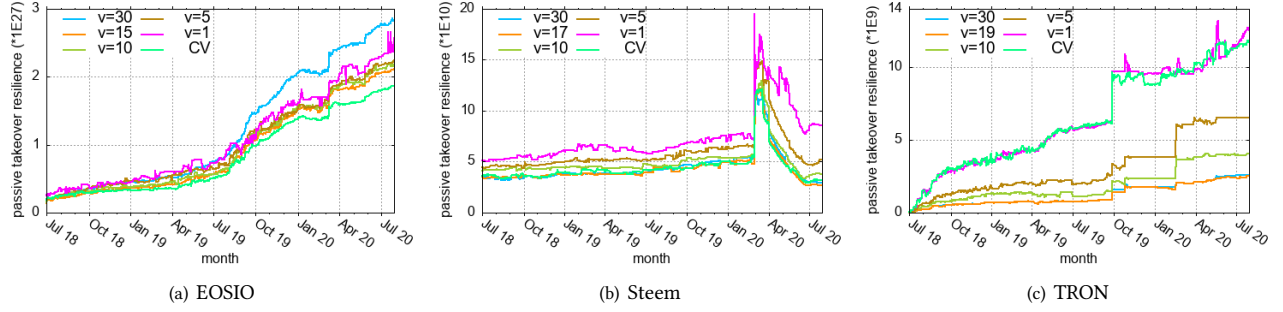
Figure 11: The daily variations of R_P, the *passive takeover resistance*.
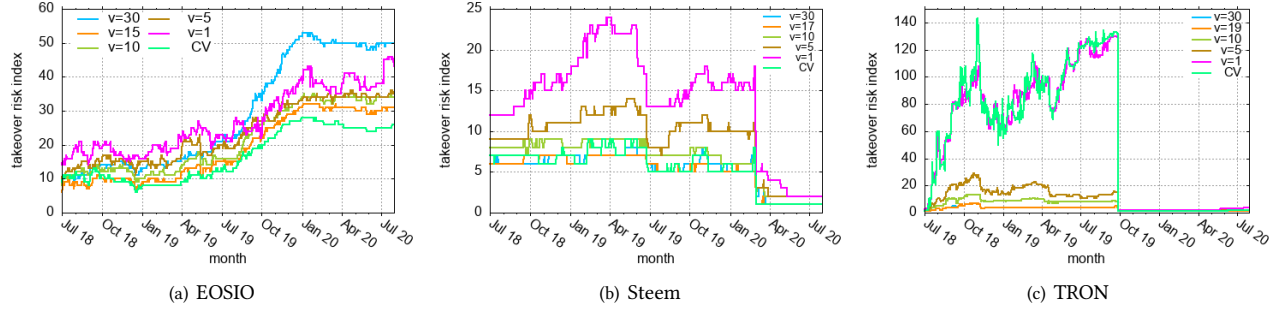

Figure 12: The daily variations of I_t, the *takeover risk index*.

corresponds to the *active takeover resistance* R_A and measures the minimum amount of voting power that an attacker needs to have to take over a blockchain. Recall that for active resistance, as illustrated in Lemma 3, we have $\zeta_a R_A = t p^r_{n-t+1}$ in equilibrium. However, for passive resistance, instead of assuming that the $(n-t+1)^{th}$ BP is controlled by the co-resisters, we disregard the BP's attitude towards takeovers and directly measure the actual voting power of the $(n-t+1)^{th}$ BP, denoted as p_{n-t+1}. Therefore, in a voting system with parameters (v, t, n), R_P can be computed as $\zeta_a R_P = t p_{n-t+1}$, namely $R_P = \frac{t p_{n-t+1}}{\zeta_a}$.

The second metric, referred to as the *takeover risk index*, is denoted as I_t. It measures the minimum number of voters whose combined voting power can successfully take over a blockchain. To compute I_t for a given day, we first sort voters based on the power snapshot of that day and obtain a vector of sorted voters. We then determine the minimum value of i as I_t such that the sum of voting power owned by the top-i voters in the vector exceeds R_P. Based on R_P and power snapshots, the *takeover risk index* captures the risks associated with top voters who, although individually incapable of taking over a blockchain, may collude to do so. A metric with a similar purpose, called the Nakamoto coefficient [13, 41], has been widely used to evaluate blockchain decentralization. It essentially measures the minimum number of resource holders (e.g., mining power) required for their combined resources to surpass a threshold (e.g., 50%). However, we could not directly employ the Nakamoto coefficient in our study because each voter in a DPoS blockchain may vote for a different set of BP candidates while only the votes received by the top $(n-t+1)$ BPs may affect the difficulty of a takeover attack. As a result, we propose the *takeover risk index*.

Evaluation. We start by evaluating the *passive takeover resistance*, $R_P = \frac{t p_{n-t+1}}{\zeta_a}$, for EOSIO, Steem, and TRON under a range of voting

system design choices. Theoretically, based on Equation 1, we can categorize these design choices into three groups:

- Group 1: For an approval voting (AV) system with $1 < v \le t$, we have $\zeta_a = v$ and thus $R_P = \frac{t}{v} p_{n-t+1} \propto \frac{p_{n-t+1}}{v}$.
- Group 2: For an approval voting (AV) system with $t < v \le 30$, we have $\zeta_a = t$ and thus $R_P = p_{n-t+1}$.
- Group 3: For a cumulative voting (CV) system, $\zeta_a = 1$, so $R_P = t p_{n-t+1} \propto p_{n-t+1}$.

For design choices in the first group, we aim to understand how resistance R_P changes when both the reduction of the $(n-t+1)^{th}$ BP's actual voting power p_{n-t+1} (i.e., voting power decay, detrimental to resistance) and the reduction of the MaxVote parameter v (i.e., weakened amplification effect, beneficial to resistance) occur. In the second group, we can anticipate that resistance R_P will increase when the MaxVote parameter v is larger. However, we want to determine if the design choice of employing a large t, say $t = 30$, is the optimal option among the three groups. Finally, we aim to compare the two types of voting systems.

The daily variations in *passive takeover resistance* R_P are depicted in Figure 11. Using the notation of (AV, i) for an approval voting system with a MaxVote $v = i$ and CV for a cumulative voting system, we can rank the design choices according to their respective resistance R_P, from best to worst, using the symbol '$\ge$' to indicate 'slightly better,' and '$\approx$' to signify 'almost the same':

$$EOSIO : \underbrace{(AV, 30)}_{\text{group 2}} > \underbrace{(AV, 1) \ge \cdots \ge (AV, t)}_{\text{group 1}} > \underbrace{CV}_{\text{group 3}}$$

$$Steem : \underbrace{(AV, 1) > \cdots > (AV, t)}_{\text{group 1}} \approx \underbrace{(AV, 30)}_{\text{group 2}} \approx \underbrace{CV}_{\text{group 3}}$$

$$TRON : \underbrace{CV}_{\text{group 3}} \approx \underbrace{(AV, 1) > \cdots > (AV, t)}_{\text{group 1}} \approx \underbrace{(AV, 30)}_{\text{group 2}}$$

The results effectively address our questions. For design choices in the first group, namely $(AV, 1)$ to (AV, t), we can see that the resistance R_P tends to be higher for a smaller v across all three blockchains. This implies that the impact of weakened amplification effect outweighs that of voting power decay. However, since EOSIO voters exhibit the most diverse preferences from a micro perspective, the voting power decay in EOSIO is the most pronounced, leading to all design choices in the first group offering relatively similar levels of resistance for EOSIO. For the second group, we note that $(AV, 30)$ is the best choice of EOSIO but the worst choice of both Steem and TRON. This suggests that an approval voting system with a large v might be more suitable for blockchains with more diverse voter preferences. In contrast, we find that CV is generally the worst choice for both EOSIO and Steem, while being the best choice for TRON. This indicates that CV may be more fitting for blockchains where the voter preferences are highly consistent.

Finally, we present the daily variations of the *takeover risk index* I_t in Figure 12. Recall that I_t is related to both R_P and the power snapshots, resulting in similar trends over time for I_t and R_P, as demonstrated by comparing Figure 12 with Figure 11. It is evident that EOSIO generally exhibits a larger I_t than Steem, due to the more skewed distribution of voting power in Steem. During the month of TRON's takeover, we note that I_t of Steem drops to 1, which highlights the capability of I_t to detect known events. More interestingly, we find that I_t of TRON reaches 1 in Oct. 2019, indicating the presence of a single voter with sufficient voting power to take over TRON, which demonstrates the effectiveness of I_t in identifying unknown takeover risks.

In summary, our findings indicate that the approval voting rule with a small MaxVote parameter v is a suitable choice for all three blockchains examined in this work. The consistency of our findings across multiple blockchains demonstrates the robustness of our conclusions and implies that our recommendations may serve as a foundation for optimizing voting system design choices in diverse DPoS blockchain environments. Furthermore, for blockchains with more diverse voter preferences, such as EOSIO, the approval voting rule with a large v may also help improve passive resistance. Conversely, for blockchains with less diverse voter preferences, like TRON, the cumulative voting rule may be a viable alternative.

7 DISCUSSION

A hybrid approach. As illustrated by our measurements of the *takeover risk index*, the passive resistance alone may be inadequate to resist takeovers as the voters may not be capable of understanding and taking advantage of the voting rule. Besides the selection of the most appropriate voting system design choices, the community of a blockchain may also arrange an amount of dedicated voting power, which is only used for actively preventing takeovers without affecting the election. Specifically, the dedicated voting power can be delegated or transferred to a smart contract or a trusted party, which will continuously rank BP candidates based on the distribution of voting power excluding the dedicated part and leveraging the dedicated voting power to vote for exactly the top $(n - t + 1)$ candidates. For instance, in Steem, by setting $v = n - t + 1 = 4$ and assigning an amount of dedicated voting power, p_r to the top-4 candidates, the overall takeover resistance would become the sum of the passive takeover resistance R_P and the upper-bounded active

Consensus protocols	Blockchains
DPoS+PoA: Proof of Staked Authority (PoSA), HPoS	Binance Coin (BNB, #4), Huobi Token (HT, #56), KuCoin Token (KCS, #57)
DPoS+BFT: Tendermint, Delegated Byzantine Fault Tolerance (dBFT)	Cosmos (ATOM, #20), OKB (OKB, #29), Terra Classic (LUNA, #44), Neo (NEO, #72), Osmosis (OSMO, #80), Kava (KAVA, #98)
Liquid Proof of Stake (LPoS)	Tezos (XTZ, #48)
Nominated Proof of Stake (NPoS)	Polkadot (DOT, #12)
XinFin DPoS (XDPoS)	XDC Network (XDC, #95)

Table 4: Recent variants of DPoS that implement coin-based voting governance and their associated Top 100 cryptocurrencies on coinmarketcap.com as of Jan. 15, 2023 [51].

takeover resistance $R_A = 4.25p_r$. We believe that a hybrid approach that combines both passive and active resistance may provide a promising solution to improve takeover resistance.

Generalization. In general, our analysis in this paper is applicable to any blockchain that employs the coin-based voting governance model introduced in Section 3, including but not limited to the ones listed in Table 4. On the one hand, these blockchains inherit the core coin-based voting governance model from DPoS, making them vulnerable to takeover attacks. On the other hand, they have made improvements upon the original DPoS [37], either by combining DPoS with other consensus protocols (e.g., PoA [9], BFT [4]) or by refining specific steps in the original DPoS (e.g., the committee size is dynamically adjustable in LPoS [1], BP candidates require nomination by others in NPoS [54]). We believe that the work presented in this paper will lay out the foundation for enhancing the takeover resistance of these blockchains and can provide valuable insights for future research on the impact of new features on takeover resistance in potential new variants of DPoS.

Limitation and future work. In Section 6.2, we adopt certain assumptions for the sake of simplicity and manageability in our simulation. These assumptions are based on empirical data and applied uniformly across all voters. However, we recognize the potential for more accuracy in future studies by diversifying these assumptions, such as classifying voters into strategic and non-strategic categories. Besides, in terms of future research directions, we have identified various recent DPoS variants that implement coin-based voting governance in Table 4. A careful evaluation of the enhancements these variants bring to the original DPoS could provide valuable insights into their alignment or divergence with the governance model we propose. This method helps to understand the direct applicability of our models and findings and potentially reveal new research problems. For instance, our insights could be directly applied to DPoS+PoA blockchains as their governance model remains unaffected. In contrast, applying our findings to LPoS blockchains, which have an adjustable committee size, might require adjustments and hence present a new research problem.

8 RELATED WORK

Decentralization on blockchains. As the most prominent PoW blockchains, Bitcoin and Ethereum's decentralization have attracted sustained interest from researchers. In 2014, Gervais et al. conducted an empirical study of Bitcoin data in [17], and their results showed

that many key processes in Bitcoin are substantially controlled by a few entities. Subsequently, Feld et al. analyzed the peer-to-peer network of Bitcoin and focused in [12] and concluded that the network is highly centralized. Miller et al. further investigated the topology of the Bitcoin network in [43] and found that a small number of top 2% nodes essentially controlled about 75% of the effective resources. Zeng et al. measured the decentralization in Ethereum at the level of mining pool participants in [55]. After that, researchers have conducted a comparative analysis of the decentralization of different PoW blockchains. In 2018, Gencer et al. compared the actual degree of decentralization of Bitcoin and Ethereum in [16]. The results showed that Bitcoin and Ether were similarly decentralized. In 2019, Kwon et al. studied the gaming of Bitcoin and Bitcoin Cash in [35]. This work modeled the mining game of these two systems and demonstrated that the Nash equilibrium of the game leads to severe centralization of the disadvantaged system. Recently, blockchains based on non-PoW consensus protocols have gained a lot of attention. Kwon et al. analyzed the decentralization in various blockchains including PoW, PoS and DPoS in [36]. Li et al. compared the decentralization between Steem and Bitcoin in [40].

Attacks on blockchains. In 2014, Eyal et al. questioned whether Bitcoin incentives can achieve incentive compatibility in [11]. Their paper proposes a selfish mining attack, in which a selfish mining pool does not disclose new blocks mined to maintain its advantage, but discloses new blocks mined when it is about to lose its advantage. Since then, Sapirshtein et al. optimized the selfish mining attack method in [48] and proposed an algorithm to define a lower bound on the resources an attacker needs to hold to benefit from selfish mining. Gervais et al. proposed a quantitative framework that helps devise optimal adversarial strategies for double-spending and selfish mining in existing PoW-based deployments and PoW blockchain variants [18], which inspired our efforts to enhance takeover resistance in DPoS blockchains and their variants.

Besides selfish mining and double-spending, blockchains are vulnerable to other types of attacks. In 2015, Eyal proposed the "miner's dilemma" theory in [10]. From a theoretical perspective, this research argued that rational mining pools have an incentive to send members to join competing pools and launch a block withholding attack. Kwon et al. proposed a novel fork-after-withholding attack in [34] and showed that the attack is very profitable and that a large pool can definitely win by launching the attack against a small pool without getting into a miner's dilemma. Gao et al. investigated two novel attack methods, power-adjusting-withholding (PAW) and bribery-selfish-mining (BSM) in [14], and showed that PAW could evade miners' dilemmas, while BSM increases attackers' gains by 10% over selfish mining. Gaži et al. further analyzed the impact of resource centrality on security thresholds in Bitcoin at a theoretical level in [15].

Recently, the security of decentralized governance has attracted a lot of attention. In [31], Jeong et al. theoretically studied the optimal number of votes per account in DPoS blockchains that employ the approval voting rule. In [19], Monday Capital and DappRadar investigated the decentralized governance of six DAOs (Decentralized Autonomous Organizations) where decisions are made through stake-weighted votes and demonstrated that these projects tended to be extremely centralized. In [13], Fritsch et al. empirically studied the distribution of voting power in three prominent DAOs and showed that the governance is dominated by a few voters. In this paper, inspired by these recent works, we have formally modeled coin-based voting governance, takeover attack/resistance and the *takeover game*. Our work demonstrates the theoretical upper bound of active resistance for blockchains that employ different voting rules, and we presented the first large-scale empirical study of the passive takeover resistance of EOSIO, Steem and TRON.

9 CONCLUSION

In this paper, we demonstrate that the resistance of a DPoS blockchain to takeovers is governed by both the theoretical design and the actual use of its underlying coin-based voting system. After modeling the coin-based voting system and formalizing the takeover attack and resistance model, we theoretically model a game between an attacker and the cooperative resisters and demonstrate that the current active takeover resistance is far below the theoretical upper bound. We then present the first large-scale empirical study of the passive takeover resistance of EOSIO, Steem and TRON. The results demonstrate the diversity of voter preferences, which significantly affects the passive takeover resistance when the parameters of the coin-based voting system change. Our study suggests potential ways to improve the takeover resistance of DPoS blockchains, including the recommended configuration settings of the system based on our theoretical and empirical analyses and a hybrid approach in which both passive and active resistance are combined to improve takeover resistance. We believe the study presented in this work provides novel insights into the security of coin-based voting governance and can potentially facilitate more future work on designing new voting rules for decentralized governance that provide more compliance with resistance to takeovers. Additionally, we suggest further investigation into a broader range of voting systems (e.g., Single Transferable Vote) could potentially uncover voting methods that improve the security of coin-based voting governance. We also recommend researching other governance models that combine coin with reputation and contribution as the weight for voting, which could potentially improve the overall security and fairness of the governance model.

ACKNOWLEDGMENTS

REFERENCE

[1] Victor Allombert, Mathias Bourgoin, and Julien Tesson. Introduction to the tezos blockchain. In *2019 International Conference on High Performance Computing & Simulation (HPCS)*, pages 1–10. IEEE, 2019.
[2] Steem based DAPPs [Internet]. Available from. https://steem.com/developers/. Accessed Apr. 2023.
[3] Sanjai Bhagat and James A Brickley. Cumulative voting: The value of minority shareholder voting rights. *The Journal of Law and Economics*, 27(2):339–365, 1984.
[4] Ethan Buchman. *Tendermint: Byzantine fault tolerance in the age of blockchains*. PhD thesis, University of Guelph, 2016.
[5] Vitalik Buterin et al. A next-generation smart contract and decentralized application platform. *white paper*, 2014.
[6] The call-to-action in Steem [Internet]. Available from. https://steemit.com/steem/@theycallmedan/call-to-action-earn-upvotes-to-vote-for-witnesses. Accessed Apr. 2023.

[7] Kyle Croman et al. On scaling decentralized blockchains. In *International Conference on Financial Cryptography and Data Security*, pages 106–125, 2016.

[8] Andrea De Salve, Andrea Lisi, Paolo Mori, and Laura Ricci. Measuring eos. io dapp resource allocation and costs through a benchmark application. In *2021 4th International Conference on Blockchain Technology and Applications*, pages 24–30, 2021.

[9] Parinya Ekparinya, Vincent Gramoli, and Guillaume Jourjon. The attack of the clones against proof-of-authority. In *Network and Distributed Systems Security (NDSS) Symposium*, 2020.

[10] Ittay Eyal. The miner's dilemma. In *2015 IEEE Symposium on Security and Privacy*, pages 89–103. IEEE, 2015.

[11] Ittay Eyal and Emin Gün Sirer. Majority is not enough: Bitcoin mining is vulnerable. In *International conference on financial cryptography and data security*, pages 436–454. Springer, 2014.

[12] Sebastian Feld, Mirco Schönfeld, and Martin Werner. Analyzing the deployment of bitcoin's p2p network under an as-level perspective. *Procedia Computer Science*, 32:1121–1126, 2014.

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

[14] Shang Gao, Zecheng Li, Zhe Peng, and Bin Xiao. Power adjusting and bribery racing: Novel mining attacks in the bitcoin system. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, pages 833–850, 2019.

[15] Peter Gaži, Aggelos Kiayias, and Alexander Russell. Tight consistency bounds for bitcoin. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, pages 819–838, 2020.

[16] Adem Efe Gencer, Soumya Basu, Ittay Eyal, Robbert Van Renesse, and Emin Gün Sirer. Decentralization in bitcoin and ethereum networks. In *International Conference on Financial Cryptography and Data Security*, pages 439–457. Springer, 2018.

[17] Arthur Gervais, Ghassan O Karame, Vedran Capkun, and Srdjan Capkun. Is bitcoin a decentralized currency? *IEEE security & privacy*, 12(3):54–60, 2014.

[18] Arthur Gervais, Ghassan O Karame, Karl Wüst, Vasileios Glykantzis, Hubert Ritzdorf, and Srdjan Capkun. On the security and performance of proof of work blockchains. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pages 3–16, 2016.

[19] Decentralized governance in DeFi: Examples and pitfalls [Internet]. Available from. https://dappradar.com/blog/decentralized-governance-in-defi-examples-and-pitfalls. Accessed Apr. 2023.

[20] Ronald L Graham, Donald E Knuth, Oren Patashnik, and Stanley Liu. Concrete mathematics: a foundation for computer science. *Computers in Physics*, 3(5):106–107, 1989.

[21] Barbara Guidi, Andrea Michienzi, and Laura Ricci. A graph-based socioeconomic analysis of steemit. *IEEE Transactions on Computational Social Systems*, 8(2):365–376, 2020.

[22] Ningyu He, Ruiyi Zhang, Haoyu Wang, Lei Wu, Xiapu Luo, Yao Guo, Ting Yu, and Xuxian Jiang. {EOSAFE}: Security analysis of {EOSIO} smart contracts. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 1271–1288, 2021.

[23] EOS Network Foundation [Internet]. Available from. https://eosnetwork.com/. Accessed Apr. 2023.

[24] Fork 0.22.2 [Internet]. Available from. https://steemit.com/steem/@softfork222/soft-fork-222. Accessed Apr. 2023.

[25] Fork 0.22.5 [Internet]. Available from. https://github.com/steemit/steem/pull/3618. Accessed Apr. 2023.

[26] Interactive Steem API [Internet]. Available from. https://developers.steem.io/. Accessed Apr. 2023.

[27] Interactive TRON API [Internet]. Available from. https://developers.tron.network/docs/trongrid. Accessed Apr. 2023.

[28] Steemit [Internet]. Available from. https://steemit.com/. Accessed Apr. 2023.

[29] Steemit Inc. [Internet]. Available from. https://www.steem.center/index.php?title=Steemit,_Inc. Accessed Apr. 2023.

[30] TRON Super Representative [Internet]. Available from. https://tronprotocol.github.io/documentation-en/mechanism-algorithm/sr/. Accessed Apr. 2023.

[31] Seungwon Eugene Jeong. Centralized decentralization: Does voting matter? simple economics of the dpos blockchain governance. *Simple Economics of the DPoS Blockchain Governance (April 21, 2020)*, 2020.

[32] Ghassan O Karame, Elli Androulaki, and Srdjan Capkun. Double-spending fast payments in bitcoin. In *Proceedings of the 2012 ACM conference on Computer and communications security*, pages 906–917, 2012.

[33] Christine Kim. Ethereum 2.0: How it works and why it matters. *Coindesk: https://www. coindesk. com/wp-content/uploads/2020/07/ETH-2.0-072120. pdf*, 2020.

[34] Yujin Kwon, Dohyun Kim, Yunmok Son, Eugene Vasserman, and Yongdae Kim. Be selfish and avoid dilemmas: Fork after withholding (faw) attacks on bitcoin. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 195–209, 2017.

[35] Yujin Kwon, Hyoungshick Kim, Jinwoo Shin, and Yongdae Kim. Bitcoin vs. bitcoin cash: Coexistence or downfall of bitcoin cash? In *2019 IEEE Symposium on Security and Privacy (SP)*, pages 935–951. IEEE, 2019.

[36] Yujin Kwon, Jian Liu, Minjeong Kim, Dawn Song, and Yongdae Kim. Impossibility of full decentralization in permissionless blockchains. In *Proceedings of the 1st ACM Conference on Advances in Financial Technologies*, pages 110–123, 2019.

[37] Daniel Larimer. Delegated proof-of-stake (dpos). *Bitshare whitepaper*, 2014.

[38] Kevin Leyton-Brown and Yoav Shoham. Essentials of game theory: A concise multidisciplinary introduction. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 2(1):1–88, 2008.

[39] Chao Li and Balaji Palanisamy. Incentivized blockchain-based social media platforms: A case study of steemit. In *Proceedings of the 10th ACM conference on web science*, pages 145–154, 2019.

[40] Chao Li and Balaji Palanisamy. Comparison of decentralization in dpos and pow blockchains. In *International Conference on Blockchain*, pages 18–32. Springer, 2020.

[41] Qinwei Lin, Chao Li, Xifeng Zhao, and Xianhai Chen. Measuring decentralization in bitcoin and ethereum using multiple metrics and granularities. In *2021 IEEE 37th International Conference on Data Engineering Workshops (ICDEW)*, pages 80–87. IEEE, 2021.

[42] Jieli Liu, Weilin Zheng, Dingyuan Lu, Jiajing Wu, and Zibin Zheng. From decentralization to oligopoly: A data-driven analysis of decentralization evolution and voting behaviors on eosio. *IEEE Transactions on Computational Social Systems*, 2022.

[43] Andrew Miller, James Litton, Andrew Pachulski, Neal Gupta, Dave Levin, Neil Spring, and Bobby Bhattacharjee. Discovering bitcoin's public topology and influential nodes. *et al*, 2015.

[44] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. 2008.

[45] Hoang H Nguyen, Dmytro Bozhkov, Zahra Ahmadi, Nhat-Minh Nguyen, and Thanh-Nam Doan. Sochaindb: A database for storing and retrieving blockchain-powered social network data. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 3036–3045, 2022.

[46] ENF Statement on EOS Network Actions Taken on December 8th [Internet]. Available from. https://eosnetwork.com/blog/enf-statement-on-eos-network-actions-taken-on-december-8th. Accessed Apr. 2023.

[47] First ECAF ruling [Internet]. Available from. https://eosauthority.com/approvals/view?scope=libertyblock&name=chngkeyha4ta&lnc=z&network=eos. Accessed Apr. 2023.

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

[49] Jaelle Scheuerman, Jason Harman, Nicholas Mattei, and K Brent Venable. Modeling voters in multi-winner approval voting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 5709–5716, 2021.

[50] Jaelle Scheuerman, Jason L Harman, Nicholas Mattei, and K Brent Venable. Heuristics in multi-winner approval voting. *arXiv preprint arXiv:1905.12104*, 2019.

[51] Historical snapshot on coinmarketcap.com [Internet]. Available from. https://coinmarketcap.com/historical/20230115/. Accessed Apr. 2023.

[52] TRON white paper [Internet]. Available from. https://tron.network/static/doc/white_paper_v_2_0.pdf/. Accessed Apr. 2023.

[53] Gavin Wood. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum project yellow paper*, 151:1–32, 2014.

[54] Gavin Wood. Polkadot: Vision for a heterogeneous multi-chain framework. *White paper*, 21(2327):4662, 2016.

[55] Liyi Zeng, Yang Chen, Shuo Chen, Xian Zhang, Zhongxin Guo, Wei Xu, and Thomas Moscibroda. Characterizing ethereum's mining power decentralization at a deeper level. In *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*, pages 1–10. IEEE, 2021.

[56] Yuzhe Zhang and Davide Grossi. Power in liquid democracy. In *Proceedings of the AAAI conference on Artificial Intelligence*, volume 35, pages 5822–5830, 2021.

[57] Weilin Zheng, Zibin Zheng, Hong-Ning Dai, Xu Chen, and Peilin Zheng. Xblock-eos: Extracting and exploring blockchain data from eosio. *Information Processing & Management*, 58(3):102477, 2021.

#	post	comments
1	call-to-action-earn-upvotes-to-vote-for-witnesses	449
2	steemit-witness-voting-policy	385
3	an-open-letter-to-the-community-hf22-5	370
4	my-resignation-from-steemit	182
5	an-update	139

Table 5: The top 5 posts with the most comments after the takeover.
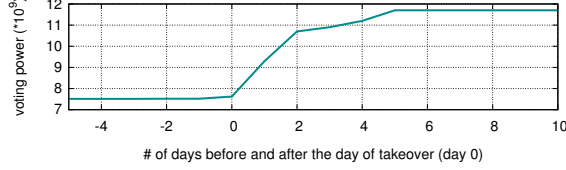

Figure 13: The increment of the author's voting weight.

A THE CALL-TO-ACTION

On the day of the takeover, a prominent Steem community member posted a call-to-action [6]. As illustrated in Table 5, the call-to-action (#1) attracted a remarkable number of comments (449 comments) within ten days, surpassing the two open letters about the takeover posted by TRON (#2, #3) and becoming the most-discussed post during this period.

The call-to-action presented two recommendations:

> Either PROXY ME (i.e., the author).
> Or VOTE HERE: https://steemitwallet.com/~witnesses
> Vote for 22-42 at a minimum, we need to vote for the same witnesses to maximize our votes! Use all 30 of your votes!

In other words, it advised voters to either utilize liquid democracy to delegate voting power to the author of the call-to-action or employ approval voting to cast votes for the 21 BP candidates belonging to the Steem community who were ranked 22-42 at the time.

As shown in Figure 4 of Section 4.2, the call-to-action yielded exceptional results, and the recommended BP candidates quickly received substantial voting power support. Further investigation revealed that many voters also followed the first suggestion of the call-to-action. As depicted in Figure 13, they delegated their voting power to the author of the call-to-action, thereby increasing the author's voting weight to 156% of that prior to the takeover.

B COMMUNITY-TO-COMMUNITY TAKEOVER

In the analysis of Section 5, we have made an implicit assumption that $\mathcal{A}$ (or $\mathcal{R}$) is always capable of evenly distributing ζp_a (or ζp_r) across t (or $n - t + 1$) candidates. For instance, in a system that $t = 7$ and $v = 5$, $\mathcal{A}$ (or $\mathcal{R}$), who owns $\zeta_a p_a = 5 \times 70\delta = 350\delta$, needs to assign an amount of 70δ to 5 candidates. To do that, they need to create 7 accounts as voters, transfer an amount of 10δ voting power to each voter and use each voter to vote for 5 different candidates. This process thus becomes highly complex, which may be difficult to implement in practice. For example, if Steem's call-to-actions included the aforementioned complex steps, it may be difficult to attract co-resisters who are willing or able to follow them.

In community-to-community takeovers, the two players, $\mathcal{A}$ and $\mathcal{R}$, might represent two different communities within the same blockchain (e.g., two distinct national communities) or from different blockchains (e.g., members of blockchain A exchanging tokens

Chain	R_A (current)	R_A (upper)
EOSIO	p_r	$3p_r$
Steem	p_r	$5p_r$
TRON	p_r	$3p_r$

Table 6: The current active resistance R_A and the theoretical upper bound of R_A in community-to-community takeovers.

from blockchain B to attack blockchain B). In these scenarios, the simplification of the attack and resistance processes, namely call-to-actions, becomes crucial. Therefore, we assume that both $\mathcal{A}$ and $\mathcal{R}$ are communities and employ a minimum number of *simple* call-to-actions, denoted as z, such that $vz \geq t$ for $\mathcal{A}$ and $vz \geq (n - t + 1)$ for $\mathcal{R}$. We can consider each *simple* call-to-action as a pool with an upper limit, which simply accumulates voting power from voters. Once the limit is reached, the pool casts its voting power toward v candidates, and is then replaced by a new pool created by another *simple* call-to-action, which votes for another v candidates. In this way, $\mathcal{A}$ (or $\mathcal{R}$) could leverage a minimum number of *simple* call-to-actions to vote for each candidate at least once but at most twice. Specifically, in an approval voting system, $\mathcal{A}$ would employ only $z = \lceil \frac{t}{v} \rceil$ *simple* call-to-actions and assign an amount of $\frac{p_a}{\lceil \frac{t}{v} \rceil}$ voting power to each candidate. Similarly, $\mathcal{R}$ would employ $z = \lceil \frac{n-t+1}{v} \rceil$ *simple* call-to-actions and assign an amount of $\frac{p_r}{\lceil \frac{n-t+1}{v} \rceil}$ voting power to each candidate. This gives the following theorem:

THEOREM 2. *In an approval-voting supermajority-governing system, if both $\mathcal{A}$ and $\mathcal{R}$ are communities and employ a minimum number of simple call-to-actions, the resistance R_A would be upper-bounded by $\lceil \frac{t}{n-t+1} \rceil p_r$.*

PROOF. Based on two properties of ceiling functions [20], P1 ($x_1 \leq x_2 \Rightarrow \lceil x_1 \rceil \leq \lceil x_2 \rceil$) and P2 ($\lceil mx \rceil = \lceil x \rceil + \lceil x - \frac{1}{m} \rceil + \cdots + \lceil x - \frac{m-1}{m} \rceil$ for positive integer m), we have:

$$R_A = \frac{\lceil \frac{t}{v} \rceil}{\lceil \frac{n-t+1}{v} \rceil} p_r = \frac{\lceil \frac{t}{n-t+1} \cdot \frac{n-t+1}{v} \rceil}{\lceil \frac{n-t+1}{v} \rceil} p_r$$

$$\leq \frac{\lceil \lceil \frac{t}{n-t+1} \rceil \cdot \frac{n-t+1}{v} \rceil}{\lceil \frac{n-t+1}{v} \rceil} p_r \text{ (based on P1)}$$

$$\leq \frac{\lceil \frac{t}{n-t+1} \rceil \cdot \lceil \frac{n-t+1}{v} \rceil}{\lceil \frac{n-t+1}{v} \rceil} p_r \text{ (based on P2)}$$

$$= \lceil \frac{t}{n-t+1} \rceil p_r$$

□

Then, based on Lemma 4 and Theorem 2, we can easily prove Lemma 5 by injecting $v = n - t + 1$ into $R_A = \frac{\lceil \frac{t}{v} \rceil}{\lceil \frac{n-t+1}{v} \rceil} p_r$.

LEMMA 5. *Given a pair of parameters (t, n), by setting the MaxVote parameter $v = n - t + 1$, the active takeover resistance R_A can reach the upper bound whether or not the players are communities that employ a minimum number of simple call-to-actions.*

Finally, Table 6 illustrates the theoretical upper bound of R_A when players are assumed to be communities employing a minimal number of *simple* call-to-actions, by setting $v = n - t + 1$. The results indicate that R_A achieves even higher values compared to those presented in Table 2.