POLITECNICO DI TORINO

Master's Degree in Engineering and Management (LM-31)

Master's Degree Thesis

The impact of Decentralized Autonomous Organizations on Corporate Governance

Supervisor

Prof. Riccardo CALCAGNO

Candidate

Francesco CAMPISI

March 2022

Abstract

Blockchain is a technology that can provide smart solutions to the classical corporate governance inefficiencies, particularly in the relationship between the shareholders and the company. The advancements of the blockchain technology in fact gave birth to a wide variety of use cases, among which, a novel form of organizations called Decentralized Autonomous Organizations (DAOs), is driving a paradigm shift in the organizational design, by introducting governance mechanisms that are independent by a central authority and by enabling permissionless participation. On the other side, actual governance theories do not encompass the alienation of trust between parties and the union of ownership and control that DAOs are enabling. The aim of this study is thus to explores how the use of the blockchain can shape the decision-making process inside an organization and providing, through a game theoretical approach, a model to describe the shareholders' behavior in a DAO. The model focus on the relationship between voting costs, bias and the concentration of the ownership in determining the approval or rejection of proposals inside a DAO.

Table of Contents

List of Tables

List of Figures

Chapter 1

Fundamentals of blockchain

1.1 The blockchain technology

This section provides a brief description of the key elements of blockchain technology and the main advantages and potential issues that characterize the use of digital artifacts and distributed ledgers with respect to the traditional centralized protocols. [1]

A blockchain is a specific form of the broad family of Distributed Ledger Technology (DLT), which encompasses all the types of databases that are shared across several locations or among multiple participants, with the aims of eliminating the need for a central authority or an intermediary to process, validate or authenticate transactions. In contrast with centralized protocols, in which a central authority grants the validity and originality of the information contained therein, thus constituting a vulnerable Single Point of Failure (SPOF), within a distributed ledger, there is a network of participants which validates transactions, and each of them maintain its own version of the ledger. Therefore, a distributed ledger will be less likely to go down because of interruptions of the activity by some participants.
Decentralization of the ledger also implies that its validation should not be controlled or manipulated by a single participant or a small number of colluding participants, thus it requires to associate all participants to the validation of transactions, for example, through majority voting. In a distributed ledger in fact, whenever a transaction occurs, the information is sent by the trading parties to

the network, so that the network participants, after having verified the integrity of the information transmitted, can add it to their own ledger.
On the other side, since the validation of transactions depends on the network participants and generally, they use efforts and resources for doing this, it is necessary, for maintaining the sustainability of the network, to provide incentives to the network contributors. [2]
Moreover, using a distributed ledger to contain information about the ownership of a digital asset, such in the case of cryptocurrencies, is another critical point. In fact, using digital artifacts, which are, by-design, easy to replicate, give rise to the risk of double spending, that happens when the same digital currency is spent more than once by creating a new amount of the copied currency, leading to inflation, and undermining the reliability and trust on the system. While such problem is easy to be prevented in a centralized system, with an online central trusted third party that can verify whether a token has been spent, in a decentralized system, the problem became much more difficult to solve. In fact, considering that, servers must store identical up-to-date copies of a public transaction ledger, to avoid the need for a trusted third party and that, when transactions are broadcasted, they arrive at each server at slightly different times, if two transactions attempt to spend the same token, each server will consider the first transaction it receives as valid, and the other invalid. Once the servers disagree, there is no way to determine true balances, as each server's observations are considered equally valid, unless a consensus mechanism is used. [3]
Specifically, blockchain is a mechanism that employs cryptography and uses a set of specific mathematical algorithms to create and verify a continuously growing data structure to which data can only be added and from which existing data cannot be removed, that takes the form of a chain of "transaction blocks", which functions as a distributed ledger. [1]
Despite the first blockchain-like protocol was firstly introduced by the cryptographer David Chaum in 1982 [4], his modern conceptualization is commonly granted to the anonymous researcher (or group of researchers) who worked under the pseudonym of Satoshi Nakamoto and published the Bitcoin Whitepaper in 2008 [5]. Satoshi Nakamoto is considered the first to conceive a decentralized blockchain which, based on a peer-to-peer network, is capable of timestamping transactions

without requiring the need of a trusted party who sign them, thus avoiding the intermediation of financial institutions. Bitcoin is a decentralized digital currency, structured as a peer-to-peer network architecture on the top of Internet. The term peer-to-peer, or P2P, means there is no hierarchy among the computers that participate in the network, there are no "special" nodes, and that they all share the efforts for providing network services. "*Nodes in a peer-to-peer network both provide and consume services at the same time with reciprocity acting as the incentive for participation*". [6]
The key innovation of Bitcoin was to combine timestamping i.e., a sequence of encoded information that identifies when a certain event has occurred, with a consensus mechanism called Proof of Work, introduced by Back in 2002. [7]The technology produced an immutable ledger that eliminated a crucial problem of digital assets i.e., making perfect copies and spending them multiple times. Blockchains allowed for the key features desirable in a store of value, which never before were simultaneously present in a single asset. In fact, blockchains allow for cryptographic scarcity (Bitcoin has a fixed supply cap of 21 million), censorship resistance and user sovereignty (no entity other than the user can determine how to use funds), and portability (can send any quantity anywhere for a low flat fee). These features combined in a single technology made cryptocurrency a powerful innovation.

1.2 State of arts in the blockchain industry

This section provides an overview of the current trends in the blockchain industry, the player involved, and the actions undertaken by governments around the world.

The last decade has seen the blockchain industry constantly evolving, with practitioners of many fields exploring and researching application scenarios in various sectors such as finance, supply chain, healthcare, energy, manufacturing, and smart city. At the same time, many countries and regions, such the USA and Canada have actively encouraged blockchain-based innovations through financing research and regulations, while El Salvador has recently become the first country to adopt the cryptocurrency Bitcoin as legal tender. [8, 9]. In Europe, the European Union

has committed to providing reliable data security and privacy for international business by establishing a unified Blockchain service infrastructure. From a company perspective, IBM, Oracle, and SAP are the three largest players in the business application domain of blockchain technology.
In the financial services industry, blockchain is driving the change from payments and deposit to lending, investing, trading and insurance and, with an always growing interest of both major institutions and individuals, funds continue to flow into the digital asset market. [10]
However, blockchains have some technical shortcomings that still need to be resolved. First of all, it has limited scalability. Indeed, while the Visa network is capable of processing at least 40,000 transactions per second, Ethereum and the Bitcoin blockchain are capable of processing approximately 15 and 7 transactions per second, respectively. Second, the adoption of a universal browser-based API still needs greater collaboration among all browser makers including Microsoft, Google, Facebook, Apple and Mozilla. On the other hand, blockchain has shown some maturity for supply chain management and the technologies have been implemented in some industries. In the marketplace, many enterprise platforms including Ethereum, Hyperledger, R3 Corda, Ripple, Quorum, and others are readily implemented, while the Blockchain Interoperability Alliance was formed in 2017 for facilitating interoperability. It is reasonable to say then, that blockchain seemingly will not reach the maturity phase soon, but its technology surely is not in its infancy any longer and has reached a point that actual enterprise implementation is made possible by collaborative efforts. [11]

1.3 Blockchain architecture

In this section, the building blocks of a blockchain are analysed. For the sake of simplicity, the focus will be on the blockchain underlying the digital currency network Bitcoin, despite since its inception, a wide type of blockchains has been developed, using different protocols and having different purposes. The analysis includes the description of the two different paradigms around which blockchains are built and of the main elements required to run a blockchain i.e., the nodes (participant) who share the efforts for securing the network, the structure of the

blocks that contain the information and the methods generally used to ensure the integrity of the information i.e., the consensus algorithm and the use of asymmetric cryptography.

1.3.1 Permissionless vs. Permissioned blockchain

First of all, it is necessary to distinguish among the two main paradigms that characterize the blockchain technology: the permissionless paradigm, which found wide application on the cryptocurrencies industry, and it is characterized by an open and transparent architecture, and the permissioned paradigm, which is instead private and generally less decentralized.
Permissionless blockchains, also known as trust-less or public blockchains, are open networks available to everyone to participate in the consensus process that blockchains use to validate transactions and data. Participants can join or leave the network at will, without having to be approved by any central entity and it is possible to join the network and add transactions to the ledger by only having a computer on which the relevant software has been installed. It is also characterized by the absence of central owner of the network and software, and identical copies of the ledger are distributed to all the nodes in the network. This kind of blockchain is also more exposed to attacks such as Sybil attack, where the attacker creates a large number of pseudonymized identities that are then used to gain a disproportionately large influence in the consensus process, undermining the reputation of the system. Permissionless blockchains tend to be used in applications with a strong financial component or that require highly decentralized blockchains, such as digital asset trading, crowdfunding and distributed file storage, such as blockchain storage. [1]
In contrast, a permissioned blockchain, also called private blockchain or permissioned sandbox, is a closed network in which validators (the nodes) must be pre-selected by a network administrator, who sets the rules for the ledger. This allows to easily verify the identity of the network participants but, on the other side, decrease the transparency of the ledgers, that could be controlled based on the organization's purposes, and increase the risk of collusion among members, especially in the case of few participants. Permissioned blockchains have enabled

new applications that depend on privacy and security, including supply chain tracking, claims settlement and identity verification (Know Your Customer). Figure 1.1 provides a resume of the main characteristics of the two paradigms.

	Permissionless	**Permissioned**
Overview	Open network available for anyone to interact and participate in consensus validation. Fully decentralized across unknown parties.	Closed network. Designated parties interact and participate in consensus validation. Partially decentralized (i.e., distributed across known parties)
Also known as	Public, trustless	Private, permissionless sandbox
Key attributes	▪ Full transparency of transactions, based on open-source protocols. ▪ Development via open source ▪ Mostly anonymous, with some exceptions ▪ Privacy depends on technological limitations or innovations ▪ No central authority ▪ Often involves digital assets or token for incentives	▪ Controlled transparency, based on organizations' goals ▪ Development via private entities ▪ Not anonymous ▪ Privacy depends on governance decisions ▪ No single authority, but a private group authorizes decisions ▪ May or may not involve digital assets or tokens
Benefits	▪ **Broader decentralization**, extending access across more network participants ▪ **High transparency** ▪ **Censorship resistant**, due to accessibility and participation across locations and nationalities ▪ **Security resilience**, since attackers cannot target a single repository, and it is costly and difficult to corrupt 51% of the network.	▪ **Incremental decentralization**, but participations from multiple businesses helps mitigate risks of highly centralized models ▪ **Stronger information privacy** because transaction information is only available based on permissions ▪ **Highly customizable** to specific use cases through diverse configurations, modular components and hybrid integrations ▪ **Faster and more scalable**, since fewer nodes manage transaction verification and consensus
Pitfalls	▪ **Less energy efficient** because network wide transaction verification is resource-intensive ▪ **Slower and difficult to scale**, as high volume can strain network wide transaction verifications ▪ **Less user privacy** and information control	▪ **Limited decentralization** because a network with fewer participants increases risk of corruption or collusion ▪ **Risk of override**, since owners and operators can control or change the rule of consensus, immutability, or mining ▪ **Less transparent to outside oversight**, since participants are limited, and operations determine privacy requirements
Market traction	▪ Peer-to-peer ▪ Business-to-consumer ▪ Government-to-citizens	▪ Business-to-Business ▪ Business-to-consumer ▪ Government-to-organizations

Figure 1.1: Main characteristics of permissionless and permissioned blockchain.

1.3.2 Nodes

A blockchain node is a storage device that carries out the functions of validating and propagating transactions and blocks and discovering and maintaining connections to peers. Some nodes, called full nodes, maintain a complete and updated copy of the blockchain and can autonomously and authoritatively verify any transaction without external reference. Other nodes, called lightweight nodes, use a method called Simplified Payment Verification (SPV) which, rather than downloading the entire blockchain, use only the block headers, to verify the authenticity of the transaction. [6]

1.3.3 Block structure

As stated previously, the blockchain is essentially a sequence of blocks, container data structures that aggregates transactions, or more in general, information, for their inclusion in the public ledger. The first block is called genesis block and it is the one upon which all the additional blocks are connected through the chain. The chain, which is a hash representation of the transactions made up till the latest block, ensures the integrity of transactions, thus that all the transactions made in the past are not manipulated.
In the case of Bitcoin, the structure of a block consists of a block header, and a block body, composed of transactions and a counter. The block header consists of three sets of block metadata. In the first set, there is a reference to a previous block hash, which connects the current block to the previous block. The second set of metadata, namely the difficulty, timestamp, and nonce, are related to the mining competition while the third piece of metadata is the Merkle Tree Root, a data structure used to efficiently summarize all the transactions in the block. [6] The description of the metadata contained in a block header of the Bitcoin blockchain is reported in Table 1.1. Normally, the new block verified is added to the latest block of the chain, using the Longest Chain Rule (LCR), according to which nodes accept as the valid version of the blockchain the longest one. However, it may happen, intentionally (ex. as a result of collusion among a group of nodes) or accidentally (ex. when two nodes find the solution of a new block at nearly the same time and broadcast two different versions of the ledger) that alternative

Field	Size	Description
Version	4 bytes	A version number to track software/protocol upgrades. It indicates the set of rules that should be followed for block validation.
Previous Block Hash	32 bytes	A reference to the hash of the previous (parent) block in the chain.
Merkle Root	32 bytes	hash of the root of the Merkle tree of this block's transactions.
Timestamp	4 bytes	The approximate creation time of this block (seconds from Unix Epoch).
Difficulty Target	4 bytes	The proof-of-work algorithm difficulty target for this block.
Nonce	4 bytes	A counter used for the proof-of-work algorithm.

Table 1.1: Description of the metadata in a Bitcoin's block header.

chains emerge, causing a fork. In this case there could be more competing branches, each registering a potentially different version of the ledger. Such forks could make the ledger less stable, reliable, and useful, as they could create uncertainty about the distribution of property rights. [2]

For accidental forks, Satoshi Nakamoto suggested that the problem would be solved if nodes always chain their block using the LCR criteria: "*Nodes always consider the longest chain to be the correct one and will keep working on extending it. If two nodes broadcast different versions of the next block simultaneously, some nodes may receive one or the other first. In that case, they work on the first one they received, but save the other branch in case it becomes longer. The tie will be broken when the next proof-of-work is found, and one branch becomes longer; the nodes that were working on the other branch will then switch to the longer one.*" [5]

For the intentional forks, it is possible to distinguish between hard forks, and soft forks. Hard forks are backward-incompatible software updates. Typically, these occur when nodes add new rules in a way that conflicts with the rules of old nodes. Thus, the new nodes can communicate only with other nodes that operate the new version and no longer with the nodes that decide to keep the old software. As a result, the blockchain splits, creating two separate networks: one following the old

rules, and one with the new rules. A soft fork is instead a backward-compatible upgrade, meaning that the upgraded nodes can still communicate with the non-upgraded ones.
Various example in the history of Bitcoin, such as the accidental fork occurred in March 2013 [2] shows the important role that the coordination of nodes (or the lack thereof) and their expectations plays in the emergence and resolution of forks. Figure 1.2 shows a simplistic representation of the blockchain structure.

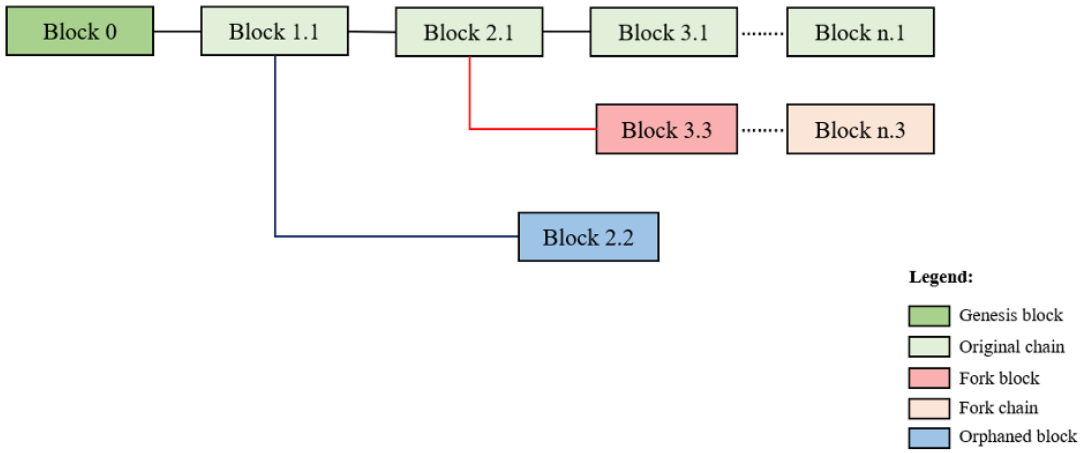

Figure 1.2: Representation of a blockchain structure and forks.

1.3.4 Asymmetric cryptography

Blockchain networks generally use an asymmetric cryptography mechanism to validate the authentication of transactions. Each user owns a set of digital keys: a public key, which is publicly known and essential for identification, and a private key, which is kept secret and it is used for authentication and encryption. The digital keys are not actually stored in the network but are instead created and stored by users in a file, or simple database, called wallet. An example of how the asymmetric cryptography works is provided in Figure 1.3. In such system, the sender can sign a transaction by using its own private key, that grants the validity of the transaction (for a cryptocurrency it grants the ownership of the funds in the sender's address) and the intended receiver's public key, which represents the address of the user. Trough the presentation of the public key and signature,

everyone in the network can verify and accept the transaction as valid, confirming that the person transferring the currency owned the amount at the time of the transfer.

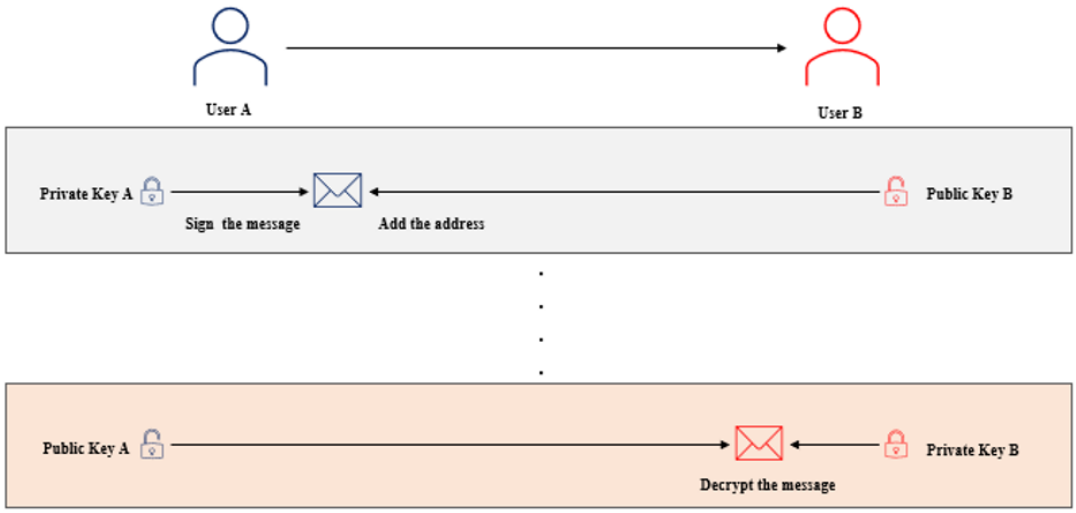

Figure 1.3: Asymmetric cryptography flow.

1.3.5 Consensus algorithm

In the applications of blockchain it is required to solve two main problems: the double spending and the Byzantine Generals Problem. [12] Blockchain solves the double spending problem, as stated in the previous section, by assigning the verification of the transactions to many distributed nodes together. Byzantine Generals Problem instead is a game theory problem of distributed system which describes the difficulty that decentralized parties have in reaching the consensus, as they have no reliable source of information and no way of verifying the information they receive from other members of the network. The data transmitted between different nodes through peer-to-peer communications may originates from nodes maliciously attacked, leading to changes of communication contents. Thus, normal nodes need to distinguish the information that has been tampered and obtain consistent results with the other normal nodes. This requires the design of a corresponding consensus algorithm (also called consensus mechanism), that can

prevent the two problems described above. A consensus mechanism is a predefined specific cryptographic validation method that ensures the correct sequencing of transactions on the blockchain by allowing the reach of a common agreement about the present state of the distributed ledger and incentivizing participants to behave properly. In this way, consensus algorithms achieve reliability in the blockchain network and establish trust between unknown peers in a distributed computing environment. Essentially, the consensus protocol makes sure that every new block that is added to the blockchain is the one and only version of the truth that is agreed upon by all the nodes.
Hereinafter, the main examples of consensus mechanisms will be briefly discussed: the Proof of Work ("PoW") mechanism, the Proof of Concepts mechanisms (PoC) and the Proof of Stake ("PoS") mechanism.

1.4 Proof of Work (PoW)

The earliest blockchain networks were developed based on Proof-of-Work (PoW) mechanism. PoW is, for example, the consensus algorithm used in the currency network Bitcoin. Generally, the nodes in a PoW-based blockchain network reach consensus by participating in solving a computational problem with adjustable difficulty (referred to as "moderately-hard puzzle" in computer science). This puzzle-solving process is commonly referred to as "mining" and requires high energy consumption. The problem to solve has nothing to do with the economic transactions included in a block and it requires computing power to perform independent trials (similarly to draws under replacement) until one node finds a solution to a numerical problem. In particular, the solution should be such that given as input the nonce, the previous block hash and the transactions contained in the block, the output of the hash function must be in a target range so that the block can be accepted. The flows according to which nodes draw the solution is represented in Figure 1.4. Due to the property of the hash function, the nonce can only be found by continuously trying different nonce values until the output is within the target range. As nodes randomly draw candidate solutions, one of them eventually gets lucky and solves the problem before the others, so that he can create the next block and get a certain amount of reward. When a participant finds

the nonce, it will broadcast the block along with the transactions to other nodes. Then, if the new block is verified and determined to be the first block mined after the last block in the chain, it will be integrated into the current chain and become the latest block in the chain. In PoW, the participants compete to be the firsts to find the correct nonce, and the ones with a higher hash rate (computational power) have higher chances to be the block winner and receive the reward. The probability p_i that the participant i is selected as the block winner in a network of N participants is expressed by the Equation 1.1, where c_i is the hash rate of participant i.

$$p_i = \frac{c_i}{\sum_{j=1}^{N} c_j} \tag{1.1}$$

This computation leads to the large amount of energy consumption for blockchains using PoW consensus mechanisms, as the participants try to increase their hash rates to have a higher chance to be the leader and receive rewards. Since participants with low hash rates have very low chances to win a block and receive rewards, they often join mining pools i.e., groups of miners who share their computing resources, to have more opportunities to get revenues. In this way, mining tasks will be distributed to the miners, and due to huge computing resources, mining pools often get much higher opportunities to win a new block than individuals. While joining a mining pool provides more stable incomes, the nodes in the pool often do not contribute to the transaction validation and propagation since they only perform the nonce search process in a specific range. Thus, mining pools have been dominating processes making new blocks in most of current blockchain networks. At the time of writing, the top five mining pools control up to 73% total hash rate of the Bitcoin network [13]. This is a key issue of PoW-based blockchain networks because it might compromise the decentralized spirit of the blockchain technology. Another issue of PoW protocols is due to the possible delay in the transaction confirmation caused by the possibility of two participants that find the solution to a block simultaneously. For this reason, a block in PoW is finalized only when it is a certain number of block k (usually six) deep in the chain. Moreover, PoW mechanism is also vulnerable to 51% attack, that happen when a single party control more than 51% of the network's total computational power, thus gaining the power of adding conflicting blocks to the chain. While 51% attacks might not

be a serious problem for large blockchain networks, the newly established networks with small and limited total computational power are especially vulnerable. [2, 12, 14]

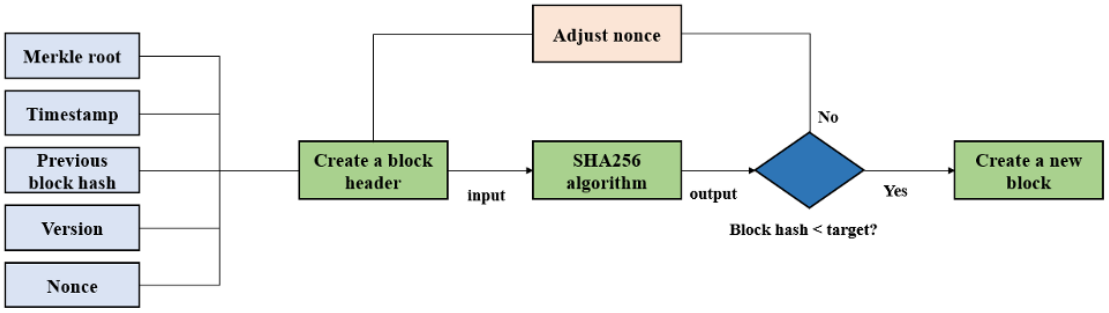

Figure 1.4: Flow of PoW.

1.5 Proof of Concept (PoC)

Proof-of-Concepts (PoC) consensus mechanisms were developed with the aims of replacing the PoW solution searching with useful calculations and to improve the performance of PoW in terms of security, incentives, and resource usage.
In [14], Nguyein et al. published a detailed list of several PoC consensum mechanisms. Among these, Proof-of-Exercise, Proof-of-Useful-Work and the blockchain Primecoin were developed to make a better use of the computational resources by substituting the nonce research with other mathematical problems.
Additional PoC consensus mechanisms were designed for distributed data storage service such as Permacoin, KopperCoin, and Filecoin. Generally, these consensus mechanisms divide the data files into segments and distribute them to multiple participants in the network and to participate in the mining process the nodes have to provide proofs of storage. Similarly to the PoW, the more storage volume a node offers, the better chances it owns to be selected as a leader.
Further PoC consensus mechanisms have been developed with the aim of solving the problem of mining pool formation. This issue was addressed by designing non outsourceable puzzles that replace the PoW solution searching process, structured in such a way that the node who find the solution can steal the reward, thus disincentivizing the formation of mining pools.

Moreover, other consensus mechanisms have been developed to reduce the computational requirement of PoW. As example, the Spacemint network employs a Proof-of-Space protocol, in which the consensus nodes must provide proof of storage when participating in the solution searching process or the Proof-of-Human-Work protocol, in which the Completely Automated Public Turing-Test to tell Computers and Humans Apart (CAPTCHA) is employed to involve human activities and reduce computational requirements in the solution searching process.

1.6 Proof of Stake (PoS)

Proof-of-Stake (PoS) protocols were developed as energy-saving alternatives to PoW. Instead of computational power resources, winner nodes are selected based on their stakes, namely the number of digital tokens that they hold or deposit. The flows according to which PoS work is represented in Figure 1.5. The winner node will then perform the mining process and will add the new block to the chain, thus avoiding the consumption of energy for the searching process required in the PoW. To simulate the stake-based leader selection process, the Follow-the-Satoshi (FTS) algorithm has been adopted in many PoS-based blockchain networks. The FTS algorithm is a hash function that takes a seed (a string of arbitrary length such as the previous block's header or a random string created by some other selected nodes) as the input and give as output a token index. Using the index, the algorithm searches the transaction history to find and select the current owner of that token to be the leader. Therefore, the probability p_i that the node i is selected to be the winner node in a network of N participants can be expressed by Equation 1.2, where s_i is the stake of participant i.

$$p_i = \frac{s_i}{\sum_{j=1}^{N} s_j} \tag{1.2}$$

Consequently, the more stake a node holds, the higher chance it has to be selected as the leader. Besides the advantage related to the low energy consumption, the PoS mechanisms have also a faster transaction confirmation speed than with respect to the PoW mechanisms. In fact, the confirmation of a transaction in a blockchain network, depends on two main factors, the transaction throughput and the block confirmation time. The transaction throughput is the number of transactions

per second $T_{x/s}$ a network can process, which is crucial for the performance of the network, especially when there are many pending transactions. $T_{x/s}$ can be calculated according to Equation 1.3.

$$T_{x|s} = \frac{Block_{size}}{Tx_{size} * Block_{time}} \tag{1.3}$$

For example, for the Bitcoin network, which has $\text{Block}_{\text{size}} = 1$ MB, $\text{Tx}_{\text{size}} = 250$ bytes, and $\text{Block}_{\text{time}} = 600$s, the number of transactions that can be processed per second is around 7. The $T_{x/s}$ determines how quickly a transaction is added to the chain, whereas the block confirmation time dictates how fast the transaction is confirmed after it is added. The block confirmation time depends on $\text{Block}_{\text{time}}$, i.e., the average time it takes for a new block to be added to the chain, and the finality of the consensus mechanisms. In the Bitcoin network, a transaction usually has to wait for $k = 6$ blocks before it can be confirmed, so the average confirmation time is $k \times \text{Block}_{\text{time}} = 3600\text{s} = 1\text{hr}$. Typically in PoS networks, the block size is larger, and some of them can achieve immediate finality ($k = 1$) so their transaction confirmation time is significantly shorter, leading to an higher transaction throughput. However, some PoS protocols, similarly to PoW, adopt the LCR, which ensures that when there are multiple versions of the chain (forks), the honest participants will only adopt the longest fork. As a result, the finality in these protocols is delayed. The security of PoS protocols depends on various factors such as the network synchrony and the incentive mechanisms. Network synchrony is crucial to the security of many PoS protocols because the leader selection processes are simulated by voting rounds, where the voters send their votes to other participants and the network cannot guarantee that all the messages are properly sent in practice due to network delay and connection complexity. Some PoS protocols are proven to be secure as long as the network is partially synchronous, where messages sent will reach their destinations within a certain time limit, or asynchronous, where messages may not reach their destinations. On the other side, the reward scheme must incentivize consensus participation by rewarding block creators and validators, but it also has to penalize malicious behaviors and prevent attacks that specifically target PoS. Moreover, since in the PoS networks, the probability that an individual stakeholder with a small stake amount is selected to be the leader is low and that, participating in the consensus

process, a involves operational costs, small stakeholders often pool their stakes together to increase their opportunities to win blocks and share operational costs, which results in the formation of stake pools. Similar to the mining pools in PoW networks, a stake pool is considered to be a single node, and thus it poses a threat of centralizing the PoS networks. [14]

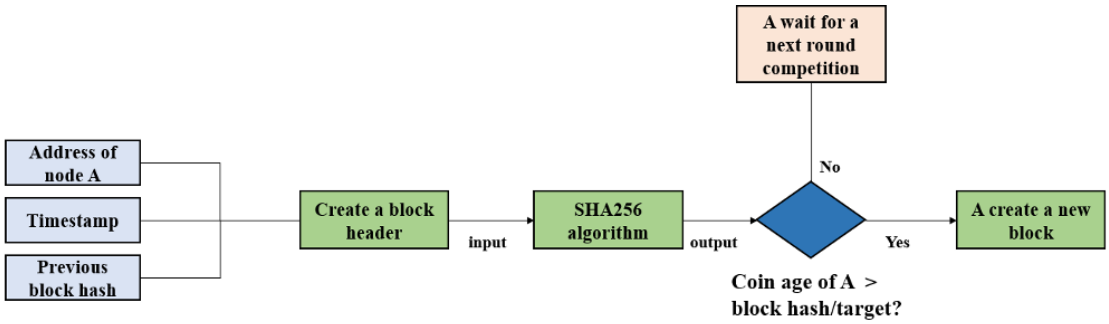

Figure 1.5: Flow of PoS.

1.7 Smart Contracts

The concept of smart contract was firstly proposed by Nick Szabo in 1994 [15], but it found wide exploitation with the creation of the blockchain Ethereum [16] in 2015. Smart contracts are computer programs or transaction protocol which automatically execute contracts or agreements according to the terms settled between the interested parties and are written in the form of program codes that exist across a distributed, decentralized blockchain network. Smart contacts have a wide variety of use-case, that range from the real estate industry, where they can be used, for example, to transfer the ownership of a real estate once a certain number of resources are transferred to the seller's wallet, to the music industry, where smart contacts may resolve ownership disputes on artistic work, by crediting the royalties to the artist whenever his work is used by other parties. The whole life cycle of smart contracts can be resumed into four consecutive phases, as illustrated in Figure 1.6.

1. Creation of smart contracts. Firstly, the involved parties, after multiple rounds of discussions and negotiations, reach an agreement on the obligations,

rights and prohibitions on the contract that has to be created. Lawyers and counsellors will help parties to draft an initial contractual agreement while software engineers will then convert the agreement into a smart contract written in computer languages including declarative languages and logic-based rule languages. Similar to the development of computer software, the procedure of the smart contract conversion is an iterative process composed of design, implementation and validation.

2. Deployment of smart contracts. The validated smart contracts can then be deployed to platforms on top of blockchains. Since the contracts stored on the blockchains cannot be modified due to the immutability of block-chains, any emendation will require the creation of a new contract.

3. Execution of smart contracts. After the deployment of smart contracts, once the contractual conditions reach, the functions will be automatically executed, following the logic underlying the smart contract. When a condition is triggered, the corresponding statement will be automatically executed, consequently a transaction will be executed and validated by miners in the blockchains.

4. Completion of smart contracts. Once the smart contract has been executed, new states of all involved parties are updated and stored in the blockchain, while the digital assets, after having been transferred from one party to another, are unlocked.

From a comparison with the conventional contracts, the main advantages emerging from the use of a smart contracts can be resumed in:

- Reduction of the risk of alteration, due to the immutability of blockchains. Since all the transactions are stored and duplicated throughout the whole distributed blockchain system are traceable and auditable, malicious behaviours like financial frauds can be greatly mitigated.

- Reduction of administration and service costs. Smart contracts stored in blockchains can be automatically triggered in a decentralized way, saving the administration and services costs due to the intervention from the third party.

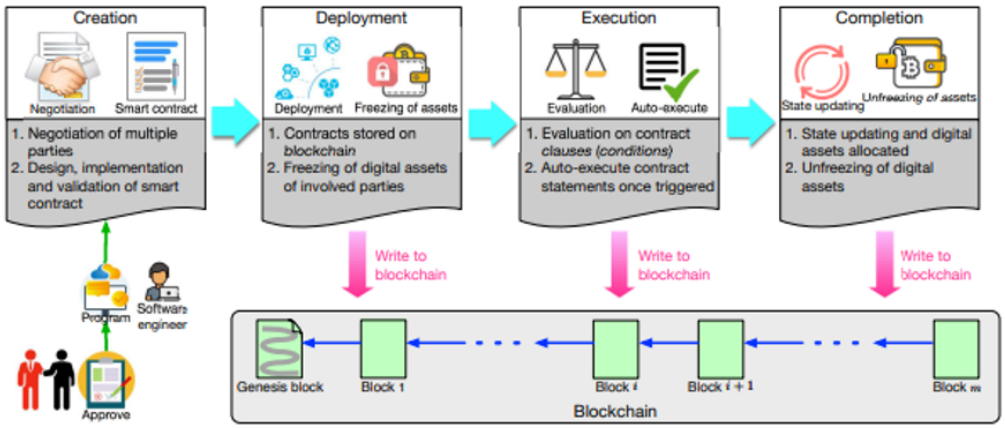

Figure 1.6: Smart Contract Lifecycle.

- Improvement of the efficiency in business processes, due the elimination of the dependence on the intermediary. [17]

On the other side, smart contracts cannot process by their self "off-chain" information and events i.e., that do not directly happen on the blockchain, thus the trusted environment provided by the blockchain technology can be tampered by the information acquired from external channels.
From a corporate strategic perspective, the willingness of a company to enter into a smart contract may be limited as it represents a pre-commitment not to behave opportunistically in the future, due to its intrinsic irreversible nature.
Another issue of using smart contracts concern the data protection of the parties, since the storage of the information by all network participants may conflict with legal requirements such as the right to be forgotten.
Moreover, being software, smart contracts are prone to bug, and the exploitation of these can cause severe consequences for the contracting parties, such as in the case of TheDAO, where a vulnerability in the smart contract's code allowed a hacker to steal a third of the funds of the investors.[18]

Chapter 2

Decentralized Autonomous Organizations

2.1 Taxonomy of DAOs

A Decentralized Autonomous Organization (DAO) is a blockchain-based system that enables people to coordinate and self-govern themselves mediated by a set of self-executing rules deployed on a public blockchain, and whose governance is independent from a central control.
Participation or affiliation within a DAO often is evidenced through a blockchain-based "token" that is coupled with the smart contracts that govern the organization. Individuals can either purchase tokens or receive them as a reward for some other contribution, such as computing power. Through smart contracts, tokens can be associated with specific rights for their holders, such as the right to receive a portion of the organization's income or the right to use the network, software, or other service offered by the organization. DAO tokens are also increasingly designed to provide their holders with the right to govern underlying software through a vote. By extending the logic underlying the working principle of a DAO, it may be constructed in a manner that allows to function without human managerial interactivity, so long as the underlying smart contracts are made robust and substantiated by Turing-complete mechanisms. [19]
DAOs are organizations since they mediate the interactions of a group of people, which typically takes the form of an open community. In fact, as stated above,

generally the membership of a DAO is granted by holding tokens that enables DAO participation, similar to corporation shares. DAOs are considered autonomous because their operations follow the rules embedded in its code, together with the human governance of its members. Autonomy can be thus described as the degree to which smart contracts govern activities by determining a priori binding, formalized, transparent and software-encoded set of rules. DAOs are decentralized because they rely on a server-less decentralized infrastructure (a public blockchain) and on certain decentralized governance mechanisms, so that the decision-making process depends on the collective agreement of its members. This process typically is translated on some form of voting, in which the DAO members participate. The decisions may refer to the allocation of resources (funding projects or payments to members), but also to changes in the DAO code. Thus, upon the agreement of its members, a DAO may be updated to operate differently, with a new set of encoded rules. However, updates may be critical because they often require intense bug-fixing but are necessary to enables DAOs to adapt to community needs and demands.

Moreover, being deployed on a public blockchain, DAOs are censorship-resistant, since there is no central controller that may turn off the DAO and its provided service. Thus, as long as there are members willing to execute their code, DAOs will continue operating by providing services, purchasing/selling resources or hiring people.

A DAO is fundamentally communitarian in orientation and the individuals are typically bounded by a charter or bylaws encoded on the blockchain, subject to amendments when approved by a majority, or some other portion, of the validator nodes. The vast majority of Blockchain networks and smart contract-based apps are organized as DAOs, although some are governed less formally than others.

Because of its characteristics, structuring an organization in the form of a DAO may not be effective for every kind of organizations. However, its field of possible application remains quite wide, ranging from charity communities and no-profit organization to venture capitalists and investment funds.

In order to systematically cluster and classify the DAOs it is possible to use the taxonomy proposed by [20] that is essentially built on four main elements: the scope of multilateral agreements, resource management, discussion process and voting

process. A resume of the main features of the structure of a DAO is represented in Figure 2.1 Design choices in these elements are likely to have repercussions on several other aspects of DAOs that will be briefly mentioned but are not considered crucially important.

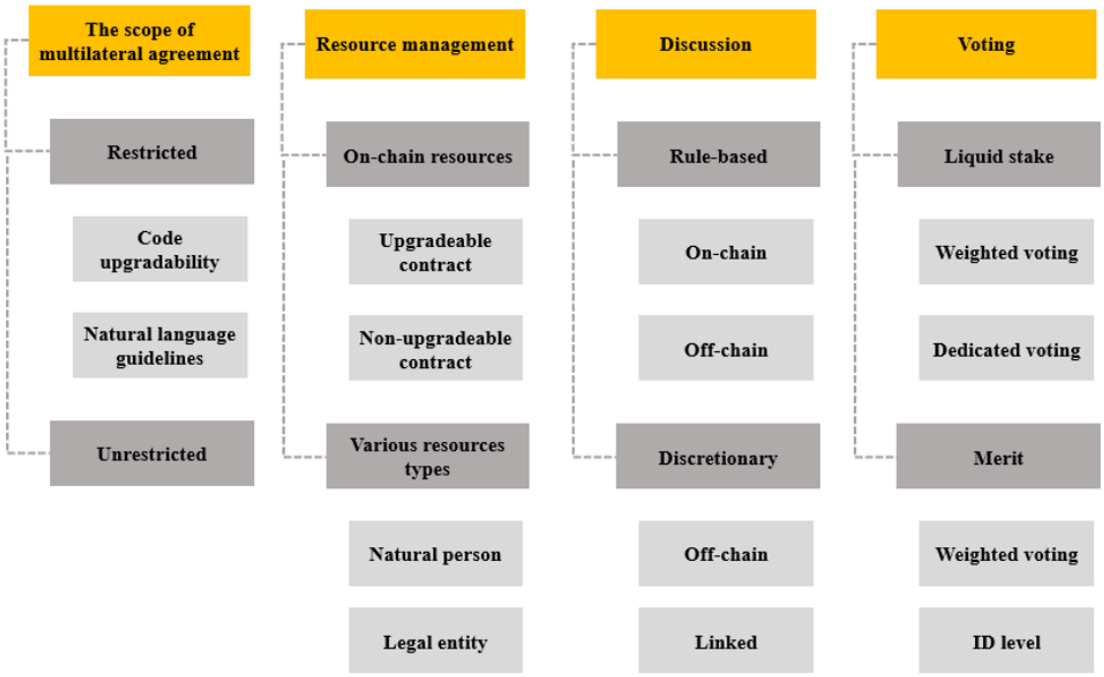

Figure 2.1: Elements of DAO's architecture.

2.1.1 The scope of multilateral agreements

The "scope of multilateral agreement" determines the degree to which all the decisions are agreed upon jointly i.e., whether the scope of multilateral agreements is restricted or unrestricted.

In case of restricted multilateral agreement, exemptions can either be incorporated within natural language or software. This constitutes a relevant distinction, since the choice of the medium defines how changes to the content thereof are enforced. Software upgrades are typically stringent, whereas natural language texts lack hard enforcement and act rather as soft guidelines. On other side, changes to natural language texts, such as a constitution defining specific core values, could be deemed to be less critical than software upgrades. Moreover, natural language texts may

still be required to express a nuanced idea or account for and link to existing legal contracts. Software upgrades, on the other hand, usually relate to core operational elements, such as the deployed smart contracts, or proposed changes to a native blockchain protocol.

2.1.2 Resource management

The resource management is a key aspect of the architecture of a DAO since it profoundly impacts the degree of autonomy that the organization can reach. The exclusive usage of on-chain resources allows a DAO to directly exert control and initiate action via a smart contract while avoiding the counterparty risk. However, a trade-off between the level of multilateral agreement and practicability i.e., the possibility to upgrade the contract, should be ensured. On the other hand, using off-chain assets, such as traditional currencies, implies that a natural person or legal entity must be assigned to the control over the respective assets. This constitutes a legally identifiable point of contact that has legal control over off-chain assets and thus introduces counterparty risk, since it would be possible for her to abuse the legal authority over the managed assets and disregard decisions of the DAO.

2.1.3 Discussions

Discussions over a certain proposal can be accomplished through different mechanisms. Rule-based discussions are generally executed on-chain by pre-voting and are based on a smart contract that is used for the implementations of token curated registries or prediction markets. As these implementations are based on smart contracts, they all are binding, formalized, and transparent. Alternatively, a rule-based discussion could also be executed off-chain by linking a smart contract event to a legal contract. If the discussion is not rule-based, it must be discretionary and, by definition, based on an off-chain medium. These could either be legal contracts that are not initiated or otherwise directly linked to smart contracts of the DAO, or various fora or messenger services.
Linked discretionary discussions are usually involving an identity-based off-chain discussion that will be then referred to in an on-chain transaction, once a rough consensus has been reached. By doing this, an informal and open discussion can

be integrated into a rules-based system. Most DAO frameworks have made use of, mostly pseudonymous, fora to enable a natural language discussion of members.

2.1.4 Voting structures

The voting structure of DAOs is based on tokenized voting rights and it can assume different features whether voting is based on a liquid stake or merit.
The concept of liquid stake describes a voting process that is executed with a transferable token that is usually locked for a certain amount of time. The implementation can follow a "dedicated voting" structure, further defining whether the "stakeable" token must be native i.e., issued by the DAO to be a bearer right to vote, or foreign i.e., merely representing a financial commitment. It is essential to note that for native tokens, liquidity can have a significant impact on the ability of a user to enter or exit a DAO and that in the case of low/null demand or supply in primary or secondary markets, the system is inherently flawed. Furthermore, votes can be also weighted in order to make the system more robust and avoid malicious behaviour. Among the possible alternatives to the simplest set-up where one token represents one vote, Vitalik's Quadratic Voting [21], according to which the costs for acquiring voting rights increase quadratically, or other functions where n token represent m votes can be used to disincentivize centralization of the voting power. On the other side, the voting structures that are based on merit instead of a stake still involve the use of tokens. As the name suggests, such tokens would usually be issued based on the merit and activity of an individual. However, in this case tokens would not be transferable freely, but merely display the salience of any owner. The concept behind a merit-based voting structure is thus to have tokens that cannot be moved for any purpose besides voting and are restricted to internal usage. Similarly, to stake based tokens, non-transferable merit tokens could still be weighted.

2.1.5 Other factors

The respective design choices in the above elements are likely to influence numerous other areas, that will be briefly mentioned below.

- Compounding smart contract risk. The intrinsic risk associated to a smart

contract increase as the set-up became more complex, requiring multiple interaction among different contracts. Nonetheless, formal verification of smart contract code or insurances might be able to remedy or manage this risk.

- Added value functionalities. They express to what extent the chosen structure allows a subsequent enhancement with elements such as delegated or secret voting.
- Social cohesion. Among the many factors that contribute to enhance the participation within a DAO, the difficulties encountered in the procedures and the possibility to easily exit, play a key role.
- Dispute resolution capacities. They are likely to be influenced by design choices for the discussion and voting processes.
- Degree of autonomy. It is most likely additionally influenced by the broader environment of the DAO. As example, whether the DAO is operating a proprietary or "foreign" blockchain or what the entity of the costs for conducting a censorship attack would be, can deeply impact on the degree of autonomy a DAO can achieve.

2.2 Decision-Making process in DAO: a problem of scalability and resilience

Typically, electronic voting systems focus on providing a secure infrastructure with encrypted and verifiable votes, which is trustworthy for voters. Under a blockchain, those technical issues are generally reduced, due to its nature as a distributed ledger without central management, and with its transactions validated cryptographically. However, while designing the decision-making process of a DAO, there are two crucial aspects that are necessary to be addressed: the scalability i.e., the possibility to scale direct democratic participation to large group of people, and the resilience i.e., the tolerance and resistance of a governance system to faulty behaviours. [22] Scalability can be intended both as the possibility to scale the number of participants and the possibility to scale the number of decisions that

should be taken. In fact, as a community grows, the number of voters required to to satisfy the majority increases together with the number of decisions that demands attention from voters in a given time frame. Thus, scaling an organization, either in terms of members or decisions, hinders the governance of the organization. On the other side, reducing the number of voters required to take decisions, for example by requiring only a certain percentage of the total member to vote, compromise the organization resilience since it will increase the possibility of malicious behaviours by small minorities. Such duality between scalability and resilience is addressed in different ways, since specific solutions are developed depending on the number of members, the number of proposals that have to be addressed and the scope of the organizations. Some of the solutions developed, although still largely in an experimental phase, will be now examined in detail.

2.2.1 Token-based Quorum voting

Quorum voting is a governance mechanism widely tested and it has a long history in political systems. It requires a certain threshold of voters to be reached for a proposal to pass (as example: 60% quorum means that 60% of voting power needs to vote) and once the threshold has been met, the decision which has more votes wins. On the other side, if the quorum threshold is not reached, the proposals will fail, independently from the percentage of positive votes. The threshold is typically based on the total number of votes, although some protocols have a quorum threshold only for the votes in favour of a proposal passing instead (a 20% quorum in this case would mean 20% of voting power voting positively for a proposal in order for a proposal to be considered for passing). Quorum voting is also generally characterized by a simple UX for users. On the flipside, pure token-based voting can lead a DAO to a plutocracy, in which the richer users determine the volunteer of the community and requires a high amount of participation from members to pass proposals, which is expensive and time consuming. Moreover, from a tactical perspective, quorum-based voting leads people against a proposal to abstain since they are more likely to prevent a proposal from passing with respect to voting no. Another issue related to quorum voting is the selection of the “right” quorum requirement since a low threshold make the proposals very easy to pass and the system easy to attack while high quorum make them very difficult to pass

proposals, making the system ungovernable. Lastly, token-based voting is sensitive to certain kinds of attacks, like the flash governance attack that happened to Maker [23, 24].

2.2.2 Holographic Consensus

The mechanism of Holographic Consensus tries to solve the issues of attention related to large-scale online communities i.e., that after reaching certain scale, participants cannot review and vote all submitted proposals. Thus, the idea behind Holographic consensus is to establish some sort of proposal filtering, so that participants give their attention to the proposals which are most aligned to the community ends.

In order to understand how Holographic works, it is necessary to introduce the concept of absolute majority and relative majority. Absolute majority is reached when more than half of the total voting power agree (or disagree) with a given proposal, representing thus the global opinion of the DAO, while relative majority constitutes only an approximation of the DAO's opinion since the deliberation or rejection of proposals is based on involving only a sufficient number of the total agents. Thus, relative majority, beside increasing the scalability factor of a DAO, it also increases the threat of potential attacks, since it would be possible for a small number of agents to deviate from the real DAO's global opinion.

Holographic Consensus tries to overcome the tension between scalability and resilience by setting an absolute majority to approve a given proposal, and only if some conditions are met, a relative majority suffices. The condition that allows to turn an absolute majority into a relative majority a certain proposal is called "boosting" and implies spending token to promote the proposal, until a certain threshold is reached before the end of the boosting period. Boosting can be done, not only by the members of the DAO but also by foreign agents. This mechanism establishes thus some sort of prediction market that acts as filter for the community, enabling them to make predictions about which proposal will be accepted, rewarding those that were accurate. With this method, high-quality proposals aligned with the purpose of the DAO are supposed to be selected and will require a simple majority vote instead of absolute majority.

When a proposal is created, a boosting period starts, and anyone can bet a certain

number of tokens that the proposal will pass (upstake) or not pass (downstake) after being voted. If the upstaked tokens of a proposal reach a dynamic threshold, which grows exponentially with the number of active boosted proposals, it is considered that the proposal is relevant enough, and then it is boosted. On the other side, a non-boosted proposal is a proposal that has not received enough upstakes due to downstakes or inaction. Finally, when the proposal has an outcome (the result voted by the DAO members), stakers who were aligned with the DAO's opinion will gain tokens, while the ones who were not, will lose them. Thus, since stakers act as predictors to maximize their return, they need to anticipate the DAO's global opinion. Therefore, if stakers are rational, they will boost 'good proposals' and filter out the one which are not aligned with the DAO's scope. Consequently, proposals boosted would need less votes to be approved, allowing the community to saves attention and voting effort, while those not boosted will require a majority vote. As a result, DAO members need stakers for the sake of scalability, and stakers need to anticipate DAO's opinion in order to earn a reward. [25]

2.2.3 Conviction voting

The concept of Conviction Voting is designed specifically for the allocation of resources and budgeting decisions. The decision-making process of Conviction Voting funds proposals based on the aggregated preference of community members, expressed continuously. Also, instead of ordering proposal in a transitive way, the community members divide all their voting power among the different proposals according to their respective proportions. Rather than casting votes in a single time-boxed session, voters constantly assert their preference for which proposals they would like to see approved. Members can change their preference at any time, but the longer they keep their preference for the same proposal, the "stronger" their conviction gets. This added conviction gives long standing community members with consistent preferences more influence than short term participants merely trying to influence a vote. Conviction Voting sidesteps sybil attacks, provides collusion resistance, and mitigates many of the attack vectors of time-boxed voting mechanisms. [24]

Chapter 3

Decentralized Autonomous Organization and Corporate Governance

3.1 DAOs in Corporate Governance Theory

Corporate governance is founded on the agency relationships between the various actors of a corporation. Modern finance and corporate governance in fact, attempt to optimize the incentives between principals and agents, optimize the risk preferences between them, minimize information asymmetries, and control costs, adverse selection and moral hazard. The Corporate form remains the most popular form of a governance mechanism, despite the unresolved agency problems associated with the separation of ownership and control, and the incomplete and sub-optimal rules that govern such conflicts.

In this framework, shareholder value maximization has emerged as the dominant corporate governance solution for reducing the risks of managerial misbehavior and the associated agency problems, by increasing the alignment of the interests of the stakeholders within those of the investor-shareholders, so that it has become the dominant implementation of the shareholder primacy doctrine. The doctrine suggests that by aligning the interests and incentives of the various actors with those of the investor-shareholders, all of the stakeholders in a firm and the public benefit. Following this logic, the primary goal of corporate governance is then to

increase the shareholder control over other actors within the firm, so that correct corporate governance measures are seen as naturally resulting in shareholder value. [26] However, despite decades of governance experiments and extensive rule revisions, the existing scope of agency problems suggest that the core underlying agency problems cannot fully be resolved within the existing theoretical and legal infrastructure.

In this context, blockchain-based solutions can offer new alternatives to the traditional corporate governance mechanisms. DAOs, for example, can be more efficient than hierarchical organizations and their governance. In a hierarchical organization in fact, power is divided according to a hierarchical tree structure by necessity and governance strengthen such power structure further with agency constructs. By contrast, a DAO, ideally gives autonomous power to each member, enabling greater flexibility and enhanced options for efficiency using the decentralized power structure of interacting domains of expertise. DAOs can also ending up in being more productive with respect to hierarchical organizations because the information allocation and feedback effects allow them to distribute the optimal amount of power to the optimal talent at the optimal point in time. The successful implementation of such dynamic power organization requires a decentralized governance structure that motivates DAO token holders to collaborate productively, by fairly rewarding development and work and policing any diminishments.

However, current governance theories do not account for the alienation of trust between parties and the union of ownership and control when a number of stakeholders with competing interests exist, such in the case of a DAO. For example, Agency Theory poses an ideal case where a single "entrepreneur-manager" makes optimal decisions and then executes them, acting both as principal and agent in his own interest. While in the case where there are multiple principals and agents, with the absence of any incentives, agents will pursue their own interests separate to the interests of the principals, the DAO raises the possibility of a "next-best-case" of Agency Theory, where there are multiple entrepreneur-managers who have no need to trust each other and may function as a single-minded entrepreneur-manager. Moreover, existing experimentation with DAO designs already contradicts fundamental assumptions about firms, such as the hierarchical organizational structure, the separation of firm members from market members, the cultural or technical

homogeneity of members, and many other natural definitions for a firm. Indeed, decentralized transaction cost economics call into question the theory of the firm, according to which firms exist to reduce transaction costs and as a response to the high cost of using markets. In fact, if the high costs of using markets, that legitimize the existence of the firms, can be lowered ad infinitum, the role of the firm itself becomes questionable. Peer-to-peer transactions in DAOs, facilitated by trust through decentralized consensus protocols and other emerging decentralized technology, can significantly lower transaction costs. Efficient smart contracts enable trustworthy transactions of any size with minimal transaction costs and high levels of transaction security so that monitoring costs and the cost of agent supervision may become largely superfluous in effective DAO designs. Overall, emerging DAO designs can be expected to continue to remove transaction costs, making them a more efficient business organization vehicle, especially in comparison with traditional firms.

3.2 Legal and governance issues of DAOs

While providing a transparent environment for enabling permissionless participation and a dynamic decision-making process that evaluates preferences in nearly real time, [27] the revolutionary mechanism behind DAOs has also incurred in several legal challenges and governance issue, that has to be overcome to reach widespread adoption.

First, the intrinsic risks of a distributed governance system have to be taken into account when designing the governance structure. In fact, even if smart contracts may provide certain operational efficiencies by streamlining decision making procedures, they do not eliminate the social and political dimensions of governance. Consequently, this may result in increased costs and difficulties for reaching group consensus, which could in turn limit the ability of DAOs to take actions. A distributed consensus mechanism is in fact subordinated to the member's participation and requires people to remain consistently engaged and attentive to an organization's activities on an ongoing basis. For many, in fact, collecting all of the information necessary to make a well-informed decision could be too laborious and complex, discouraging participation and may ultimately, limits their ability to

generate social and economic gains.

Moreover, while incorporating more traditional legal documents into smart contracts has some appeal, it creates several downsides. In fact, using legal text to accommodate or describe the underlying mechanics of smart contracts could end in potential mistranslation or ambiguity about how the underlying smart contracts of a DAO work. Such translation errors create more opportunities for dispute amongst members and courts, who may be tasked with administering a DAO-related dispute, may lack the instruments for assessing whether the code or the natural language provisions in the operating agreement should prevail for resolutions. On the other side, legal agreements increase the costs of creating and establishing a DAO, which undermine the potential efficiencies given by the use of blockchain technology and related smart contracts to create, set up, and administer a DAO. Moreover, the need of support from a lawyer or other legal service to assist the creation of an agreement that aligns the intent of DAO members may go against the purpose of private ordering of some members.

Beyond questions of governance, DAOs still lack any formal legal recognition, and this create potential liabilities for members and expose them to the organization's liabilities and responsibilities. Moreover, the lack a legal entity, may also limit the ability of these organizations to conduct affairs with more traditional legal enterprises and to protect the personal assets of owners from creditors. For instance, in the U.S., DAOs formed for the purpose of making a profit are likely to be classified as "general partnership" and consequently lack the ability to protect members' assets in case of a third-party injury or inability to repay its creditors. These factors may restrict the potential member-base of DAOs, particularly excluding those with significant assets and capitals. However, state law efforts are already underway to adapt traditional business entities to DAOs and it is possible to argue that as blockchain-based enterprises become more mainstream, the creation of a path to limited liability and legal personhood will become a more important objective for entrepreneurs and investors. As example, In Vermont, the legislature has passed an amendment to the state's limited liability company statute which would allow a limited liability company to designate itself a "Blockchain-Based LLC.", authorizing the creation of an LLC that substitutes "blockchain technology" for traditional governance tools.

Additional challenges with DAOs derive from their ability to represent interests in these organizations as tokens. Typically, when companies raise money from the public, they issue securities that may take several forms such as common stock, preferred stock, bonds, or convertible bonds, and which are clearly identified as debt, equity, or a hybrid of the two. Through the use of smart contracts and blockchain-based tokens, however, businesses have the ability to sell tokens to the public and raise money in novel ways. Tokens can be in fact associated with economic rights, participation rights, governance rights, and utility right, and their issuance to the public is somewhat similar to the traditional Initial Public Offerings (IPOs). The explosion of Initial Coin Offerings (ICOs) in the recent years has demonstrated the ability of blockchain-based enterprises to raise large sums of money through the sales of these tokens, but from a regulatory perspective, it is still uncertain if such tokens should be considered as securities or not. Tokens, in fact, may implicate interests related to both investment and consumption by entitling the holder to use a particular platform or network and also holding out the possibility of generating economic gains through the resale on the secondary market. [28]
The lack of proper incentivization is a core common denominator for all existing blockchain governance structures and especially on-chain governance structures. The participants in any institution need to be incentivized to improve their own utility while at the same time benefiting the entirety of the institution for the long run. Without that duality of incentivization, rational and opportunistic internal and external constituents will attempt to game the governance design.

3.3 A case study: The DAO

"The DAO", whose name has not to be confused with the generic acronym of decentralized autonomous organization, was established on the Ethereum blockchain in April 2016 and was aimed to be a form of investor-directed venture capital fund. With more than $120 million raised in digital currency equivalent from more than 11.000 investors, The DAO's crowfunding campaign was recognized to be one of the largest in the history of the blockchain industry. The DAO used smart contracts on the Ethereum blockchain network to manage its trustless environment

and make corporate, management or governance decisions. Smart contracts in fact, both granted investors voting rights according to their level of investment and managed their subsequent votes on investment proposals accordingly, so that all the decisions regarding the distribution and management of funds, risk, residual claims, voting rights, and voting itself, were achieved through the consensus of the investing community. Therefore, The DAO can be considered as a unique case of a company whose corporate governance consisted entirely of information technology governance.

However, on June 17, 2016, a hacker found a bug in the code of the smart contracts that allowed him to steal funds from the company. In the first few hours of the attack, the equivalent of $70 million in digital currency was stolen. As a response to this, the Ethereum's founder Vitalik Buterin initially proposed a soft fork, in order to prevent the hacker to move the stolen funds and create a mean for the investors to reacquire their funds. This opened many debates about the morality and philosophy of the action, since altering the "history" of the blockchain unilaterally undermines the pillars of censorship-resistance and non-reversibility and poses question about the degree of effective control of the Ethereum blockchain. In the end, a majority of the investors who made up The DAO agreed to introduce a hard fork to return the funds, where many users agreed to alter their copy of the blockchain to a new version where the hack had never happened. This created a parallel blockchain where there had been no money stolen and the bug had been patched. Unsurprisingly, the hack was the beginning of the decline of the organization. However, the hack highlighted the controversial nature of the "algorithmic authority" of the blockchain code. In this regard, it is important to highlight the opinion of the "hacker", according to who, he acted in line with the rules and the rights determined by the smart contract, as he declared in an anonymous letter:

"I have carefully examined the code of The DAO and decided to participate after finding the feature where splitting is rewarded with additional ether. I have made use of this feature and have rightfully claimed 3,641,694 ether, and would like to thank The DAO for this reward... "

and also

"I am disappointed by those who are characterizing the use of this intentional feature as "theft." I am making use of this explicitly coded feature as per the smart contract

terms and my law firm has advised me that my action is fully compliant with United States criminal and tort law... " [29]

Under this prospective, the terms of the smart contract dictate that the code itself and the members of The DAO community all agreed to be bound exclusively by the code of the smart contract, and anyone who utilizes the code, ethically or not, is merely exercising his rights under the contract.

The DAO raised legitimate questions also about whether someone should be accountable in DAOs, whether trustless systems are really "trustless," and what organizations like The DAO will look like in the future when and if the details of governance, legalities, ethicalities, and the logic flaws in the code are corrected. In fact, the key issue that arises by removing governance from people and placing it in the hands of a smart contract is the inability to hold individuals accountable when things go wrong. Morover, despite the transparency provided by the blockchain, a large number of information asymmetries existed between participants in The DAO. Participants did not know who each other were, their ambitions or motivations to invest in The DAO, or their values and priorities. It is evident thus that some were unable to trust the various proposed solutions to their problem in a way that would allow them to effectively and efficiently vote in favor of or against it. Their priorities and values did not align and there were no contingencies to define, manage, or control these conflicts. The consequence was chaos in a time of crisis and the splitting of an organization.

"*The lack of centralized authority needed to make quick decisions was felt strongly throughout the history of DAO. This is however the nature of decentralized systems, and is both a blessing and a curse. This is exemplified by the fact that even little posts by Vitalik were interpreted as decisions, even though he just gave his opinion.*" [30]

By the way, the voting design adopted by The DAO introduced also some points of centralization and corruption. Just as with the traditional voting stock of a corporation, The DAO's token allowed its holders to vote on investment proposals with a voting power proportional to the token owned and voting proposals did not involve staking of any non-fungible tokens towards a vote outcome and did not have a randomization element, so that all the voting revolved around centralized voting power. As such, its voting design introduced opportunities to corrupt the

DAO and its governance structure. The governance structure in fact relied on centralized leadership in the form of "curators". The curators created and controlled the whitelist of those pre-selected contractors that were authorized to receive ether from the DAO. Because the smart contract on its own could not distinguish real from fake proposals, the curators had approval power and vote prioritization power over contractor proposals before the proposals were put up for the community vote. DAO token holders who were selected as contractors through a DAO member vote needed the curators' verification that a given smart contract submitted by the token holder matched the public smart contract on the Ethereum blockchain, and that the contractor's identity matched. Curators, thus, were the de-facto the core point of centralization in the design of The DAO. The payment structure for smart contract work proposals also introduced a point of centralization. Despite any token holder was eligible to submit a proposal via smart contract to become a contractor for the token holder community, he was required to pay an deposit that he would forfeit if the proposal failed to achieve the quorum requirement, introducin possible points of attack and corruption.

Chapter 4

Voting model in a DAO

4.1 Introduction

The management of the operations in a DAO recalls the voting mechanism used in publicly-traded companies, in which shareholders are called to express their opinion over certain questions such as electing the Board of Directors, merger and acquisition, executive compensation packages or stock split, with a voting power that is proportional to shares they own. In DAO, such governance mechanism is extended to the entire administration of the organizations, trough proposals that have to be submitted and voted on a frequent basis in order to address decisions over the operations that have to be followed. Indeed, DAOs are characterized by the fact that there is not a separation between ownership and control i.e., members own a part of the company and at the same time act as self-managers, since they can vote the proposals that has to be implemented by the DAO by committing their voting power, that could be proportional in some way with respect to the amount of token they hold. Similar to corporations quoted at the stock exchange, DAOs' ownership is highly fragmented among different people, with different goals and preferences with respect to the proposals that has to be voted. Differences with respect to a proposal can arise from private benefits, different social or political views, time horizon, risk aversion or other factors. All these factors make the blockchain-based corporate governance functions only to a limited extent in traditional corporate forms, since they offers dynamic regulatory features that are partially incompatible with the rule-based traditional legal environment with

which traditional limited liability entities are required to comply. DAO model could be in fact more suitable for certain kind of organizations such as investment funds or charity communities but could result in immobility for companies operating in other industries, since the governance is subordinated to high participation of the members. The model described in the sections below aims to describe the impact of bias, concentration of the ownership and voting costs in reaching the approval of a proposal, by interpreting the voting process of the token-holder as a two-stages game.

4.2 Model

Consider a DAO with a continuum of risk-neutral token-holders, indexed by i, each of them owning a quantity $t_i > 0$ of tokens with $t_i \in (0, T^*)$, such that $\sum_{i=1}^{n} t_i = T$, where T is the fixed total token supply and $T^* < T$ the maximum quantity each member can own. Token-holders should vote a proposal, that can be either accepted (a=1), if a portion t_a /T or more than $\tau \in (0,1)$ of all the tokens are assigned in favour of the proposal or rejected, otherwise (a=0).

$$\begin{cases} a = 1, & \text{if } t_a/T \geq \tau \\ a = 0, & \text{if } t_a/T < \tau \end{cases} \tag{4.1}$$

The value of the token for the member i-th, upon the announcement of a new proposal, can be expressed by

$$v_i(a, \theta, b_i) = v_0 + a(\theta + b_i) \tag{4.2}$$

and depends on whether or not the proposal is accepted $a \in \{1,0\}$, on the state θ, on his bias b_i, such that $\theta + b_i > -v_0$ and on the initial value of the token $v_0 > 0$ before the announcement. The added value of the proposal for each member will be then

$$v_i(1, \theta, b_i) - v_i(0, \theta, b_i) = \theta + b_i \tag{4.3}$$

and each of them will be willing to accept the proposal as long as his expectation of $\theta + b_i$ is positive. The parameter θ contains the part of the value of the proposal that is common to all the token-holders. The more the proposal is aligned with

the objectives of the DAO, the higher θ will be. The parameter b_i represents token holder's bias for the proposal and it can be positive or negative. Bias might be influenced by the token holder private opinion, but also from eventual private benefits that he could enjoy with the approval of the proposal. By shifting the focus from the member to the token, it is possible to represent each token with the parameters b_i and t_i that represent respectively the token-holder's bias and the amount owned. Supposing then that biases have a density function f(b), continuous in $[-\bar{b},\bar{b}]$, with $\bar{b}>0$, it is possible to define F(b) as the cumulative distribution function of the tokens with respect to their corresponding token-holders' biases.

$$F(b) = P(x \leq b) = \int_{-\bar{b}}^{\bar{b}} f(b)\, db \tag{4.4}$$

To simplify the exposition, it is also useful to introduce the tail distribution

$$T(b) = 1 - F(b) \tag{4.5}$$

Same considerations can be applied to the distribution of the tokens among the members, since members are ordinated somewhat, defining f(t) as the probability density function of the token owned by the members, continuous in $(0,T^*)$, and F(t) and H(t) as the respective cumulative and tail distribution.

$$F(t) = P(x \leq t) = \int_{0}^{T^*} f(t)\, dt \tag{4.6}$$

$$H(t) = 1 - T(t) \tag{4.7}$$

We assume also that trading is not allowed, so that the distribution of the tokens remain constant. Under these conditions, it is possible to represent the decision-making process of a DAO through a two-stage game in which players take a decision on whether or not participating to the discussion in the first phase and vote in favour or against the proposal in the second phase.

4.3 Discussion

We assume that only the members who participate to the discussion, will express their vote on the subsequent stage. Participating in the discussion, before the

voting stage takes place, involve some voting costs V_c, that can be explained by the time and efforts spent in evaluating and discussing the proposal. On the other side, such voting costs allows participants to reach a more accurate opinion on the real value of the proposal. Thus, after the discussion stage ends, participants will observe a public signal c about the state θ, where c = E(θ|discussion). During the voting stage, participant will evaluate the proposal according to their value of $c+b_i$ and will respectively vote in favour if $c+b_i > 0$ or against if $c+b_i < 0$. It is possible then to conclude that, if the proposal is approved by the token holders vote, there should exist in any equilibrium, a value c* such that the proposal is approved when $c \geq c^*$.

4.4 Model without discussion

We first consider the simplest case in which players vote without discussing so that all the token are caste in favour or against the proposal, without abstentions. The density function f(b) represents the token base at the voting stage, and the proposal is approved if and only if at least fraction τ of the token are casted in favour of it. Since member with a larger bias value the proposal more, this will be approved if and only if the τ-th token is casted in favour for the proposal by his holder with a bias $b_i = T^{-1}(\tau)$ and since the τ-th token will be casted in favour if and only if $\theta > -b_i$, the condition that allows the approval of the proposal is

$$\theta > \theta^* = -T^{-1}(\tau) \tag{4.8}$$

i.e., when

$$1 - \int_{-\bar{b}}^{\theta^*} f(b)\, db = \tau \tag{4.9}$$

Figure 3.1 illustrates the equilibrium of the model without discussion and plots the tail distribution T against the biases b. The tokens associated with a bias b = $-\theta^*$ represent the marginal voters, whose vote on the proposal determines whether it is approved. The identity of this members (and the tokens associated to) is crucial for the decision on the proposal. If $\theta = \theta^*$, there are 1-τ tokens associated to holders for whom $b_i + \theta < 0$, who will vote against ("Vote No" region of the

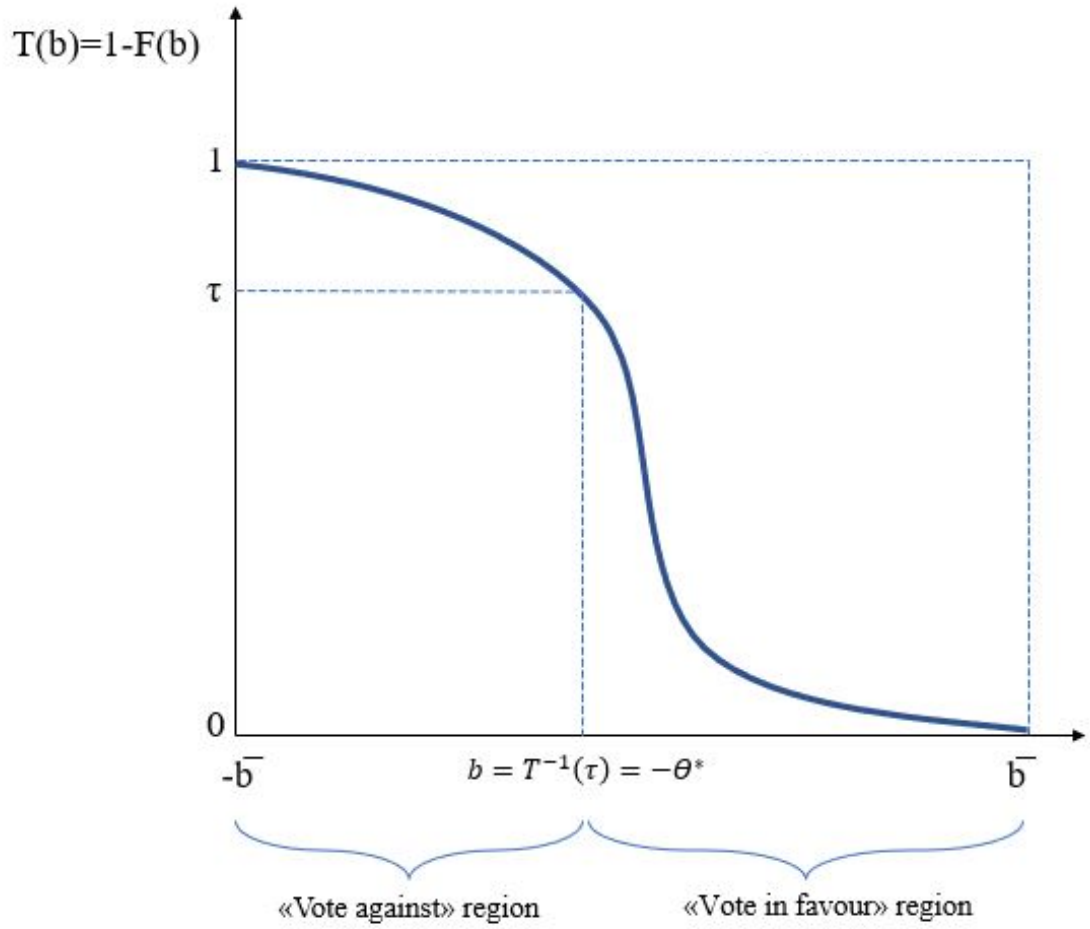

Figure 4.1: Equilibrium without discussion

Figure), and τ tokens that will be casted in favour of the proposal ("Vote Yes" region). Thus, the marginal voters are the holders who are indifferent between accepting and rejecting the proposal if exactly τ token are casted to accept it.

Voting Stage	
$\theta + b_i$	Result
>0	Vote in favour
<0	Vote against

Table 4.1: Strategies of players in one-stage game.

4.5 Two-stages model

We consider now that members, before voting, have to participate to a discussion, which entail some costs. The assumption here is that all the token holders who had taken part to the discussion stage will vote, while who decided to not participate to the discussion will abstain. Members decide to participate to the discussion on the basis of the voting cost and their expected payoff of the proposal, that is thus influenced by the amount of tokens they own. The participation condition can be written then as:

$$t_i|v(1,\theta,b_i) - v(0,\theta,b_i)| > V_{\mathrm{c}} \tag{4.10}$$

from which follows

$$t_i|v_0 + (\theta + b_i) - v0| > V_{\mathrm{c}} \tag{4.11}$$

and thus

$$t_i|\theta + b_{\mathrm{i}}| > V_{\mathrm{c}} \tag{4.12}$$

The equation above relates the number of tokens owned by the players and their ex-ante expectation with respect to the proposal with the voting costs that they have to face when deciding to participate to the discussion. In fact, if the total payoff/lose obtained from the approval of the proposal exceed the voting costs, players will decide to participate to the discussion while they will decide to abstain if their expected payoff/lose is not higher than the voting costs. This reflects the fact that members with low stakes are in general less interested in taking part to the governance of the DAOs although some of the low-stake members may participate, because of higher bias, that in this case may be identified as "hype". Given the voting cost condition, only a fraction of all the holders will decide to participate and only a portion α of the total tokens will be casted in favour or against the proposal.

The approval is then dependent on the concentration of the tokens, the ex-ante expectation for the proposal, the voting cost, the public signal and the distribution of the bias.

Table 3.1 represent the strategies available to the players in the two stages, depending on the the value of their bias and the amount of token owned.

Discussion Stage			**Voting stage**	
$\theta + b_i$	$t_i \mid \theta + b_i \mid$	Result	$c + b_i$	Result
>0	$>V_c$	Participate	>0	Vote in favour
	$>V_c$	Participate	<0	Vote against
	$<V_c$	Not participate	>0	Abstain
	$<V_c$	Not participate	<0	Abstain
<0	$>V_c$	Participate	>0	Vote in favour
	$>V_c$	Participate	<0	Vote against
	$<V_c$	Not participate	>0	Abstain
	$<V_c$	Not participate	<0	Abstain

Table 4.2: Strategies of players in the two-stages game.

Defining as f(t,b) the joint probability density function of the number of tokens owned by the same member and the corresponding bias and A as the subset of R^2 for which $t_i(b_i + \theta)$ - $V_c > 0$, it is possible to write the fraction of tokens that participate to the discussion α as

$$\alpha = \int_A f(t,b)\, dtdb \tag{4.13}$$

In order for the proposal to pass, such fraction should necessarily be $\geq \tau$.

4.6 Uniform distribution of tokens

In order to provide some insights about the effect of the discussion in the equilibrium, we consider the case in which the token are distributed uniformly, with n members owning the same constant quantity $t_n = \frac{T}{n}$, the condition to participate in the discussion stage will be:

$$b_i > \frac{V_c}{t_n} - \theta \tag{4.14}$$

$$b_i < -\frac{V_c}{t_n} - \theta \tag{4.15}$$

So that the portion of participants will be

$$\alpha = \alpha_1 + \alpha_2 = \int_{-\bar{b}}^{-\frac{V_c}{t_n}-\theta} f(b)\, db + \int_{\frac{V_c}{t_n}-\theta}^{\bar{b}} f(b)\, db \tag{4.16}$$

In Figure 3.2 are represented the regions of participation for the members. In

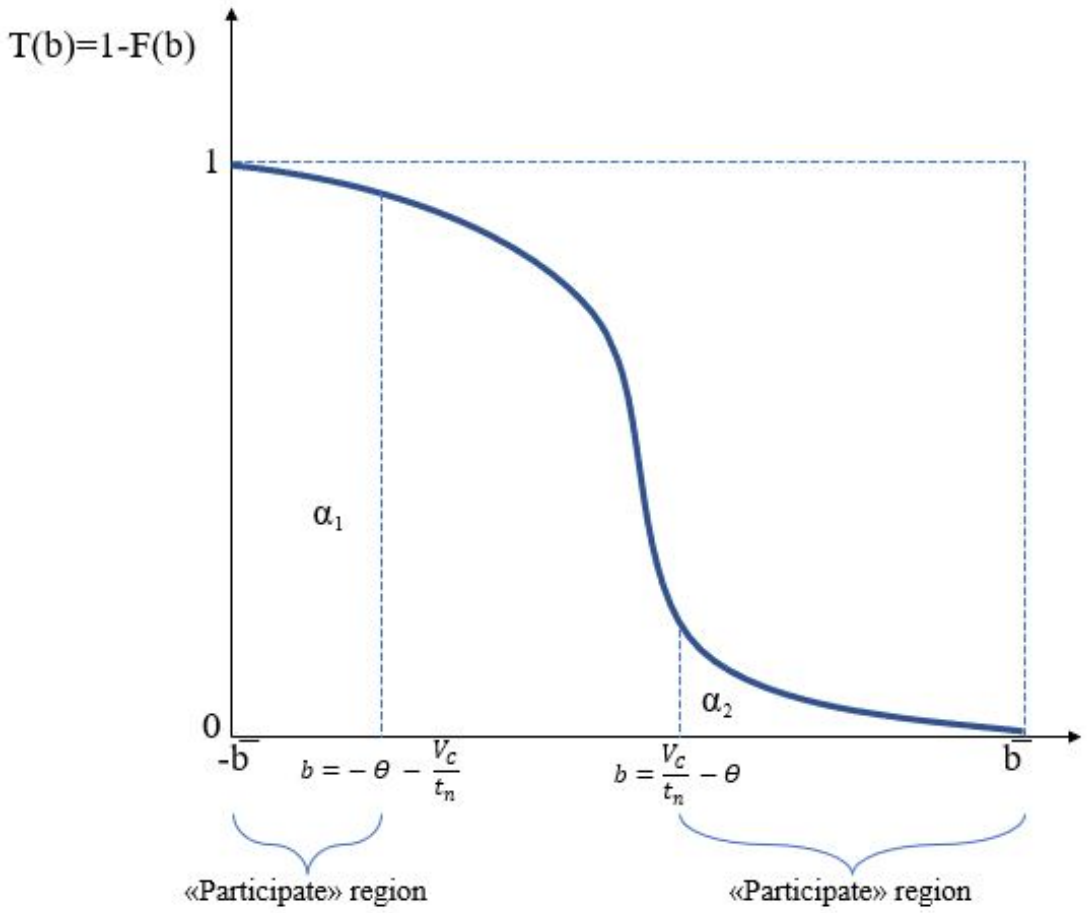

Figure 4.2: Equilibrium of participants with uniform distribution

particular, it is possible to note that the tokens associated to a bias in $[-\bar{b}, \frac{V_c}{t_n} - \theta]$ belong to players who decided to participate because they think the proposal will disadvantage them and so are willing to vote against it (α_1), while the token associated to a bias in $[\frac{V_c}{t_n} - \theta, \bar{b}]$ belong to players who participate to the discussion because of a positive expectation on the proposal (α_2). The "not participation" region contains instead the tokens owned by players for which the voting costs exceed their expected pay-off/loose and may include both players for which $\theta + b_i > 0$ and $\theta + b_i < 0$.

Finally, in the voting stage, players will vote in favour or against the proposal depending on their value of $c + b_i$ and it is possible to distinguish different situations,

depending on the value of c:
c $> \bar{b}$: All the players will vote in favour.

$$V_{\text{favour}} = \alpha_1 + \alpha_2 \tag{4.17}$$

$\theta + \frac{V_c}{t_n} < \text{c} < \bar{b}$: all the players of the region α_2 and part of α_1 vote in favour.

$$V_{\text{favour}} = \int_{-c}^{-\frac{V_c}{t_n}-\theta} f(b)\, db + \alpha_2 \tag{4.18}$$

θ - $\frac{V_c}{t_n} < \text{c} < \theta + \frac{V_c}{t_n}$: only the players α_2 will vote in favour

$$V_{\text{favour}} = \alpha_2 \tag{4.19}$$

-$\bar{b} < \text{c} < \theta$ - $\frac{V_c}{t_n}$: only part of the players α_2 will vote in favour

$$V_{\text{favour}} = \int_{-c}^{\bar{b}} f(b)\, db \tag{4.20}$$

c < -$\bar{b}$: All the players will vote against the proposal.

$$V_{\text{favour}} = 0 \tag{4.21}$$

By imposing then that and $\text{V}_{\text{favour}} = \tau$ it is possible to find the equilibrium value of c^* such that the proposal is approved.

4.7 Observations

The model proposed represents the behavior of the members of a DAO when they have to face decisions concerning a proposal. By comparing the choices available to the players in the two cases, with and without the discussion stage, it is possible to derive the following considerations:

- Without the discussion stage, all the members vote only on the basis of their ex-ante expectation of the proposal, given by their bias and the common value the proposal ante-discussion. This, from one side leads to higher participation ($\alpha = 1$) but also to results that could be less beneficial for a good management of the organization, since θ is a less accurate value for taking decisions.

- By introducing the parameter Vc, representing the voting costs, the model want to address the situations for which some of the members abstain from voting. In particular, members for which $t_i|\theta + b_i| < V_c$ prefer to not participate in the voting process since their estimated impact of the proposal is not enough relevant for them. The presence of voting costs, reduce the participation of the members in the decision making process of the DAO, consequently increasing the space for collusion and misbehavior by the players on one side, and increasing the likelihood of rejection of proposals that could be beneficial for DAO's members. However, players who take part of the discussion will base their vote on c rather than θ, substantially improving the accuracy of their opinion and the outcome of the collective decisions.

- The approval or rejection of the proposal is highly influenced by the parameter τ, that express the percentage of voting power which should agree for a given proposal to pass. Lower value of τ, beside favouring the scalability in the decision making, increase also the threats of collusion among members with higher stake in the organization.

- An high dispersion of ownership, which is seen as a favorable trait in decentralized organization, reduce members incentive in participating to the voting process

Designing an appropriate voting structure thus requires to make trade-offs between the level of resilience against misbehavior and collusion and the practicability and smoothness of the governance mechanism but also with regard to the concentration of the ownership and the incentives to participate in the decision-making process.

4.8 Conclusions

The purpose of this work was to describe the new role of the actors in a DAO and the new mechanisms of governance that blockchain-based solutions entail, by analysing the different features that a decentralized decision-making process gives, and providing a simplistic model to highlight the trade-offs that members face when deciding to participate or not in such process.
DAOs are by no means mainstream and will be subject to a significant emerging

development process in the next decade. The distinctive case of The DAO highlights the role of trust in current governance structures and identifies it as an underlying assumption in key governance theories: trust, or the lack thereof, is revealed to be the underlying issue dictating how organizations manage themselves in order for actors to be made trustworthy. Still, The DAO represented a new species of governance characterized by the alienation of trust from parties and union of the ownership and control inside the organization. Because the blockchain industry is still in its infancy and the core decentralized infrastructure elements will remain lacking for the foreseeable future, DAOs are less likely to disrupt existing corporate structures and the associated governance solutions in the near future. However, DAOs have the potential to create significant decentralized equivalents of corporate structures and such development is contingent on workable governance solutions for DAOs. However, the blockchain industry has begun to recognize the need for DAO infrastructure and governance solutions, although such they are still largely lacking, even in the developmental phases.

Moreover, the success of DAO governance designs is determined by several core factors. First and foremost, the level of decentralization in the DAO governance design is decisive for its future resilience and attack resistance. Incentive alignment of DAO constituents is a core function of successful decentralization and has a significant impact on the success of the governance design. Dynamic feedback effects in the DAO governance design also impact its ability to provide solutions as the circumstances of the DAO change. The structure and extent of on-chain DAO governance is also an important evolving design factor. Finally, decentralized autonomous and anonymous reputation verification systems offer much needed governance design input parameters that so far could not been provided otherwise. In conclusion, further efforts are needed to fully understand this new phenomenon, to articulate its implications for corporate governance, and examine how organizations may overcome these new challenges in the future.

Bibliography

[1] Robby Houben and Alexander Snyers. *Cryptocurrencies and blockchain Legal context and implications for financial crime, money laundering and tax evasion STUDY Requested by the TAX3 committee Policy Department for Economic, Scientific and Quality of Life Policies* (cit. on pp. 8, 9, 12).

[2] Bruno Biais, Christophe Bisière, Matthieu Bouvard, and Catherine Casamatta. «The Blockchain Folk Theorem». In: *Review of Financial Studies* 32 (5 May 2019), pp. 1662–1715. ISSN: 14657368. DOI: 10.1093/rfs/hhy095 (cit. on pp. 9, 15, 16, 20).

[3] Ivan Osipkov, Eugene Y Vasserman, Nicholas Hopper, and Yongdae Kim. *Combating Double-Spending Using Cooperative P2P Systems*. 2007 (cit. on p. 9).

[4] Alan T Sherman, Farid Javani, Haibin Zhang, and Enis Golaszewski. *On the Origins and Variations of Blockchain Technologies*. 2018 (cit. on p. 9).

[5] Satoshi Nakamoto. «Bitcoin: A Peer-to-Peer Electronic Cash System». In: (). URL: www.bitcoin.org (cit. on pp. 9, 15).

[6] Andreas M. Antonopoulos. *Mastering bitcoin : unlocking digital cryptocurrencies*, p. 272. ISBN: 9781449374044 (cit. on pp. 10, 14).

[7] Adam Back. *Hashcash-A Denial of Service Counter-Measure*. 2002 (cit. on p. 10).

[8] Z. Li, Ray Y. Zhong, Z. G. Tian, Hong Ning Dai, Ali Vatankhah Barenji, and George Q. Huang. «Industrial Blockchain: A state-of-the-art Survey». In: *Robotics and Computer-Integrated Manufacturing* 70 (Aug. 2021). ISSN: 07365845. DOI: 10.1016/j.rcim.2021.102124 (cit. on p. 10).

[9] *El Salvador becomes first country to adopt bitcoin as legal tender* (cit. on p. 10).

[10] Claudina Castro Tanco Linda Pawczuck Richard Walker. *Deloitte's 2021 Global Blockchain Survey* (cit. on p. 11).

[11] Jeannie Jo Penn P. Wu. «Topics of blockchain technology to teach at community college». In: () (cit. on p. 11).

[12] Shijie Zhang and Jong Hyouk Lee. «Analysis of the main consensus protocols of blockchain». In: *ICT Express* 6 (2 June 2020), pp. 93–97. ISSN: 24059595. DOI: 10.1016/j.icte.2019.08.001 (cit. on pp. 17, 20).

[13] *Pool Stats* (cit. on p. 19).

[14] Cong T. Nguyen, Dinh Thai Hoang, Diep N. Nguyen, Dusit Niyato, Huynh Tuong Nguyen, and Eryk Dutkiewicz. «Proof-of-Stake Consensus Mechanisms for Future Blockchain Networks: Fundamentals, Applications and Opportunities». In: *IEEE Access* 7 (2019), pp. 85727–85745. ISSN: 21693536. DOI: 10.1109/ACCESS.2019.2925010 (cit. on pp. 20, 23).

[15] Zibin Zheng, Shaoan Xie, Hong-Ning Dai, Weili Chen, Xiangping Chen, Jian Weng, and Muhammad Imran. «An Overview on Smart Contracts: Challenges, Advances and Platforms». In: (Dec. 2019). DOI: 10.1016/j.future.2019.12.019. URL: http://arxiv.org/abs/1912.10370%20http://dx.doi.org/10.1016/j.future.2019.12.019 (cit. on p. 23).

[16] Vitalik Buterin. *Ethereum: A Next-Generation Smart Contract and Decentralized Application Platform.* (Cit. on p. 23).

[17] Bhabendu Kumar Mohanta and Debasish Jena. *An Overview of Smart Contract and Use cases in Blockchain Technology* (cit. on p. 25).

[18] Quinn Du Pont. *Experiments in algorithmic governance*. 2017 (cit. on p. 25).

[19] Usman W. Chohan. «DAO and governance issues». In: () (cit. on p. 26).

[20] *Binance DAO Theory* (cit. on p. 27).

[21] Steven P Lalley and E Glen Weyl. *Quadratic Voting How Mechanism Design Can Radicalize Democracy*. URL: https://ssrn.com/abstract=2003531 (cit. on p. 30).

[22] *An Operating System for Collective Intelligence*. 2018 (cit. on p. 31).

[23] *Urgent flash loans and securing the maker protocol* (cit. on p. 33).

[24] *Voting options in DAOs* (cit. on pp. 33, 34).

[25] Youssef Faqir-Rhazoui, Javier Arroyo, and Samer Hassan. *A scalable voting system: Validation of holographic consensus in Daostack*. 2021, pp. 5557–5566. DOI: 10.24251/hicss.2021.676 (cit. on p. 34).

[26] Wulf A Kaal. *Stanford Journal of Blockchain Law Policy Blockchain-Based Corporate Governance*. 2021 (cit. on p. 36).

[27] Nathan Schneider. *Cryptoeconomics as a Limitation on Governance* (cit. on p. 37).

[28] Aaron Wright, Benjamin N Cardozo School Of Law, Stanford Journal, and Blockchain Law. *Stanford Journal of Blockchain Law Policy The Rise of Decentralized Autonomous Organizations: Opportunities and Challenges License: Creative Commons Attribution 4.0 International License (CC-BY 4.0)*. 2021 (cit. on p. 39).

[29] Anonymous. In: (2016) (cit. on p. 41).

[30] Christoph Jentzsch. «The History of the DAO and Lessons Learned». In: (Aug. 2016) (cit. on p. 41).