Understanding scalability in distributed ledger technology*

JONATHAN CLARK, University of Cape Town, South Africa

Distributed ledger technology (DLT) stands to benefit industries such as financial services with transparency and censorship resistance. DLT systems need to be **scalable** to handle mass user adoption. Mass user adoption is required to demonstrate the true value of DLT. This dissertation first analyses scalability in ethereum and EOS. Currently, ethereum 1.0 uses proof of work (PoW) and handles only 14 transactions per second (tps) compared to Visa's peak 47 000 tps. Ethereum 2.0, known as Serenity, introduces sharding, proof of stake (Casper), plasma and state channels in and effort to scale the system. EOS uses a delegated proof of stake (DPoS) protocol, where 21 super-nodes, termed 'block producers' (BPs), facilitate consensus, bringing about significant scalability improvements (4000 tps). The trade-off is decentralisation. EOS is not sufficiently decentralised because the BPs yield significant power, but are not diverse. This dissertation conducts an empirical analysis using unsupervised machine learning to show that there is a high probability collusion is occurring between certain BPs. It then suggests possible protocol alterations such as inverse vote weighting that could curb adverse voting behaviour in DPoS. It further analyses whether universities are suitable BP's before mapping out required steps for universities to become block producers (leading to improved decentralisation in EOS).

Additional Key Words and Phrases: Distributed ledger technology (DLT), EOS, ethereum, scalability, decentralisation, delegated proof of stake (DPoS), Block Producer (BP)

Contents

*This dissertation forms part of the MPhil in Financial Technology degree at the University of Cape Town (2019).

1 INTRODUCTION

Distributed ledger technology[1] (DLT) has garnered much attention and hype over the past few years as early adopters, institutions, and governments alike speculate on the exact value this immature and rapidly growing technology can deliver. In 2017 the market capitalization of the universe of crypto assets grew 30x to 550 billion US$ before it saw a sharp decline to 125 billion US$ in 2018[38]. The euphoria of profits and extreme market volatility has often led to an overshadowing focus on crypto asset prices as opposed to underlying system protocols delivering real value[45].

This dissertation first focusses on scalability protocols in ethereum and EOS. It then examines the trade-off between scalability and decentralisation, particularly in EOS. In then conducts an empirical analysis on an EOS voting dataset and suggests possible protocol alterations. Finally, it determines the feasibility of universities becoming block producers to aid decentralisation in EOS.

1.1 Background

One of the major challenges in DLT is scalability[13, 31, 53]. Scalability implies that the system can handle an increasing number of users and transactions, while still functioning as originally intended. The scalability issue is best illustrated by examining the bitcoin network and its current inability to function as a payments network handling global e-commerce traffic.

Given bitcoin's block size limit of 1 000 000 bytes, 10 minutes per block confirmation, and an average of 300 bytes per transaction, theoretically it can support 7 transactions per second (tps) [44, 48]. In contrast, the e-commerce giant, Visa, recorded a peak of 47 000 tps in the 2013 festive season[52]. For bitcoin to function as a global payments network and handle the equivalent amount of transactions as Visa, bitcoin's block size would need to increase to 8 gigabytes [44]. Given the average block confirmation time of 10 minutes, this would lead to over 400 terabytes of data per year[44]. To put that into context, one terabyte stores the same amount of text as the paper made from 50 000 trees. The openness of the bitcoin protocol dictates that every single transaction sent in the system must be communicated and executed by every single node participating in the network. One can imagine that requiring a node to process over 400 terabytes of data a year, would lead to a small and select set of powerful nodes operating the network.

The above illustrates a major challenge of scaling DLT networks. Given every single node has to process every single transaction in the network, the fundamental limitation on transactional throughput is the processing ability of the weakest node. This dissertation explores the different approaches taken by ethereum and EOS to ensure scalability while still aiming for system security and decentralisation.

The high level approach by ethereum is to 'shard' the network, hence not every node will have to process every transaction. Given the network is split into many shards, it can become very complicated to ensure that the system is still secure.

In contrast, EOS relies on a small subset of 21 powerful 'super-nodes' to process all network transactions. With just 21 nodes responsible for actively processing transactions, there is a concern

[1]The broader term distributed ledger technology (DLT) is used, as opposed to blockchain throughout the report. This is in accordance with the distributed ledger technology systems report[45] issued by the University of Cambridge that aims to standardize DLT nomenclature in order to provide more meaningful and standardized research in this area.

around the centralisation of the EOS system. This dissertation analyses the extent to which the system can be considered centralised, through a number of techniques, most notably using unsupervised machine learning techniques. As a result, certain protocol modifications and a proposal how universities could be part of the 21 actively producing nodes are suggested. University participation could lead to a more diversified and trusted pool of nodes, aiding system decentralisation. Participation can also enhance student learning.

1.2 Motivation

To motivate academic research in DLT scalability, it is important to understand the purpose and benefits of the technology. DLT exists to create a **shared data structure** with the following important attributes[45]:

- **Transparency.** The contents of the data structure and transactions used to change the state of the structure are completely open and viewable by anyone.
- **Persistence.** Since the data structure is replicated and stored by many nodes worldwide, the data should 'persist', and not be affected by the failure of a single node.
- **Cryptographic-validation.** Every action affecting the shared data structure has a cryptographic link pointing toward the source of the action.
- **Censorship resistance.** No single entity can alter system rules, block transactions, or restrict a users account.

The result is a data structure that "provides an authoritative version of records at a moment in time that is both shared amongst the users of the system and updated over time as users engage with one another via the system" [45].

A straightforward example of the societal benefits DLT can present is through facilitating fair democratic voting. Censorship resistance ensures that no single party can alter the voting rules or manipulate votes, and therefore affect the outcome of the vote. Cryptographic-validation ensures that each person is mathematically restricted to only a single fair vote, while transparency allows stakeholders to easily verify the outcome of the vote. Numerous academic papers have detailed how a voting system using DLT might function in practice [34, 39]. An interesting paper termed 'decentralising privacy', authored by Sandy Pentland, explores the problem of privacy in a decentralised context, using the example of voter privacy within a DLT voting system [61].

DLT also stands to benefit the financial services industry significantly. Work from the Cambridge Journal of Economics, suggests that 'New financial architecture' (NFA), the "integration of modern day financial markets with the era's light government regulation" was largely responsible for the 2008 global economic crisis[20]. In particular, the NFA introduced financial instruments that were complex and opaque, leading to a difficulty in the pricing of these instruments. An example of one such opaque instrument is the collateralized debt obligation (CDO). Each CDO can be compromised of up to 150 Mortgage Backed Securities (MBS), and each MBS can contain thousands of mortgages[20]. This reveals the clear lack of transparency in the instrument, and how it is often challenging to identify the correct owner and current state of the asset (especially given the complexity of some assets). Consequently, the difficulty of pricing these instruments often termed 'marking to magic', rendered the instruments illiquid when the boom cycle ended. This illiquidity coupled with the lack of demand caused prices to plummet, consequently acting as a catalyst for the global economic crisis. This example highlights how DLT stands to benefit financial services through the added transparency it affords. A recent review by the MIT Sloan Business school further

talks about "how [DLT] will change organizations" particularly concerning fair value exchange, a concept central to financial services[50]. Two further publications make thought-provoking remarks on how DLT stands to impact the financial services industry positively[17, 28].

Not just the private financial services industry stands to benefit from DLT. The South African Reserve Bank (SARB) initiated project Khoka in late 2017 to assess the potential of DLT in the context of interbank payment settlements in South Africa[6]. The SARB is not the only central bank exploring DLT. Project Ubin initiated by the Monetary Authority of Singapore and project Jasper initiated by the Canadian central bank also investigate the use of DLT for clearing and settlement of payments and securities[6]. These projects demonstrate that governments have recognised the potential of DLT, and are eager to harness the benefits for their respective jurisdictions.

Projects that are aiming to deliver financial services through DLT will either require their own purpose built DLT system; or some form of decentralised application (DApp), or smart contract hosted on a DLT platform. An example of a project that designed its DLT system from scratch is the Ripple network. Their system was designed for the express purpose of being a fast real-time gross settlement network. Although designing a unique DLT system for a project is possible, the significant cost, complexity and lead time of this process mean most emerging projects will instead opt for the much simpler option of using a DLT platform (such as ethereum or EOS) on which they can deploy their DApp or smart contract.

The performance of new projects deployed on DLT platforms will largely be contingent on the performance of the underlying DLT platform. For example, if a payments service is developed and deployed on the ethereum platform, given ethereum can only process 14 tps, this newly deployed payments service will also theoretically be limited to 14 tps at best. Since many financial services products involve large volumes of transactions and will require significant user adoption for success, it will be necessary for the underlying DLT platforms to be scalable. If DLT platforms are not scalable, then businesses are left with one of the following two options:

(1) Build their own DLT system. Complex, costly and represents a large barrier to entry.
(2) Build their project on a non-scalable DLT platform. The platform may not sufficiently handle the operational needs of the business.

Since both of the above options are not attractive, there will likely be minimal entrants into the market offering financial services through DLT. It is clear that scalable DLT platforms will unlock the value DLT, as it will easily allow businesses to harness the technology, and provide possibly improved financial services at scale to institutions and individuals.

The above helps to provide context around the following summarised rationale of this dissertation:

- DLT can provide significant societal benefits. Specifically, the financial services industry can benefit from greater transparency.
- New financial services offerings using DLT will most likely deploy a DApp or smart contract on a DLT platform (such as ethereum or EOS). Since cost and complexity will most likely prevent them from developing their own DLT systems.
- The entry of new DLT based financial service offerings is contingent on the underlying DLT platforms ability to meet the functional requirements of the service. One of the primary requirements for many financial services is mass-user adoption, hence scalability.
- To handle mass user adoption, underlying DLT platforms need to have the necessary protocols in place allowing them to scale.

- This motivates that research and proposals regarding DLT scalability protocols are a valuable area of research.

1.3 Objectives

This dissertation firstly examines existing protocols in DLT that limit scalability. It aims to show how protocols such as proof of work (PoW) and Nakamoto consensus place fundamental limitations on the scalability of DLT networks.

The next objective is to research and understand the different DLT scaling solutions proposed by the ethereum and EOS platforms. The aim is to understand the prospects and limitations of these solutions, as well as their trade-off's experienced between scalability and decentralisation. Given decentralisation is at the core of why DLT exists, this dissertation aims to clarify the term and critically analyse the level of decentralisation present in EOS. An empirical analysis using unsupervised machine learning techniques will be used to better understand whether centralized groups of block producers are likely to be colluding. Protocol alterations that can enhance the EOS voting landscape will be suggested.

Given that EOS provides a promising and scalable DLT platform for future businesses, improving decentralisation in EOS is a primary objective and will likely increase confidence in the system. The next objective is to propose a way in which a greater number and more diverse set of actors can be part of the EOS ecosystem, leading to a more decentralised system. This proposal aims to show that universities are capable of joining the EOS system and making it more decentralised, as universities are diverse actors with a strong reputation, very different from the current pool of participants. It also shows how university participation can enhance the learning experience while funding academic research.

The final objective is to create the proposal such that it is easily replicable by universities around the globe, to enable competition between universities, and between universities and other block producers, further promoting decentralisation.

1.4 Methodology

The dissertation takes an analytical approach by assessing theoretical and practical implications of varying protocols and suggestions. Although solutions often work in theory, real-world dynamics introduce many subtle nuances often previously unconsidered. It is therefore essential to consider how protocols function in practice, in order to better understand the prospects of scalability in DLT.

A data-driven approach is used to analyse the voting and block producing dynamics present in EOS. In particular, unsupervised machine learning techniques wil be used on the data to gain a greater understanding of this unstructured data set. The methodology used for the proposal of university DPoS participation is based on research and the understandings of an ongoing attempt to implement the suggested model through the University of Cape Town (UCT). Research combined with practical experiences will help to better formulate a realistic proposal that can be used as a starting point by universities going forward.

1.5 Dissertation structure

The first section of the dissertation involves a detailed literature review of distributed ledger technology (DLT), and its characterization in detail, before exploring two specific implementations of it

(ethereum and EOS). The remaining sections form the majority of the academic contribution as enhancements to current DLT protocols are proposed.

The detailed structure of the dissertation is outlined below:

Section 2 consists of the literature review. It will build knowledge as a foundation for understanding the problem of scalability. First a DLT system is characterized before discussing how basic protocols such as PoW and Nakamoto consensus are needed to ensure Byzantine Fault Tolerance. It is then discussed how this ensures system security while still achieving consensus. Based on this understanding, the DLT tri-lemma (security, decentralisation, scalability) is introduced and explored. It then explores ethereum. The current network implementation is discussed , before examining the various scalability proposals outlined in Serenity (ethereum 2.0). Finally it explores EOS. The delegated proof of stake (DPoS) protocol is described and shown to be scalable.

Section 3 defines and discusses decentralisation. The decentralisation dynamic in the DPoS protocol (used in EOS) is critically analysed using various approaches to determine its consequences. It shows that EOS shows strong signs of centralisation, particularly with regards to voting power and a non-diverse pool of BP candidates.

Section 4 uses unsupervised machine learning techniques to better understand the extent to which collusion is occurring in practice in the EOS. Partitioning and hierarchical clustering methods are both used.

Section 5 features proposals to improve the current DPoS system. First, certain voting modifications such as inverse vote weighting are explored. Next, a proposal for university participation in the DPoS protocol as a block producer is outlined. It motivates why participation could improve network decentralisation while also enhancing the students learning experience.

Section 6 deals will the more practical aspects of actually participating in the DPoS protocol as a block producer (BP). It investigates the requirements to participate as a BP in the EOS network and how best the university could create a compelling BP candidacy.

Section 7 concludes the report summarising the main findings and highlighting the academic contribution.

Section 8 suggests shortcomings in the report and future work necessary for the project to succeed.

2 LITERATURE REVIEW

2.1 Distributed Ledger Technology (DLT)

2.1.1 System characterization. To better understand this dissertation and protocols discussed, it is appropriate to define and outline some important concepts. A recent report by the Judge Business School, University of Cambridge [45], proposes a conceptual framework for distributed ledger technology systems to bring a degree of standardisation to DLT research. This dissertation will utilise the suggested framework and definitions from this report to support research standardisation.

Recall, DLT exists for the express purpose of creating a shared data structure. More formally defined, DLT is a system of electronic records that [45]:

- enables a network of independent participants to establish consensus around
- the authoritative ordering of cryptographically-validated[2] ('signed') transactions. These records are made
- persistent by replicating the data across multiple nodes, and
- tamper-evident by linking them by cryptographic hashes.
- The shared result of the reconciliation/consensus process - the 'ledger' - serves as the authoritative version for these records.

It is important to clearly define the following DLT terminology:

- The **ledger** is the shared result of the consensus process that serves as the authoritative version for the shared data structure.
- **Nodes**[3], are network participants that store a copy of the ledger, propose records to be added to the ledger, and check that proposed transactions and records are valid before propagating these on to the rest of the network.
- **Records**, often referred to as 'blocks', are groups of transactions that have been included in the ledger. Proposed records have yet to be added to the ledger. In most DLT architectures, records are added linearly in an append-only fashion leading to a chain of records (hence the popularized term 'blockchain'). Transactions contained in R_t were executed before R_{t+1} and after R_{t-1}.
- **Transactions**, are operations grouped in records and executed by nodes resulting in updates of the shared ledger.
- **State**, refers to a specific node's view of the ledger.
- **Consensus**, refers to the ideal scenario upon which all nodes agree on the state of the ledger.

2.1.2 Transaction execution life-cycle. Initially, it is assumed the system is in a state of consensus. As users interact with the system, they create certain transactions. Nodes group proposed transactions, creating a proposed record, and propagate this through the network. Once a new proposed record has been deemed valid and accepted by all nodes, the computational work involved with the transactions in the record, is then executed by each node resulting in a common update[4] of the shared ledger, and hence a newly reached state of consensus.

In short, DLT functions through:

(1) Users creating transactions.
(2) Nodes in the network deciding (in the absence of a central party) which transactions execute, and the order in which to execute them (via record proposal and acceptance).
(3) Nodes executing the agreed upon transactions (computational work) resulting in a common ledger update and newly reached state of consensus.

[2]Understanding the mathematics behind the cryptography is not needed to understand the issue of scaling, and hence beyond the scope of the dissertation. The "diffie hellman key exchange" and "trapdoor functions" are great topics to explore if interested in understanding how cryptography ensures transactions are valid. The following gold standard papers expand on these basics[12, 41, 56]

[3]For a more strict definition of node sub-classes one can refer to the following report[45]. In this context we simply assume that all nodes produce records, a class of nodes often termed as miners or validators.

[4]In DLT systems, it is a requirement that the computational work done is deterministic. This ensures that all nodes executing the same transactions will arrive at the same shared state of the ledger.

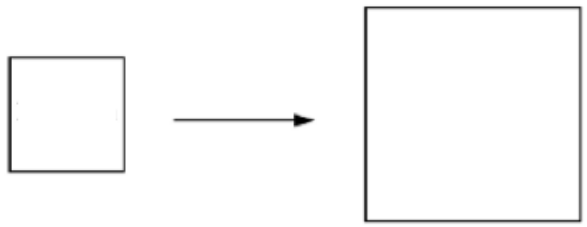


Fig. 1. Example of layer 1 scaling solution. Attempt to increase transactional throughput through modification to base protocols.

Scalability can now loosely be characterized as the rate at which the network can complete the above life-cycle of executing transactions, hence **transactional throughput**. A more formal criterion of transactional throughput (hence scalability), is that of measuring transactions per second (tps) that the network can process. One can see that transactional throughput is a function of the networks ability to perform steps 2 and 3.

Step 2, deciding the transactions to execute, is the previously described process of nodes proposing records. The transactional throughput is limited by the size of the record (how many transactions the records contains), as well as the rate at which records are proposed and accepted by the network[5].

The limitation in step 3 is more subtle. Each record that is accepted as a result of step 2, needs to have all the computational work involved with the record executed by every node in the system. This redundancy of every node having to process every transaction is part of what makes the network so safe. Consequently, the following fundamental limitation applies: transactional throughput is limited to the performance of the weakest node in the network[49].

At this point, a few high-level scalability solutions can be discussed:

A Increase the size of the proposed records to accommodate more transactions.
B Increase the rate at which records are proposed and accepted by the network (hence executing transactions more frequently).
C Currently every transaction is processed by every single node in the network. The network could be 'sharded' in such a way that only certain subsets of nodes process certain transactions.
D The underlying protocols remain as is, and a solution 'on-top' of the base protocol is designed to increase transactional throughput. This in effect amounts to us performing some computation 'off-chain'[49].

A, B and C are termed as layer 1 solutions. This is since they involve changes to underlying protocols in an effort to increase transactional throughput. The idea of A is explicitly shown in figure 1. D is portrayed in figure 2 and termed as a layer 2 solution. This instead involves utilizing innovations to perform work 'off-chain'. Layer 1 scaling solutions are explored in ethereum and EOS, and layer 2 solutions are also explored in the context of ethereum.

[5]Readers may be more familiar with the concept of block size and block proposal time.

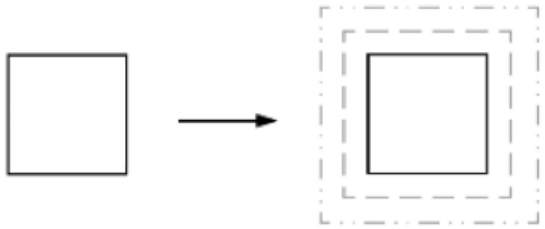


Fig. 2. Example of layer 2 scaling solution. Attempt to increase transactional throughput through solution 'on-top' of base layer. Essentially increasing throughput by performing work 'off-chain'.

Before further exploring these possible scaling ideas, it is important to note that DLT systems operate in 'adversarial environments'. This introduces various limitations that make the process of step 2, reaching consensus, a difficult task. In the next section, this issue is described in detail.

2.1.3 The Byzantine Generals Problem. Recall, distributed ledger technology allows a group of individuals to reach consensus on the state of a shared data structure without the need for a central party. Since no central authority exists, (absence of censorship), any individual can participate in the system. This simple fact that any individual can participate brings to light an immediate problem, that of malicious actors. Malicious actors aim to undermine the system by exploiting specific protocols or other loopholes often in an attempt to benefit themselves. Having to cope with malicious actors in the system is generally termed as operating in an 'adversarial environment'.

This fundamental concept of achieving consensus, while operating in an adversarial environment, is what sets DLT apart from traditional distributed database systems. Although now an area of focus more than ever, the problem of computer systems communicating in adversarial environments was conceptualized in 1982 and termed, 'The Byzantine Generals Problem'[35].

Addressing the Byzantine Generals Problem is one of the most difficult and significant challenges in DLT. Numerous nodes are required to continually communicate to establish consensus or agreement on the state of the shared data structure. A 'Byzantine node' is a node that can cheat and intentionally communicate malicious messages in order to prevent or alter system consensus. Given the openness of the system, Byzantine nodes can freely join the network. A 'Byzantine Fault Tolerance' mechanism is required to cope with malicious actors (Byzantine nodes) while still safely reaching consensus in DLT systems.

The fundamental reason why scalability is so hard to achieve in DLT, is that systems need to incorporate some Byzantine Fault Tolerance mechanism to ensure the security of the network. The incorporation of this Byzantine Fault Tolerance mechanism often constrains performance. In the next section when examining Proof of Work (PoW), it is shown how the PoW protocol, necessary to ensure Byzantine Fault Tolerance, constrains the transactional throughput of the system.

It is a fascinating problem and active area of research to understand how DLT systems can be scaled while still ensuring Byzantine Fault Tolerance, i.e. system security. The project 'Algorand' released a paper on scaling DLT systems through 'Byzantine Agreements'. This approach involved using Verifiable Random Functions (VRF) in a non-interactive way to anonymously select nodes to

participate in 'Byzantine Agreement' hence establishing consensus [25]. Other Byzantine Fault Tolerant scaling ideas suggested by ethereum and EOS are examined in section 3 and 4.

2.1.4 Proof of Work (PoW) and Nakamoto consensus. This section now demonstrates a basic mechanism used to achieve a state of consensus amongst a set of nodes operating in an adversarial environment.

It is first appropriate to point out two prominent DLT architectures, permissionless and permissioned, each dictating differing solutions to the above problem.

- **Permissionless** architectures allow any node the ability to propose records.
- **Permissioned** architectures allow only a restricted subset of nodes to propose records.

The debate on which characterization is best, is unclear, and often a matter of opinion. Through exploring both ethereum (permissionless) and EOS (permissioned), this dissertation highlights varying trade-off's represented by this architectural decision.

Since the problem of achieving consensus is significantly more difficult in a permissionless environment, first described is PoW and Nakamoto consensus[6], protocols suitable for this architecture. PoW and Nakamoto consensus is used to provide a Byzantine Fault Tolerant consensus mechanism. Understanding their necessity is the basis for understanding the fundamental limitations of scalability introduced by operating in an adversarial environment.

The Proof of Work (PoW) protocol is designed to make it difficult for a node to propose a valid record[40]. Proposing a valid record is difficult, not due to checking the validity of transactions contained in the record, but as nodes are required complete arbitrary computational work in order for the proposed record to be seen as valid by the network.

This work is in the form of computing a hashing function:

$$hash(R) \approx hash(\text{previous record hash}|\text{merkle root}^{7}|nonce^{8})$$

Where the resultant hash requires a certain number of leading zero's for the record to be deemed valid by the network. The difficulty of this task for a record, R, is quantified as follows:

$$d(R) \approx 2^{\text{number of leading zeros of hash(R)}}$$

Since the function *hash* is random[9], it simply requires brute force computational work to produce records that are actually valid, especially as the difficulty increases[59]. Importantly, it is fast for other nodes to verify the validity of the record once a valid record is found.

Nakamoto consensus is merely an agreement between nodes to build upon the chain (ledger), containing the greatest total difficulty (difficulty defined as above)[40]. This rule can be more informally stated as "build on the current longest chain of records". This is necessary in the case of forks where nodes propose equally valid records, temporarily forking the chain.

[6]These mechanisms were famous through there practical implementation in bitcoin.
[7]The transactions are stored in a data structure called a merkle tree. The merkle root therefore provides a unqiue hash of all the contained transactional data.
[8]Cryptographic one time primitve. This is the arbitrary number that is actually iterated through by the record proposer/miner as they seek to find a hash with a suitable number of leading zeros.
[9]Pseudo-random to be more precise.

Although many variants of system attacks exist, in a permissionless architecture, PoW is designed to prevent the following two attacks primarily:

(1) Double spend attacks
(2) Sybil attacks

Double spend attacks occur when a participant creates two transactions, t_{x_a} and t_{x_b}, where both transactions point toward the same source of funds, such that each transaction by itself is valid, yet both cannot be included in the ledger as they 'double-spend' the same funds[36].

Sybil attacks are a result of nodes being able to join and leave the network easily. In a Sybil attack, a malicious actor creates a large number of Byzantine nodes and attempts to gain control of the network using these nodes which account for a large fraction of the total network [58].

PoW effectively mitigates both these attacks since it makes producing valid records 'difficult', i.e. computationally expensive. Therefore, successfully performing one of the above attacks becomes very expensive i.e. economically infeasible to any rational attacker.

For example, a Sybil attack would fail for the following reason. Joining the network with a large number of nodes would not gain control of the network even though these nodes may account for a large fraction of the total network participants. This is since it is not the number of nodes that dictate system control, but rather the hashing power of a node that dictates system control. This is since hashing power corresponds to the rate at which valid records can be found in the system, rather than merely controlling a large number of Byzantine nodes.

Therefore, in a more technical sense we would consider PoW not so much as a consensus mechanism, but as a mechanism to prevent against Sybil and double spend attacks[59], hence providing Byzantine Fault Tolerance. The consensus mechanism in this case (Nakamoto consensus), is just the decision to always append records to chain with most work done (containing the greatest total difficulty).

Although PoW ensures a safe system that is Byzantine Fault Tolerant, it fundamentally limits transactional throughput. The process of finding a valid record, and hence executing a group of transactions, becomes a difficult task where nodes compete to do this. Therefore, the rate at which valid records are found and executed in the system ($\approx$ transactional throughput), is restricted by the difficulty of finding a valid record in PoW.

2.1.5 Permissioned architectures. Permissioned architectures are less complex than permissionless architectures because record proposal is restricted to a subset of pre-vetted nodes. By design, Sybil attacks fail, since arbitrary nodes cannot simply produce records. Restricted record proposal also mitigates double spend attacks as a conflicting transaction t_{x_b}, would not be included by a record producing node after the initial transaction t_{x_a} has already been accepted and included in the ledger[10].

Although permissioned architectures do not need a Sybil resistance scheme such as PoW, they still require some consensus mechanism to elect which of the pre-vetted nodes should propose

[10]This assumes honest behaviour by the group of pre-vetted nodes. Later the dissertation explores how permissioned systems achieve this.

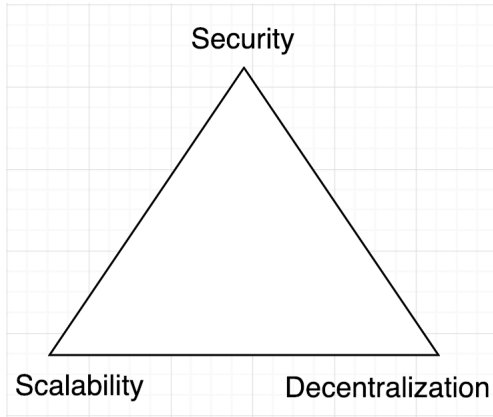


Fig. 3. DLT tri-lemma: security, scalability, decentralisation. Refers to that fact that it is 'very difficult' to create a DLT system with all three of these core tenets.

records. The simplest mechanism is a round-robin cycle, where nodes, turn by turn, produce records. Other algorithms, such as Raft or Paxos all temporarily elect a node to produce records[45]. Note that mechanisms are required for the election of the pre-vetted nodes, and to ensure that these nodes behave honestly. Sections 4 and 5 discuss this in more detail.

The following major differences between permissionless and permissioned architectures are again emphasised. Permissionless environments are open, and any node can compete to propose records. Permissioned environment have a restricted subset of nodes working together to create records. Therefore, permissioned environments show greater efficiency (and scalability). Furthermore, permissioned environments show greater centralisation as only a restricted subset of nodes propose records. The implication is trusting that these nodes perform their work honestly. In reality, collusion between these record producing nodes could result in network censorship; something DLT was expressly designed to protect against.

As with most architectures, trade-off's exist and the above trade-off is very apparent. It is essential to understand which trade-off variants may perform best. Next, the 'DLT tri-lemma' is more formally discussed.

2.1.6 DLT tri-lemma. The tri-lemma, shown in figure 3, has been cited in numerous academic and peer-reviewed research papers[18, 19, 33, 46, 57, 60]. The tri-lemma states that it is not impossible, but rather 'very difficult' to create a system with all three of the following properties: security, scalability, and decentralisation.

Without the system being secure, the value of the network would cease to exist. Therefore, security is a given choice in the tri-lemma. This choice leads to the trade-off between decentralisation and scalability. More explicitly, the choice of a scalable network that is centralised, or a decentralised network unable to scale. The following earlier mentioned examples depict this:

(1) **Visa**: A completely centralised payments system able to scale and handle 47 000 tps[52].
(2) **Bitcoin**: A decentralised payments solution processing 7 tps[48].

This trade-off leads to arguably the biggest and most obvious question in DLT right now: can a system be designed to be secure, decentralised and scalable?

It is apparent that further exploration of the tri-lemma is needed to understand how best to allow sufficient scalability and decentralisation while still ensuring the system security.

2.1.7 Platforms analysed: ethereum and EOS. This dissertation particularly examines the EOS and ethereum platforms. As mentioned earlier, entrants to the DLT market are not likely to build their own DLT systems from scratch, due to the significant overhead (time, cost and complexity). Instead they would use existing platforms, like EOS or ethereum, that have a demonstrated track record of security and functionality. Therefore, it is appropriate to investigate the scalability of these platforms, as it allows a better sense of understanding as for when to expect a DLT system capable of handling mass user adoption. This being necessary for the deployment of financial service applications using DLT, and requiring a significant user base or transactional volume.

Ethereum and EOS are examined as they are the two largest DLT platforms by market capitalisation, and they represent fundamentally different views on scalability. While ethereum is permissionless, EOS is a permissioned architecture. These different approaches allow broad and diverse insights into the current state of DLT scalability solutions. To keep the scope of the dissertation as narrow as possible, it is unfortunate that many other platforms posing interesting scaling solutions were not examined. Section 10, future work and shortcomings, elaborates on this point.

2.2 Ethereum

2.2.1 Overview. The initial aim of the ethereum platform was to create a decentralised network with a Turing complete virtual machine[30]. Unlike previous decentralised networks (bitcoin) which only allowed for limited actions through basic scripting, a Turing complete virtual machine can solve any reasonable computational task. The result was the first network developers could use to create arbitrary decentralized applications[55]. Ethereum extended the realm of use-cases for DLT beyond just payments.

Although currently still using a PoW protocol, ethereum has a clear road-map of improvements to ensure scalability of the system. Before analysing these ideas in later sections, recall the following two solutions suggested to increase transactional throughput:

A Increasing the size of the proposed records to accommodate more transactions.
B Increasing the rate at which records are proposed and accepted by the network (effectively executing groups of transactions more frequently)

In case A, if the size of the record were to increase 10x, transactional throughput would hypothetically increase 10x as each record would now contain $\approx$ 10x the amount of transactions[11]. Recall that in open DLT systems such as ethereum, every node in the network has to execute all the computation in each record independently. Therefore, each node would be required to compute 10x the amount of work for every record processed. This requirement means nodes would need to be much more powerful, hence increasing the barrier of entry to the network. The consequent effect is that weak nodes would drop out of the network, making the network consist of a smaller set of powerful nodes, and leaving the network more centralised[49]. An interesting question is the number of nodes, or hashing power distribution required to keep the network sufficiently

[11] The nuances around this proposal are outlined in more detail in a 2016 paper from Cornell University termed, Bitcoin-NG: A Scalable Blockchain Protocol[23]

decentralised. This point is discussed further in section 5.

Case B involves increasing the rate at which records are proposed, accepted and executed in the network. In a PoW system, this rate is kept relatively constant through adjusting record difficulty periodically based on the observed hashing power of the network. The major issues with increasing the rate at which records are added are security and orphaned records. A 2015 publication in the International Conference on Financial Cryptography and Data Security details how increasing the record rate reduces system security against double-spend attacks[48]. The publication also addresses these security concerns through the GHOST protocol, a protocol implemented by ethereum currently[48]. This improvement has allowed ethereum to reduce record production time to 14 seconds intervals, as opposed to the 10 minute intervals present in the bitcoin network.

Orphaned records refer to the case where two valid records are found in the network at the same time, leading to a temporary fork in the chain. When the next valid record gets proposed and appended to one fork of the chain, the other record will be 'orphaned', i.e. all transactions contained in the record will return to the unconfirmed transaction pool and wait to be executed by the network at a later stage. This significant network inefficiency represents a waste of work.

The takeaway from the above is that serious thought, research, and engineering is required to develop solutions that allow decentralised and scalable systems. It is clear that PoW, the current Sybil resistance scheme ensuring Byzantine Fault Tolerant consensus, is not sufficiently scalable. For permissionless architectures like ethereum, it is important to consider what mechanisms can replace PoW, improving efficiency whilst maintaining system security. Permissioned architectures may also provide an answer to scalability. Section 4, EOS, explores this possibility.

2.2.2 Serenity (ethereum 2.0). Serenity is the name given to the envisaged ethereum 2.0 platform incorporating many major protocol changes to achieve the goal of scalability. Casper, Sharding, State channels, and Plasma are all examples of the multi-pronged approach to scale ethereum while maintaining security and decentralisation. The following sections analyse these ideas to understand their limitations and likelihood of success better.

2.2.3 Casper - Proof of Stake (PoS). The idea behind PoS is that record proposers 'stake' the native crypto asset (eth), with the size of the stake proportional to the chance of being selected to produce the next record in the system. Staking refers to the fact that the amount of the asset staked, is locked up in a security deposit, and held as collateral. If a node were to propose records maliciously, it would lose its stake, therefore incentivising honest behaviour[11]. The other major economic incentive underlying this protocol is that participants with greater stake will be inclined to produce valid records, as the value of their staked asset is largely dependant on the integrity of the system.

PoS exists in many variants. Originally Vitallik Buterin and Virgil Griffith authored the paper Casper FFG (Friendly Finality Gadget)[16], recommending this variant for the etherem network. Casper FFG introduces PoS while ensuring Byzantine Fault Tolerance and consistent finality[16]. Finality is the assurance that past transactions can never change, and can, therefore, be considered 'final'. Casper FFG sits above a record proposal mechanism (such as PoW) and ensures finality in the ledger. In practice, it was suggested that while other records would still be proposed through PoW, every 50th record in the ethereum chain will be a PoS checkpoint, ensuring consistent finality. This checkpoint is particularly important in e-commerce where finality of payments is required for

vendors to safely process orders.

A major advantage of the Casper FFG protocol, and other PoS variants is that less electricity is required to run the network when compared to PoW protocols. There are also stronger finality assumptions (i.e. stronger security), although finality can typically never be absolutely guaranteed[16]. The main value-add of the Casper FFG protocol is that it is better suited[12] to the scaling solution, sharding. The original specification of Casper FFG is currently under modification in order to tailor its compatibility with sharding, the basis of Serenity.

On the 5th of November 2018, lead ethereum researcher Vlad Zamfir along with three colleges released a draft paper titled 'Introducing the "Minimal correct-by-construction (CBC) Casper" Family of Consensus Protocols'[5]. This comprehensive research paper outlines numerous PoS variants all offering Byzantine Fault Tolerant consensus along with other benefits[5]. The following variants of Casper are all defined:

- Casper the Friendly Binary Consensus
- Casper the Friendly Integer Consensus
- Casper the Friendly GHOST
- Casper the Friendly CBC Finality Gadget
- Casper the Friendly CBC Sharded Blockchain

Further details regarding Casper are beyond the scope of this dissertation. The following papers outline Casper in more detail [5, 16].

2.2.4 Sharding. Recall that in DLT systems every node in the network needs to process every transaction, creating a fundamental limitation. We either need to increase the workload of each node (forcing weak nodes out the system and risking centralisation), or somehow devise a scheme such that every node is not required to process every transaction. Sharding is the technique used to implement the latter, where groups of nodes are responsible for validating certain sub-chains (i.e. shards of the main chain), periodically referencing back to the main chain.

Figure 4 visually illustrates the system implementation. The main chain will provide a point of reference and hold the secure management contracts (SMC) that accept ethereum deposits as stake in response to nodes participating in the system. The beacon chain will most importantly provide a continuous 'heartbeat' of random numbers that are consistent across the network (since determinism is required in distributed networks)[4]. Although used for various things, the numbers generated from the beacon chain will be used to effectively assign participating nodes to differing roles, such as record producing or attestation, in differing shards. Because a node will now only be required to complete the work in a single shard, this allows simple consumer grade nodes to participate, promoting the core tenet of decentralisation[9]. Since many shards can now exist in the network, there is theoretically infinite horizontal scalability, as new shards can be created dynamically to handle increased transaction activity.

Sharding is an active area of research with many complex problems needing solutions before a working implementation is realised. Given the complexity involved in sharding, a successful sharding implementation will probably only be expected in the next 2-5 years.

2.2.5 EVM replacement. In many cases ethereum is described as the largest decentralised computing network or supercomputer. This is since all transactions are executed by every node in the

[12] Although PoW is not impossible in sharded systems, PoS presents substantially more synergies.

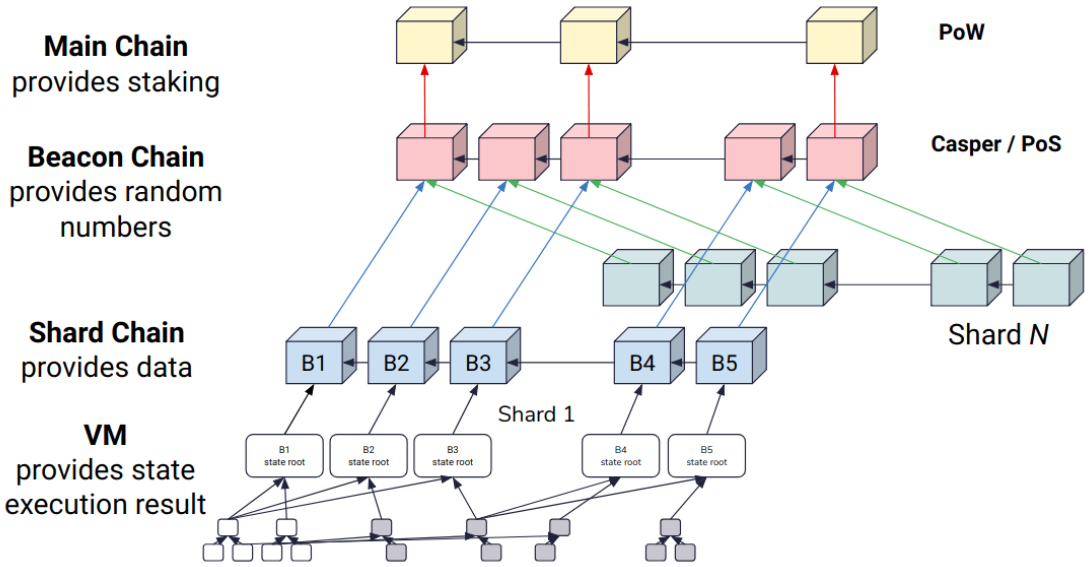


Fig. 4. Various layers present in the sharding proposal[4]. Sharding is a mechanism such that every node is not required to process every transaction. Groups of nodes are responsible for validating certain sub-chains (i.e. shards of the main chain), periodically referencing back to the Beacon chain and main chain.

network using the ethereum virtual machine (EVM). Transactions in this case can effectively be any payment, code or logic since the EVM is Turing-complete. Already noted is the fact that every node in the network is required to execute every transaction using the EVM. Therefore, to a great extent the speed of the network is reliant on the efficiency at which the EVM executes transactions.

The EVM's function is to execute byte-code that has been complied from high level smart contract languages like Solidity and Vyper. As it stands, the EVM executes transactions in a sequential manner as it is single threaded. The ethereum 2.0 roadmap outlines a successor for the EVM, eWASM (ethereum Web Assembly), a deterministic version of Web Assembly (WASM) [29]. The objective is to introduce a more efficient and multi threaded execution engine that will allow nodes to execute transactions at a faster rate, while also taking advantage of the new POS and sharding structure. The faster execution of transactions correlates to nodes being able to handle and execute greater workloads faster, leading to an increase in overall network activity.

2.2.6 Layer 2 solutions. Casper, Sharding and improvements to the underlying execution engine of ethreum were examples of 'layer 1' scaling solutions, where changes in underlying protocols and infrastructure are used to scale the base capacity of the system. Layer 2 solutions instead do not change underlying protocols, but rather examine what scaling solutions could be built 'on-top' of base protocols using the ethereum virtual machine (EVM) environment. This approach shifts more work 'off-chain', reducing 'on-chain' computation, and therefore creating capacity for further 'on-chain' work, hence greater scalability. Since layer 2 solutions are built 'on-top' of base protocols, they are relatively modular solutions that can easily integrate with a variety of base protocols (like Casper), allowing multiple ways to tackle the issue of scalability.

Layer 2 solutions are possible according to the following logic. First, numerous components make DLT possible: economic incentives, cryptography and networking theory. All of these are used to align stakeholders incentives and ensure that the network can achieve consensus regarding the state of the shared data structure. By achieving this state of consensus, a kernel of truth exists that the network can refer to, and take advantage of by creating another layer of economic mechanisms referring back to this kernel of truth when necessary[49]. Through programming the logic of these additional economic mechanisms and publishing these on-chain as modular mechanisms, there now exists a layer 2 solution where network interactions do not necessarily need to use the main-chain, but merely refer to it as a kernel of truth. Next, two examples of 'layer 2 solutions', state channels and plasma, are described.

2.2.7 State Channels. State channels are:

- Communication channels set up between two users in the network such that;
- the two users in the network can interact and perform state updates through signing and sending transactions between one another;
- effectively allowing updates to occur 'off-chain'.
- A smart contract on the main-chain manages the logic of the process;
- where in the case of a dispute between the users, the contract existing on the main-chain (kernel of truth), can be referred to.

Although many use-cases exist for state channels, one prominent example is that of payment channels. This is best illustrated through an example. Next described is a scenario where two fictitious banks, Alpha and Beta, constantly have settlements occurring between them.

Imagine Alpha needs to send a 2 eth payment to Beta. Typically, this transaction would be created and processed on-chain. The next day, Beta needs to send 5 eth to Alpha, and again this transaction is processed on-chain. With a payment channel, Alpha and Beta would set up an initial smart contract in which they, for argument's sake, each contributed 5 eth (this is important as collateral). The contract would also contain the desired rules of their business transactions, and act as a kernel of truth for disputes.

Now that this initial contract is set up using an on-chain transaction, Alpha and Beta can begin to effectively transact off-chain. Now if Alpha needs to send a 2 eth payment to Beta, Alpha will sign this transaction and send it directly to Beta. The next day Beta needs to send 5 eth to Alpha, Beta signs this transaction and sends it directly to Alpha. The state has now been updated, i.e. net-net Alpha has received 3 eth, yet no transactions to the chain have been required.

Imagine now Alpha and Beta have finished settling and want to close the channel, the latest transaction sent from Beta to Alpha, is now submitted by Alpha to the main-chain contract. After a set resolution period, Alpha will now receive 8 eth from the contract, and Beta will receive 2 eth (remember each party initially contributed 5 eth).

The resolution period exists in the case that one party tries to cheat. After Beta sent the signed 5 eth transaction direct to Alpha, Beta could then try submit the earlier 2 eth signed transaction received from Alpha to the main-chain, so that the naive contract would erroneously payout Beta 7 eth and Alpha only 3 eth. The resolution period exists, so Alpha has sufficient time to submit her version of signed transactions to the smart contract, with the contract facilitating resolution dispute and honouring to the full transaction history.

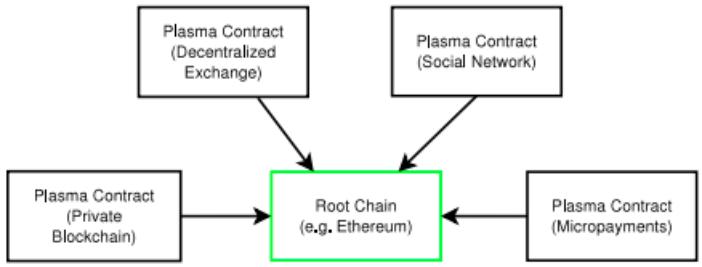


Fig. 5. Plasma chain [43]. Illustrates the concept that business use-cases may exist as plasma child-chains

The following list highlights the benefits of payment channels:

- **Scalability** by shifting more work off-chain. This reduces on-chain transactions, effectively allowing more to be done with the same underlying protocols.
- **Instant finality** is achieved, as once a transaction is signed and sent, the receiving party can consider it final as this can now be submitted to the main-chain in the case of conflict dispute.
- **Anonymity**. Since transactions occur off-chain, they are no longer visible to the entire network. Furthermore, as channels start to connect multiple parties, intricate webs of transactions between parties provide a strong degree of anonymity to users.
- **Low cost**. Since most of the work is occurring off-chain where gas fees are not required, these channels are a cheaper form of interaction.

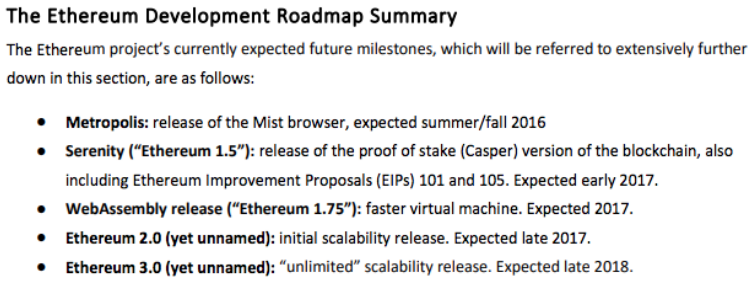
Although state channels sound somewhat like a silver bullet, significant challenges are still present. No fully functional commercial implementation of state channels currently exists[13] on the ethereum network, due to complexity of the solution. More information about state channels can be found here[21, 22].

2.2.8 Plasma. Joseph Poon and Vitalik Buterin released a paper titled Plasma: Scalable Autonomous Smart contracts in 2017[43], detailing this potential layer 2 scaling solution. Core to the idea of Plasma, is that instead of just creating state channels pegged to the main chain, plasma 'child-chains' can be created and pegged to the main chain. These plasma child-chains are cheaper to use as the computation performed in the child-chain will not need to be performed by the entire network, in contrast to regular transactions[49]. Like state-channels, plasma enables scalability in ethereum through reducing on-chain computation by instead performing this in separate plasma child-chains.

Figure 5 is taken from the Plasma proposal paper illustrating the concept that business use-cases may exist as plasma child-chains. Further details regarding plasma are beyond the scope of this report.

2.2.9 Scalability timeline. It is clear that the current realisation of the system, ethereum 1.0, does not achieve the long term vision of scalability. Accordingly, major research groups have been funded in a strong initiative to reach the scalability goals set forth by the ethereum 2.0 roadmap. Figure 6, a 2016 roadmap for ethereum shows how the timeline for many key updates has lapsed without a working implementation. There is significant risk that complexities associated with these solutions, sharding, in particular, will further delay key milestones and create doubt around the

[13] Although projects such as Raiden are making great progress, commercially functioning implementations are not yet available.



The Ethereum Development Roadmap Summary

The Ethereum project's currently expected future milestones, which will be referred to extensively further down in this section, are as follows:

- **Metropolis:** release of the Mist browser, expected summer/fall 2016
- **Serenity ("Ethereum 1.5"):** release of the proof of stake (Casper) version of the blockchain, also including Ethereum Improvement Proposals (EIPs) 101 and 105. Expected early 2017.
- **WebAssembly release ("Ethereum 1.75"):** faster virtual machine. Expected 2017.
- **Ethereum 2.0 (yet unnamed):** initial scalability release. Expected late 2017.
- **Ethereum 3.0 (yet unnamed):** "unlimited" scalability release. Expected late 2018.

Fig. 6. The (now lapsed) ethereum roadmap outlined in 2016 by Vitalik Buterin [15]. This creates cause for concern regarding the timely delivery of future milestones.

timely delivery of Serenity. The latest timeline indicates the roll out of Serenity will happen in three separate phases expected to be completed in 2022. Phase 0 will include the beacon chain (discussed earlier), phase 1 will include sharding and phase 2 will introduce the new execution engine (eWASM).

Ethereum has typically been seen as the go to DApp developement platform. It presented the first functional and robust environment where developers could easily deploy customised DApps. Since then, many different DLT platforms have emerged, all aiming to better existing offerings and provide a unique value proposition, most often, scalability.

The ethereum ideal of perfect security, decentralisation and scalability is one that is beginning to be challenged. Instead, other platforms are viewing the tri-lemma through a different lens. While security remains the overarching tenet, in order to scale now and hit the consumer market, there may be an acceptable threshold of decentralisation that can be sacrificed. This logic surmounts to delivering secure and scalable technology now, while rather working on ways to improve system decentralisation over time. The effect of this is to draw developers to this new technology that can meet the required scalability demands of businesses **now** - rather than in a circumstantial 2-5 year window.

2.3 EOS

2.3.1 Overview. 'eosio' is a powerful infrastructure for smart contracts and decentralised applications, with 'EOS' being the native token used within the system. For simplicity, the platform is referred to as 'EOS' throughout the dissertation. The platform was developed by block.one capital who raised 4 billion $US over a year-long Initial Coin Offering (ICO) to develop the software and grow the ecosystem. The EOS main-net has been live since June 2018 and continues to grow and change. The analysis in this section is primarily concerned with how the DPoS protocol enables scalability in EOS. Section 5 will analyse the decentralisation dynamic in EOS as a result of the DPoS protocol.

2.3.2 Delegated Proof of Stake (DPoS) . While the cryptography underpinning various DLT systems remains largely the same, EOS makes use of a protocol known as delegated proof of stake (DPoS) as opposed to more well known PoW protocol. DPoS is relatively similar to PoS - it also uses the crypoteconomic incentive, staking, to encourage desirable system behaviour. In DPoS, staked resources give you the ability to vote for the nodes you believe should produce records, with

your stake proportional to the weight of your vote. Continuous voting results in 21 nodes being selected to actively produce records. The current protocol is now outlined in more detail before discussing how it ensures scalability.

All users holding EOS can stake their EOS, i.e. lock their EOS into a secure smart contract, with the resulting stake[14] allowing them to a vote for block producers[15]. For every 1 EOS staked, 1 vote can be cast for between 1 to 30 different block producing candidates[16]. Since the user holds EOS, they have at the very minimum, a financial interest in the system. Based on this assumption, the user should be inclined to vote for block producing candidates that add value to EOS ecosystem, hence adding value to their investment in the native EOS token. To ensure the above economic incentive mechanism functions as intended, users are required to stake their EOS to vote, a more stringent requirement than simply holding EOS. By staking EOS, tokens are locked in a secure smart contract and upon request, take 3 days to 'unstake', rendering them illiquid. This mechanism ensures that only users who have a vested interest in the system, i.e. illiquid EOS, determine the block producers. The above action purposely excludes traders and short term speculators from voting, actors without a vested interest in the system.

Any node is allowed to join the system and register as a block producer (BP) candidate. In this sense, the system is still open. BPs are incentivised to partake in the system as part of the inflation in the EOS token supply is allocated as a reward for this service. Even if a BP candidate is not part of the top 21 active producers, they may be still be rewarded as a standby producer if they receive sufficient votes. At the time of writing, the top 80 BP candidates were all actively receiving rewards[8]. This encourages more entrants into the system, hence greater competition, something desirable for decentralisation. It further acts as a critical redundancy, since BPs in the top 21 that unexpectedly fail or shut-down, can almost instantly be replaced by standby producers.

It is the job of the BP candidate to provide a compelling narrative to the EOS community as to the value they will add to the EOS ecosystem, and hence why they should be voted for by EOS token holders. More so, since financial rewards are consistently received, BPs need to show that rewards received are not solely for profiteering, but also being reinvested back into projects that are growing and adding value to the entire ecosystem.

To encourage EOS token holders to continuously assess the climate of current BPs and actively change their vote to those best performing, the mechanism of vote decay was introduced. Vote decay implies that the weight or strength of your vote decreases over time from the point at which you cast your vote. The weight of your vote decreases to 50% its original strength over 1 year, and 0% over 2 years[3]. This combats 'set it and forget it' behaviour and encourages all participants to recast their votes constantly. Votes can be recast as frequently as desired. With the re-election of candidates happening every 63 seconds, the system is extremely dynamic and can swiftly respond to vote out BPs acting maliciously[3].

The DPoS protocol essentially ensures:

- **users** with a **vested interest** in the system;

[14]Note: Staking also gives users right to a proportionate amount of network resources to use for computation.
[15]Note: we now use the term block producers, as opposed to the standardized term record producers used throughout the report. This is consistent with terminology in the EOS whitepaper.
[16]i.e. 1 EOS can be used for 1 vote to candidate x_1, or 1 vote to candidate x_1, x_2, ..., x_{30}. This mechanism encourages a diverse vote distribution, as opposed to simply 'backing one horse'

- **continuously elect** a set of 21 **service providers**;
- where each service provider constantly needs to demonstrate their **value addition** to retain their votes.

This positive feedback loop stimulates competition between BPs to grow the value of the community.

2.3.3 Why DPoS is scalable. Only a restricted subset of pre-vetted nodes, 21 to be precise, can actively produce records. Recall that this is an example of a permissioned system since record production is restricted. By design, the network no longer needs to introduce a mechanism, such as PoW, to actively guard against Sybil and double spend attacks. This is because arbitrary Byzantine nodes cannot just join the network in an attempt to control the majority of the network or produce malicious records. Therefore, records can be produced in a coordinated round-robin fashion by the 21 BPs, as opposed to competing to produce a valid record. The result is a less energy intensive process and one in which records can be produced at a much faster rate. Each of the BPs takes a turn to produce 6 consecutive records, over a 3 second period, hence a record is produced every 0.5 seconds. Once a record is produced, a super majority of nodes (15 out of 21) are required to verify and sign the record, deeming it irreversible[3].

Since coordination rather than competition exists in record production, temporary forks are less of an issue[17] and the record production rate can increase to the noted 0.5 second time-frame. This is a sizeable improvement over bitcoin's 10 minute and ethereum's 14 second rate of record production (granted ethereum is planning to change their PoW protocol).

Recall that in DLT systems every node in the network needs to process every transaction, creating a fundamental bottleneck. Either the workload of each node needs to increase (the rate and amount of work being done), or every node should not be required to process every transaction in the network (sharding). While ethereum takes the latter option, EOS instead opts for the earlier option. The 21 nodes running the network are required to be 'super-nodes' with state of the art hardware capable of handling high transactional throughput.

Consequently, EOS has ensured system scalability by increasing the rate at which records are produced and increasing the computational ability of the nodes in the system (i.e. each record is 'bigger' allowing more work and greater transactional throughput). This innovation has allowed EOS to achieve 4000 tps [3].

Perhaps a better measure of the scalability of a platform is the concept of activity. Transactions per second (tps) is a rather crude measurement as it does not take the complexity of the transaction into account. Given these systems have a rich Turing complete virtual machine, transactions could be just one, or several complex operations. Activity measures the number of operations as opposed to transactions, a more accurate criterion in evaluating the processing capability of the system. Figure 7 shows the relevant activity statistics. It is easily seen that EOS has the greatest activity, processing nearly 50x the total activity of the ethereum network.

The trade-off associated with EOS is obvious. By substantially increasing the hardware requirements for a node to participate in the system, weaker nodes cannot compete, and hence only a select elite of super-nodes can become BPs. Decentralisation is traded for scalability. This loss of

[17] Dan Larimer highlights why forking is not an issue with DPoS in the following resource[1]

			Activity			Value	Index	
#		Name	Activity	Average (7d)	Record	Market Cap	AVI	CUI
1		EOS	41,257,770 Op	42,039,819 Op	74,568,958 Op	$ 2.5 B	4,753	
2		TLOS	29,917,513 Op	15,160,794 Op	32,217,207 Op	$ 0.016 B	546,297	
3		STEEM	996,699 Op	996,699 Op	2,522,380 Op	$ 0.047 B	6,106	
4		TFD	990,221 Op	334,675 Op	990,221 Op	$ 0.003 B	112,485	
5		KIN	871,516 Op	1,042,644 Op	5,258,216 Op	$ 0.005 B	51,403	
6		TRX	797,102 Op	886,469 Op	5,306,869 Op	$ 0.950 B	239	
7		BSV	638,225 Op	635,064 Op	900,436 Op	$ 1.7 B	105	
8		ETH	615,577 Op	451,255 Op	1,372,918 Op	$ 16 B	11	
9		IOST	506,862 Op	397,699 Op	874,225 Op	$ 0.064 B	2,249	
10		BTC	461,882 Op	461,315 Op	1,178,080 Op	$ 132 B	1.0	

Fig. 7. Activity measurement of top 10 DLT systems. Activity is the defined as the number of operations performed by the system over a 24 hour period. CUI - (Capacity Utilization Index) is the ratio of the systems daily activity to total systems capacity. AVI (Activity Valuation Index) is the ratio of the systems activity to market cap valuation[10].

decentralisation is undesirable as the express purpose of DLT was to create a system that was decentralised, and hence censorship resistant among other things. Section 5 presents an in-depth analysis of the decentralisation dynamic in EOS, in order to better understand the ramifications of this trade-off.

3 EOS DECENTRALISATION

The following section clearly defines the concept of decentralisation before examining the decentralisation dynamic existing in EOS. It first uses an EOS voting data set to understand the block producer dynamics in EOS at a high level. It then presents a comparative approach comparing EOS to ethereum. Finally, an economic approach assesses the extent of competition between the pool of block producers.

3.1 Decentralisation definition

The idea of decentralisation is major point of debate and often misunderstood. Presented from a philosophical standpoint, many believe that decentralisation is a binary ideal, shifting the power dynamic from major corporations and governments to individuals. Decentralisation is more accurately defined in the *DLT systems conceptual framework report* released by *the University of Cambridge*, as a system designed to "allow free and **open participation** and encourage vibrant

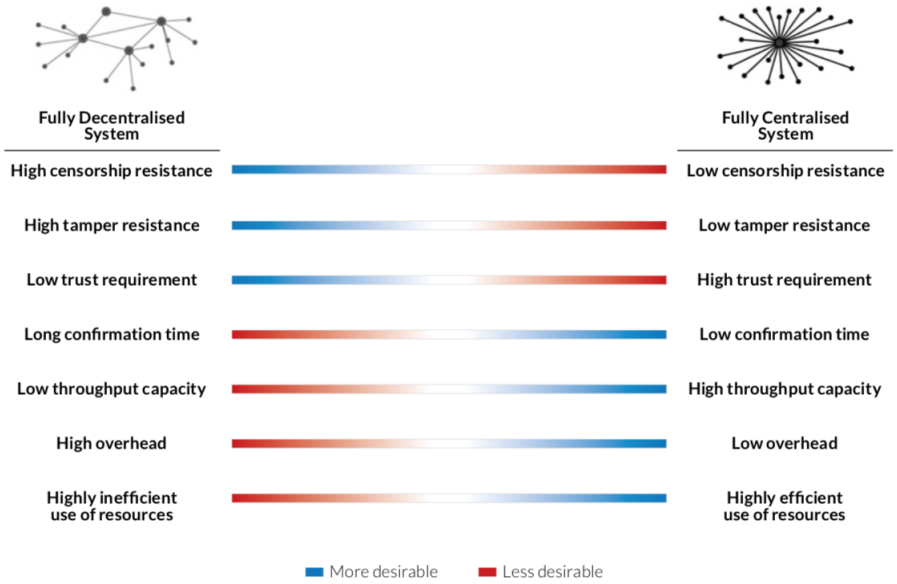


Fig. 8. Centralised vs decentralised systems. This figure shows the wide array of trade-off's present in decentralised systems. It also shows how decentralisation is not binary, but exists more as a continuous scale over a number of important system aspects[45].

debate, rather than relegating decision-making or system management to a fixed set of entities" [45].

With this definition in mind, note that decentralisation may exist at differing levels in a system, to various extents. For example, a system's ledger may be open for any node to validate transactions (more decentralised), yet record proposal may be restricted to nodes with a minimum stake in the system (more centralised).

Accordingly, it is beneficial to examine the various set of trade-off's experienced between more and less decentralised systems. This helps to better highlight system efficiencies gained, and what ideals may be lost, through creating a system that is more centralised. Figure 8 shows some of the most prominent trade-off's experienced in fully decentralised and centralised systems.

Following from figure 8, an example of a fully decentralized[18] system would be bitcoin. It possesses the desirable characteristics of minimal trust requirements and censorship resistance, albeit at the cost of a low throughput capacity (7 tps). Visa, a fully centralised system, represents a single central party responsible for all transactional processing. Therefore, its transactional throughput is only limited by the processing constraints of its state-of-the-art hardware and network latency. Since it is not operating in an open adversarial environment, no intricate protocols (possibly limiting throughput) are required to ensure Byzantine Fault Tolerance. The clear trade-off, in this case, is the high trust requirement of the Visa corporation and the complete control they have over the network.

[18] Although bitcoin is not fully decentralised, it is a good representation of a system striving for that ideal.

3.2 Is EOS centralised? The important questions to answer

Recall, **decentralisation**: a system designed to allow free and open participation and encourage vibrant debate, rather than relegating decision-making or system management to a fixed set of entities.

First, it is important to have a basic answer to the following question: *Is EOS a system allowing free and open participation and encouraging vibrant debate?* Fortunately, this answer is fairly straightforward. EOS does allow free and open participation. Anyone can easily create an EOS account, stake tokens, use the EOS system, deploy smart contracts, vote for BPs, register to become a BP, along with everything else that is also available. There is no restriction on any account, and everyone has an equal opportunity to participate in the system. The EOS telegram community boasts over 70 000 users, where vibrant debate exists on the future of the EOS platform.

The difficult question is: *To what extent is decision-making or system management controlled by a fixed set of entities?* Recall that in DPoS, users with staked tokens have the power to vote in BPs, and BPs effectively control the system. Therefore, it is important to examine two different dynamics. First, the concentration of voting power. This will determine whether decision-making (election of BPs), is relegated to a few key individuals. Second, the BP dynamic, to understand the current pool of BPs, whether they are diverse, and the extent to which they can control the EOS system.

3.3 Voting dynamic

It is important to first get perspective with some high-level numbers, before further analysing this dynamic. *Note: The numbers used for voting analysis have been extracted and calculated from the full EOS account staking data set as at 2019-02-06, supplied by EOS Authority. These numbers have also been used to conduct analysis and create all of the figures below (unless otherwise stated). The full dataset and accompanying code can be found in the projects github repository: https://github.com/moose-code/eos-voting*

- Total EOS supply: 1 032 633 828 EOS (100%)
- Total staked: 489 396 200 EOS (48%)
- Actual EOS votes: 258 251 663 EOS (25%)
- Total number of EOS accounts: 702 156 accounts (100%)
- Number of voting accounts: 45 623 accounts (6.5%)

It is interesting to note that only about $\approx$ 25% of all EOS is used to vote. This could be thought of as translating to a 25% voter turnout in an election. This number is low in comparison to traditional voter turnout in national elections, where turnout most often exceeds 50%[24]. This begs the question as to whether the incentive of holding staked tokens (a vested financial interest in the system), is strong enough to motivate voter participation. Interesting to consider is $\approx$ 50% of all EOS tokens are not currently staked, and thus cannot be used to vote. Of the remaining $\approx$ 50% EOS tokens that are staked, approximately 50% of these are used to vote. This statistic, 50% of staked tokens being used to vote, may better reflect the actual voter turnout rate and is closer to what could be expected in national elections. To better illustrate this, unstaked tokens can be thought of as tourists being present in a country (EOS), but not participating in the election (voting), since they are just passing through and do not technically have a long term vested interest (staked tokens). This argues that voter turnout percentage is somewhat similar to national elections, and although not perfect, what can typically be expected. Voting percentage would likely increase over time, given the system is relatively new, and the voting process is not yet as streamlined as it should be in the future.

Before further examining the voting landscape, it is essential to mention the concept of proxy voting. Users who feel they do not have sufficient time to conduct extensive research and choose desirable BP candidates may instead delegate their voting power to a proxy. Anyone can register a proxy, with a proxy generally representing the voting ideals of a particular person or organization. For example, if a user wishes to vote for BP candidates that demonstrate transparency, they could delegate their voting power to a proxy voting for BP candidates that have historically been transparent. This significantly reduces a users workload by shifting the problem of researching hundreds of BP candidates to simply researching proxies aligning with their ideals[19].

The following split exists between votes via proxies and direct votes:

- Total EOS votes: 258 251 663 EOS (100%)
- Votes via proxy: 142 340 740 (55%)
- Direct votes: 115 910 923 (45%)

With proxies accounting for more than half of the total votes, it is intriguing to briefly consider the dynamic proxies introduce. On the one side of the table, proxies amalgamate voting power and hence could be seen as contributing to centralisation much like mining pools. On the other hand, proxies have detailed mandates, showcasing the ideals they stand for, the research conducted, and how that translates into the candidates chosen. This educated research on BP candidates as well as having a clear motivation as to why certain candidates were voted for, can be considered as healthy for the system.

Further, just as voters will no longer vote for BPs they perceive underperforming, voters will no longer delegate their power to proxies straying from their values. Critically, vote decay ensures that all voters have to reconsider the above point constantly (weekly if users are to keep their vote at its full 100% strength). In accordance with this logic, proxies will not necessarily be fixed entities with constant voting power, a desirable outcome for decentralisation. Overall proxies can be considered a valuable addition to the voting dynamic.

Recall that for every EOS token staked, between 1 to 30 BPs can be voted for. This method of voting is termed 'approval voting', and is well documented in academic literature[14, 54]. The main benefit it brings is through mitigating the issue of voters only 'backing one horse'. Imagine Alice had 100 EOS tokens staked to vote. Alice believes Alpha and Beta both to be good BP candidates, but believes Alpha is better. If each token counted as only a single vote, Alice might cast all 100 votes for Alpha. Since 21 BPs need to be elected, this is not optimal as Beta was also a good candidate. Approval voting means Alice can cast 100 votes for both Alpha and Beta, a desirable system outcome. In EOS it is especially important to mitigate 'backing one horse' behaviour, as 21 producers are elected, and it is crucial voters get the chance to consider a strong pool of candidates as a collective, without having to trade-off votes between candidates.

Figures 9 and 10 show the distribution of votes, concerning the number of BP candidates voted for using each vote. Figure 10 shows that there are a large number of accounts casting votes for just a single block producer. Fortunately, this does not translate into a vast majority of the actual voting power, voting for a single block producer. Figure 9 shows this positive picture, the vast majority of EOS used to vote, is for a large number of candidates. This is desirable as 'backing one horse' behaviour has been mitigated in practice, pointing toward the success of the approval voting

[19]This fact will likely lead to proxies increasing voter turnout.

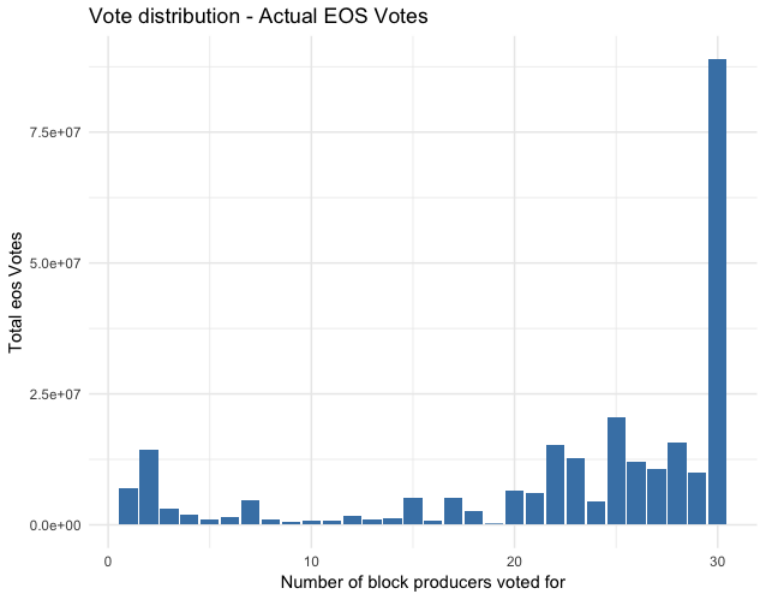


Fig. 9. This plot shows the vote distribution according to the total number of EOS votes. With a total of 258 251 663 EOS voted, 88 951 717 EOS (35%) was used to vote for the maximum allowed 30 block producers and 203 131 637 EOS (79%), was used to vote for 20 or more block producers.

mechanism.

Now the actual voting power dynamic is discussed. Figure 11 gives a high-level overview of this seemingly alarming scenario where voting power is controlled by a small set of entities. Table 1 shows that the top 35 most powerful voting accounts account for 50% of all votes cast. Figure 12 further reinforces that the voting power of the network is held by a small set of individuals or organizations.

It is clear that voting in EOS is centralised. A very small subset of individuals determine the BPs in the system. The BPs control the system. It is important to understand the ramifications of this fact. *Does this mean a subset of individuals control the system?* This does not necessarily mean a small subset of individuals control the system if there exists a sufficient disconnect between the BPs and voters. Although the centralised voters may determine the set of BPs, they do not control the BPs or their actions, and therefore do not control the system. However, it is difficult to hypothesize whether this disconnect exists. Since the key entities controlling the voting power are anonymous, it is difficult to determine whether there exists sufficient disconnect between them and the BPs. However, if sufficient disconnect does exist, the system could still be considered decentralised.

It is important to acknowledge that the skewed voting power distribution is by design. Users with significant voting power, hold significant amounts of EOS tokens. The top 35 accounts that currently comprise of 50% of the total votes, collectively hold at least $300 000 000 worth of EOS tokens[20]. This is a significant vested interest in the system, and accordingly, the top 35 accounts holding this

[20]Given the spot price of EOS tokens at $2.73 per token, when the data was collected on 06/02/2018.

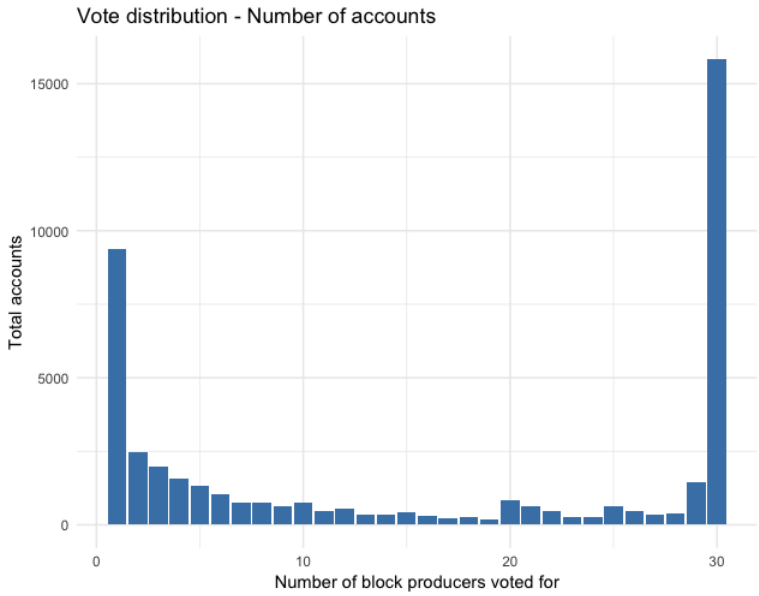


Fig. 10. This plot shows the vote distribution according to the accounts that voted. With 45 447 accounts voting in total, 20% of voting accounts voted for just 1 block producer. 35% of the accounts voted for the maximum allowed 30 block producers.

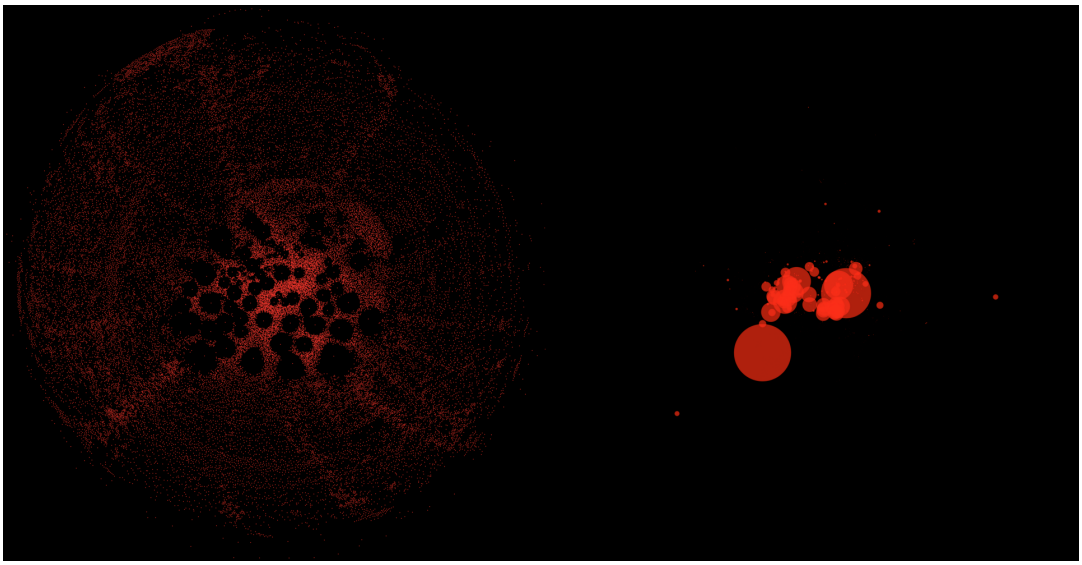


Fig. 11. The left side shows a hypothetical visualisation of the EOS voting network if every account had exactly one token for voting. The right side shows the true voting power distribution (i.e. a perfect democracy). Voting power per account (equals number of staked tokens). Circle radius represents percentage of total number of staked tokens.[47].

No. voting accounts	Percentage voting power controlled
10	22 %
20	35 %
35	50 %
50	59 %
100	73 %
200	82%
500	89%
1 000	93%
5 000	98%
20 000	99%
45 623	100%

Table 1. The table above shows that the top 35 voting accounts, control 50% of the current voting power in the system. This shows that voting power is extremely concentrated in the EOS network.

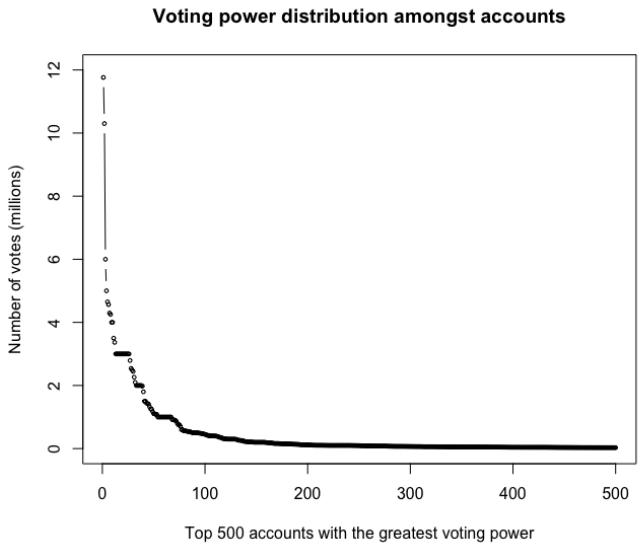


Fig. 12. This plot shows the voting power distribution. The x-axis corresponds to the accounts having the most voting power, in decreasing order. This graph clearly shows that voting power is very concentrated.

power, should be very strongly incentivised to vote in a way that will not harm system value. This is expressly how the system was designed. Strong economic incentives exist to encourage desirable system behaviour.

Unfortunately, the above logic does not hold true in one acute case. Large cryptocurrency exchanges hold large amounts of EOS, on behalf of users who are trading. Since these assets do not belong to the exchange, the exchange does not have a strong vested interest in system, yet they control significant voting power. Accordingly, this power can directly be used to vote for themselves as BP candidates, while they contribute little to the EOS ecosystem. Again, due to the anonymity

of voters, it is difficult to understand the exact extent of this problem without purely speculating.

The key takeaway from the voting dynamic analysis is that voting power is strongly concentrated (hence centralised). Sufficient disconnect between the concentrated voting power and BPs, means that voting power should not necessarily translate into controlling the system. Strong economic incentives should ensure that entities controlling the voting power, vote in a fair manner beneficial to overall system value.

3.4 Block producer (BP) dynamic

Figure 13, created by Peter Ruppel[21], shows an excellent visualisation of the current BP landscape (the radius of the circle is representative of the votes received by each candidate). Although 453 BP candidates exist, one can see visually that only a small subset of candidates receive enough votes to be considered relevant. Figure 14 further highlights this fact. It can easily be seen that only the top 100 BP candidates receive enough votes to be considered relevant. The discussion on the competitive landscape in block producing is deferred to the economic analysis section.

With only 21 BPs controlling the system, it is important to note the following: for any action to be accepted as valid and final in the network, a super-majority, 15 out of 21 BPs, are required to sign and confirm this action. This ensures that no single BP can unduly censor the system. This supports the argument of system decentralisation, as no single entity controls the system. In practice, BPs may collude[22] to censor the network, alter system rules or vote exchange. Therefore, to ensure the EOS network is sufficiently decentralised, it is fundamental that a diverse group of actively competing BPs exist.

One example of collusion is covered in article IV of the current EOS Constitution titled ‘No Vote Buying’, which states that: "No Member shall offer nor accept anything of value in exchange for a vote of any type, nor shall any Member unduly influence the vote of another." This talks to the behaviour of ‘mutual voting’, where BPs vote for each other in order to remain in power and keep their passive income. Having a diverse pool of BP candidates will significantly lower the chances of this collusion harmful to the EOS system.

The diversity of the BP candidates is currently questionable. Figure 15 clearly shows that the geographical diversity amongst candidates is lacking. A strong concentration of candidates in North America and China show two opposing nationalities largely control the system. Entrants from emerging nations will greatly help to diversify the pool of BPs.

The current pool of BPs are exclusively for-profit private organisations. Private organisations can be considered a desirable addition to the pool of BP candidates, as profit-seeking encourages healthy competition between entities to provide compelling BP services to increase profits further. There are two issues with only having private profit-seeking organisations forming part of the BP pool. First, since profit is the primary objective, collusion between these private BPs to increase profits, becomes a more likely scenario. Second, the volatility of the EOS token price means that downward price fluctuations in bear markets may cause many BPs to operate at a loss and quickly

[21] Image created by Peter Ruppel 2018, a senior researcher at Technische Universitat Berlin/Telekom Innovation Laboratories. This image has been licensed for use under the Attribution 4.0 International (CC BY 4.0) license.
[22] Later in the economic approach it is motivated why collusion should often fail.

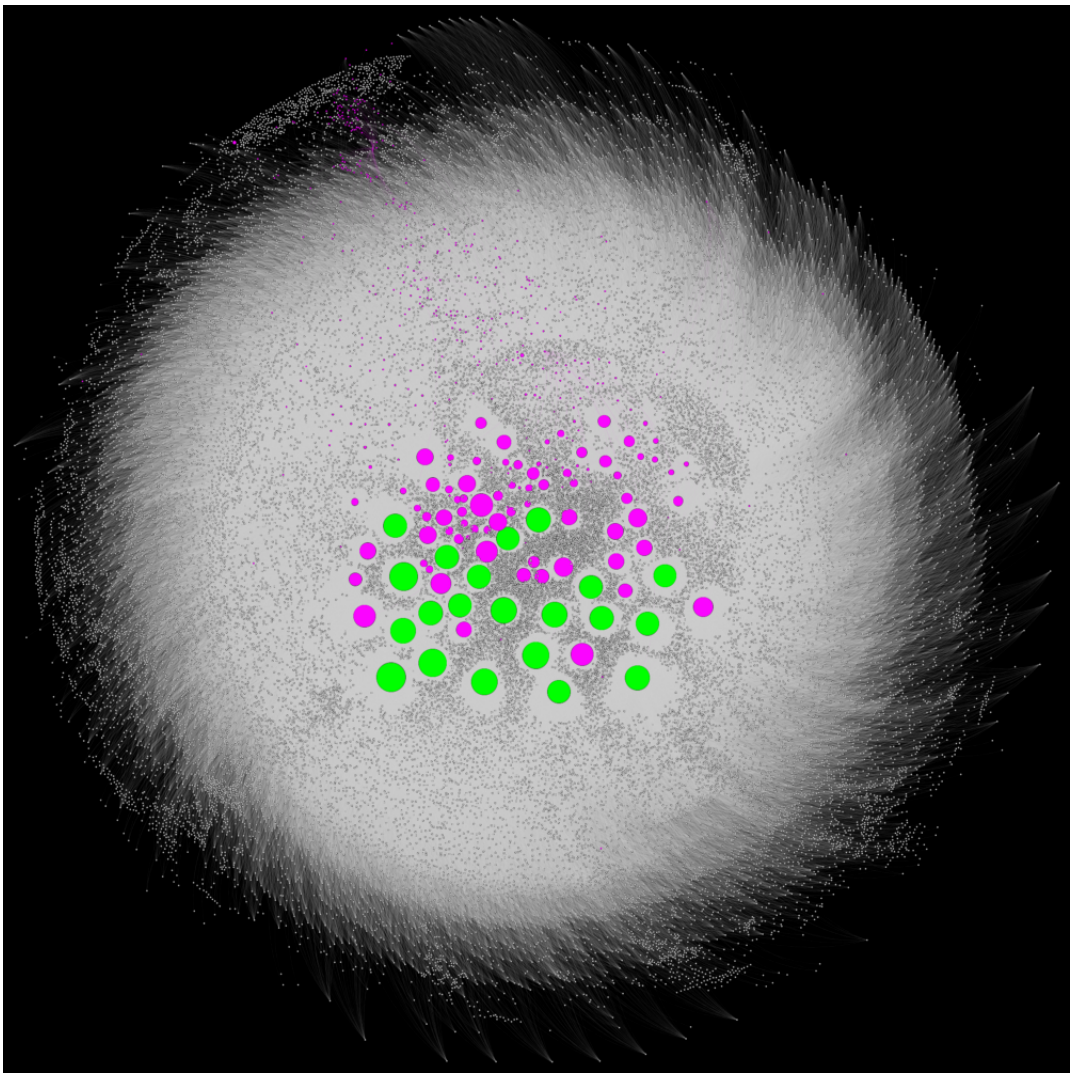


Fig. 13. EOS voting network. Grey nodes: < 35K EOS account who voted for BP candidates. Pink nodes: 432 producer candidates who received votes, but did not get elected as producers. Green nodes: 21 producer candidates with highest amount of received votes, thus being elected as producers. Circle radius represents percentage of received total votes. Edges represent a 'voted for' relationship. There are 500K edges in the graph [47].

shut down. This may leave the network vulnerable with only a few BPs still operating.

Through diversifying the pool of BPs by introducing actors such as universities, the above scenarios are less worrying. Given universities already have the human and networking infrastructure, they would operate regardless of the token price fluctuations, while also avoiding collusion to maintain their long-standing reputations. This makes for a strong addition to the diversity of the BP pool.

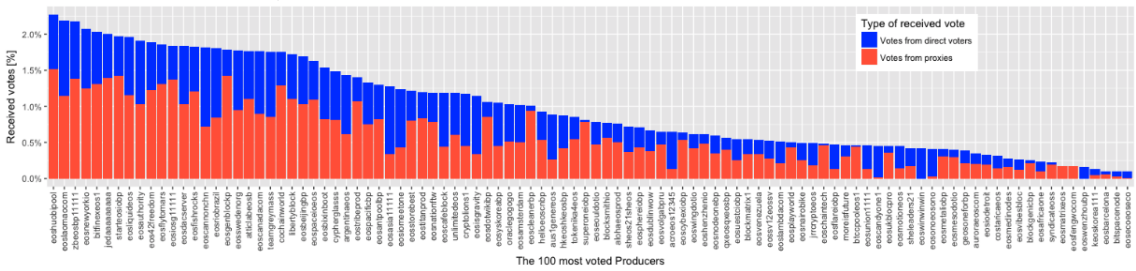


Fig. 14. Distribution of direct votes and proxy votes for the top 100 most voted block producer candidates. Blue indicates direct votes while red indicates proxy votes. [47].

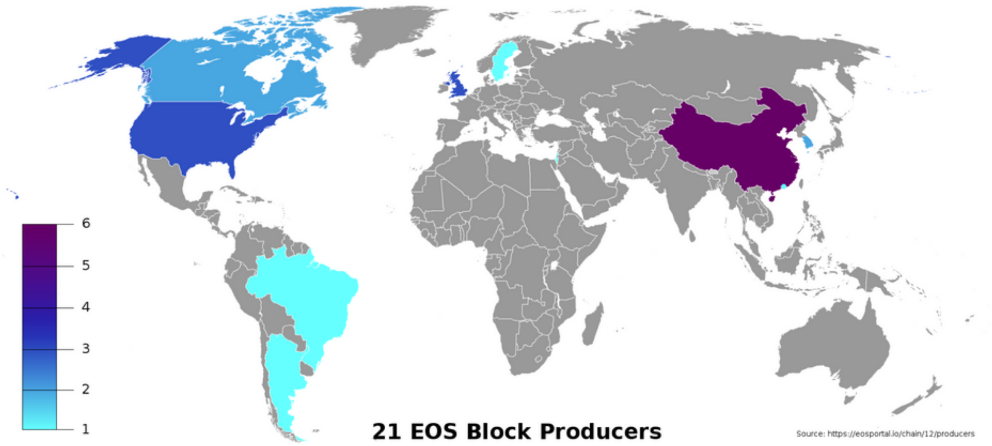


Fig. 15. Map of the global distribution of the top 21 block producers.

3.5 Comparative analysis

Here the observed node distribution in EOS and ethereum is compared. The EOS network is run by 21 nodes. While it is impossible to calculate the exact number, it is hypothesized the ethereum network has closer to 10 000 nodes actively producing records (mining). Based on the above, it is tempting to conclude ethereum is more decentralised than EOS prematurely. However, there are some very interesting factors still needing to be considered.

Due to economic realities in PoW, most miners join centralised entities known as mining pools, to ensure a more stable stream of income. The result is a landscape where a few key mining pools control the hashing power of the ethereum network. Figure 16 illustrates this, with just four mining pools together controlling more than 75% of the networks hashing power.

There are some intriguing comparisons regarding control in the EOS and ethereum. EOS has 21 BPs operating the system with 15 out of 21 BPs needed to control the network. Ethereum is controlled by a few mining pools, with theoretically 51% of the hashing power needed to somewhat control the network. One could argue that while you would need 15 BPs to collude in EOS, the collusion of just three mining pools could compromise the ethereum network. Based on the idea of centralisation being the ability of a small set of entities being able to control a system, ethereum

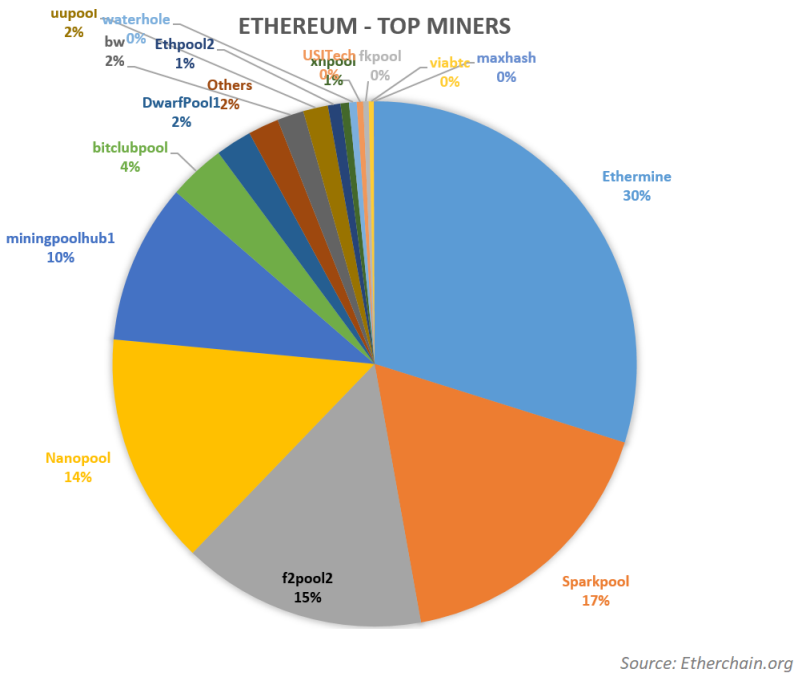


Fig. 16. Percentage of ethereum network power controlled by each mining pool.[7].

could be considered more centralised. This conclusion would be premature. It is essential to dig a layer deeper and examine who controls the BPs and mining pools.

As shown earlier, in EOS the BPs are mostly determined by a select number of entities holding a large amount of voting power. In ethereum, the mining pools are comprised of closer to 10 000 nodes in the network and their respective hashing power. In effect, this amounts to the following interesting dynamic:

- EOS - Small set of entities determining the BPs (less decentralised)
- ethereum - Large set of miners determining mining pool power (more decentralised)
- EOS - Control 15 of 21 BPs to control the network (more decentralised)
- ethereum - Control only 3 mining pools to control the network (less decentralised)

Although a possible oversimplification, the above reality represents the crux of the comparison. Both systems contain elements of centralisation at different levels.

3.6 Economic analysis

Key to the economic approach is better understanding competition; the associated market dynamic; and how this maps to decentralisation. In Keynesian economics, it is proposed that the following four basic market structures exist[42]:

(1) Monopoly
(2) Oligopoly
(3) Monopolistic Competition
(4) Perfect Competition

In a monopoly, there is no competition, generally due to legal or natural barriers preventing others from entering. A single dominant firm supplies the market, and as a result can influence output, prices, profit, and efficiency[42]. This is a clear example of centralisation; a single fixed entity with the ability to control a 'system' (output, prices).

On the opposite end of the spectrum, there is perfect competition. As the name suggests, fierce competition exists between a large number of firms offering an identical product or service. Further, there are no restrictions on entering the industry and competing. Since many firms exist and competition is strong, no single entity has the ability to influence the market. This market environment is more representative of the decentralised ideal. There are greater societal, and market efficiencies as no single entity can profiteer through limiting output and hiking prices[42]. Ethereum mining shows aspects of perfect competition as miners provide the identical service of record production, there are no restrictions on miners entering, and old miners do not have any advantage over new miners.

The EOS block producing landscape is neither a monopoly nor a perfectively competitive environment. The economic approach allows an investigation into where in the spectrum, oligopoly or monopolistic competition, the EOS BP industry could be considered. This answer will give more insight into EOS decentralisation.

Monopolistic competition is a market structure where[42]:

(1) A large number of firms compete
(2) Each firm produces a differentiated product
(3) Firms compete on product quality, price and marketing
(4) Firms are free to enter and exit

BPs can be thought of as firms. They are providing the service of producing records in the EOS network. Point 1 is satisfied as around 100 serious BPs are competing. Although each BP provides the service of producing records, one could argue they provide differentiated services. This is since each firm has its compelling narrative of why they should receive votes. To create this narrative, they are required to provide additional services to the community such as creating educational portals, block explorers, or contributing to core protocol development. They compete for votes based on the quality of their service provided amongst other things. Therefore points 2 and 3 are satisfied. Finally, point 4 is satisfied as BPs are allowed to enter and exit freely.

The above provides a relatively compelling argument that EOS block producing is a monopolistic competition market structure, a structure more representative of decentralisation.

One caveat weakens this argument. Since monopolistic competition contains many firms, theoretically collusion should be impossible. Although hundreds of BPs are competing, it only takes collusion between 15 of the top 21 BPs to control the system. This aspect is more representative of an oligopoly, as, in oligopolies, a small number of firms face the temptation to collude to increase their joint profit. This could mean the top BPs collude to ratify code changes that increase their revenue stream. It could also mean the BPs collude through 'vote exchange' agreements, in order to make there position more permanent.

It is well documented in game theory that collusion will likely fail. The prisoner's dilemma is a classic example demonstrating this. The logic put forth in the prisoner's dilemma should hold with

regards to colluding BPs. If all BPs colluded, they could maybe profit (even then unlikely since the token price may crash due to the act of collusion undermining system integrity). However, if one of the BPs cheated by alerting the community to the collusion, this would bring them a greater pay-off (increase there ranking). Therefore the Nash equilibrium would be all BPs cheating, acting in self-interest and not colluding.

The above only holds if the voting power distribution is such that colluding BPs can get punished and voted out of the top 21. The next section presents evidence that suggests that collusive behaviour is occurring, yet the voting power held by these BPs is too large such that they cannot be punished and voted out of the top 21.

The possibility of collusion among BPs (where the empirical evidence of this is explored in the next section), is more representative of an oligopoly and is cause for concern.

4 EMPIRICAL ANALYSIS OF BLOCK PRODUCER COLLUSION

The previous section painted an interesting picture of the landscape of block producing in EOS and highlighted the fact that collusion amongst block producers is a cause for concern. While opinions vary on whether block producer collusion is possible or will occur, this section aims to bring empirical evidence to light that suggests that collusion is occurring.

This section will primarily use unsupervised machine learning techniques in order to better discover structure and patterns concerning voting for EOS block producers. This will be prefaced by anecdotal evidence (leaked spreadsheets) suggesting that collusion is occurring.

4.1 Anecdotal evidence

The first alarming discovery is that of a vote for vote strategy apparent amongst top Chinese block producers. In this strategy, BPs group together and keep each other in power through an exchange of votes. The concept is simple, I vote for you, you vote for me, both parties benefit from the exchange. In and of itself, voting for another BP that you believe is doing a good job while they vote for you is not an issue. However, when both BPs make little contribution to the community and simply exchange votes to stay in power, this becomes a concern. Figure 17, a spreadsheet accidently released (leaked) by Huboi, one of the top block producers in EOS, highlights this alarming behaviour occurring in practice. It can be seen that at the latest time point, documentation indicating that vote exchanging occurs between Huboi and 16 other BPs.

The authenticity of this spreadsheet can obviously be called into question, and therefore it is critical to empirically examine voting in the EOS landscape to confirm that this behaviour is actually occurring in practice. While this is difficult as votes can be cast from pseudonymous accounts, general voting patterns and other complex relationships can be discovered through unsupervised machine learning techniques to determine whether collusion is occurring.

4.2 Unsupervised overview

In order to conduct a full scale unsupervised machine learning analysis of the EOS voting landscape, it is important to follow a rigorous methodology to avoid any statistical errors or modelling biases and ensure meaningful results.

火币投票节点	火币投票数	对方回投数	火币投票数	对方回投数	火币投票数	对方回投数
节点名称	9月4日		9月5日		9月10日	
eoshuobipool	1400		1400		1400	
starteosiobp	1000	1300	1000	1300	1200	1400
zbeosbp11111	1400	1500	1400	1500	1400	3705
eosflytomars	700	678	700	678	1700	2142
eostitanprod	200	456	200	440	200	484
bitfinexeos1	1000	4750	1000	4750	1000	4800
eosgenblockp	1400		1400		2000	
eoscannonchn			800	1490	1100	2007
eosfishrocks	300	318	300	318	300	458
eosstorebest	400	200	700	200	700	200
eosbeijingbp	600		600		600	
eosbixinboot	500	200	900	200	900	300
jedaaaaaaaaa	500	300	500	300	500	759
eoshenzhenio	500	50	500	50	500	98
eoseouldotio	500		500		500	
atticlabeosb		500		500		500
sheleaders21	500		500		500	527
eospacificbp					2000	
eoslaomaocom					200	
qxeosqxeosbp					150	280
eoscybexiobp					150	272
geosoneforbp					100	112
cryptokylin1	500		500		500	
eosiosg11111	1400		1400		2000	
cochainworld	1400		1400		2000	
eospaceioeos	1400		1400		2000	
总计	15600	10252	17100	11726	23600	18044

Fig. 17. The following spreadsheet was accidently released (leaked) by a cryptocurrency exchange in China (Huboi), also one of the top block producers in EOS. The column on the far left lists a number of different block producers. The 3 remaining double columns showcase 3 different time periods where the left column in white dictates the number of votes Huboi is giving to that BP (in millions), while the column on the right (in yellow) indicates the number of votes being received from other block producers in return. This graphic shows that at the most recent time point, vote exchanging is clearly occurring between Huboi and 16 other block producers all with Chinese origin.

First, the data being used will need to be cleaned and explored, accounting for various intricacies such as accounts that do not vote. Modelling decisions also need to be made on which data points to include in the analysis. The primary unsupervised method being used will be clustering analysis (k-means and hierarchical) which will allow us to determine whether meaningful groups or clusters of block producers exist that are receiving similar votes. Before starting the clustering analysis, several basic tests are performed to ensure that the data is in fact appropriate for this technique.

Voter name	Staked	Producers (up to 30 can be selected)
b1	90199990.0101	
guzdkmrtgage	10300189.6551	alohaeosprod, argentinaeos, aus1genereos, cryptolions1 ...
g4ydgmjug4ge	5666234.005	atticlabeosb, big.one, bitfinexeos1, blockpooleos ...

Table 2. The table above shows the first 3 entries of the raw EOS voting data. Staked represents the amount of votes stemming from that account. Producers is a list of up to 30 different block producers that that account currently votes for.

Based on observed clusters we can hypothesize the extent to which voting patterns and collusion may be occurring in EOS.

4.3 Data preparation

The data set being used is an EOS voting data set that consists of a list of all the accounts in EOS with staked EOS, the amount of EOS staked, and list of producers being voted for by that account. This dataset has been pulled directly from a full EOS node and made available by EOSAuthority. The data, along with all the accompanying code used in the analysis can be found in this projects github repository[23].

Table 2 shows the first 3 entries of the raw dataset that will be used to conduct analysis. The data set consists of around 500 000 entries, containing all the EOS accounts that have staked EOS and are eligible to vote. The first step in the data preparation is to remove all non-voting accounts. These can be thoughts of the null values in the data set. It is appropriate to simply remove these entries as we are conducting analysis on voting accounts only.

The next important modelling decision is to decide which subset of data to use in the modelling. Recall that the top 35 voting accounts control more than half the networks voting power. Therefore, for a more clear and accurate analysis, we will not include all 500 000 voting accounts, but rather a smaller and more meaningful subset. With the last 499 000 accounts only accounting for around 5% of the voting power, excluding the majority of accounts will remove a lot of unwanted noise and create a more computationally friendly dataset.

Figure 12, found in section 3, was used to create a visual cut off of the top 200 voting accounts. The top 200 accounts represent approximately 80% of the total voting power, and figure 12 clearly shows the voting power distribution to plateau at this point. Therefore, as it stands, the dataset being used now consists of 200 entries of the top voting accounts and the respective block producers they are voting for.

The next modelling step involves distance measurement. Distance measurement is crucial in clustering analysis. This is since the distance between two objects x and y represents the similarity of these two objects. Consequently, we need to obtain a distance matrix or as it is more commonly known, a dissimilarity matrix from our given data. Generally, if all features in the dataset are numeric, a distance based measurement such as Euclidean or Manhattan distance will be used. It is also possible to use a correlation based distance measurement such as Spearman or Pearson. Given our current data set, further manipulation and modelling thoughts are needed before we are able to calculate the dissimilarity matrix. This is since we currently have an interesting non-numeric

[23] github.com/moose-code/eos-voting

Block producer	voter 1	voter 2	...	voter 200
alohaeosprod	TRUE	FALSE	...	FALSE
atticlabeosb	FALSE	TRUE	...	TRUE
argentinaeos	TRUE	FALSE	...	TRUE

Table 3. The table above shows the first 3 entries of the final data set being used for clustering analysis. Voter 1 to voter 200 are boolean features representing whether that voter, voted for the relevant block producer.

feature, producers, which contains a list of up to 30 block producers.

The first step is to encode this feature, producers, as a categorical variable. This means will we now, instead of the producers feature, have alhoaeosprod, argentinaeos ... (a list of the all the block producers), with either TRUE or FALSE, a binary variable representing whether that voter did or did not vote for that block producer. The next step is removing the staked feature which is no longer necessary, as we have already used this feature to filter out and keep the top 200 voting accounts, and now we are rather concerned with discovering similarity in voting.

The final and least obvious modelling step is to invert the current matrix. Currently, the data is organized in a fashion where we would be doing clustering analysis to determine or cluster similar voters. This is less meaningful to us as voter names are pseudonymous. Rather, we are interested in grouping block producers based on whether they receive similar votes. This will be meaningful to isolate and determine whether a certain group of block producers receive very similar votes.

The final resultant dataset from the data preparation phase consists of 96 observations, the 96 different block producers, with 200 features, each a boolean indicating whether or not that top 200 voter, voted for the block producer. The first three entries are shown in table 3.

Given the current data, we are now in a position to create the dissimilarity matrix required for clustering analysis (distance measurement). Recall that while simple methods such as euclidean distance can be used for numeric features, all the features in our data set are boolean variables. In 1971, Gower authored a paper where he outlined Gower distance, a distance measure that can be used when dealing with numerical and or categorical variables[26]. Gower distance is computed as an average of partial dissimilarities across features, where a partial dissimilarity in a qualitative feature simply equals 1 if and only if the observations each have a different value for that feature. In the case of our data set, if one block producer receives a vote from voter 1, and another doesn't, this would result in a partial dissimilarity of 1. This score would then be averaged across all the features (voters).

Next, we will use a heat map to visually understand the result of the dissimilarity matrix. This is displayed in figure 18. Red in the heat map represents that there is high similarity (low dissimilarity), while blue indicates the opposite. The result is striking and exactly the what would we hope to see. Figure 18 shows that two distinct red squares are present in the heat map. This indicates that all the block producers within this square, receive votes of an extremely high similarity. What is alarming, is that the smaller of the 2 red squares, shows a group of 8 block producers that seemingly receive exactly identical votes. Among them, eosantpoolbp and dopsclubbp, are known mining pools, that contribute little to nothing to the EOS community. The consequences of these results are discussed in greater detail after the clustering analysis results are also presented.

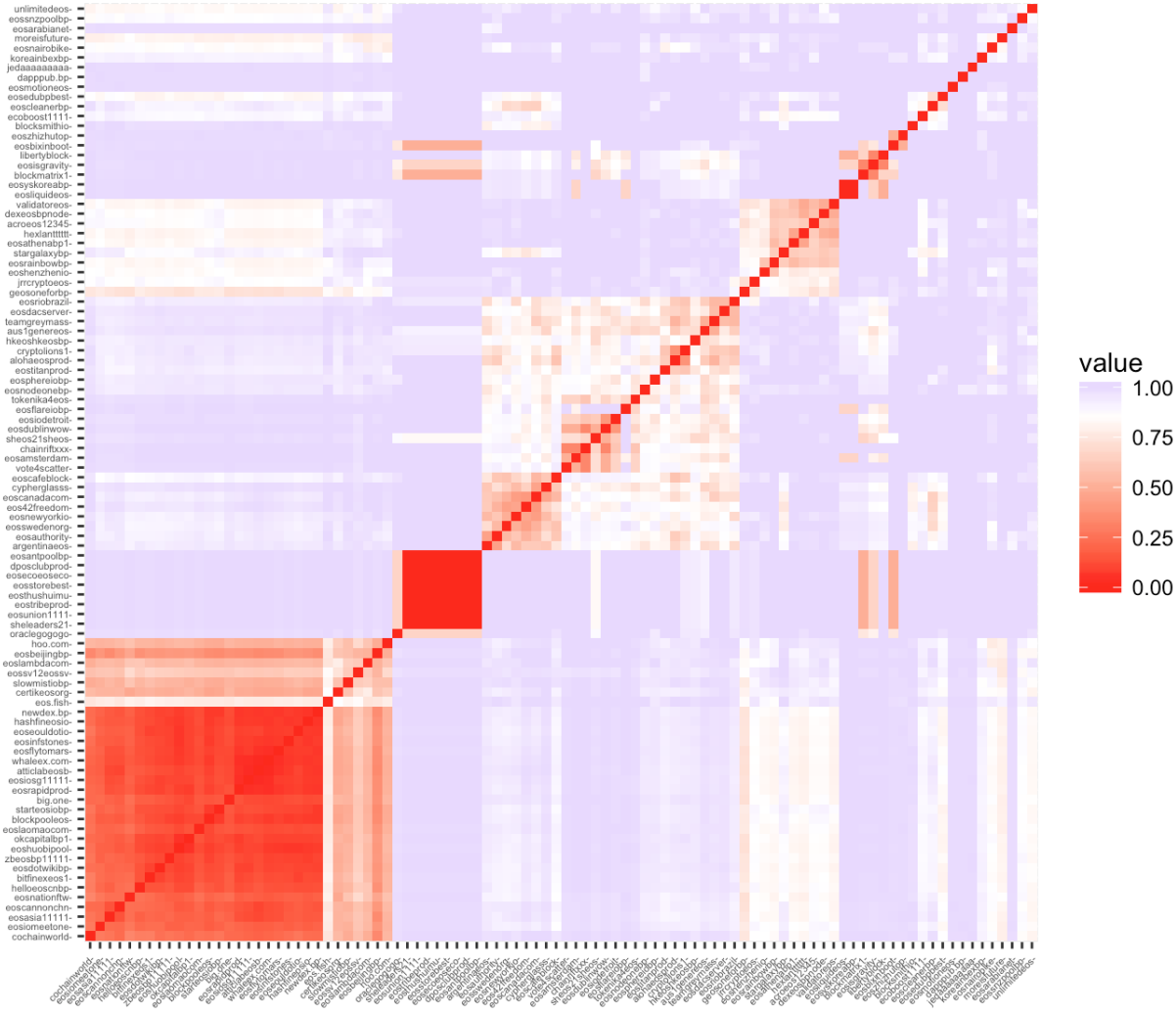


Fig. 18. The following heat map shows the various block producers and the similarity of votes they have received. Red indicates that there is a high similarity, with pure red indicating the votes received where identical. Blue indicates a low similarity on votes received. Figure 18 shows 2 distinct blocks of red where it appears these groups of block producers receive highly similar and in some cases, identical votes. The labels are not necessary. The focus of the diagram is showing that distinct blocks, representing groups, occur.

Next, it is important to determine whether clustering analysis is an appropriate and valid unsupervised machine learning technique on the given dataset.

4.4 Feasibility of clustering analysis

Before we begin with our clustering analysis, it is important to determine whether the given data contains any meaningful clusters (there is a non-random data structure). This is important as clustering methods will produce clusters for our data regardless of whether there are or are not

meaningful clusters[32]. We use both statistical and visual methods in order to further confirm that our data has meaningful clusters before beginning our cluster analysis.

4.4.1 Statistical methods. In this section we use the *Hopkins statistic*. The *Hopkins statistic* measures the likelihood that our data set was generated from a uniform distribution. We could also describe it as testing the spatial randomness of the data [32]. The Hopkins statistic is calculated as follows:

$$H = \frac{\sum_{i=1}^{n} y_i}{\sum_{i=1}^{n} x_i + \sum_{i=1}^{n} y_i} \tag{1}$$

When $H = 0.5$, we have that $\sum_{i=1}^{n} x_i$ and $\sum_{i=1}^{n} y_i$ are similar and thus the data has no meaningful clusters.

- Null hypothesis: No Meaningful clusters
- Alternative hypothesis: Meaningful clusters exist

We obtain $H = 0.2968359$ and thus we can reject the null hypothesis and conclude that meaningful clusters do exist.

4.4.2 Visual methods. For visually determining whether clusters are present we simply compute and display the dissimilarity matrix of our data. This has been done already in figure 18 and definitely confirms the presence of clusters as we saw those distinct red blocks. We can now move on toward performing our clustering analysis.

4.5 Partitioning clustering

The main idea behind clustering is classifying data observations into multiple groups based on similarity [32]. Partitioning clustering involves using clustering algorithms where it is required to specify upfront the number of clusters used in the analysis. The k-means clustering algorithm will be used, which creates clusters such that the total within cluster variation is minimised [32].

$$Total\ Within\ Cluster\ Variation = \sum_{k=1}^{k} \sum_{x_i \epsilon C_k} (x_i - \mu_k)^2 \tag{2}$$

The algorithm begins by selecting k different data points as the centres for the cluster. The remaining data points are then assigned to the various clusters. Iteratively the algorithm computes the new mean for each cluster then continues to update and assign data points to the different clusters until convergence is reached[32]. In order to accurately execute this algorithm, it would independently run 50 times so that 50 different starting permutations are initially used. This avoids cases where initial random permutations result in local as opposed to global minimums. Consequently, the best result is selected.

To estimate the optimal number of clusters for this technique we will use the average silhouette method. Figure 19 plots the average silhouette distance against the number of clusters. The average silhouette width can be thought of as an approximation to the average distance between clusters. According to Kaufman and Rousseeuw (1990), the best number of clusters should be that which maximises the average silhouette width [32], hence clusters would have the greatest dissimilarity. This corresponds to $k = 2$ in our diagram. It can be seen that $k = 3$ is also a near optimal number and we should also consider using 3 clusters.

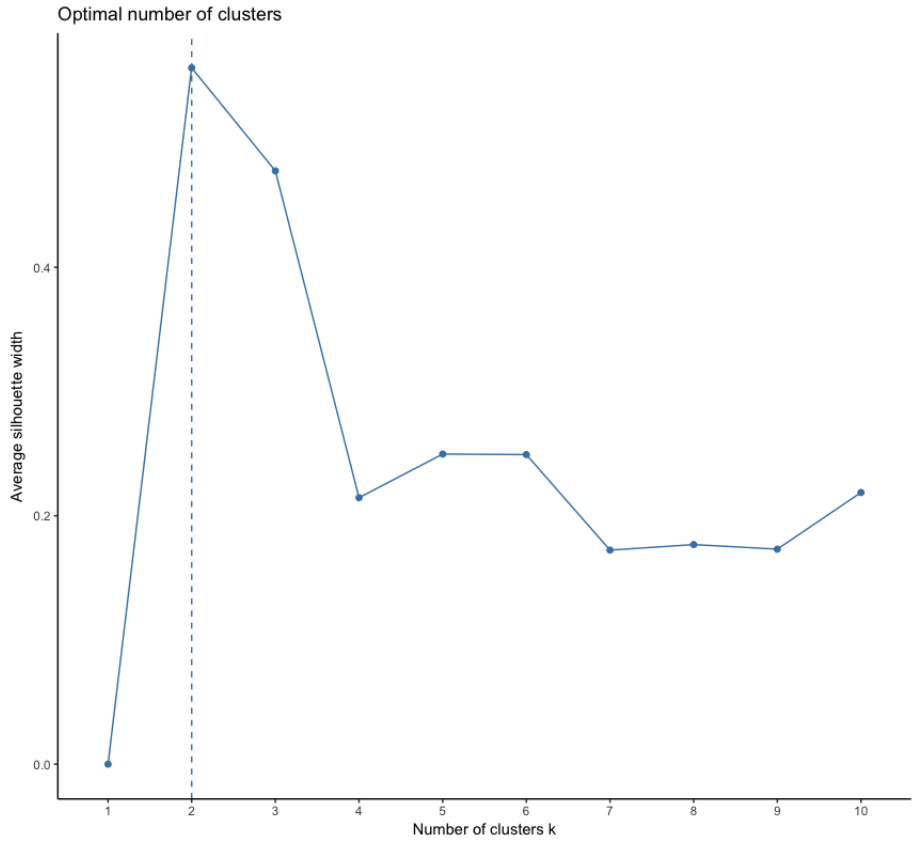


Fig. 19. Silhouette method used to determine number of clusters for analysis. The average silhouette width can be thought of as an approximation to the average distance between clusters. The best number of clusters should be that which maximises the average silhouette width, which is 2 in this case.

Figures 20 and 21 both clearly show the distinct clusters that are present, that is, groups of block producers are receiving similar votes. In these plots, the x-axis label, Dim1 (64.7%), implies that the first principal component accounts for 64.7% of the total variation.

PCA is a visualization and dimension reduction technique[51]. The first principal component direction of the data is that of which the given observations will vary the most. These principal components together help cumulatively to explain the variation in the observations.

In order to better understand the cluster plot and interpret what these two principal components represent, we refer briefly to figure 22, which details what exactly comprises of the first and second principal components. Since we are dealing with 200 binary variables, this plot is particularly uninteresting. It is less important for us to understand what comprises of the principal components, and rather more important to show distinct clusters and understand what they are (which has been shown). However, it would be incomplete to not show this plot and comment on how Dim1 is constructed in the clustering analysis.

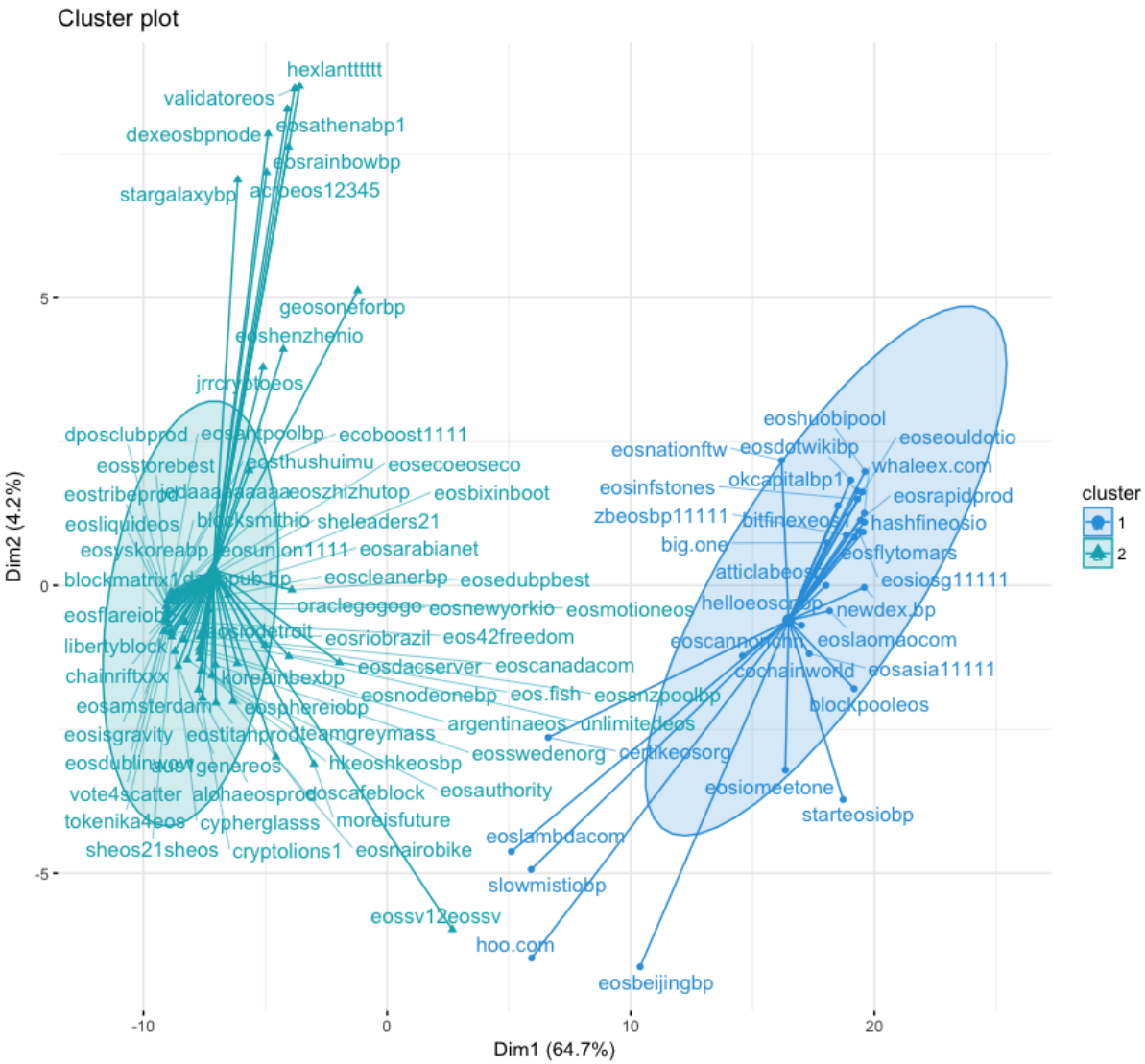


Fig. 20. Cluster plot generated using the unsupervised machine learning technique k-means clustering. Assumes $k = 2$ (2 clusters are present). Dim1 (64.7%), implies that the first principal component accounts for 64.7% of the total variation, while Dim2 (4.2%) shows the second principal component accounts for only 4.2% of the total variation. The first and second principal component of the first two orthogonal components derived from principal components analysis on the original feature set (200 voter boolean variables) that account for the most variation.

Before commenting further and concluding based on these cluster plot results, we also look at the hierarchical clustering technique. This will help to visually bring further insight the resulting conclusion.

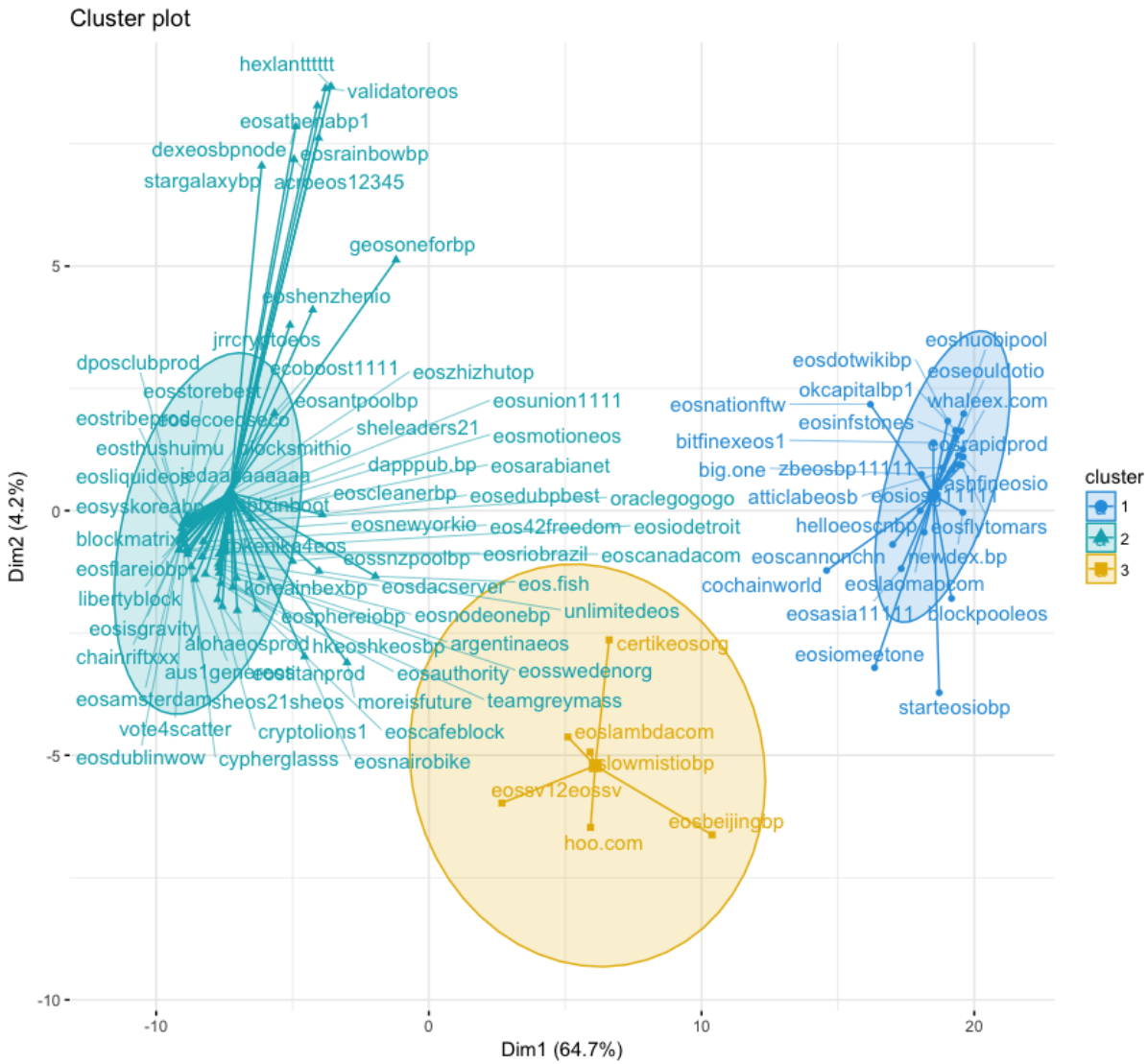


Fig. 21. Cluster plot generated using the unsupervised machine learning technique k-means clustering. Assumes $k = 3$ (3 clusters are present). Dim1 (64.7%), implies that the first principal component accounts for 64.7% of the total variation, while Dim2 (4.2%) shows the second principal component accounts for only 4.2% of the total variation. The first and second principal component of the first two orthogonal components derived from principal components analysis on the original feature set (200 voter boolean variables) that account for the most variation.

4.6 Hierarchical clustering

Hierarchical clustering is an alternative to partition clustering, with the notable difference being that the analyst doesn't need to specify in advance the number of clusters. Hierarchical clustering can be split down further into two approaches, namely agglomerative and divisive clustering[32]. In agglomerative clustering, each data point starts off being considered as its own cluster before

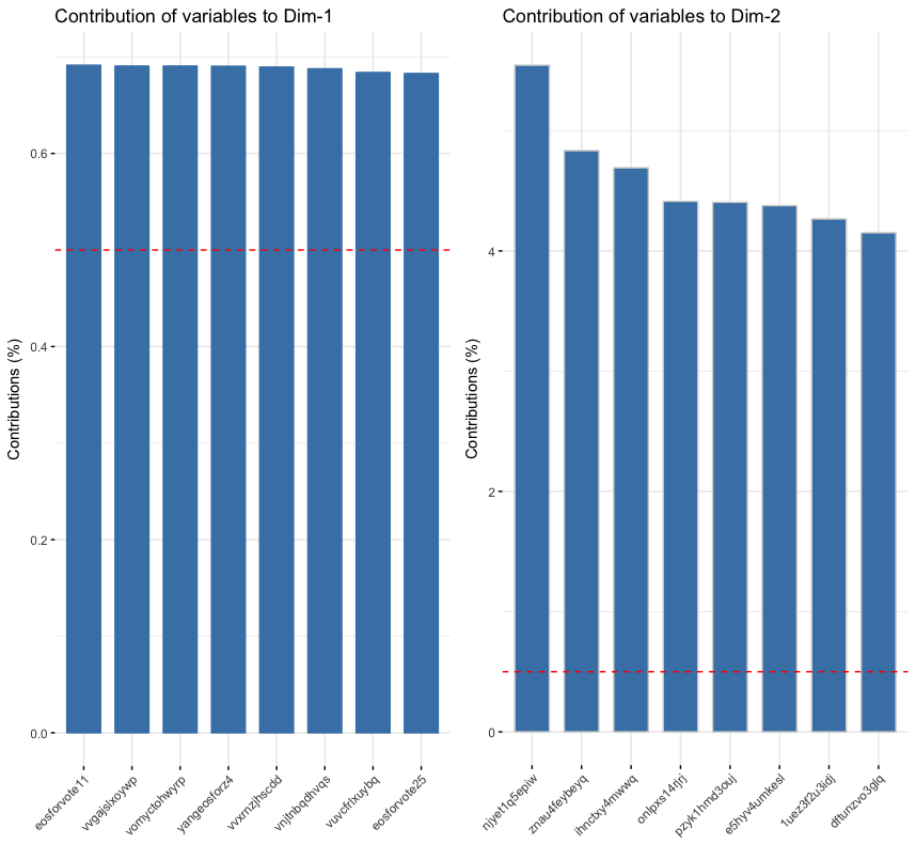


Fig. 22. Voters contributing the most variation to the first and second principal component.

the algorithm commences. In contrast, divisive clustering considers all data points to be part of the same single cluster before proceeding. For our analysis we use agglomerative clustering and we will present our results with a tree based figure known as a *dendogram*.

We expand further on the algorithm used in agglomerative clustering. As mentioned, each data point is initially considered as a its own cluster. At each step of the algorithm, the *two clusters* that are the most similar are combined [32]. This process is repeated until finally all the clusters are combined into one final cluster.

The result of this exercise is a tree based figure called a dendogram. The resulting dendogram from the agglomerative clustering in our analysis can be seen in figure 23. The diagram is interpreted as follows. Initially, each leaf is considered its own cluster. The two most similar leaf nodes (BPs), are then combined into a single cluster. The height at which these these clusters are joined in the denodgram is symbolic of how similar the relevant clusters are. If the clusters are joined at a very low height, this indicates that the clusters are extremely similar (hence the votes received by these clusters are extremely similar). These clusters are continually joined using this algorithm until two final clusters are shown (in turquoise and blue). Based on the average silhouette width

Fig. 23. Dendogram as a result of agglomerative clustering. Each block producer is initially considered as its own cluster. At each step of the algorithm, the two block producers receiving votes the most similar are then combined into a cluster. This process is repeated until finally all clusters are combined into a single cluster. The height at which clusters are joined on the dendogram is representative of similarity between the two clusters. I.e. If you look at 3 o'clock on the diagram, hashfineosio and newdex.bp are joined at a very small height where at 6 o'clock, cycpherglass and eoscafeblock are joined at a much bigger height indicating there votes received are less similar. The dendogram is also coloured to show two specific clusters (as in the last section we determined 2 as the optimal number of clusters).

test conducted earlier, the denogram was colour coded to show these 2 distinct clusters.

Figure 23 clearly shows that a large constituent of BPs have a join height that is very small (between 12 and 3 o'clock on the diagram), especially in comparison to the rest of the diagram. This indicates that this group of BPs receive votes of a very high similarity in comparison to the rest of the BPs receiving votes. This unusual vote similarity suggests some degree of control by a group of voters to elevate this cluster of BPs. The next section outlines these results and comments that this is likely indicative of a collusive voting relationship, where a vote for vote strategy is present and as a result we see this distinct cluster present where votes received are near identical.

4.7 Results

Based on the empirical analysis above we now discuss results that showcase a strong possibility of collusion amongst top block producers. The two most telling figures from the analysis are figure 18, the heat map generated from the dissimilarity matrix, and figure 23, the dendogram from the hierarchical clustering. Both of these figures showcase evidence that there are distinct groups of block producers receiving almost identical votes.

Figure 18 first clearly shows this with the distinct red blocks indicating high similarity of votes received between certain groups. The dendogram, figure 23, again clearly shows this fact. Notable, is that in the blue cluster in figure 23, the height at which the constituents of the blue cluster are joined together is remarkably small in comparison to the rest of the BPs. This demonstrates just how abnormally similar the votes received by these BPs. In particular, it is interesting to note the likes of eoshuboipool, zbeosbp11111, strateosbp, eosfishrocks, eoshenzhenio, bitfinexeos1 and atticlabeosb amongst others, featured in the leaked spreadsheet voting scandal, are found in this cluster receiving almost identical votes. Another discovery from the dendogram is the group of 8 block producers (5 o'clock on the diagram), that receive the exact same votes. This was the same group of BPs in shown in the heat map to have received identical votes.

These abnormally high similarities of votes received by certain groups of BPs is likely evidence of collusive relationships occurring. Since these block producers receive almost the exact same votes, we can infer that there exists numerous voters, voting for this very specific group of BPs. Since cryptocurrency exchanges and mining pools have large amounts of EOS and hence large voting power, they are likely constituents of these groups of voters, that are voting for themselves amongst others specific to their group. In a deliberate vote for vote relationship, these large players each have agreements to vote for each other, resulting in a group of specific voters (the exchanges and mining pools), voting for a very specific groups of BPs (the exchanges and mining pools). The result of this behaviour is a tight cluster where the votes received by these BPs have a high similarity as they almost all originate from the same specific group of voters (themselves).

Since voting accounts are pseudonymous and we cannot be certain who controls them, we cannot be 100% sure that a vote for vote strategy is occurring between top BPs. However, if this behaviour were to be occurring, tight clustering of vote similarity received by certain BPs would be a definite symptom, and since our empirical analysis showcases this result, there is strong reason to believe that vote collusion is occurring. This is in addition to to the leaked spreadsheet that suggests this behaviour.

We therefore hypothesize that there exists a large level of vote for vote collusion amongst the top Chinese block producers seen in the blue cluster (eoshuboipool, okcapitalbp, zeosbp11111 etc...).

The results of the unsupervised machine learning analysis indicate that there is a high likelihood that this hypothesis hold true, although one cannot deduce this with certainty because of the pseudonymous nature of voting accounts.

The idea of the EOS voting system, is that as users discover or suspect collusion, they would vote these block producers out and rather vote for honest block producers. Unfortunately, this fails, as these top block producers have enough voting power amongst themselves, to continue to remain in power. This is obviously undesirable and leads one to believe that EOS may not represent the best solution to scalability, as collusion is occurring between a subset of the small minority that control the integrity of the system. Therefore we can infer that the system is centralized (small set of block producers actively participating in vote collusion) and vulnerable to the possible ramifications this may introduce, collusion on varying levels (system censorship etc...).

The next sections aim to present some possible protocol modifications that could potentially make the EOS system more robust, in light of this discovery.

5 DPOS PROTOCOL ALTERATIONS

The following section explores possible protocol alterations to the EOS DPoS system. The protocol alterations are aimed at eliminating or reducing the current collusion present amongst the pool of block producers. Furthermore, these alterations aim to improve the overall health and diversity of the current block producing pool.

First, we will examine possible techniques such as inverse vote weighting, vote cut-off and quadratic voting as a means to reduce collusion and centralization. Finally, we will examine the addition of universities as block producing candidates.

5.1 Inverse vote weighting

Recall that for every EOS token staked, between 1 to 30 BPs can be voted for. In section 3 we discussed this method of voting, 'approval voting', and specifically talked about how its main benefit is through mitigating the issue of voters only 'backing one horse'. Recall that Alice may believe Alpha and Beta both to be good BP candidates, but believes Alpha is better. If each token counted as only a single vote, Alice might cast all 100 votes for Alpha. Since 21 BPs need to be elected, this is not optimal as Beta was also a good candidate. While approval voting solves this problem, another problem still exists. This is the fact that a large voter that is also a block producer, may only vote for themselves, as opposed to using all 30 votes. The consequence being that they significantly elevate themselves in the standings while not contributing votes to the competition. Returning to examine figure 10, shows clearly that this behaviour currently exists with a large number of accounts voting for just a single block producer.

Inverse vote weighting is a mechanism that could be used to incentivise a voter to cast more than just a single vote. The basic principle is that the power of the vote diminishes when less block producers are voted for. The actual implementation of the rate at which voting power diminishes as a fewer number of block producers are selected is an interesting question. Figure 24 shows two possible implementations where voting weight changes in accordance to number of block producers voted for.

The left side of figure 24 is represented by the following trigonometric equation:

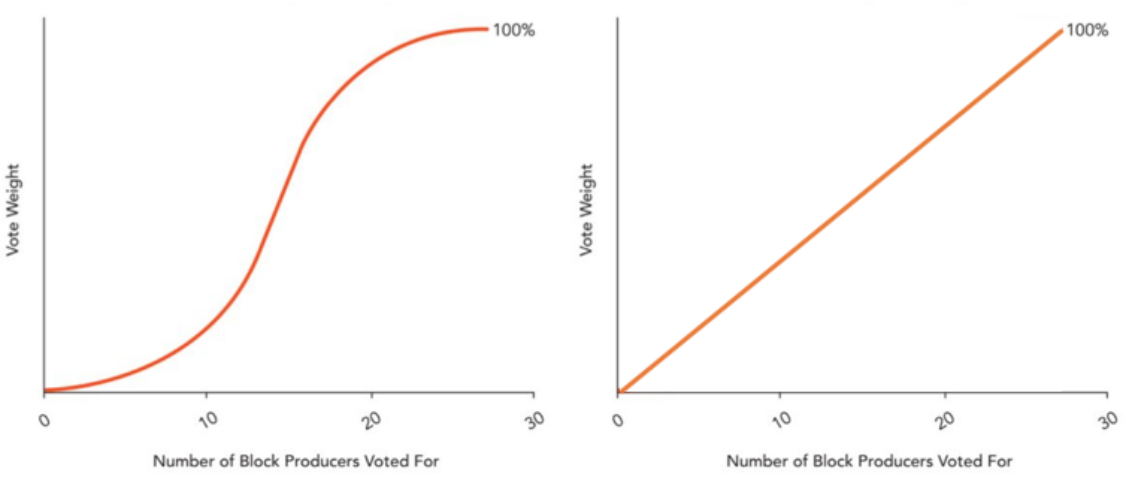


Fig. 24. Inverse vote weighting functions.

$$Votestrength = normalvotingweight \cdot sin(\frac{\pi}{2} \cdot \frac{number\ of\ BPS\ voted\ for}{maximum\ number\ of\ BPs\ possible\ to\ vote\ for}) \quad (3)$$

While the right side of figure 24 is represented by the much more simple linear function:

$$Vote\ strength = normal\ voting\ weight \cdot \frac{number\ of\ BPS\ voted\ for}{maximum\ number\ of\ BPs\ possible\ to\ vote\ for} \quad (4)$$

As can be easily seen from figure 24, if one would like to maximize the strength of their vote, a greater number of block producers should be selected. The trigonometric function would likely be preferred as it offers greater incentive to initially add more BPs to any given vote, while also penalizing one less harshly for not including a final few block producers. This is desirable as users will be incentivised to research and include more block producers, while also not being burdened with having to worry about including a full 30. Once 21 block producers are reached, the full power of the vote is almost reached.

While inverse vote weighting can be seen as a valuable addition to the DPoS voting protocol, there are caveats. For a malicious block producer to still attain a vote at full strength without voting for any meaningful competitors, this BP could randomly cast the other 29 votes to low ranking block producers, or spin up 29 Sybil BPs and simply vote for these. In order to reduce this behaviour, an interesting modification would be such that for a full strength vote to qualify, a certain subset of the BPs voted for should rank in some percentile. While this might mitigate voting for random Sybil nodes, it encodes a hard and fast rule where a selection of top BPs need to be continually voted for to keep vote strength at its fullest. This is of course undesirable when wanting a healthy BP pool where new entrants can easily join, and old BPs are not entrenched in the system.

5.2 Vote levelling

Vote levelling is the term we will use that refers to mechanisms aiming to reduce the power of the major voters controlling the network. Below we explore a vote cut-off and a quadratic voting system.

Referring back to figure 12, one can see that there are a small handful of actors with between 2 to 12 million votes, whilst the vast majority have less than 50 000. In fact, only the top 350 voters have 50 000 or more votes. The most naive approach would be to limit the total power that any one voter has through having a ceiling on voting power. This ceiling could say be 50 000 votes as the maximum any account could cast. The exact optimal number of the maximum vote power for any account is a difficult question that would likely need a period of empirical analysis for a firm conclusion to be drawn.

Another form of vote levelling would be a quadratic voting power mechanism where:

$$vote\ power = \sqrt{Number\ of\ votes\ cast}$$

with effect being that the power of large votes is greatly diminished. For example, a vote previously worth 1 000 000, would now only be worth 1 000, while a vote of 1 would still be worth 1. The effect is drastically narrowing the gap between strongest and weakest voters.

While both vote levelling mechanisms seem to be attractive ways to limit extreme voting power, they are completely vulnerable to the design of the system that allows seamless creation of pseudonymous accounts. In order to bypass vote levelling, large voters could simply spread their voting power across a large number of smaller accounts that they own. In the case of vote cut-off, this would mean creating a host of accounts with 50 000 voting power. Quadratic voting power would make it a little more complicated, but still a similar strategy of numerous accounts being used would help circumvent the intended behaviour of this protocol modification.

In light of this attack vector, the implementation of vote levelling would likely add little value to the EOS ecosystem.

5.3 University participation in DPoS: Proposal

We propose that formal academic institutions, such as universities, participate in the DPoS protocol by encouraging a group of professors and students, to run a block producing node. University participation would help to diversify the pool of BPs and promote decentralisation in EOS. The model can be constructed so that the university provides valuable research into platform protocols, all while augmenting the educational experience. First, the rationale of the proposal is discussed before sketching the details of implementation.

5.3.1 Health of DPoS. Note the apparent barriers to entry in running a BP node, intensive human capital and networking infrastructure requirements. Unfortunately, these barriers are a must, since these requirements ensure that the nodes running the network are high performing, hence facilitating scalability. Therefore, instead of examining how to lower the barriers of entry, it is interesting to examine which set of entities, other than private companies, are capable of participating as BPs. Recall, new private companies are the BPs running the network. It is apparent that because these candidates lack a significant historical track record, there is an understandable lack of trust in the current pool of BP candidates.

Notably, universities have a legacy. That is, they have a long-standing reputation built on many years of academic research and education, inspiring trust and confidence in their endeavours. Having professors and students participate under a formal university banner would create confidence around the honesty and quality of the BP node. In turn, this would inspire more confidence in EOS and similar DPoS based systems, by having a long-standing, reputable institution involved in running the network. Universities also possess the skilled human capital (professors and students), and the server infrastructure, affording most universities the opportunity to relatively easily enter the BP market.

Besides adding to the diversity of current block producers, the project[24] also aims to add value to the network. This critical aspect is required to present a compelling narrative to voters. When discussing this narrative in the implementation, we describe in greater detail the value a university could aim to add to the network. We now briefly note that this includes community education, formal academic research into system related problems and protocol implementations.

5.3.2 Enhanced learning experience. "Education is the most powerful weapon which you can use to change the world."

– Nelson Mandela

The above quote is clear. Education is the cornerstone of progress. It is therefore essential to continually consider how best to educate.

Interesting research from the University of London suggests that people currently aged 20, have a 50% chance of living to more than 100 years old[27]. The ramifications of this finding, an elongated 100-year life, are that periods of work become more extensive, savings more central, and across the passage of time, major transformations occur in industries, jobs, and **education**[27]. Based on these insights, it is now more critical than ever for educators and students to consider the following questions:

(1) What should we learn?
(2) How should we learn?

When faced with the first question, individuals should think of what skills are *valuable* (i.e. skills in demand) and *rare*[27]. The implication of rare is that lack of supply dictates a premium required for that skill. Although this first question is unique to each individual, it is still important to educators as this aggregate decision translates into demand for acquiring specific skills. Given the technological developments occurring and our elongated lifespan, it is crucial to assess the continually shifting employment landscape and acquire skills that have the best possible chance of remaining valued in the future.

Distributed ledger technology certainty satisfies this criterion, with demand continuing to increase for individuals skilled in this area. Unfortunately, many universities are still slow to adapt to the demand for education in these critical skills. For example, the University of Cape Town's Computer Science department offers no courses at any level concerning DLT[25]. This statistic is alarming as the demand for these skills is outpacing any other computer science related field. In fact, Upwork, prominent freelance marketplace platform, revealed that demand for DLT related

[24]Throughout the report, 'the project' references the project whereby the university runs a BP node for a DLT network.
[25]To their credit, AIFMRM part of the University of Cape Town, has pioneered through launching a financial technology graduate degree focussed on DLT.

skills was the fastest growing amongst more than 5000 listed skills in 2018.

The above gives strong motivation for universities to introduce core courses focussed on DLT. If not, students may avoid formal academic institutions and acquire these skills through more informal online learning portals. There is undoubtedly a strong case for many individuals to learn DLT as it is both *valuable* and *rare*. The next important question is how one should learn. Here we outline why running a block producing node is one of the best ways to learn about DLT.

It should come as no surprise that we see a shift in how we learn. Given we exist in an age where information is more accessible than ever before, it is no longer our knowledge that sets us apart, rather what we have experienced with this knowledge [27]. Accordingly, we should see a shift toward experiential learning, learning through actions and experience as opposed to the current classroom/textbook norm. The implication of this would be students better acquiring tacit knowledge. Tacit knowledge is difficult to transfer to others through writing it down or verbalizing it; instead, it needs to be acquired over time through actions and experiences. Tacit knowledge is something of growing importance in the wake of rapid artificial intelligence advances, as explicit knowledge[26] becomes less and less valued.

Creating a project where a group of students are responsible for a block producing node, exemplifies the ideal of experiential learning. Students will be applying their knowledge to a complex real-world task, and in the process acquiring essential tacit knowledge. With the wide array of skills required to run a successful node, cross-departmental collaboration will be required. The result is that students will need to learn to work with other students possessing complementary skill-sets, in order to be successful. This dynamic has long existed in the workplace, yet is absent for the most part in education[27]. Additionally, we hypothesize that this real-world project will see a greater commitment from students given its real-world consequences.

The overall theme is that the project has the potential to enhance learning. Given that employers require not just 'book smarts', but real-world skills - this project ideally equips students for the workplace. The greater overhead of grading real-world projects, as opposed to grading simple assignments, may cause reluctance of educators to adopt this model. These types of challenges are discussed further in the implementation section.

5.3.3 Revenue. The possible revenue to be received as BP is another aspect making the project attractive. The number of votes received together with the overall ranking amongst BPs determines the revenue received by each BP. The top 21 BPs understandably receive a higher share of revenue compared to the other standby producers. As it currently stands, the number 1 ranked BP receives about 700 EOS tokens per day while the 82nd ranked BP receives 100 EOS per day. The price has ranged between $1 to $10 with relative stability now occurring at $2.5 per token[37]. This equates to earning $1 750 or $250 per day respectively for BPs ranked 1 and 82. Due to the price volatility, it is difficult to forecast the real purchasing power of this revenue. Nevertheless, the positive outlook of the platform together with the ability to convert EOS earnings to $US dollars daily is encouraging.

[26]Explicit knowledge or codified knowledge, can easily be transferred, through text or speech. In the age of the internet, this makes it easily accessible to all.
[27]There is a special mention to many Master's level degrees that address this discrepancy.

Conservatively estimating $250 per day, the BP node project could collect $92 000 per year. The impact of this amount of money in an emerging market is not to be understated. This revenue could pay for a year's tuition in a sciences related degree for 23 students at the University of Cape Town[28]. The above model works since the costs of running the node are minimal. The server infrastructure already exists given the university. The human capital is readily available from the university. Therefore the revenue received almost directly correlates to funds that can be deployed into research and education.

The introduction of revenue possibilities will deter many academics, as the motive for the project begins to be questioned. In response to hearing the high-level idea construed in this report, one academic head of department remarked: "we (academia) do not subscribe to the corporate thinking that with money things become possible. Things become possible with ideas". Although we echo the sentiment, it is an unfortunate reality, without money, ideas are often not conceived. Money ensures the basics, food, shelter, water, electricity and other necessitates, things that would otherwise occupy our time as opposed to ideation. It is fundamental to understand that generating revenue is not the goal of the project. Instead, revenue is an unintended and welcomed by-product that can be used to further education and ideation.

6 UNIVERSITY PARTICIPATION IN DPOS: IMPLEMENTATION

The core goal is that the BP node model is designed such that it can be easily replicated and deployed by universities around the world. The result of this would be competition, forcing all candidates to perform as best as possible. Just like the infamous Cambridge Oxford boat race, we anticipate that global competition between universities to rank among the top BPs will drive strong system growth. As the famous Roman poet, Ovid once remarked, "A horse never runs so fast as when he has other horses to catch up and outpace". We now detail the formal requirements of a BP node, before discussing more specific implementation opportunities.

6.1 BP candidate requirements

The following 8 requirements are needed to become a BP for EOS[2]:

(1) **Public presence**. *Website required and at least one social media account.*
(2) **ID on Steemit**[29]. *The following information is to be posted on Steemit:*
   - *Official block producer candidate name.*
   - *Location of company headquarters.*
   - *Expected location of servers.*
   - *Type of servers (cloud, bare metal, hybrid).*
   - *Current employee list and pictures of at least 67% of members.*
   - *Relevant background qualifications for at least 67% of members.*
(3) **Technology Specifications**.
(4) **Scaling plan**. *How the node infrastructure will scale over time to handle the increasing demands of the network.*
(5) **Community benefit**.
(6) **Telegram + Testnet**. *Telegram account and test-net nodes for community participation.*
(7) **Roadmap**. *Ethos, values, finances, transparency, or any other topic the candidate deems important.*

[28] Assuming fee prices of $4 000 per year for sciences tuition. This fee is in line with the 2019 UCT sciences fee handbook
[29] Steemit is a social media based public blockchain that allows for permanent and accountable posts.

(8) **Dividend Positions**. *What will be done with BP revenue received? Important that this will be not used to buy votes as this would undermine system integrity.*

While most of the requirements are relatively straightforward, some require particular attention. Having solid technological specifications are core to the value proposition of any block producer. Fortunately, the server infrastructure is generally of top quality at universities. Close contact between university network administrators and relevant stakeholders should be established to gauge the actual server specifications and hardware that may be utilised. We note that this may be one of the most difficult, and make or break steps, in ensuring project success. University staff may be very apprehensive as they could see the project as a pure money-making scheme. It is very important to clearly articulate the goals of the project to gain university buy-in. No doubt, a university professor or a similar individual with esteemed rapport associated with the project would help this process.

Requirements 5 and 7 are more central to the value proposition for the BP node and will be discussed in further detail in the coming sections.

6.2 Possible project structures

Below we elaborate on two possible structures that may suit the project.

6.2.1 Honours/Masters module/dissertation. In this more formal option, a lecturer can create a module centred around DLT, with the project being the core of the module and representing experiential learning. This could be an optional honours module in the Computer Science honours degree or a part of a year-long module in the financial technology masters degree. The idea would be that all students taking the module, are a team from day one. With say, 15 students enrolled in the module, it would be representative of a real-world business start-up and allow the students to face the same problems currently faced by start-ups today and learn through experience. It is suggested that initially minimal guidance is given, as it will give students the freedom to experiment and determine how best to start with a real-world project. Given the learning through experience and observations seen, the professor administering the course should still provide sufficient support when needed.

Through the above process, there should be a leadership structure formed, as well as distinct groups (3-4 people), working on different tasks crucial to project success. There should most likely be one group responsible for the block producing infrastructure, one for the development of community-based tools, one investigating useful additions to current protocols, and another in charge of community educational workshops. Since marketing, business and strategy are also large components of success; effort should be made for cross-departmental collaboration. A team of 2-4 postgraduate marketing students could be responsible for the digital marketing of the project. This will actively add to their knowledge stock as they work on a real-world project in a growing digital sector. Accounting and finance postgraduate students should also be involved. The accounting students would gain valuable experience in accounting for tokenized assets, while finance students can drive company strategy. The above will help students to learn the essential tacit skills of working together with individuals possessing a completely different skill set.

The above ideal is very optimistic. Convincing academics across different departments to encourage students to join a project seen as risky will be very challenging. It will again be necessary for the professor in charge to clearly articulate the benefits of experiential learning and how the project can significantly enhance the students learning experience. Instead of a module, the project could also take the form of a dissertation where different students tackle different academic problems

currently existing in the system. There are economic issues regarding voting in DPoS. There are transparency issues regarding BP earnings[30]. There are fundamental protocol issues regarding network resource usage. All of these and many more can provide fascinating academic research topics for students from many different departments. The research into these system problems, provided with the BP services offered would provide a compelling narrative for the success of the project.

Given that introducing this project into formal education may be very difficult, the structure suggested below may be more realistic to implement, and still allow for many for the same benefits.

6.2.2 Student society. Given the possible difficulty of integrating this type of project into a formal academic course, it is necessary to consider how else this project may succeed. The project taking the form of a student society is an interesting possibility. Generally, a student society is created to unite a collective body of students that have a common interest in a specific topic of activity. The chess, tennis, debating, consulting, entrepreneurship and gamers club represent examples of successful clubs formed around a common interest. Accordingly, the BP society would revolve around students common interest in distributed ledger technology and how they can unite to learn more about the technology through actively participating in a real-world project.

Just as normal university societies attract a diverse range of students, this BP society would represent the perfect opportunity to attract students from a broad range of faculties, perfect for the operation of the BP node. Ideally, the society would be able to generate revenue from the BP services and use this revenue to grow the society and provide widespread education on disruptive technologies such as DLT, given its visible role in the future. Because universities are renown for supporting student societies, it is hopeful that this society would be allowed to utilise the universities server infrastructure to provide the BP service. Later, revenue streams will allow the society to buy their own server infrastructure.

6.3 The narrative

As previously emphasised, it is vital for the project to present a compelling narrative to the EOS community as to why it should be voted for. Below we describe a few aspects that would make the candidacy of a university BP node compelling.

Academic research. Being a formal academic institution, universities are positioned perfectly to conduct academic research into active problems present in distributed ledger technology. Through adhering to stringent research requirements and producing peer-reviewed research, there will likely be desirable mechanisms and ideas generated that can aid the ecosystem. An example may be economic research paper culminating in a new voting mechanism deemed to be more robust than the current approval voting mechanism.

Although academic research is seemingly compelling, interactions with a current top 21 block producer, EOS New York, brought to light that the community is more interested in working implementations as opposed to pure research. Although research is often necessary for the resultant implementation, it is the entity contributing the raw lines of code that receives praise and votes from the community. Accordingly, it will be essential to ensure that in addition to the academic

[30]There was keen interest from one master's level students to complete their dissertation on transparent accounting DLT environments.

research conducted, students attempt to implement the ideas suggested (or contribute to ongoing implementations).

Ethos and values. This should be unique and carefully thought out by each prospective university aiming to implement this project. The primary values that would be important to show to the community are honesty, integrity, hard work, and generosity. In the next section, we outline the ethos and values of the UCT pilot project.

Community benefit. Here lies a significant aspect of the narrative. Since the BP rewards received will mostly be disposable (due to no large server and employee costs), it will be essential to use the rewards to educate the community. Given universities are environments centred around learning; the perfect facilities exist to enable learning. Funding can be used to consistently host workshops and courses designed to educate the broader community on the technology. The members of the project can better hone their skills through teaching, while the open workshop will allow communities to master a technology on the rise. Through understanding DLT, local communities can be opened up to global work opportunities. Besides facilitating education to a wide audience, rewards will also allow for tertiary education scholarships. Although the above may not directly contribute to the value of EOS, it importantly shows that participants in the ecosystem are taking a definitive step toward educating the community. To reiterate the thoughts of Nelson Mandela, education is the weapon we can use to change the world.

Value add tools. One of the key ways BPs add value to the community, is through their development of tools useful for the EOS ecosystem. For example, 'block explorers' help provide an intuitive user interface for individuals to observe the EOS chain and the current transactions taking place in it. Voting tools help users to vote with their tokens more easily, and statistics portals help users to understand the current composition of EOS BPs and their earnings. These all represent self-contained projects that can be undertaken to help better the user experience of the system. Groups of 3-4 students who are members of the project can actively develop useful tools needed by the community. This contribution of tools will be another compelling argument for receiving votes.

Finances and transparency. The issue with receiving substantial rewards for being a BP, is that the community wants to see how these rewards are being used. If they are simply used for personal profiteering, it is unlikely they will continue to be voted in. Transparency is important as it documents how and why finances are being used as they are. Given universities are already accustomed to very rigorous financial procedures, professionally handling the received funds and accounting for everything spent should be fairly routine.

It is important to motivate to the community that all the funds received are used by the university for the express purpose of furthering education. For the most part, this is through educating the community on DLT, funding research projects tackling significant DLT related problems and curating courses that enable experiential learning while operating a BP node.

6.4 Pilot: University of Cape Town node

The first attempted implementation of the suggested model is currently being undertaken by the University of Cape Town. In this brief section, we highlight the current progress of the project.

The first step to the project was finding academics who would believe in the benefit of the project and the impact it could make. The Director of the South African financial innovation lab, Co-Pierre,

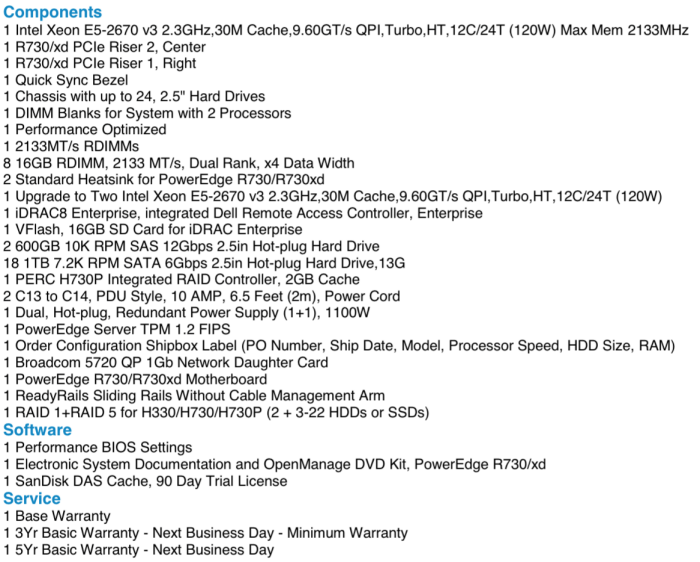


Fig. 25. Specifications of the UCT server to be used for the BP project.

liked the idea and saw its benefits. Accordingly, Co enabled one of his master's students to pioneer the project, through securing the use of a high-performance networking server. The exact server infrastructure used for the project is detailed in figure 25. This was the first major milestone, as the infrastructure is an absolute must for the success of the project.

In order to install the required eosio software, register to become a producer, and start the actual service, specific permissions and requirements are needed from the university's Information and Communication Technology Services department to perform these steps on the server. As it stands, the break in the academic year has not allowed further progress with accessing the requisite permissions necessary to set up the software on the server. In preparation, the process of cloning the software and simulating a virtual eosio environment has been completed on a regular laptop, in anticipation of later access to the server.

The next major milestone would be finding the necessary human capital to embark on the project and ensure its success. Although some students showed strong initial interest in producing academic papers on EOS related topics, unfortunately, other topics were chosen. Currently, the only academic research conducted in association with this project is this current dissertation that assesses the state of DLT scalability, motivates why the DPoS protocol is scalable and desirable and proposes the BP node project (DPoS participation) to enhance student learning and EOS decentralisation. With a new academic year on the horizon, the findings in this dissertation can motivate to relevant academics how embarking on such a project can benefit students, amongst a host of other benefits. For example, the masters in financial technology degree has a financial engineering module in which this project would help foster experiential learning.

6.5 Risks

Like with any nascent technology, there are significant risks. The biggest risks worth highlighting is the fact that rapid technical advances may lead to a new system implementation, no longer utilising the DPoS protocol. Since the project is built around being a BP in the DPoS protocol, a new system realisation could render previous work and effort on becoming a BP redundant. Even in light of this risk, the entire experience will again still help students to learn the valuable lesson of how technology can rapidly shift the working landscape. There will be room for the project to pivot and decide how best to continue going forward.

7 CONCLUSION

7.1 Summary

Distributed ledger technology (DLT) is valuable. We have shown that properties afforded through DLT, such as censorship resistance and transparency, are valuable for a variety of reasons, notably to the financial services sector. DLT can enable a faster, cheaper and more transparent distribution of financial services to a global population.

Distributed ledger technology is currently not scalable. We explored the DLT tri-lemma - it is very difficult to provide all three tenets: security, decentralisation and scalability. Since every single node has to process every single transaction in an open DLT system, throughput is fundamentally capped by the processing ability of the weakest node. This redundancy ensures system security and decentralisation, but has left current systems without sufficient scalability to handle mass global adoption, a requisite to ratify the true value of the technology. To scale these systems at a base level, we can either devise a mechanism such that not every node has to process every transaction, or we have to increase the workload required by each node in the system.

Ethereum 2.0 is on its way aiming to provide scalability through a number of complex solutions. Sharding is a mechanism to be implemented that shards the network so that not every node has to process every transaction. Proof of Stake (Casper) is the consensus mechanism implemented in tandem to incentivise the correct behaviour in a sharded network and reduce total electricity consumption of the network. State channels and Plasma are other layer 2 solutions also aimed at scaling the network. The highly complex nature of these solutions leaves a circumspect timeline on the actual delivery of these scalable solutions.

EOS is a scalable DLT platform. The Delegated Proof of Stake (DPoS) protocol allows 21 powerful Block producers (BPs) to run the network. BPs coordinate in a round-robin fashion to produce blocks and ensure a scalable system.

EOS voting power is centralised and the pool of block producers are not diverse. This presents concern around the decentralisation of the platform. The skewed voting power distribution is by design. Strong economic incentives theoretically ensure this is not an issue. Current BP candidates are not sufficiently diverse and this brings to light concerns regarding collusion between BPs.

Vote collusion is likely occurring between top EOS BP candidates. Unsupervised machine learning techniques showed tight clusters indicating a highly abnormal similarity in the votes received by 2 distinct groups of BPs of Chinese origin.

Universities are well positioned to become BPs. The networking infrastructure and human capital present at universities enable them to join as candidate BPs. The main motivation for becoming a BP would be enhancing the student learning experience through participation in a real-world engaging project. This would help students gain valuable tacit knowledge. Furthermore, university reputation would help add to the trust and diversity of the pool of BPs, and hence decentralisation of EOS. Revenue received from the project could be used to boost education efforts across communities.

7.2 Contribution

We have provided research into the current state of scalability in DLT systems and motivated why scalability is necessary for the continued growth of the sector. Specific research was conducted on the decentralisation of the DPoS protocol in EOS. Specifically, an unsupervised machine learning analysis uncovered abnormal similarity in votes received by certain suspected collusive BPs. Based on research and understanding of the DLT space, we drafted a proposal and motivated why universities should become block producers to enhance student learning and further research in the DLT space.

8 FUTURE WORK AND SHORTCOMINGS

This dissertation failed to produce a working implementation of an EOS BP. With no continued interest in the project from any students or academics, the implementation will likely never be completed. It was envisaged that the dissertation would showcase how UCT had successfully become a BP and used this experience to enhance student learning and contribute to widespread community education. Based on the dissertation documenting the steps of this project and areas of success, universities would be more inclined to follow suit and replicate the proposed model.

Another shortcoming of this dissertation was the failure to examine scalability solutions in platforms other than ethereum or EOS. There are many other notable platforms aiming to address scalability, such as Hyperledger, Dfinity and Cardano to name just a few. Future work would should include assessing the scalability ideas presented in these projects and their respective trade-off's.

ACKNOWLEDGMENTS

I would like to thank EOS New York, for the extensive insight given regarding the intricacies of block producing on the EOS mainnet. In particular I would like to thank Warrick Fitzgerald, the lead blockchain engineer at EOS New York for his personal visit to Cape Town in an effort to help establish the EOS CT community.

Finally I would like to sincerely thank my project supervisor, Professor Co-Pierre, for his constant support.

REFERENCES

[1] 2017. DPOS Consensus Algorithm - The Missing White Paper. (2017). https://steemit.com/dpos/@dantheman/dpos-consensus-algorithm-this-missing-white-paper
[2] 2018. EOS Block Producer FAQ. (2018). https://medium.com/@bensig/eos-block-producer-faq-8ba0299c2896
[3] 2018. EOS.IO Technical White Paper v2. (2018). https://github.com/EOSIO/Documentation/blob/master/TechnicalWhitePaper.md
[4] 2018. Ethereum Sharding Biweekly Development Update. (2018). https://medium.com/prysmatic-labs/ethereum-sharding-biweekly-development-update-9-prysmatic-labs-f2b1ad55e825
[5] 2018. Introducing the "Minimal CBC Casper" Family of Consensus Protocols. (2018). https://github.com/cbc-casper/cbc-casper-paper/blob/master/cbc-casper-paper-draft.pdf

[6] 2018. PROJECT KHOKHA - Exploring the use of distributed ledger technology. (2018). https://www.resbank.co.za/Lists/News%20and%20Publications/Attachments/8491/SARB_ProjectKhokha%2020180605.pdf

[7] 2018. Top 4 Ethereum Mining Pools Account For 75Distribution. (2018). https://bitrazzi.com/top-4-ethereum-mining-pools-account-for-75-hashrate-distribution/

[8] 2019. EOS Block Producer Voting Statistics. (2019). https://eosauthority.com/voting

[9] 2019. On sharding blockchains. (2019). https://github.com/ethereum/wiki/wiki/Sharding-FAQs

[10] 2019. The real value of blockchains. (2019). https://blocktivity.info/

[11] Andreas M Antonopoulos and Gavin Wood. 2018. *Mastering Ethereum: Building Smart Contracts and Dapps*. O'Reilly Media.

[12] Steven M Bellovin and Michael Merritt. 1992. Encrypted key exchange: Password-based protocols secure against dictionary attacks. In *Research in Security and Privacy, 1992. Proceedings., 1992 IEEE Computer Society Symposium on*. IEEE, 72–84.

[13] Joseph Bonneau, Andrew Miller, Jeremy Clark, Arvind Narayanan, Joshua A Kroll, and Edward W Felten. 2015. Sok: Research perspectives and challenges for bitcoin and cryptocurrencies. In *Security and Privacy (SP), 2015 IEEE Symposium on*. IEEE, 104–121.

[14] Steven Brams and Peter C Fishburn. 2007. *Approval voting*. Springer Science & Business Media.

[15] Vitalik Buterin. 2016. Ethereum: Platform Review. *Opportunities and Challenges for Private and Consortium Blockchains* (2016).

[16] Vitalik Buterin and Virgil Griffith. 2017. Casper the friendly finality gadget. *arXiv preprint arXiv:1710.09437* (2017).

[17] Javier Sebastian Cermeño. 2016. Blockchain in financial services: Regulatory landscape and future challenges for its commercial application. *BBVA Research, Madrid, Spain* (2016).

[18] Ruzanna Chitchyan and Jordan Murkin. 2018. Review of Blockchain Technology and its Expectations: Case of the Energy Sector. *arXiv preprint arXiv:1803.03567* (2018).

[19] Mauro Conti, Ankit Gangwal, and Michele Todero. 2019. Blockchain Trilemma Solver Algorand has Dilemma over Undecidable Messages. *arXiv preprint arXiv:1901.10019* (2019).

[20] James Crotty. 2009. Structural causes of the global financial crisis: a critical assessment of the 'new financial architecture'. *Cambridge journal of economics* 33, 4 (2009), 563–580.

[21] Christian Decker and Roger Wattenhofer. 2015. A fast and scalable payment network with bitcoin duplex micropayment channels. In *Symposium on Self-Stabilizing Systems*. Springer, 3–18.

[22] Stefan Dziembowski, Sebastian Faust, and Kristina Hostáková. 2018. General state channel networks. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 949–966.

[23] Ittay Eyal, Adem Efe Gencer, Emin Gün Sirer, and Robbert Van Renesse. 2016. Bitcoin-NG: A Scalable Blockchain Protocol.. In *NSDI*. 45–59.

[24] Alan S Gerber, Donald P Green, and Christopher W Larimer. 2008. Social pressure and voter turnout: Evidence from a large-scale field experiment. *American political Science review* 102, 1 (2008), 33–48.

[25] Yossi Gilad, Rotem Hemo, Silvio Micali, Georgios Vlachos, and Nickolai Zeldovich. 2017. Algorand: Scaling byzantine agreements for cryptocurrencies. In *Proceedings of the 26th Symposium on Operating Systems Principles*. ACM, 51–68.

[26] JC Gower. 1971. ILLUSTRATION OF A NEW TECHNIQUE FOR COMPARING DIFFERENT DISTANCE ANALYSES. In *AMERICAN JOURNAL OF PHYSICAL ANTHROPOLOGY*, Vol. 35. WILEY-LISS DIV JOHN WILEY & SONS INC, 605 THIRD AVE, NEW YORK, NY 10158-0012, 280.

[27] Lynda Gratton and Andrew Scott. 2016. *The 100-year life: Living and working in an age of longevity*. Bloomsbury Publishing.

[28] Ye Guo and Chen Liang. 2016. Blockchain application and outlook in the banking industry. *Financial Innovation* 2, 1 (2016), 24.

[29] Dominik Harz and William Knottenbelt. 2018. Towards safer smart contracts: A survey of languages and verification methods. *arXiv preprint arXiv:1809.09805* (2018).

[30] Everett Hildenbrandt, Manasvi Saxena, Xiaoran Zhu, Nishant Rodrigues, Philip Daian, Dwight Guth, and Grigore Rosu. 2017. *Kevm: A complete semantics of the ethereum virtual machine*. Technical Report.

[31] Ghassan Karame. 2016. On the security and scalability of bitcoin's blockchain. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 1861–1862.

[32] Alboukadel Kassambara. 2017. Practical Guide To Cluster Analysis in R, Unsupervised Machine Learning. (2017).

[33] Eleftherios Kokoris-Kogias, Philipp Jovanovic, Linus Gasser, Nicolas Gailly, Ewa Syta, and Bryan Ford. 2018. Omniledger: A secure, scale-out, decentralized ledger via sharding. In *2018 IEEE Symposium on Security and Privacy (SP)*. IEEE, 583–598.

[34] Nir Kshetri and Jeffrey Voas. 2018. Blockchain-enabled e-voting. *IEEE Software* 35, 4 (2018), 95–99.

[35] Leslie Lamport, Robert Shostak, and Marshall Pease. 1982. The Byzantine generals problem. *ACM Transactions on Programming Languages and Systems (TOPLAS)* 4, 3 (1982), 382–401.

[36] Matthias Lei. 2015. Exploiting Bitcoin's Topology for Double-spend Attacks. (2015).
[37] Bloomberg L.P. 2018. Market Capitalisation of eos from 01/1/2018 to 01/1/2019,. (2018).
[38] Bloomberg L.P. 2018. Total Market Capitalisation of cryptocurrencies from 01/1/2017 to 01/1/2019,. (2018).
[39] Patrick McCorry, Siamak F Shahandashti, and Feng Hao. 2017. A smart contract for boardroom voting with maximum voter privacy. In *International Conference on Financial Cryptography and Data Security*. Springer, 357–375.
[40] Satoshi Nakamoto. 2008. Bitcoin: A peer-to-peer electronic cash system. (2008).
[41] Moni Naor and Moti Yung. 1989. Universal one-way hash functions and their cryptographic applications. In *Proceedings of the twenty-first annual ACM symposium on Theory of computing*. ACM, 33–43.
[42] M Parkin, M Kohler, L Lakay, B Rhodes, A Saayman, V Schöer, F Scholtz, and K Thompson. 2010. Economics: Global and Southern African Perspectives. *Pinelands: Philippa van Aardt* (2010).
[43] Joseph Poon and Vitalik Buterin. 2017. Plasma: Scalable autonomous smart contracts. *White paper* (2017).
[44] Joseph Poon and Thaddeus Dryja. 2016. The bitcoin lightning network: Scalable off-chain instant payments. *See https://lightning. network/lightning-network-paper. pdf* (2016).
[45] Michel Rauchs, Andrew Glidden, Brian Gordon, Gina C Pieters, Martino Recanatini, Francois Rostand, Kathryn Vagneur, and Bryan Zheng Zhang. 2018. Distributed Ledger Technology Systems: A Conceptual Framework. (2018).
[46] Zhijie Ren, Kelong Cong, Taico Aerts, Bart de Jonge, Alejandro Morais, and Zekeriya Erkin. 2018. A scale-out blockchain for value transfer with spontaneous sharding. In *2018 Crypto Valley Conference on Blockchain Technology (CVCBT)*. IEEE, 1–10.
[47] Peter Ruppel. 2018. eos voting. Who is pulling the strings? (2018). https://medium.com/@_tlabs/eos-voting-who-is-pulling-the-strings-8a4944c45e61
[48] Yonatan Sompolinsky and Aviv Zohar. 2015. Secure high-rate transaction processing in bitcoin. In *International Conference on Financial Cryptography and Data Security*. Springer, 507–527.
[49] Josh Stark. 2018. Making Sense of Ethereum's Layer 2 Scaling Solutions: State Channels, Plasma, and Truebit. (2018). https://goo.gl/ZhbWdA
[50] Don Tapscott and Alex Tapscott. 2017. How blockchain will change organizations. *MIT Sloan Management Review* 58, 2 (2017), 10.
[51] Robert Tibshirani, G James, D Witten, and T Hastie. 2013. An introduction to statistical learning-with applications in R. (2013).
[52] Manny Trillo. 2013. Stress Test Prepares VisaNet for the Most Wonderful Time of the Year. (2013). https://www.visa.com/blogarchives/us/2013/10/10/stress-test-prepares-visanet-for-the-most-wonderful-time-of-the-year/index.html
[53] Marko Vukolić. 2015. The quest for scalable blockchain fabric: Proof-of-work vs. BFT replication. In *International Workshop on Open Problems in Network Security*. Springer, 112–125.
[54] Robert J Weber. 1995. Approval voting. *Journal of Economic Perspectives* 9, 1 (1995), 39–49.
[55] Gavin Wood. 2014. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum Project Yellow Paper* 151 (2014).
[56] Andrew C Yao. 1982. Theory and application of trapdoor functions. In *Foundations of Computer Science, 1982. SFCS'08. 23rd Annual Symposium on*. IEEE, 80–91.
[57] Guangsheng Yu, Xu Wang, Xuan Zha, J Andrew Zhang, and Ren Ping Liu. 2018. An Optimized Round-Robin Scheduling of Speakers for Peers-to-Peers-based Byzantine Faulty Tolerance. (2018).
[58] Haifeng Yu, Michael Kaminsky, Phillip B Gibbons, and Abraham Flaxman. 2006. Sybilguard: defending against sybil attacks via social networks. In *ACM SIGCOMM Computer Communication Review*, Vol. 36. ACM, 267–278.
[59] Dmitrii Zhelezov. 2018. A primer on blockchain design. (2018).
[60] Joe Zou, Zhongli Dong, Allen Shao, Peng Zhuang, Wei Li, and Albert Y Zomaya. 2018. 3D-DAG: A High Performance DAG Network with Eventual Consistency and Finality. In *2018 1st IEEE International Conference on Hot Information-Centric Networking (HotICN)*. IEEE, 262–263.
[61] Guy Zyskind, Oz Nathan, and Sandy Pentland. 2015. Decentralizing privacy: Using blockchain to protect personal data. In *Security and Privacy Workshops (SPW), 2015 IEEE*. IEEE, 180–184.