Introduction to Voting and the Blockchain: some open questions for economists

Amrita Dhillon, Grammateia Kotsialou, PeterMcBurney and Luke Riley

WORKING PAPER SERIES

Centre for Competitive Advantage in the Global Economy

Department of Economics
Introduction to Voting and the Blockchain: some open questions for economists

Amrita Dhillon*  Grammateia Kotsialou†  Peter McBurney‡  Luke Riley§

King’s College London, UK

Abstract

This work discusses the potential of a blockchain based infrastructure for a decentralised online voting platform. When compared to paper based voting, online voting can vastly increase the speed that votes can be counted, expand the overall accessibility of the election system and decrease the cost of turnout. Yet despite these advantages, online voting for political office is subject to fraud at various levels due to its centralised nature. In this paper, we describe a general architecture of a centralised online voting system and detail which areas of such a system are vulnerable to electoral fraud. We then proceed to introduce the key ideas underlying blockchain technology as a decentralised mechanism that can address these problems. We discuss the advantages and weaknesses of the blockchain technology, the protocols the technology uses and what criteria a good blockchain protocol should satisfy (depending on the voting application). We argue that the decentralisation inherent in the blockchain technology could increase the public’s trust in national elections, as well as eliminate voter impersonation and double voting. We conclude with a discussion regarding how economists and social scientists can collaborate with the blockchain community in a research agenda on the design of efficient blockchain protocols and new voting systems such as liquid democracy.

1 Introduction

Elections are key to the democratic process. However, the integrity of elections around the world are continuously questioned both by independent observers and the voters themselves. For instance, in the latest Election Integrity Project review [38], independent researchers from Harvard and Sydney universities ranked only 19.5% of countries very high for election integrity. Furthermore, the most recent World Values survey [33] included a question on the fairness of elections, where more than 25% of voters in about 76% of the countries surveyed, believe that election officials are often unfair (biased).

The issue of questionable election integrity can affect both developing and developed countries. In the developing world, examples include the Honduras general election (26th November 2017) which suffered from major irregularities at the counting stage leading to the Organization of American States (OAS) to recommend that the election should be rerun [36]. A further example is elections in Albania that continue to suffer from vote buying allegations [19]. Neggers shows in [33] how the composition of the team of electoral officers in India can causally shift votes towards favoured parties, with magnitudes large enough to change election outcomes.

In the developed world, examples include a 2014 mayor election in Tower Hamlets, London, UK that had to be rerun as it was discovered (only after a court ordered investigation) that ballots were double

*Department of Political Economy, email: amrita.dhillon@kcl.ac.uk
†Department of Political Economy, email: grammateia.kotsialou@kcl.ac.uk
‡Department of Informatics, email: peter.mcburney@kcl.ac.uk
§Department of Informatics, email: luke.riley@kcl.ac.uk
cast or cast from false addresses [34]. Another example from a country ranked highly on institutional independence, is the 1984 grand jury investigation into voter fraud in New York, USA. This investigation uncovered large scale and systematic fraud in the primaries of two of the borough’s congressional districts between 1968 and 1982 (where 1000 to 2000 bogus registrants were discovered [28]). More recently, a US government study states that a weakness of the American system is that poll workers are not dependable or sufficiently trained [37].

To provide a high level of election integrity, an election system must balance the need for accessibility with the need to establish trust. An election authority must show that eligible voters can be easily registered and vote (to display accessibility). For countries with compulsory voting, easily accessible voting facilities are of even higher importance. Fraud can also arise from collusion between officials (entrusted with authority to run the election) such as ballot box monitors or other election insiders. Therefore trust in the system includes trust that these types of fraud will not happen. To do this, an election authority must show that every vote cast will be recorded accurately (to establish trust). The correct balance needs to be found between these two conflicting objectives because the more rigorous the identification and registration process, the less accessible the system is. On the other hand, if the system is not rigorous then it increases the scope for voter fraud.

In addition, the system should not be too costly to run and should deliver results as fast as possible. Note that all paper ballot elections take some time to count the results (usually overnight). But some elections, such as for Australia’s House of Representatives and Senate, vote counting can take an average of two weeks [6], due to the single transferable vote method that they use. Finally, the monetary cost of organising and securing the entire election can be very large. Thus we want an election to cost as little as possible but without compromising on security.

In this paper we examine how the decentralised nature of a new technology, the blockchain [32], can improve both the accessibility and trust properties of an online voting system. In Section 2 we focus on online voting systems, their general architecture and possible areas of manipulation. Section 3 describes an example of a blockchain voting system giving the general advantages and disadvantages of this technology. In Section 4 we describe the main blockchain protocols, while, in Section 5 we discuss the potential of implementing different types of blockchains in various voting applications. In Section 6, we provide a complete categorisation of the different ways that blockchain technology can be used in voting. Finally, in Section 7 we conclude with open questions in this area.

2 Electronic voting

Electronic voting (or e-voting) is the procedure of voting through the use of electronic devices. The two main categories of electronic voting is via offline voting (electronic voting machines - EVMs) and online voting, discussed in Section 2.1 and 2.2, accordingly.

2.1 Offline Voting

Electronic Voting Machines (EVMs) can be positioned at polling stations in private booths to digitally record votes of citizens. Therefore, these machines can replace the traditional paper ballot voting system. EVMs can be extended to include: (i) Voter-Verifiable Paper Audit Trails (VVPAT), which gives the voters a non-digital way to verify that their vote was recorded correctly, and (ii) End-to-End verifiability (E2EV), which means that a voter has the capability to make sure that their vote has been properly cast, recorded and tallied in the election system [24].

For election authorities, EVMs can be beneficial for the faster collection and the counting of the citizens’ votes. Adding perfectly working EVMs to an election would substantially reduce human errors,

---

1 We can view the voting system as a sequence of steps, each of which is vulnerable to some manipulation. Trust in the system implies that possibilities of manipulation are minimised at each one of these steps: Registration, Authentication at the polling booth (or electronic analogue), Casting of the vote at the polling booth (anonymity and privacy are important at this stage) and Counting.
result in more accurate outcomes, prevent fraud in polling stations, and minimise staffing costs that occurs when manually performing the election \[42\]. Additionally, enormous amount of paper and energy is saved, which would otherwise be used on ballots printing, their transportation and storage. Since EVMs allow for the automation of vote counting, the final results can be announced faster than in a paper ballot election. Note that same day results in paper-based elections can occur only with large numbers of counting staff. In some cases, manual vote counting can take even more than a week to complete due to the complexity of the counting algorithm used. For instance, the Australian national elections use the single-transferable-vote system for a large number of candidates (sometimes greater than 100), which may take up to a month to manually count.

Currently the majority of experts agree that EVMs can be built and operated in a satisfactory manner as long as they include VVPAT and a sufficient level of auditing (possibly with E2EV for the voters themselves) \[21\]. Despite the many ongoing issues with EVMs still in operation, studies show that both in India \[17\] and Brazil \[20\] there was reduced electoral fraud and even better public good provision (due to reduced government spending for elections), when EVMs were introduced. We argue however that EVMs are still a centralised online voting system where there is scope for fraud by insiders as well as by hacking of the EVMs. We come back to this issue later.

2.2 Online Voting

The use of the Internet has brought tremendous voting possibilities. Citizens can now vote digitally from any geographical location using a device connected to the Internet. One of the main advantages of online voting is accessibility for everyone: people with special needs can be part of the voting community by casting votes from their own home, or people abroad can avoid the stress related to posting their vote (searching for nearby post offices, missing postal vote letters, etc.).

Despite the tremendous convenience of using the Internet to vote, online voting for political office is difficult to implement due to the complex computer science issues that need to be resolved \[21\]. Such a system not only needs to maintain the integrity of the election, but it must do so in a way to convince the losing candidates that the winner is legitimate. At the same time, the system needs to maintain the secrecy of the voters' choices, to prevent voter coercion and minimise the possibility of vote buying.

Indeed, the main complications arising from online voting have to do with the conflicting objectives of maintaining voter secrecy and yet allowing auditability of the votes. In contrast, when the secrecy of transactions from the central authority is not important, applications such as online banking already exist and flourish. In the case of banks e.g. they keep many transaction records and audit logs, some of which are available to the user. These allow both the users and the banks to check for fraudulent activity. In voting, the more records an online system keeps on a voter, the more likely that voter secrecy will be compromised. Additionally, banks can use their transaction records and audit logs to undo the fraudulent transactions while any money lost can be written off as the cost of doing business. Undoing votes and writing off hacks in an online voting system for a political office is a much more delicate issue.

2.2.1 General Architecture

A (centralised) online voting system comprises of a series of steps which the user cannot physically observe or interact with due to its online centralised nature. To understand the main problems of online voting, we need to first understand its infrastructure. Below, we present and discuss a general architecture of an online voting system.

\[\text{There are currently multiple areas around the world without Internet coverage but this is something to be solved in the next years using new promising technological advances (e.g. see https://x.company/loon/).}\]
Eligible voters submit their votes to a server through a single point of entrance. At this stage, there is a collection of all the votes and confirmation of their validity (we assume that voters’ identity and eligibility has been validated before this stage). Then the collection of valid votes are stored in another server until they are counted at a fixed time after the end of the election. The favourite candidate is the output of the software system.

2.2.2 Areas of attack or manipulation in centralised systems

In this section, we describe some of the main areas of possible attack during the sequential process of centralised online voting. To be confident in the validity of the final result, we need to have guarantees that all areas of attack have been defended against:

1. Virus on voter’s device: Online voting implies the usage of a device connected to the Internet so that the voter is able to connect with the online voting server described previously. However, devices such as laptops or mobile phones can contain types of malware (digital viruses), which can delete or change a citizen’s vote before it is even sent to the server.

2. Denial of Service: As mentioned in the system architecture description, all votes are received by the online voting server from a single point of entrance, which can also be a single point of failure. The way a malicious entity can interfere with the election process at this stage is by sending an excessive number of requests to the server. Overloading the server can cause large delays or even cease the collecting of new votes during the duration of the attack.

3. Private code: The programming code used to execute the collection, storage and counting of the votes is usually kept private. Even though sometimes online voting providers open source their code, there is no guarantee that the code manuscript they publish matches the one running in their live product, which means that a malicious election authority could insert code to change the vote at any of the validation, storage or counting stages.

4. Invited stakeholders: During the counting stage, there is usually only a limited number of invited stakeholders to observe this process. But these invited stakeholders need to be technically knowledgeable enough to identify if any malicious action has occurred. Even if they are technically knowledgeable, they need to be given sufficient time and access to perform a full evaluation.

In the next section we discuss the main ways in which a blockchain can handle the problems above.

3 Blockchain technology

A blockchain contains groups of data (blocks) that are sequentially linked to each other using cryptography, so that these blocks form an ordered chain. The idea of a blockchain is that data is recorded into blocks in an “append only” manner. Once data is added onto the blockchain, it is extremely difficult to modify when compared to storing data in a centralised database. Blockchain technology was created for a distributed network of nodes (computers) with no single authority in charge. These nodes are connected in a peer-to-peer manner where an individual node is not necessarily directly connected
to every other node (see picture below) and may not even know the graph of connections (nodes may only know the peer to which they are connected to). Each node in the network contains a copy of the blockchain and the rules on how the nodes reach an agreement on which data will be saved onto the blockchain (which we describe in Section 4).

The data stored on a blockchain occurs from events. For example, if a user signs into a website connected to a Bitcoin node and uses the website to send Bitcoin to a receiver, the connected node will forward data of this transaction to all the nodes of the Bitcoin network. After this data (the Bitcoin transaction) is validated, then it is added into the blockchain as part of a new block. This new block is distributed around the Bitcoin node network so that eventually all the nodes have the same blocks (of data) in the same order. For another example, a more technically savvy user who is running her own Bitcoin node, would not have to sign into a third party website to transfer Bitcoin but could instead send her Bitcoin transaction data to her own node. Then, as before, this node would forward the transaction to the entire Bitcoin node network to await validation and inclusion into a new block.

3.1 Using blockchain technology in voting: an example

In this section, we describe one possible design of an online voting system based on blockchain technology. In this example, a voter needs to connect with one of the nodes of the blockchain network to submit her vote. Upon receiving the user’s vote, the chosen node forwards the vote to the rest of the blockchain network nodes (to become part of a shared unrecorded votes pool). The next block to be added in the chain will include a subset of the unrecorded votes. Every new block (with confirmed votes) that is created will be sent to every other node of the network so that all nodes update their copy of the blockchain, allowing them to agree on what valid votes have been recorded.

As discussed previously, there are two different ways to connect to a node: (i) if a voter knows the location of a particular node, then she can directly connect to that node to cast her vote. This node could be on her PC or another peer. This connection method is only recommended for users with advanced technical skills, or (ii) a voter can sign into a website interface that connects to the blockchain network.
Since blockchain node networks are distributed, there could be multiple different interfaces connecting into the same blockchain, therefore there would be multiple access points for the citizens to vote.

3.2 Advantages of the technology

There are many advantages of blockchain and distributed ledger technologies, including the following:

1. Censorship resistance: Data stored on a blockchain is very difficult to censor, due to the distributed network of nodes (that may be run by many different stakeholders of the system) and the way these nodes communicate between them to reach agreement on data (known as consensus protocols). For instance, if a user sends a cast vote transaction to a node, and the user discovers that this vote is subsequently not recorded into the blockchain (i.e. the vote has been lost or censored), then the user can continue resending the vote to another node of the network until the vote is recorded. Note that this resending process can be performed automatically to avoid overburdening the user. Furthermore, once a vote is recorded in a block, it is significantly difficult to modify it. To do so, a malicious node would have to accurately edit the following: the vote, the meta data associated to the vote’s block, and every link between successive blocks in the chain. The malicious node would have to perform this procedure on every node of the network at the same time to avoid detection, which becomes increasingly difficult as more nodes join the blockchain network.

2. Decentralised nature: Compared to standard centralised voting systems (where there is a single authority monitoring actions), a blockchain based voting system can allow some level of decentralisation on the monitoring and implementation of the system’s procedures. Examples where multiple nodes can have a role include: (i) the collection of votes; (ii) the validation of votes; and (iii) the counting of votes. Note that there should be a variety of stakeholders with conflicting interests running nodes to disincentivise collusive attacks (see discussion in Section 5.3).

3. Smart contracts: Blocks of data can also contain smart contracts. A smart contract is a piece of code that will execute functions when particular data transactions are added to a subsequent block. Thus the more smart contracts added into the blockchain, the more features this blockchain can offer to its users. Examples of useful features for voting applications, that can be contained within smart contracts, include: (i) creating and holding the list of eligible voters; (ii) obfuscating interim results; (iii) decentralising the counting of votes, i.e. every node can check that the advertised tally of the election is correct according to the data stored in the blockchain; and (iv) counting encrypted votes without decrypting them. Note that if an application desires vote secrecy, the encryption of votes needs to happen before the votes enter the blockchain network. The reason is that as soon as votes are submitted unencrypted to a blockchain network, they will be permanently recorded as unencrypted in the block.

3.3 Weaknesses of the technology

Despite the tremendous possibilities this technology can bring, we outline the main weaknesses with respect to voting applications.

1. Blockchain network interfaces: In the beginning of Section 3, we mention that a user can connect with the blockchain network by signing into a website interface connected to a blockchain node. Even though this is a solution for how the average person can use this technology, malicious entities may create interfaces that change or steal votes before the vote arrives at a blockchain node. Note that this is not a weakness of the blockchain technology, but a weakness of how a user can interact with the blockchain. To create a secure blockchain based voting system, we need to be aware of this issue and find ways around it (such as making sure a voter connects to multiple nodes to confirm that her vote was correctly recorded on the blockchain).

2. Partially blockchain-based voting systems: To get the full potential of this technology, it needs to be used in as many parts of the voting process as possible. By restricting its use to minor parts, it
also restricts the voting system's ability to resist attacks. For example, storing only the final election result in the blockchain implies that there would be a secure record of the election result. However, this example does not imply that the result has not been manipulated before it was calculated. Recall that blockchain can be viewed as an underlying level of additional security to the current online voting systems. If we fail to use blockchain technology to its full extent, the security flaws of centralised online voting systems can still occur in sections that blockchain has not been used.

3. Majority attack: In a blockchain network, it is possible that a single node or a coalition of nodes can have the majority of the network power. Depending on the consensus protocol used, blockchain network power can be defined with different methods, e.g., computational power or economic stake held. If malicious nodes have the majority of power in a blockchain voting system, then they also have the ability to discard valid votes or add invalid votes into a block. However, as long as there is at least one honest node (and an honest interface to this node), malicious activities can be discovered and publicly announced so that they can be investigated by independent third parties. Therefore, even if majority attacks can occur in a blockchain-based voting application, they can be used as a tool for identifying what type of manipulation attempts have occurred and by whom.

4. Running Cost: Validating data through one (centralised) server should generally be less costly compare to validating data through a decentralised system where the evaluation happens by multiple independent servers. However, the latter can improve the trust in the system simply because the same data is being checked by multiple parties. In other words, there is a trade-off between running cost of decentralisation and trust, therefore we can focus on what level of cost is acceptable so that a system can be considered as trusted. Note that the level of decentralisation in systems using blockchain technology highly depends on the consensus protocol used, which we discuss in the next section.

To summarise, the blockchain can be viewed as a way to increase the difficulty of electoral fraud. For example, if we focus on the validation of votes - the difference between a centralised system and the blockchain is that the latter has multiple independent validators who would be involved in making sure that votes are valid. Similarly, for vote storage, it would be much more difficult to modify results in a blockchain system than in a centralised system simply because it would require the attacker to control a large percentage of the resources on the network. The key features of the blockchain, namely the reliance on multiple independent validators and the "append only" nature of the transactions, together with the access to the code by multiple nodes can create big disincentives to potential attacks. While collusive attacks are possible, it is much more difficult to do so. There are still some open questions about the consensus protocols, such as how to incentivise validators and how to design the protocols in a way to discourage collusion between the nodes. We introduce these protocols in the next section.

4 Reaching Consensus on Data via Blockchain Technology

The set of rules on how the users of a blockchain can reach agreement on what data is in the blockchain are known as consensus protocols [3], which consist of a sybil control mechanism and a data agreement protocol (rules on how the next block is agreed). There are many consensus protocols available and each blockchain implements one. Each consensus protocol must specify how validators for transactions are chosen and how many validators are needed. We describe features of (standard) blockchain sybil control mechanisms and discuss how they can be combined with a data agreement protocol as well as their potential applicability to voting.

- **Proof of Work (PoW):** This is the first blockchain sybil control mechanism as it is used by Bitcoin [32], the first digital blockchain implementation. The idea behind this sybil control mechanism is that nodes race to solve a complex computational problem, where the first node to solve it is allowed to publish a block of data to be added to the blockchain, where a block consists of

---

3 A sybil control mechanism give no advantage to users who create multiple accounts.
a set of previously unconfirmed transactions. The likelihood of an account solving the computational problem first is proportional to the computational power that this account has access to (compared with the total computational power of the network). This means that creating multiple accounts gives a person no advantage as computational power cannot be used in parallel by multiple accounts.

The Proof of Work data agreement protocol consists of details on the exact computational puzzle to be solved⁴, how to handle multiple blocks being released at the same time⁵, and rules to identify what are valid blocks and transactions, e.g.: each block must be of a certain maximum size; each sender of cryptocurrency must have the cryptocurrency in the first place; a sender of cryptocurrency cannot attempt to send the same coins to multiple different places; etc.

In Proof of Work, anyone can become a validator -essentially this is a contest where nodes which join the blockchain solve a complex computational puzzle and the one who solves it first gets to be the first to publish. Nodes are incentivised to attempt solving this complex computational problem with some amount of the native cryptocurrency of the blockchain. But every node who loses this race, essentially has no gain from all of the energy they have wasted. Budish shows that conceptually an equilibrium requires a zero profit condition to be fulfilled: nodes enter and attempt to create the next block until there are no profits to be made, as well as an incentive compatibility condition which ensures that no one wants to carry out a majority attack. The latter implies that the cost (in terms of computing power) of doing so must be higher than the gains. He shows that these two conditions imply that the equilibrium per block payment for validation must be large relative to the one shot benefits of an attack. This constraint puts an economic limit on the applicability of this consensus protocol.

• **Proof of Stake (PoS):** In this sybil control mechanism, the likelihood of a node being assigned to create the block is random and proportional to their cryptocurrency holdings (compared with the total cryptocurrency stake in the network). This means that creating multiple accounts gives a person no advantage as their cryptocurrency holdings would have to be split between their accounts.

For Proof of Stake data agreement protocols, there does not have to be a complex computational problem for the nodes to solve and so they can be significantly more energy efficient than Proof of Work versions. A possible data agreement protocol for Proof of Stake can be a lottery among nodes where the winner’s prize is to produce the next block and each node has as many lottery tickets as it has cryptocurrency coins. Thus nodes with large stake have a higher chance to produce the next block. Other proposed proof of stake data agreement protocols are Tendermint [16], Casper [8], Oroborous [2, 16, 23] and Avalanche [40].

As in Proof of Work, nodes are incentivised to have a stake in the system because, should they cre-

---

⁴The computational problem to solve in Bitcoin is cryptographic in nature and is as follows. A node runs a pre-agreed hash algorithm, that takes as input the data of the block the node wants to create, and attempts to find an output string (series of characters) where the first $x$ digits are zeros. This string, known as hash key, links this block with the previous block in the chain. This connection is achieved as one of the input of the hash algorithm is the hash key of the previous block added in the chain. The following shows the main inputs in the hashing algorithm, where the nonce (the only non-fixed input that a node can adjust to get the desirable string) is an integer number chosen by the node running the hash algorithm.

<table>
<thead>
<tr>
<th>Hash Algorithm Input</th>
<th>$\Rightarrow$ 00…01@&amp;%@96%81</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Hash key of previous block</td>
<td></td>
</tr>
<tr>
<td>- Timestamp</td>
<td></td>
</tr>
<tr>
<td>- Transactions</td>
<td></td>
</tr>
<tr>
<td>- Nonce</td>
<td></td>
</tr>
<tr>
<td>:</td>
<td></td>
</tr>
</tbody>
</table>

⁵Bitcoin uses the longest chain rule. If multiple blocks have been released at the same time, each node should continue to attempt to build on the blockchain it believes is the longest (or randomly select a chain if it thinks there are multiple chains of equal size). If at any point a node receives a longer chain than the one it is currently building on, the node should switch to the new longer chain.
ate a block, they will be rewarded with some amount of the native cryptocurrency of the blockchain. Unlike Proof of Work, if a node is not chosen to create a block, they do not waste energy, they only have an opportunity cost of having their capital deployed elsewhere. This implies in particular that the problem pointed out by Budish [7] could be overcome with a different design.

- **Delegated Proof of Stake (DPoS):** In this sybil control mechanism, the node assigned to create the next block depends on the amount of votes received from accounts with cryptocurrency holdings, where new elections can occur every few minutes. This means that creating multiple accounts gives a person no advantage as firstly one cryptocurrency coin corresponds to one vote, and secondly creating two candidate accounts will split the electorate's votes between these two candidates.

  Possible data agreement protocols for Delegated Proof of Stake could be a round robin format between the \( k \) nodes with the highest vote in the previous election, where each of the elected nodes take turns producing the next block (in a way that is not computationally difficult to achieve) and the other \( k - 1 \) elected nodes vote on whether it is valid or not. If a pre-agreed threshold number of 'accept' votes is reached, the block is added to the chain. Once the election cycle has been completed (after each block producing node has produced a pre-agreed number of blocks), then the next set of block producing nodes are elected and this process starts again. Additionally, the \( k \) elected nodes could reuse a Proof of Stake data agreement protocol where the only participants in this election cycle are themselves.

- **Proof of Authority (PoA):** In this sybil control mechanism, a node can create a block only if it is an identified (approved) account, where it can only be approved by the current nodes with authority in the blockchain. This approval will be linked to a real life identification meaning creating multiple accounts will not be possible. Possible Proof of Authority data agreement protocols can then be not as computationally difficult as the ones proposed for Proof of Stake or Delegated Proof of Stake.

  Note that this method of consensus is suitable for a permissioned network as the trust is centralised to one node or a group of nodes (as opposed to PoW where trust is distributed). Such a consensus protocol should only be used when there is a high level of trust in each block producing node. PoA can operate through the same round robin format described for the DPoS consensus protocol. However, PoA differs (from DPoS) because only nodes with certain permissions can vote to decide if a new node can become a block producer or if a current node loses its block producing status. This consensus protocol can tolerate up to one third of the network nodes acting in a malicious or faulty manner. However, one of the mechanism's drawbacks is that by adding more nodes, the performance of the network can dramatically slow down. This is due to the large amount of messages that the nodes exchange between them, when they vote on whether they accept a new block or not [6]. A block is considered valid (accepted) only when a specific number of nodes (usually two thirds plus one) vote in favour of the block. Note that this is in contrast to PoW, where a block is considered valid until its validity is challenged.

The main criteria of a good consensus protocol for voting are the following: (1) consensus among sufficiently many nodes to make it costly for a malicious attack, including collusive attacks, (2) minimisation of computing power and (3) rewards for the block producing nodes if there is free entry in the blockchain network, e.g. in a public blockchain. At this point, there is a strong parallel to be made between mechanism design and designing the consensus protocol. The objective is to minimise the cost of running the system (including rewarding the block producing nodes) subject to incentive compatibility conditions for truth telling (including for coalitions of stakeholders) and individual rationality conditions. In contrast, for a blockchain running in a permissioned network, the main question is how to choose the approved nodes, e.g. ensuring representation of all political parties in an election.
5 Potential of a blockchain based infrastructure online voting

In this section, we compare different types of blockchains and discuss their potential on implementing various voting applications. We also argue that blockchain technology could eliminate voter impersonation and double voting as well as provide the infrastructure for more engaging voting systems.

5.1 Public versus permissioned blockchain

The main difference between public and permission blockchains is that in a public blockchain: (i) anyone can run a node on the network to read the blockchain data and create new blocks, and (ii) anyone can send and receive transactions recorded in the blocks. In a permissioned blockchain, both running a node and being able to send a transaction require special permissions. Additionally, public blockchains are characterised by a cryptocurrency reward given to the nodes that create new blocks. This reward method is used to incentivise nodes to operate truthfully as malicious behaviours should most likely decrease the value of node's cryptocurrency rewards. Permissioned blockchains do not necessarily have a cryptocurrency reward for block creation as the incentive for the nodes to act truthfully is to keep their reputation high.

5.2 Public blockchains for voting?

In a public blockchain anyone in the world could theoretically vote, or create and verify blocks that include votes. A smart contract could enforce a list of eligible voters, but a smart contract would not be able to enforce which nodes create and verify blocks. Therefore, a public blockchain is not ideal for elections where a lot of transparent data is to be placed on the blockchain and the management of the election needs to remain in a particular area, e.g. a country’s national elections. A voting system could still use a public blockchain for national elections, if the voting record was hashed onto the chain, as hashing would not reveal any specific information on voters. However, this type of blockchain voting system does not use the blockchain technology to its full extent and so the benefits of the technology are not all gained (see Section 6). On the other hand, public blockchains could be well suited for voting on matters with a global interest. Examples can include petitions on matters where any citizen of the world has an interest, e.g. air pollution, global warming, people's rights, etc. But even if we identify voting applications for public blockchains, what consensus protocols should be used?

The Proof of Work consensus protocol has significant drawbacks due to the low number of transactions that can be recorded per second and the enormous energy level that it requires. On the other hand, even for protocols such as Proof of Stake or Delegated Proof of Stake, which are considered to be energy efficient (compared to Proof of Work), we need to carefully design the incentivisation method for nodes to act truthfully, to prevent malicious disruption of the voting system when such disruption could be worth a significant amount. For example, Budish showed that there is an economic limit on how important the Bitcoin blockchain could be. Therefore, we set the following open question: Can we design energy efficient consensus protocols for a public blockchain voting systems where nodes are always incentivised to create and validate blocks containing all the valid votes cast?

5.3 Permissioned blockchains for voting?

Until the previous open question is resolved, permissioned blockchains seem to have a higher possibility to be used in important elections because the incentivisation scheme for stakeholders running nodes is easier to understand. For example, if a node acts maliciously, then other nodes can publicly announce this information and ruin an identified node's reputation. A main possible drawback of a permissioned blockchain based voting system is that all, or the majority, of the nodes in the network could still be under the control of a single authority. In this case, the difference with the current online voting

\*Above this limit, disruption in the system could worth more than the gains from preserving it.
systems will be minor. Alternatively, a permissioned blockchain with a large variety of stakeholders operating a node can allow for decentralisation of control. An open question on this topic is therefore: How many different stakeholders are required before an acceptable level of decentralisation is reached? Who should these stakeholders be? And how does this acceptable decentralisation level change for different voting applications?

Similar questions occur for corporate voting. In most corporate shareholding structures, minor shareholder do not usually have voting rights due to the large amount of bureaucracy associated with each shareholder voting. For this reason, decisions within a company are usually taken only by the main shareholder(s), which creates a more centralised and maybe undemocratic environment for the minor shareholders (especially within companies where there is a large number of minor shareholders). However, if all shareholder rights (including voting rights) can be automatically managed by a permissioned blockchain based platform, this bureaucratic burden would theoretically be minimised. But how many nodes should such a platform have to be sure that minor shareholders can participate in a company's decisions?

5.4 Voter registration

A key property underlining the integrity of democratic elections, is the concept of “one person, one vote”. Unfortunately, there currently exist electoral systems which do not guarantee that this property holds. Fraudulent votes can be cast due to exploiting either (i) the legal allowance of individuals being registered in multiple constituencies (e.g. see [12, 35]), or (ii) inaccuracies in the electoral roll (e.g. see [11, 35]). For example, a recent study in UK by the Electoral Commission [13] states that the UK electoral roll is approximately 9% inaccurate as it records incorrect or old address for some citizens. Furthermore, authors of [31] highlight that these inaccuracies may lead to voter turnout misreports (in terms of under-representing it) by almost 10%.

The source of this problem seems to be the multiple centralised databases (where each database is controlled by one constituency) which do not communicate with each other. For this reason, a blockchain based voter registration system could be ideal to resolve such issues, as it can offer the following: synchronised eligible voter databases which allow alterations under consensus among constituencies. For example, when a voter moves from constituency $A$ to constituency $B$, then constituency $B$ confirms this move with constituency $A$ before proceeding to data alterations. In addition, note that (a) all such actions would be recorded as blocks on the chain, thus there would be a history of the address changes of an individual, which can subsequently minimise the electoral roll inaccuracies due to incorrect addresses, (b) every constituency can send vote casting notifications of its own residents (to all nodes of the blockchain network) so that voter impersonation or double voting cases to be eliminated.

5.5 Fairer voting and democratic political systems

Can a blockchain based infrastructure open new ways for the implementation of fairer voting mechanisms or even new more engaging and democratic political systems? Below we focus on liquid democracy and the single-transferable-vote and discuss their applicability potential using this technology.

Liquid (or delegative) democracy allows a citizen to either vote directly on a topic or transfer her voting rights to a trustee of her choice. For this reason, such a political system would work better on continuous referenda, e.g. Switzerland is known for its democratic system, i.e. a (semi) direct democracy, where citizens are called to directly vote on some issue. Would the Swiss system be improved with a liquid democracy adoption? Apart from the fact that liquid democracy combines the advantages of representative and direct democracy, it can theoretically increase voter turnout as those who abstain...
(because they do not know what to vote for) have now an additional option to delegate their vote, which subsequently means that more informed voters influence the final outcome. Additionally, it can give the electorate more options as they can delegate their vote to different representatives for different topics and even declare a preference order over other representatives (in case their first choice trustee does not vote) [25].

Due to its ability of keeping a highly immutable record of any action within a system, a blockchain based infrastructure can allow for the implementation of a liquid democracy system as it can keep a safe and transparent record of all the delegations of a vote (the chain among voters that continuously delegate voting rights). Smart contracts can allow for the automatic delegation of votes to individuals and the automatic withdrawal of votes from their final destinations (candidate's addresses) in order to transfer them to other candidates. The latter feature would be ideal for the implementation of fairer voting mechanisms such as the single-transferable-vote, where the votes received by one candidate need to be redistributed in case she is eliminated. Note that Australia already uses this mechanism, however counting might take up to a month due to its complex algorithm [10].

6 Categorisation of blockchain based voting Systems

The first blockchain based voting system was created on the Bitcoin blockchain and is still in use today. This system has a restricted electorate (only Bitcoin nodes that create blocks can vote) and a restricted topic (they can only vote on improvement proposals for Bitcoin). The requirement of being a node in order to vote is too restrictive for real world voting systems, since it is both technically complex to set one up and can be quite costly to run. For this reason, we focus on blockchain voting systems where a voter does not have to be a node. Such blockchain voting systems can be built in different ways, which we group into three categories. A main advantage of all the three categories is that the voting records can be verified at any time in the future.

- **Using Blockchain for Vote Storage**

Voting systems in this category store the voting record on the blockchain. More specifically, the record of voters casting their vote can be added to the blockchain either as plain text, cipher text (encrypted text) produced by an encryption algorithm, or as a hash key (string of characters) produced by a hash algorithm. If an encryption or hash algorithm is used, this is run before the votes are added to the blockchain.

**Features:** This category of blockchain based voting systems allows the complete voting record to be stored in a censorship resistant manner. However, when the electorate is large, this can put a strain on the blockchain technology, which may take as little as 7 transactions per second (in the case of Bitcoin). Instead, to allow for scalability, subsections of the complete vote record can be hashed together. In this way, we can compress the entire voting record into only a few hash keys, which requires significantly less storage space compared to recording every individual vote. But note that this category of voting systems comes at a cost, as the majority of the voting system may not be transparent to other nodes of the network. This is because, the algorithms to collect votes and hash or encrypt the votes may occur in one private (and possibly malicious) location.

**Examples:** In February 2016, the Blockchain Tech Corp worked with Republican Presidential candidate Rand Paul to recorded the Iowa caucus results onto the blockchain for long term storage, via their VoteWatcher product [14]. VoteWatcher hashed the votes onto the Bitcoin blockchain (while concurrently posting them in anonymised plain text to the Florincoin chain). Later in April 2016, the Blockchain Tech Corp were once again involved with American political parties, when they recorded the results of the Liberation party’s Texas convention onto the same blockchains [18]. The largest occurrence to date of a blockchain based election in this category occurred in 2017, when X0.1 conducted a practice election to test of their SecureVote software by anchoring over a billion votes (that they generated themselves) onto the Bitcoin blockchain [22]. This occurred by grouping around 140,000 votes together, running a hash algorithm over them and putting the resulting hash string onto the blockchain.
• **USING BLOCKCHAIN ADDRESSES FOR VOTE TRACKING:** Voting systems in this category provide voters and candidates with blockchain addresses. For the election system to operate, votes can be transformed to digital tokens, which are initially placed in a voter’s address. When a voter casts a token, this token is sent as a recorded blockchain transaction to the address of the selected candidate. A variation of voting systems in this category represents a cast vote not with a digital token, but simply with an empty recorded transaction from the voter’s address to the candidate’s address.

**Features:** This category of blockchain based voting systems provides more certainty that a voter did in fact cast a vote for a candidate (compared to the vote storage category where a vote could be intercepted and changed before being recorded). This is because transactions from a blockchain address require signing by a private key, which only the owner of this blockchain address (the voter) should possess. But in this category, other issues such as voter privacy and interim results can arise, as transactions are usually public knowledge. Thus blockchain nodes are able to see which candidate a voter has selected and when this occurred. However, this issue can be resolved using an anonymous blockchain (such as ZCash[^10], Monero[^11] or Dash[^12]), as they can hide the amount of coins/tokens each candidate has and where each vote goes to. Lastly, this category is not ideal for complex voting systems such as the [single transferable vote](#) system, where a vote may move from candidate to candidate during the counting phase. When a vote token is in a candidate’s blockchain address, only this candidate can move the tokens. Additionally, the candidate would have to know where to move all of the vote tokens in her possession. Therefore this would add unnecessary required activities to eliminated candidates. However, this issue can be resolved with smart contracts, the next category of blockchain voting systems.

**Examples:** This category of blockchain based voting systems has not been used so far outside of the technology field, to the best of our knowledge. Examples of how elections of this category can work can be seen in the NEM blockchain [^9] and the counterparty protocol (built on top of Bitcoin) [^15] that allow any owner of the NEM or Counterparty cryptocurrency to create a vote and assign vote tokens to others. But both the NEM and Counterparty protocols are not anonymous blockchains, meaning that votes will be public knowledge and interim results will be known. To avoid these issues, [^41] describes a protocol to build an election of this category on top of the ZCash blockchain (which would stop information leakage on both individual ballots and interim results).

• **USING BLOCKCHAIN’S SMART CONTRACTS FOR ELECTION IMPLEMENTATION**

A smart contract is a piece of publicly readable self-executed code placed on the blockchain and is created through human understandable programming code. This is then compiled down to machine readable byte code, which is stored on the blockchain. Thus if the smart contract owner publishes the human readable version of the smart contract code, other users can check that it complies to the same machine readable byte code on the blockchain. This will increase the knowledge on how the (voting) system is operating and should in turn promote trust in it. Voting systems within this category allow key aspects of the election to be controlled by smart contracts. There are many variations of this model but the majority follows this simple pattern: a smart contract lists the election information (including the candidates) and the voters select their preferred candidate by calling a function on the smart contract.

**Features:** Such voting systems can implement any feature that can be coded into a smart contract, which is why there can be such a variation of features within this category. Examples include: managing a list of eligible voters; storing the votes encrypted; blocking interim results; allowing decentralised counting of the votes; counting encrypted votes without decrypting them; and allowing voting systems with vote transfers, such as in the single transferable vote system; etc.

[^9]: A blockchain address is a unique identifier on the blockchain network from which users can send and receive transactions. A blockchain address is protected by a **private key** operating as the blockchain address’ password.
[^10]: https://z.cash/
[^11]: https://getmonero.org/
[^12]: www.dash.org
EXAMPLES: This category of blockchain based voting systems is arguably the one with the most potential due to the transparency of the code and the multiple possible additional features that smart contracts allow. Some real-world examples follow.

In April and May of 2017, a parallel French election using the majority judgment voting system [4] occurred on a blockchain voting platform, named Cocorico [27]. A number of 52809 votes were collected in the first round while 15251 votes were collected in the second and final round of voting. Cocorico allows valid voters sending their cast vote to its own web server that would subsequently add it, in a non encrypted manner, to a smart contract in an Ethereum public blockchain. This smart contract contained the list of eligible voters and who they voted for in a non-encrypted manner. In the smart contract Cocorico had special permissions to control who could vote in the election, but the electorate could check that their vote was recorded (if they had the technical expertise to connect to the Ethereum blockchain themselves).

In December 2017, the Moscow government upgraded their online voting platform called ActiveCitizen[13] to include a private blockchain voting system based on Ethereum [22], where all votes were added to a smart contract in the blockchain. Users must go onto the ActiveCitizen website to send their vote to the ActiveCitizen servers before it gets logged onto the blockchain. Therefore, due to the fact that ActiveCitizen has a private blockchain implementation, their electorate must trust that at least one of the nodes is trustworthy enough to accurately report if a person’s vote has been recorded.

Another prominent example is [30], which produced open source code to show how votes can be held in encrypted form in the smart contract and re-tallied by anyone after the election has ended.

Scholars have identified many flaws in online voting systems, but blockchain technology has potential to solve at least some of the security flaws. However, blockchain is a developing technology and there are still questions that need to be answered before we deploy a large scale online voting system dependent on it. Most recent small scale trials using blockchain-based online voting systems have only been focused on very limited aspects of what this technology can achieve, e.g. gathering vote data from users and storing them on a blockchain. From our analysis, we believe that the whole process of an online voting platform (voter registration, vote collection to vote counting) should be underlined by a blockchain based infrastructure in order to increase security compared to the platforms without such an infrastructure. If some areas remain exposed (without being underlined by a blockchain), then these areas are as vulnerable to attacks as the current online platforms. In other words, if blockchain technology is partially used, it does not guarantee any improvement towards making online voting secure.

From the environmental point of view, we need to identify what would be the most energy efficient consensus protocol that satisfies desirable properties for our election systems, such as: finding an acceptable level of decentralisation of the system so that public trust can be established, and how election systems can be recovered in case of a majority attack. Given that we have good indicative answers for the above, the next step should be to make this transition smooth for organisations and governments that want to use blockchain-based online voting, i.e. what should the transition steps be from a centralised election authority (possibly using paper ballots) to a decentralised technological implementation (blockchain) in online voting in order to minimise disruption?

For corporate voting, implementing an online voting system is an easier use case due to the fact that votes are normally public. Therefore, even if we are a few steps away from a blockchain implementation for political office elections, corporate blockchain based online voting systems are more developed [26, 39]. Last, we believe that smart contracts are going to play an important role in this area, as they can offer multiple additional features for any voting platform using a blockchain based infrastructure.

---

13 ActiveCitizen (originally released in 2014) allows citizens to influence management decisions related to Moscow’s urban design.
7 Conclusion and open questions

In this paper we describe how online voting and in particular the blockchain technology can help to reduce electoral fraud. Our main objective is to provide a conceptual basis for understanding the key ideas behind the design of this technology. Essentially, the blockchain technology is a decentralised ledger which has the property that all transactions in the ledger must be approved by multiple nodes in the network and these transaction records cannot be modified once they are accepted in a block (therefore in the blockchain), unless all of the relevant nodes agree. In a sense, something is true only when the threshold of agreement is reached among the nodes. Unlike an EVM or a centralised voting system, it is much more costly to commit electoral fraud because one needs to hack not just one server but many independent servers, all at the same time. When the costs of doing so are high relative to the benefits, then fraud is discouraged. The system also provides transparency since each node has access to the code being run (whereas in a centralised system the code is run only from a single location). The combination of the above key features could significantly contribute to increase the public’s trust on an online voting platform as well as the platform’s accessibility levels. The latter is not only due to its online nature but also due to the multiple nodes (entry points) that voters can use to cast their vote.

The ideas underlying the workings of such a decentralised system raise many open questions for economists and social scientists in general: How to design the consensus protocol in a way to minimise costs while taking into account strategic incentives of validators? Who should be the stakeholders in elections (where permissioned blockchains are likely to be the norm)? Finally, a blockchain based infrastructure opens up the possibility of exploring new voting systems, e.g. it allows secure and transparent delegations of votes. Of course, a blockchain-based voting is not yet implemented on a large scale election but, in the meantime, economists have the opportunity to help in designing such a system.

References


[12] I have two homes. can i register to vote at both addresses?, <https://goo.gl/ajykJA>.


Frank Lynn, Boss tweed is gone, but not his vote, 1984, <https://goo.gl/GEYkRN>.


Jonathan Mellon, Geoffrey Evans, Edward Fieldhouse, Jane Green, and Christopher Prosser, Opening the can of worms: Most existing studies of aggregate level turnout are meaningless, SSRN’s eLibrary (2018).


