CS 390 Capstone Paper: Blockchain-based Voting

Mazen Abusharkh[1], Syver Johansen[1], Deepak Shah[1], Elif Ilaria Yurtseven[1]

Abstract

"Those who cast the votes decide nothing. Those who count the votes decide everything." - Joseph Stalin (alleged in 1923, Bazhanov, 1992)

This research paper proposes a voting system solution based on the blockchain technology, the same technology powering the widely known cryptocurrencies, such as Bitcoin. As a secured and transparent ledger, the blockchain introduces a potential solution to a decentralized vote validation and a distributed vote counting mechanism, such that the voters need not trust a third party to facilitate the democratic process. This paper also demonstrates a sample voting application that can be deployed on the Ethereum blockchain. The paper goes further into explaining why blockchain is better than in-person or e-voting and it finally concludes by shading light on the future of blockchain-based voting and potential future works using the same technology in voting systems.

Keywords:
Blockchain, Ethereum, E-voting, SHA-256, Solidity, blockchain voting, smart contract, decentralized application.

1. Introduction

Prior to Bitcoin and the blockchain technology, all online transactions relied on trusting a third party, like Visa Inc., to be truthful when administering transactions. Similarly, we trust email service providers to deliver our emails to the intended parties exclusively; a governmental authority to issue birth certificates for newborns; or social networks, such as Facebook, to share our post with who we particularly wish for them to share it with. The fact is that we live our life precariously in the world by relying on a third entity for the security and privacy of our records. The fact remains that these third party sources can be hacked, manipulated, compromised, or simply unnecessary. This is where the blockchain technology comes handy. The blockchain has the potential to revolutionize such services by enabling a distributed consensus where every online item, in the past and the present, involving digital records, to be verified at any time in the future. Blockchain does this without compromising the privacy of the records or of the parties involved, making anonymity and distributed consensus the two most important characteristics of this technology. Therefore, the blockchain can be used for tasks that traditionally require a trusted middleman or a facilitating party, such as an election.

The major issue with voting systems is the need to trust a central authority to receive, validate, and tally the votes. Traditional e-voting systems allow the users to vote using machines that take the vote and cast it in an internal system, usually in a private network. While encrypted, votes in e-voting systems are still susceptible to hacking because of the nature of their network connection. Moreover, votes in e-voting systems are stored and counted in one place, requiring people to trust one agency to count all the votes correctly. On the other hand, a blockchain-based voting system is virtually unhackable because of the nature of the blockchain, and it is more trustworthy as the middle-man authority is eliminated. In a blockchain voting system, all voters can receive votes, validate them, and then tally them without the need to trust a facilitating party. In our project, we will analyze a blockchain-based voting system for small populations, such as a company's boardroom election and list the positive and negative outcomes of a blockchain voting system.

In this paper, we will explain the blockchain mechanism under which the Bitcoin cryptocurrency functions, the research we have on voting systems through the blockchain, our reasoning as to why the blockchain technology is an essential technology to aid the democratic process, and based on that analysis we will build a design for a distributed voting app through the blockchain using a property called "smart contract." Thereafter, we will analyze the design of the blockchain voting system and find the issues in that design, after which we will discuss the implementation and the implementation issues that accompanied the building process of our blockchain voting application. To end the tangible part of our research, we will discuss whether the blockchain is indeed an essential technology for democratic process and the possible future improvements to our design. Our future improvements for the blockchain voting system introduce an improved design of the voting system that is directly inspired by our findings from the ethical analysis of the project's design.

Email addresses: abusha1@stolaf.edu (Mazen Abusharkh), johans1@stolaf.edu (Syver Johansen), shah2@stolaf.edu (Deepak Shah), yurtse1@stolaf.edu (Elif Ilaria Yurtseven)
[1]St. Olaf College, Northfield, MN

2. Background

2.1. What is the blockchain

A blockchain is a distributed database or public ledger of records that is shared among participating parties. Verified by some participants in the system, information in the blockchain is indelible. Bitcoin, the most popular and controversial example of the blockchain at work, enables a multibillion-dollar global market of anonymous transactions without any governmental control. Therefore, Bitcoin, alongside other cryptocurrencies, prove the blockchain's dependability and robustness as a technology when it comes to operations that require a record keeping. Explaining how Bitcoin executes transactions will provide us with an explanation of how Bitcoin employs the blockchain, and thus a succinct explanation of the blockchain at work.

2.2. How does the Bitcoin blockchain work

Bitcoin is an emerging peer-to-peer based cryptocurrency which allows people in the network to send and validate transactions without the need for a traditional bank with a central ledger. Once data is entered into the blockchain, it becomes nearly impossible to change. Each block contains some data, the hash of the block, and the hash of the previous block. The data that is stored inside a block depends on the type of the blockchain. The Bitcoin blockchain, for example, stores the details about a transaction: the sender, the receiver, and the amount sent. A block also has a unique hash that identifies the block and all of its contents. Once a block is created, its hash is calculated. Changing something inside the block will cause the hash to change, in other words: hashes are very useful when you want to detect changes to blocks. If the hash of a block changes, it no longer is the same block. Another element inside each block is the hash of the previous block, which effectively creates a chain of blocks that is tamperproof. For example, in a chain of 3 blocks, each block has a hash detailing its own contents and the hash of the previous block. So block number 3 points to block number 2 and number 2 points to number 1. If a fraudulent person tampers with the second block, the hash of the second block will change, which in turn will cause the hash of the 3rd block and all following blocks to change deeming all blocks following the second one invalid because they no longer store a valid hash of the previous block. So changing a single block will make all following blocks invalid, but using hashes is not enough to prevent tampering because hashes can be calculated very quickly on advanced computers. One could effectively tamper with a block and recalculate all the hashes of other blocks to make their blockchain valid again with the tampered block ingrained in it. To mitigate this, blockchains have something called proof-of-work, which is a mechanism that slows down the creation of new blocks. In Bitcoin's case: it takes an average of 10 minutes to calculate the required proof-of-work and add a new block to the blockchain. This mechanism makes it very hard to tamper with the blocks, because if someone tampers with 1 block, they will need to recalculate the proof-of-work for all the following blocks.

Bitcoin is a decentralized, peer-to-peer digital payments system based on the first public key cryptography proposed by Satoshi Nakamoto in 2008. Bitcoin uses a consensus protocol called Proof of Work (PoW) based on cryptocurrency to ensure only legitimate transactions are allowed within the system. Each transaction is calculated its hash value and entered into a database called Blockchain as described in fig.1. To connect one block with another, a copy of the previous block's hash value sits in the next block. The hash value must meet certain requirements called difficulty in order to be considered a legitimate block and thus be added to the blockchain. Searching for hash values that match those requirements is called Proof Of Work, and it requires guessing due to a fundamental property of hash values - randomness. The process of creating a block, doing the PoW protocol, and broadcasting it to the blockchain network is called mining.

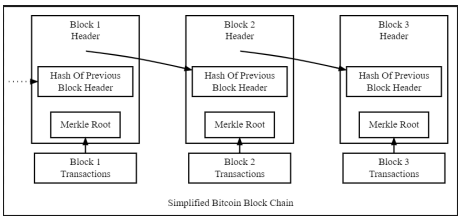

Figura 1: Simplified Bitcoin Blockchain

Miners compete against each other to create a new legitimate block in accordance with the specified difficulty. And since mining takes a lot of computing power and time, the effort to change the information in the blockchain is extremely difficult because changing one value from a verified block in the past results in a requirement to do mine that block and all the blocks following it, which then becomes a race against other miners, as blocks are mined - on average - every 10 minutes. A new block is generally generated by a miner but there are times when multiple miners generate new blocks at the same time. If this case occurs, then different miners would receive different blocks at different times, and the miner that mines the following block chooses which block to attach their block on and thus produce a longer chain. Then, as the entire Bitcoin system uses the longest chain, the blocks outside that chain are removed and the transactions inside them are reassigned to a new block in the future.

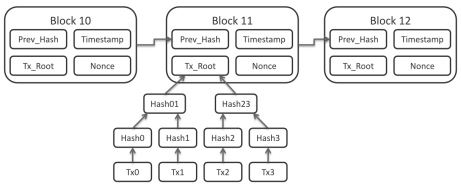

Figura 2: Merkle tree

2.3. Alternative uses of the blockchain

Due to its ability to keep data secure and optimize business decisions, the blockchain is deemed to be the future of governmental processes. By securing data, the blockchain can reduce costs associated with fraud, compliance, and financial reporting. It also adds an extra level of security for banking, medical

records, ownership records, invoices, and other services requiring record-keeping - usually - through a third party. Moreover, the blockchain can keep data secure, visible, trackable, and tamper-proof. It also enables real-time transactions across multiple parties. Security aside, other benefits potentially unlocked by this platform include speed and other efficiencies.

One key emerging use case of blockchain technology involves "smart contracts". Smart contracts are basically computer programs that can automatically execute the terms of a contract. When parties involved in a contractual agreement meet the preconfigured conditions, they can automatically make payments as per the contract in a transparent manner. According to Smart Dubai, a government entity charged with exploiting smart technologies to .[em]power, delivery and promote an efficient, seamless, safe and impactful city experience for residents and visitors", Dubai's blockchain strategy could save 25.1 million man hours, akin to 5.5bn dirhams, or 1.5bn, in savings per year. Much of this enhanced productivity will stem from moving to paperless government, with electronic document processing removing around 100 million paper transactions a year.

Another interesting thing about the blockchain is that it can be used for many additional non-financial tasks that traditionally require a trusted middleman or a facilitating party, such as counting votes in elections. The major issue with voting systems is the need to trust a central authority to receive the votes, validate them, and tally them. Traditional e-voting systems allow the users to vote using machines that take the vote and cast it in an internal system, usually in a private network. While encrypted, e-voting systems are still susceptible to hacking because of the nature of their network connection. Moreover, votes in e-voting systems are stored and counted in one place, requiring people to trust one agency to count all the votes correctly. On the other hand, a blockchain-based voting system is virtually unhackable because of the nature of the blockchain, and it is more trustworthy as the middle-man authority is eliminated. In a blockchain voting system, all voters can receive votes (excluding the voters' information), validate the votes received, and tally them. In our project, we will analyse the design of a blockchain-based voting system for small populations, such as a company's boardroom election.

3. Literature Review

3.1. Electronic Voting System

Electronic voting systems, which have been developed throughout democracies worldwide, constitute one of the conveniences of the Internet. Although electronic voting has increased the speed of voting and allowed people to vote anywhere, struggles to ensure secure and unbiased ballots have occurred. One emerging way to address this is through blockchain technology. Using Blockchain for Enabling Voting discusses Estonia's internet voting and how blockchain technology could confront the security and bias issues. Currently, the system works so that each person receives a signing key and a verification key through their national-ID card. When the person votes, they send their signature to a server and their vote is stored. While this system keeps anonymity of voters, the government still has the power to corrupt the voting. Blockchain voting could improve the current system as it would decentralize the vote counting and take away the potential for corruption [Kubjas, 2016].

3.2. Security Issues

Security issues that arise with public systems account for most of the concerns of the blockchain voting initiative. With election security dominating national discourse, the security issues of blockchain constitute a vital issue. The literature on security concerns with blockchain technology heavily promotes blockchain as the safest way to carry out electronic elections. A proposed solution to the double voting problem is to use a peer-to-peer network. The network timestamps elections by hashing them into an ongoing chain of hash based proof-of-work and proof-of-tally forming a record that cannot be changed without redoing the proof chain [Bitcongress, 2016]. Additionally, in the Handbook of Digital Currency, voters can verify their vote counted after the voting process has ended. If each voter receives a public key and nonce that is used as a secret key to log onto their account, a Merkle tree will represent the voters list linking the public key to the root of the Merkle root. With the Merkle tree published by the application, the voter's list can be independently verified by all voters and they can check the path from their' leaf node to the Merkle root [Noizat, 2015].

3.3. Uses of Blockchain

The versatility of blockchain makes for one of the main reasons of its sudden emergence. Although cryptocurrency such as Bitcoin has put blockchain in the news, its other uses may diversify the technology to ensure its relevance. One of the draws of blockchain is its anonymity.

Protection of privacy and personal information has drawn experts in technology to blockchain largely due to incidents with surveillance and the public's discomfort with mass data collection [Zyskind, Nathan, and Pentland, 2015]. In the healthcare industry, the value of anonymity cannot be understated and concerns with the vulnerability of an electronic database for medical records can be addressed with a blockchain solution. A system that uses Ethereum software can build a private blockchain that connects healthcare providers to share data, even in a decentralized healthcare infrastructure such as the United States [Mattila, 2016]. In real estate, implementations of blockchain-based registries offer an anti-fraud advantage as false transactions cannot be added or changed in the ledger without the consent of all parties. Furthermore, transparency across the system allows for consistent transaction data throughout the network and available for several organizations. For this reason, blockchain based registries are considered efficient as it does not allow for conflicting transaction information [Spielman, 2016]

3.4. Follow my Vote

The grassroots, open-source movement of Follow My Vote strives to create an online voting platform using the security of blockchain technology. Follow My Vote was founded in order

to provide a technological solution to accusations of illegitimacy in elections. As technology becomes more integrated in elections worldwide, discussions of vulnerability in the technology continues to a topic of discourse. Follow My Vote believes that blockchain technology addresses the vital concerns of e-voting as it requires a unique "wallet" form of identification for each voter and the votes are not sent to a central database, but rather distributed and approved by miners. Furthermore, Follow My Vote claims that their decentralized system will save governments from unnecessary expenditures, citing the millions of dollars the United States spends on ballots annually [Follow my Vote, 2018].

3.5. Model Implementations of Blockchain Voting from Dissertations and Graduate Papers

Several findings from dissertations and graduate papers on blockchain elections played an important role to our research. The most extensive model, labeled "The Votebook", initiates a proposal for a blockchain voting system and uses diagrams to lay out the process for how to effectively implement a blockchain election. The paper divides the voting process in three stages: Before Voting, During Voting and After Voting. Before voting, the voter receives a voter ID for security reasons. While voting, the voting ID and ballot ID are hashed together along with the plain text candidate choice of vote and together they are written to a block which after consensus is added to the blockchain. After the vote is casted, the voter can check their vote was counted [Kirby, 2016].

Similar to the Votebook, An E-Voting Protocol Based on Blockchain, laid out the protocol for blockchain voting. Prior to voting, a person would register as an eligible voter by submitting their personal information with their public key, and after verification would choose their candidate and sign a message confirming their hash. However, dissimilar to e-voting, this system sends a message to a voting center to verify the vote, and thus cannot guarantee anonymous voting. Furthermore, the system allows for corruption as each vote requires the signature of an inspector [Liu and Wang, 2017].

4. Advantages of using Blockchain for Voting

4.1. Why use the blockchain technology in a voting system

The foundation of any democracy sits on voting and therefore the voting mechanism must be accessible and secure. It is difficult to deny the accessibility and chapness of today's most common voting system, paper-based voting. However, a paper-based voting system has two major problems: the reliance on a time consuming mechanism, which leads to issues in accuracy and convenience, and the reliance on a central authority, which presumes the trust in that central authority. In other words, "Paper-based voting systems rely on the procedural security of officials conducting their jobs correctly and honestly," according to Nathan Hourt, the founder and CTO of Follow My Vote. Therefore, to address those two major problems, governments and organizations decided to develop and adapt e-voting systems. However, to ensure a fair election, e-voting systems must remain anonymous yet auditable, and tamper-proof yet non-intermediated, which seemed impossible until the invention of the blockchain. Because of properties such as transparency, decentralization, irreversibility, and non-repudiation, the blockchain technology is deemed as the missing piece to the e-voting system puzzle[Liu Y., 2017]. A blockchain-based voting system, in essence, will work the same way as a cryptocurrency system. The main difference is that instead of transferring tokens of monetary value between accounts, voting tokens will be transferred from a voter to the ballots. Since blockchain remain public yet anonymous, in a blockchain-based e-voting system, anyone can easily verify the electoral outcome, and check whether their vote has been counted correctly or not, all while still maintaining the ballot's secrecy and automation. The blockchain technology can, therefore, act as a safeguard against rigged elections. In this paper, we investigate the idea of a blockchain-based voting system and propose an implementation of it[Liu, Y., 2017].

4.2. Issues solved by the blockchain voting system

4.2.1. Voter Anonymity

Voters' ability to vote without revealing their names or real world identity make it so that after the votes have been casted, only the hashes (without names) get recorded in the blockchain. With the use of one-way functions SHA-256, it becomes virtually impossible to track back[Kirsby, K., 2016].

4.3. Tamper-Proof

New blocks must receive consensus from the miners before recording themselves on the blockchain. This system makes it virtually impossible to previous blocks to be changed because if that happened all the following blocks would become incorrect that the other miners would not allow for such thing as they have more power as a consensus[Kirsbky, K., 2016].

4.3.1. Needless trust on third party

Its peer-to-peer structure makes it so that many people in the network need to be part of a consensus in deciding how honest (not fraudulent) and correct each block is. This means that this system does not at all depend on a third party authority or on anyone else in the system for that matter.

4.3.2. Convenience

Unlike in-person voting, voters can cast votes from their own comfort zone. They can cast from their houses, offices or literally anywhere as long as they have access to a computer.

4.4. Scalibility

With the number of voters, the number of proposed blocks will get much larger. This would make it computationally more and more expensive to try and solve the cryptographic puzzle which nodes need to solve in order to add each block. Moore's law guarantees processors will continue to be powerful and hence keep solving the complex cryptographic puzzles.

4.5. *Issues Introduced by blockchain voting systems*

4.5.1. *Convenience*

While being able to vote from own houses is a convenient process for the voter, it also invites coercion by terrorists, less like to be prevented as the number of voters increase.

4.5.2. *Lack of trust in new technology*

Blockchain as a concepts, though older than its public use, still cause a lot of ambiguity and mustrust due to the very limited understanding of how the technology works.

4.5.3. *Computing power*

A powerful computer that can easily solve the cryptographic puzzle, would possibly have the capability of creating fake transactions (voter in this case) and mine them. Hence, in case quantum computing, for example, in the future, would have the power to disrupt the voting process. A case of a Russian nuclear scientist ended in him getting arrested for using a super computer in Bitcoin mining.

5. Design of Blockchain Voting

5.1. *How will it work*

Voters vote through a web application and unlike the traditional centralised application where user input is stored on a database, in this case the votes will be recorded on a single blockchain that is accessed by multiple computers. Each voters' votes are recorded on the blockchain and there is no way for any voter to modify the votes on the blockchain. After voters have voted, the blockchain is published, votes are counted and winner is determined. It is a high level overview of how blockchain will be used in voting. Below is a pictorial explanation of how a blockchain interacts with the web application.

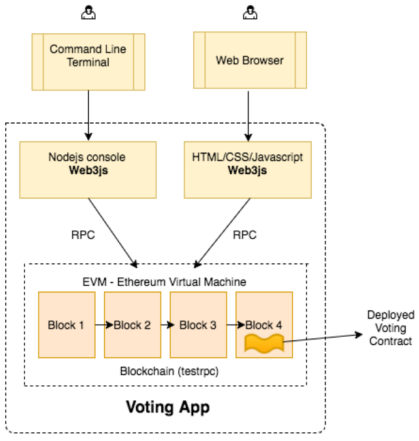

Figura 3: Web3js interacts with ethereum blockchain

Web3js is a javascript API that interacts with ethereum blockchain. In the diagram above, user sees the candidates profile on the web browser. The underlying javascript interacts with ethereum blockchain via Web3js to record the user's votes. After all the users have voted, the blockchain is published. However there are lots of details that have not been discussed, namely:

- How can the web app find out if it is an authentic voter?
- How to prevent double voting?
- How does the votes get added on the blockchain?

To design a blockchain system that solves problems mentioned above, the voting process is divided into three stages: before election, during election and after election.

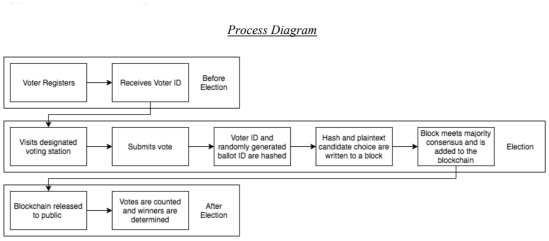

Figura 4: Voting process explanation from Votebook

5.1.1. *Before election:*

Voter registers to vote and they receive voter ID after registration.

5.1.2. *During election:*

Voter visits the voting station and submits the vote. The voter ID and randomly generated ballot ID are hashed. The hash and plaintext candidate choice are written to a block. The block is proposed by nodes to be added to the blockchain and if the block meets majority consensus, it is added to the blockchain.

5.1.3. *After election:*

Blockchain which contains the votes are released to public. Public can verify if their votes appear on the blockchain. Votes are counted and winners are determined.

Let's dive in further on during election stage. The voter first sees a webpage such as Fig 1 below:

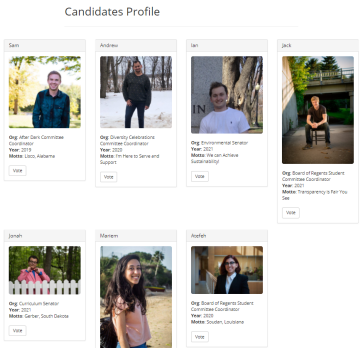

Figura 5: Candidates for Spring 2018 available from Oleville.

Clicking on "vote" under the name of a candidate will create a "transaction" which has several fields including the hash of the voter and the hash of the candidate. After several voters have voted, a block is created which is basically a list of voters

and candidates hashes. After few mins, the block is proposed by nodes to be added to the existing blockchain. The node which first solves the cryptographic puzzle is successfully able to add the proposed block to the blockchain.

5.2. Design Concerns

In order to ensure that the votes can only be counted after the ballot is closed and all votes are tallied, we will have two parallel blockchains running at the same time. The voter's names will be held in a database of names, as they register they will be hashing their names with a private key which only they know. During voting, voters will be committing their votes by hashing their votes with the timestamp and the public key, the candidate's name and the voter's signature. The voter signature can only be generated from their private keys and hence is unique as it is only related to the specific transaction. This signature is also verifiable. The hashed commits will be broadcasted on the Ethereum blockchain and verified. When committing closes, people will need to "vote" - submitting their votes by hashing a vote similar to how they hashed the commits -, these votes will be compared to their commits. If the candidate voted for is different, their vote will not count. In case they will want to opt out because they have seen the results of the commits, they will have to pay a large amount of money since we do not want that to happen. If voters do not vote after their commits, the candidate their commit;s was for will be accepted. The verification of these votes (mining) will ensure no double voting, that voters have committed a vote and that commits have been reinforced with the votes and create blocks based on voter's signatures and timestamps.

6. Implementation

6.1. The Stack

We used Votebook as a guide for the implementation of our blockchain voting system. Our software stack consists of the Ethereum blockchain, Solidity programming language, Truffle framework, Javascript, Ganache, and CSS for front end design. The decentralized design of the Ethereum blockchain which uses a peer-to-peer network, makes Ethereum faster than Bitcoin blockchain which increases its scalability and accessibility. The Ethereum blockchain has two accounts. The first, externally owned, user-controlled one generates the public-private key pairs and enables the user to purchase the 'coin' which they need to 'pay' in order to cast one vote. The second - contract - account has two contracts: the voting contract and the cryptography contract. The voting contact implements the voting protocol and the election process. The cryptography contract however, distributes the code that can be used locally without interacting with the real Ethereum blockchain. We use the Solidity programming language as a smart cryptography contract of the Ethereum blockchain. This means that Solidity has the same cryptography as Ethereum but it does not interact with it in real time[Murthy, Mahesh., 2017]. To deploy our project onto the real Ethereum blockchain we would have needed to pay for it. The Truffle framework is a development and testing framework which has asset pipelines for Ethereum as well as built-in contract compilation, linking, deployment and network management for public and private networks; all of which are essential in our program. Ganache supports a graphic user interface for the server that displays the transactions - which in our system coincide with one vote. Javascript and CSS display the web browser that interacts with the Ethereum blockchain in the background.

6.2. Why we chose to implement this way

The Ethereum blockchain ensures that each individual can check whether their vote was counted as valid without the ability to discern who anyone else voted for. The system also ensures reduced possibility of coercion with the authentication process. Since this system doesn't allow retroactive editing the validity of the votes relies on consensus. In a small scale (company boardroom elections) Ethereum allows us to eliminate proof of work which requires excessive computing power within a trustless system[Liu, Y., 2017]. In our system, only authenticated voters (e.g citizens in president elections, board members for CEO election, etc.) can make a change to the ledger. VPN and firewall take care of the possibility of other countries or companies tampering with the elections.

6.3. Voting Mechanism

Server generates public-private key pairs, stores the public key and sends out the private keys to voters' machines. Once polls are open server periodically organizes votes into blocks in a time-based protocol. Once the user clicks on "vote", a transaction is created to be added to the tamper-proof blockchain. Every transaction has two fields: From and To, which are Keccak-256 hashed addresses in the Ethereum blockchain. The use of hashes prevents voters from voting twice. For example suppose voter A wants to cast vote with 256-bit hash. Due to the deterministic property of hashes: same input would yield the same output. Each block consists of a timestamp, rows representing the voter and the vote, the hash of the previous block and a digital signature (private key to encrypt a hash digest of the rest of the block). Once the block is in the network, other blocks need to check its validity according to the hash. Other nodes use public key that corresponds to propose a node's unique identifier to decrypt the hash of the proposed block and verify a match in which case the block gets added to the blockchain and the vote validated.

Similarly the 'collision-resistant' property of hashes guarantees that the hashes of two different candidates cannot be same. So when A and B cast their votes, the blockchain correctly adds two transactions and each transaction has different hashes h(A) and h(B) each referring to different candidates.

6.4. Vote verification/SHA-256

The main reason for us to choose blockchain, is the distribute the ledger to all of the user and eliminate central authority. The vote verification process can be done after the election results are published and the entire blockchain is made available. Each voter knows their own hash and after the publication of

the blockchain would be able to see if their has appears on the blockchain. This is how the voter can verify whether their vote was counted for the right person. This ability to verify one's vote separates blockchain from other voting systems. In e-voting, votes are recorded in the database and an authority tallies the votes until a winner is declared. Voters can never be sure if their votes were correctly recorded in the database and if they were taken into consideration when tallying. However with blockchain, voters can confirm their votes made into the tamper-proof blockchain. This is what makes the blockchain voting experience truly democratic.

6.5. Implementation Concerns

Understanding a concept theoretically and implementing the same concept in codes can be quite difficult. We had some understanding of how double voting could be prevented with private keys however when beginning to implement, solidity contracts do not really have a private key. So although the theory was clear, it was still difficult to convert that to implementation. We used the contract addresses as public key. So if a voter had already voted, their public key would be recorded and hence they cannot vote again. In addition to the issues with private key, some understanding of bytecode would have helped accelerate the pace of the implementation. Since solidity codes are finally compiled to bytecode before it gets deployed on the blockchain, understanding bytecode helps debug the application. Working with a decentralized application needs some in-depth understanding about networks. In the voting application, a port listens to connections from several nodes, and when new blocks arrive, it arrives in packets. Understanding about packet transmission would have helped with the development process. In addition to that, deploying a decentralized application is not as easy as deploying a web application on Heroku. We did not have prior experiences with IPFS, the peer-to-peer deployment platform.

7. Results

We were able to implement an example blockchain voting system that can be deployed on the real Ethereum blockchain. We developed a decentralized application that makes use of Truffle framework with the Web3js javascript API that interacts with the blockchain to record votes. The source code is available on stogit, the private Gitlab for St. Olaf College and can be cloned from https://stogit.cs.stolaf.edu/cap-s18/bc. Currently the application has a list of candidates from SGA Spring election and the list of candidates can be changed by adjusting the json file which contains – the candidates name, position they are nominated for and their picture in the source directory. After voting for a candidate, a user can see the addition of a block on ganache which clarifies the understanding of the voting process using blockchain.

8. Future Works

The current implementation lacks a proficient method to tally the votes. As of now, the votes are received, tallied, and displayed in real time. Problems arise with this design, as real-time election results leads to lower voting turnouts and even worse, the spoiler effect, where voters end up voting for a candidate they didn't wish to vote for in the first place. To address this concern, we theoritized about the implementation of a system that mines the votes to the blockchain, but does not release the results until a given time after all votes have been casted and no more votes can be casted.

Based on our research, our ethical analysis, and feedback from our supervisor, we came up with two possible solutions to this problem. The first solutions is implementing a ranking-choice voting system where the voters would rank the candidates based on their preference, as opposed to making only one choice. This solution would address the spoiler effect but would not address the main concern of our implementation, which is the voters' ability to tally the votes before the end of the election period. To address our main concern, we developed a new design for the blockchain voting application that will not require a redesign of the Ethereum blockchain. The solution is a three-phase blockchain voting system on the Ethereum blockchain. The first phase in the design, the voter registers to vote using their key, similar to how they register in our current system implementation. In the second phase, the voter commits to voting for a candidate. This commitment is then broadcasted in a block to be mined and included in the blockchain. However, the vote inside this commitment cannot be "seen" by anyone as the details of the commitment are encrypted, where only the voter has the key for this commitment (encrypted vote). Anyone on the blockchain will only be able to see that the person's key submitted a potential vote – without knowing where this vote is going. The third phase is for the voter to confirm their vote, selecting the same candidate as in the second phase. This confirmation vote is simply a broadcast of the voter's public key, the encryption key, and the decrypted vote details. Blockchain miners and validators will "open" their original encrypted vote using the decryption key and check it against the voter's vote details sent in the third phase. If the encrypted vote from the second phase matches that sent in the third phase, the vote will be counted. If they are not, the vote will be discarded. This way people will not be able to change their mind when they see the tally during the third phase. These solutions are added to the paper as they introduce potential remedies to the problems of the blockchain voting system; however, they will be incorporated in the future works section due to their complexity.

includes a similar setup step, the second and third step consist of voting and tallying respectively. The setup (and registration) step consists of a database of names of potential voters, who would receive a hashed public key which indirectly relates to their name and information. Each voter would then have a private key that only they have access to, and a public key that is generated from their private key, and is in turn verifiable. The second step would be the committing step, where voters would commit their hashed votes anonymously by broadcasting a verifiable hashed value of the candidate's name alongside their unique signature. The commits are verified by checking.

The second and theoretical way to avoid the tallying problem assumes that the way to see the tally as votes are being

casted requires having a way to decrypt the encrypted information on votes. We can assume that enough people or enough computational power could lead to people having access to such results. However, the easiest way to inhibit this access would be to give half of the time necessary for the decryption to be found to the voters to vote. We do understand that this may not be ideal in large scale elections since people need multiple hours in order to find time during their day to vote. If the total voting time cannot be reduced, the solution requires the decryption code to be changed every half time (the faster necessary time for the decryption algorithm to be found by anyone).

9. Conclusion

The emergence of blockchain technology has the potential to cause a drastic change in industries such as commerce, government, and health care. Recently, blockchain technology has entered the news cycle through the form of cryptocurrencies, such as Bitcoin and Ethereum and their wildly volatile valuations. While cryptocurrencies may receive the spotlight, the technology has arguably more practical uses in other fields such as health care records, real estate, and politics. This paper sought to design and implement a Blockchain election that followed Votebook guidelines and used the Ethereum network. The goal was to develop a system that optimized the democratic voting system.

From the model developed, an electoral system that allowed for convenience, security, and anti-corruption was achieved. The model calls for an online system where votes can be made remotely, the implementation of key encryption and hashing to make it secure and anonymous, and the full use of a decentralized mining system to count votes to eliminate a government's power to corrupt the results. However, the system still has some concerns. Since blockchain elections are new and untested there exist a higher probability of unknown security risks. Furthermore, implementation and maintenance could prove costly as it requires contracting engineers to ensure credibility. Lastly, as any democratic process goes, the deployment requires the trust of the population. If people distrust or are unable to use the updated voting system, the technology can not be implemented. Overall, the development of this model marked an advancement in the search to improve the democratic electoral system in a way that optimizes convenience, security, and anti-corruption. While the model requires further implementation, it has the potential to create a better democracy.

Referencias

Kubjas, I., 2017. Using Blockchain for Enabling Voting. University of Helsinki Computer Science.
DOI: https://pdfs.semanticscholar.org/8d92/1dbfe6bebefa2599ca6afc7eeae82210a71d.pdf

Moura, T and Gomes, A. 2017. Blockchain Voting and its effects on Election Transparency and Voter Confidence. Proceedings of the 18th Annual International Conference on Digital Government Research (June 2017), 574–575.

Tarasov, P. 2017. Internet Voting Using Zcash. Trinity College Dublin.
DOI: https://eprint.iacr.org/2017/585.pdf

Liu Y. and Wang.Q, 2017. An E-Voting Protocol Based on Blockchain. Department of Computer Science and Engineering Southern University of Science and Technology (October 2017).
DOI: 10.3923/ijbc.2010.190.202

Kirby, K. 2016. Votebook: A Proposal for a Blockchain Based Electronic Voting System. New York University (September 2016).
DOI: https://www.economist.com/sites/default/files/nyu.pdf

Rifa Hanifatunnisa and Budi Rahardjo. Blockchain Based E-Voting Recording System Design. School of Electrical Engineering and Informatics.
DOI: http://ieeexplore.ieee.org/document/8272896/

Ryan Osgood. 2016. The Future of Democracy: Blockchain Voting. Computer Science and Engineering(December 2016).
DOI: http://www.cs.tufts.edu/comp/116/archive/fall2016/rosgood.pdf

Juri Mattila. 2016. The Blockchain Phenomenon - The Disruptive Potential of Distributed Consensus Architectures. Berkeley Roundtable on the International Economy (May 2016).
DOI: http://www.brie.berkeley.edu/wp-content/uploads/2015/02/Juri-Mattila-.pdf

Anna Lali Tsilidou and Georgios Foroglou. 2015. Further Applications of the Blockchain. University of Macedonia (May 2015).

Anon. BitCongress Whitepaper.
DOI: https://bravenewcoin.com/assets/Whitepapers/BitCongressWhitepaper.pdf.

Pierre Noizat. 2015. Blockchain Electronic Vote. In Handbook of Digital Currency. Amsterdam, NL: Elsevier Inc, 453–461.
DOI: https://www.sciencedirect.com/science/article/pii/B9780128021170000229

Guy Zyskind, Oz Nathan, and Sandy Pentland. 2015. Decentralizing Privacy: Using Blockchain to Protect Personal Data. IEEE CS Security and Privacy Workshops (August 2015).
DOI: http://ieeexplore.ieee.org/abstract/document/7163223/

Mahesh Murthy. 2017. Full Stack Hello World Voting Ethereum Dapp Tutorial. (January 2017).
DOI: https://medium.com/@mvmurthy/full-stack-hello-world-voting-ethereum-dapp-tutorial-part-1-40d2d0d807c2