CBS COPENHAGEN BUSINESS SCHOOL
HANDELSHØJSKOLEN

Smart Voting: Using Blockchain Technology to regain Trust in Democratic Elections

Foundations of Blockchain, Cryptocurrencies and Smart Contracts (CDSCV1001E) - Oral exam based on written product (IC)

Student numbers: 150257, 156551, 149872

Date of submission: 02/01/2023

Number of characters: 26,141

Number of standard pages: 12

EFMD EQUIS ACCREDITED AACSB ACCREDITED ASSOCIATION AMBA ACCREDITED CEMS PIM

Abstract

This paper discusses how blockchain technology can be used to improve the integrity and transparency of democratic elections. The authors first describe the requirements and problems of traditional election systems, including the need for accurate recording of votes, accessibility, voter privacy, reliability, and transparency. They argue that blockchain technology, with its decentralized and secure nature, can address these issues and restore trust in the electoral process. To demonstrate this, the authors present a proof-of-concept for a blockchain-based voting system using Hyperledger Fabric and Fabcar. They discuss the benefits and limitations of this approach and provide an overview of related work in the field. The authors conclude that the proof of concept is a viable and applicable introduction to the implementation process of an election system that would lead to restoring voter confidence in elections.

Table of Contents

1. Introduction

An election is the cornerstone of democracy, and the integrity of the voting process is of the utmost importance. However, studies show that in recent years people's trust in their government and the associated process of forming a government through elections has decreased in a lot of countries (Zimmermann et al, 2020). Most notably in the United States after the 2020 presidential election, where some people's mistrust in the election result lead to an attack on the capitol in Washington.

Traditional voting systems, using paper ballots and boxes or letters, have always been plagued by potential issues such as vote tampering, fraud and a lack of transparency to a certain extent. These concerns have led to a search for more secure and transparent alternatives, and one promising solution is the use of blockchain technology. In this paper, it will be elaborated on *how blockchain technology can be leveraged to regain trust in democratic elections*. The research question will be embedded into context by defining requirements and issues of traditional voting systems, where trust is lost. They will be contrasted to properties of blockchain applications that could potentially fix issues and restore trust in the election process. Additionally, an overview of related work that has already been done in that area will be given. A working proof of concept will be presented to show the benefits and limitations of the approach discussed.

Blockchain technology generally refers to decentralised, distributed databases that enable secure and transparent record-keeping of transactions. It usually implies the existence of a network of computers, also known as "nodes", that validate and record transactions onto a digital ledger, which is organised into blocks. Blocks are cryptographically linked, which aims to ensure the integrity and immutability of recorded transactions. However, the paper at hand will only focus on presenting a conceptual blockchain architecture and implementing chain- and calling-code that enables the casting and polling of electoral votes onto a digital ledger. The setup of a network of nodes as well as the registration of voters and candidates will not be part of this project and is deemed to be out of scope.

2. Motivation

Generally, it is clear that any computer-based electronic voting system needs to be able to fulfil all the requirements put against traditional voting systems. These would include:

- The ability to accurately record and count votes in a way that results are easily verifiable.
- Being accessible to all eligible voters, regardless of their location or physical abilities.
- Protecting the privacy of voters and allowing for their individual votes to remain completely anonymous.
- Being reliable and withstanding various types of failures and disruptions.
- Being transparent and open to audits.

Additionally, it must be secure and extremely robust towards cyber security threats of all kinds and disruptions that specifically affect electronic machines such as power outages and technical issues.

Apart from these general requirements, the main issues that fuel mistrust in traditional voting schemes should be addressed even more particularly to justify the new system.

First and foremost, there seems to be a lack of transparency - voters cannot know for sure if their vote actually made it to the counting phase and if it has been counted in the way they intended. This directly links to a general mistrust towards election management and the traditionally manual process as a whole. For example, after the 2020 presidential election in the US, it was feared by republicans that some people cast multiple votes via letter ballots. People also believe that political parties or other groups try to influence the outcome of elections by directly interfering with the electoral process in ways that are not even apparent to the broad public (Pennycook et al., 2021).

A blockchain-based voting system would have properties that intrinsically fulfil many of the key requirements for an electronic voting system and provide solutions to the issues where trust is lost the most:

- By using a distributed ledger, a blockchain-based voting system can ensure that all votes are accurately recorded and counted, with a clear audit trail that can be used to verify the results.
- Blockchain technology is inherently secure, as it uses cryptography to protect against tampering and unauthorised access. This can help to ensure the integrity and confidentiality of the votes. These cryptographic techniques can also help to ensure that individual votes remain anonymous and cannot be traced back to the voter.

- A blockchain-based voting system can provide a transparent and open voting process, as all transactions (i.e., votes) are recorded on a ledger that can be audited by anyone within the network.
- The application could potentially be accessed by eligible voters from any location, using their mobile devices or other secure devices. This could make it easier for people to cast their votes, even if they are not able to physically go to a polling station.
- The voting system could be designed to be highly reliable, with multiple copies of the ledger distributed across a network of nodes.

It needs to be noted, however, that some of the reasons why people lose trust in elections lie outside of the actual voting system itself. Media coverage of candidates, misinformation or intimidation of minority groups are issues that cannot be addressed by the voting system alone. Still, the potential benefits for voting systems introduced by properties of blockchain technology clearly motivate the research question.

3. Related Work

As stated in the introduction, the necessity for a safe and viable voting system is at an all-time high due to the continual danger of cybersecurity attacks and vote recount paranoia across the world. This section will describe the different variations of advancements toward a voting system using blockchain technology backed by literature.

In 2015, Czepluch et al. (2015) took the first step toward the potential and capabilities of blockchain technology as a decentralised voting system. Meanwhile, Zhao et al. (2015) has developed a voting system based on Bitcoin that focuses on secrecy, verifiability, and irrevocability. To protect individual voters' anonymity, this system included a stage in which secret random numbers were distributed using zero-knowledge-proofs based on a lottery system. As a result, Lee et al. (2016) proposed and presented a four-party voting system on a national level using real-life examples.

Continuous research and testing resulted in the development of Agora, a blockchain-based voting system designed to automate the voting process for a recognized institution or government. This method uses a token to differentiate eligible voters for elections by distributing tokens to each eligible voter. Agora has limitations, it relies on trusted third parties to oversee the voting process, which might lead to candidate cooperation via vote tampering (Gailly et al., 2018).

Smart contracts and encryption techniques were utilised in blockchain-based voting to replace third-party enabling. Using Ethereum, the Open Vote Network was employed within

smart contracts to establish the first-ever voting mechanism that does not rely on third parties or a trusted authority to count election results (McCorry et al., 2017). Ali Kaan et al. (2018) explored the deployment of a voting application as a smart contract on Ethereum, allowing voters to vote. This lacked an automated address verification mechanism due to the leakage of obtaining their voting rights from a centralised body in order to become eligible voters (Ali Kaan et al., 2018).

To preserve a person's identity and privacy, a recent trend in this field known as digital signatures has been embraced. In terms of digital signature schemes, the following are the most common: homomorphic encryption, blind signing, and public-key cryptography. Homomorphic encryption and secret sharing systems work together to deploy an electronic voting system that not only preserves the voter's anonymity and voting verification but also does so without the use of a trusted third party (Hsiao et al., 2018). Chillotti et al. (2016) use LWE homomorphic encryption to verify the authenticity of votes and the accuracy of election outcomes without requiring zero-knowledge proof. Further research is needed for homomorphic encryption to be successfully utilised at a government scale. In comparison to homomorphic encryption, blind signatures have been used to encrypt ballot information to prevent the leaking of election results and to enable larger-scale voting (Fujioka et al., 1992). However, it will fail to secure voters' privacy since an administrator would know the voter's bitcoin address, which will lead to the voter's identity. Finally, public-key cryptography may be used to ensure a voter's anonymity. This will generate a hash value based on the voter's ID number and the preceding ballet's hash (Ayed, A. B. (2017). The main disadvantage of this scheme is a waste of computational power and voters can’t change their vote if they made a mistake.

This led to the development of the AMVchain by Li et al. (2021), an efficient and scalable voting system that combines blockchain and smart contracts to allow transparent and decentralised voting based on the requirements for a trustworthy and efficient electronic voting system. In the voting process, linkable ring signatures are employed to sever the link between voters and ballots and ensure voter anonymity. These ring signatures can be interpreted as no indication of knowledge of a key inside a set of keys, allowing the voter to remain anonymous. In addition, if a voter changes their mind about voting, they can use the ring signature's linkability to cast a fresh ballot.

Alvi et al. (2020) propose assembling a digital voting architecture based on smart contracts to address the issues of anonymity and authenticity. A hash will be generated and logged on the chain inside their system given by the voters to facilitate this operation. The voter gains immensely from the hash since it ensures secrecy and privacy.

Khan et al. (2020) address the challenges surrounding blockchain elections and how to implement a decentralised e-voting approach rather than a centralised one utilising blockchain technology. A simple voting method that ensures the security of voter identity, data transfer, and verification. Ganache, truffle framework, and metamask are used in the system.

4. Methodology

In the following chapter, the architecture of the voting system of this paper as well as the code implementation is explained in detail.

4.1 Conceptual Design and Architecture

Figure 1 shows a general overview of the proposed architecture of this paper.

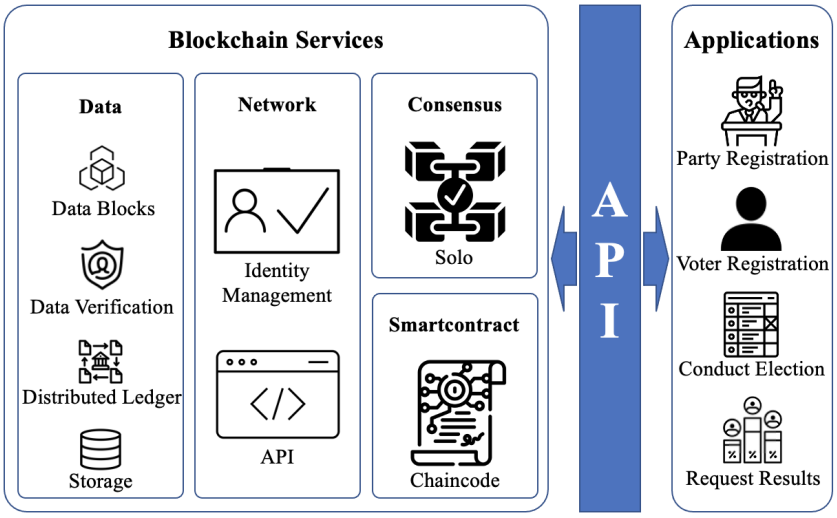

Figure 1: High-level view of the proposed architecture

The two major categories shown are the blockchain services and the applications connected through the API.

The blockchain is the technological centre of the architecture that performs the core tasks for the voting system: it verifies and stores the votes in a distributed ledger, contains different smart contracts, verifies identities and makes all this accessible to all stakeholders through RESTful APIs. The consensus algorithm adopted is the "SOLO" algorithm.

Three entities have been identified as potential users of this voting system: a voting office, political parties and voters. All these participants interact with the blockchain through an API and can run different applications. With the help of smart contracts, parties and voters can be registered, but also elections can be conducted and results can be queried. For simplicity

reasons, this paper only focuses on the latter two applications. The details of the election process are shown in Figure 2.

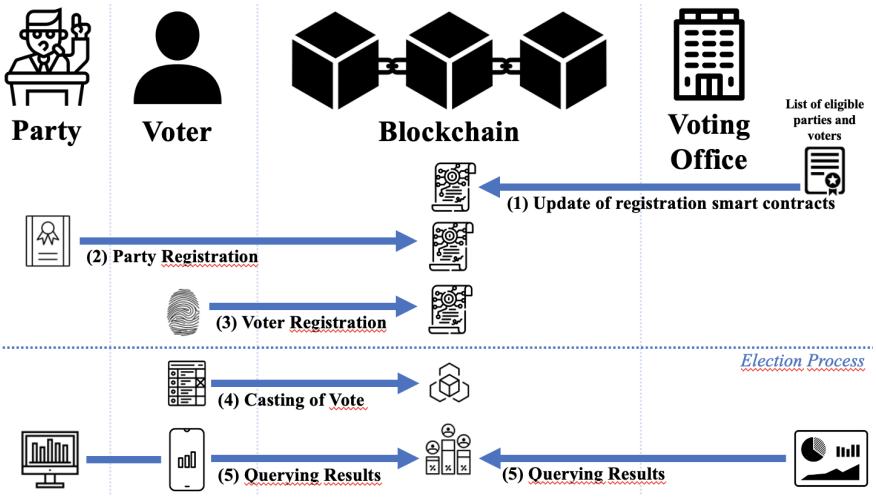

Figure 2: Detailed illustration of an election procedure

The voting office has lists of eligible voters and parties. (1) According to these lists, registration smart contracts are updated. (2 & 3) Parties and voters can then register themselves in the next step using the API, which can be done using a website or app on a smartphone and require a comparison of, for example, the ID number. (4) After registration, a voter can then cast their vote. The vote is added to the blockchain as a new block (how this works in detail is described in the following paragraph). (5) After the election, all actors have the possibility to query current election results.

How exactly a vote is registered in the blockchain is described in more detail below. After a voter has been registered, voters can cast their vote using their smartphone app or a website. For voters who do not have the option of voting digitally from home, additional polling stations with computers could be offered. When a voter casts their vote, it is written to the blockchain as a new block using the API. Such a block consists of a unique key and five elements, as shown in figure 3.

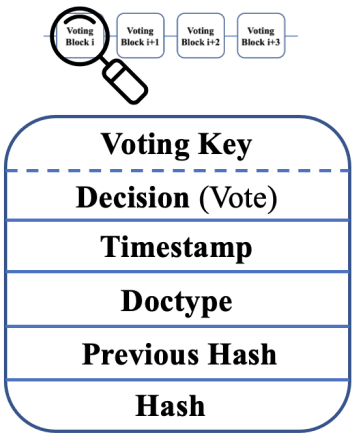

Figure 3: Voting Block in detail

While the voter only casts their vote, the system also generates a voting key, a timestamp, a doctype and two hash values. The hash is the result of the SHA256 hash function, which takes as an argument timestamp, decision and previous hash. The previous hash corresponds to the hash value of the previous block in the blockchain. This concatenation and calculation of hash values ensure that the blockchain cannot be changed, because a change of a decision would lead to a change of the hash value, which would no longer match the previous hash in the next block. This circumstance leads to an immutability of the blocks, which makes election fraud almost impossible and establishes trust in the smart voting system.

By evaluating the blockchain, i.e. the individual votes, the voter himself can also check whether their vote was cast correctly and see the current election results.

4.2 Code Implementation

The code for this project was written in the javascript code language. The chaincode is based on the structure of the fabcar example of hyperledger-fabric. For simplicity reasons, the structure of the fabcar network (consisting of an orderer organisation and two peer organisations) with the consensus mechanism "SOLO" was kept. Of course, both of these defaults could or would have to be changed in an implementation. The smart contract created in this project only covers the voting process, but not the voter or candidate registration process.

In the following, it will be explained in more detail how the chaincode for the voting smart contract is structured. A "Vote" class with five functions was defined: "initLedger", "queryVote", "queryAllVotes", "castVote" and "makePoll".

The initLedger function is used to initialise the ledger and fills the ledger with a first default vote. A vote always consists of a key and five elements: a timestamp (date and time of the vote), a decision (the actual vote), a doctype (always "vote"), a previous_hash and a hash. The hash is calculated using the SHA256 function and takes the timestamp, decision and the previous hash as input. All elements are stored as strings. The first (default) vote written to the ledger with the initLedger function has predefined values: a timestamp of "Dec 20 2022, 20:00:00", no decision ("-"), an empty previous hash ("0"), a corresponding first calculated hash value, a doctype "vote" and the key value "VOTE0". This block is shown in figure 4.

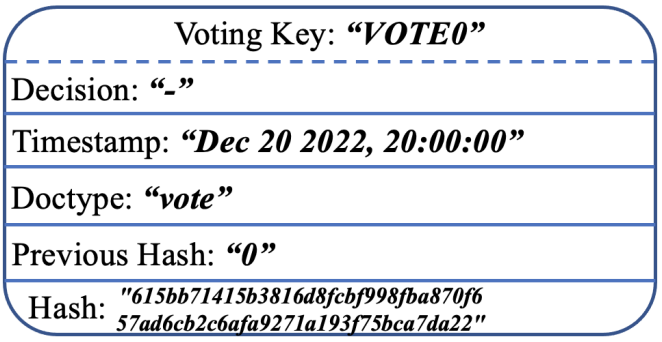

Figure 4: First Voting Block

This initiation thus writes the first block of the blockchain (after the genesis block generated during network creation).

The queryVote function takes a voteNumber (i.e. the key of a vote) as input, queries the vote from the chaincode state and returns the five elements of this specific vote as a result.

The queryAllVotes function works similarly. This function does not need any input and returns elements and keys of all votes. It should be noted that "all votes" currently refer only to the first 1,000 votes, but this number could of course be arbitrarily larger (or smaller) depending on the application.

The castVote function is the heart of this chaincode and needs only one decision as input. The function queries the hash of the previous vote and writes it into the new vote. This previous hash is then used together with the automatically generated timestamp and the decision made by the voter to generate the new hash using the SHA256 function. Furthermore, the default doctype "vote" is written into the vote and a new unique key is developed for this vote. Finally, the key is written to the ledger along with all the elements of the vote, or added as a new block.

The last function of the chaincode named makePoll allows the user to query the current result of the election. For this purpose, the votes cast so far are evaluated from the ledger and a vote count is returned. In this paper, four political ideologies (representing parties)

were used. Thus the vote count sums up all votes for the socialist party, the progressive party, the liberal party and the conservative party. In addition, the function returns all invalid votes (in this case, all votes that differ from these four parties) and the sum of all votes cast. The first (default) vote cast with the initLedger function has been removed from this calculation to avoid possible bias. Again, it must be mentioned that these political parties were only used for demonstration purposes, they could be supplemented or replaced by other voting options.

Also, the code files of the calling code were partly changed or added. The enrollAdmin.js and registerUser.js files were taken over unchanged from the fabcar example to keep the network structure. As described in the beginning, voter and candidate registration could be included in further smart contracts, but this would have been outside the scope of this project.

The query.js file was modified to request input from the user. The user can then decide if he wants to query a specific vote using its unique key or all votes. According to this input, the previously described queryVote or queryAllVotes functions are then used to funnel information about the respective vote(s) is returned.

A new poll.js file was created to trigger the makePoll function of the smart contract and retrieve current election results.

Last but not least, a new vote.js file was created. This also gives the user the possibility to make a choice after activation. The user input is then used as an argument for the castVote function of the chaincode and the vote is cast.

Attached to this paper is a separate pdf-file ("Election Guidelines"), which contains a link to the project code and guidelines for the use of the code, as well as a txt-file ("execution.txt"), in which the described guidelines were followed and the code was executed accordingly.

5. Results

In relation to the research question, the result of the code implementation leans towards a hypothetical nature. The presented proof of concept works, and it is rather easy to argue that a blockchain-based voting system would indeed regain trust in democratic elections. However, to actually answer the research question, further research must be conducted in relation to this field over a stretch of time.

Some aspects were identified to cause mistrust in traditional election systems are now clearly addressed by attributes of the blockchain applications. By being able to query the blocks on the blockchain, a voter can undeniably check if their vote is cast as they intended to. Additionally, the almost guaranteed immutability of the blockchain makes it impossible to tamper with casted votes once they are stored on the ledger. This for one assures a

complete audit trail for any revisions of the election at a later stage in time and also makes it very hard for individuals or parties to manipulate the election in the first place.
Also, any electronic voting system and especially one that is blockchain-based can be designed to be reliable by distributing copies of the ledger across multiple physical devices. Further benefits of electronic voting systems arise from the fact that they can be accessed from anywhere if a secure device and an internet connection are present. Here, the intrinsically secure nature of the blockchain towards cyber threats is a strong argument for a blockchain-based system over any other form of a database.

6. Discussion

Many details that would be important in an actual comprehensive implementation of the entire voting system are up for discussion.

Moreover, the practical question of whether people do have trust in any electronic voting system at all rather than pen and paper is always up for debate. As mentioned before, this can be answered by comprehensive research.

One of the main points that would need thorough consideration when implementing full-scale applications of the voting system is the architecture. Specifically, if a centralised or decentralised voting system is required. Both bring benefits and disadvantages, and it will likely be dependent on the actual application scenario and which option is preferable. In the case of democratic elections within a country, a fully decentralised approach like the blockchain network will not be feasible. The legal requirements bounding such an election are far beyond what can be done without a central organisation that needs to be trusted and that is responsible for conducting the election. This is already obvious at the stage of voter and party registration. Some form of voting office needs to make sure that only people who by law are eligible to vote are registered and know who these people are. This implies that any blockchain-based voting system will be operated as a private blockchain, so it cannot be fully decentralised. In the end, people need to trust their government to act upon the results obtained by any form of an election system, essentially, which will make decentralised elections of any form impracticable.

In case no voter registration is required for an election, it can be beneficial to implement fully decentralised blockchains that do not require any trust within the network. Here, additional considerations regarding the consensus algorithm and who is responsible for maintaining code and network of the application are required.

In either of the cases, the advantage of having multiple nodes should not be given away. Higher fault tolerance is of good use for private and public electronic voting systems equally. One central organisation can maintain multiple nodes across the country for example to achieve this.

7. Limitations

Upon further analysis, using blockchain technology in a voting system is a feasible and impenetrable way to regain trust within our democratic elections at a government scale. However, these are the initial steps toward a framework for the deployment of a blockchain-based centralised voting system rather than a decentralised one.

This project derives from a previous class of fabcar only containing two organisations and two peers. Due to the limited nature of the project, it was decided to use only two peers to act as our governmental representation.

A different consensus algorithm should also be considered when conceptualising a voting-based system. Although the “SOLO " algorithm fits the scope of the project, it is used for experimenting with HLF networks and is not used within a standard production environment. Thus, a new consensus algorithm has been created and adopted in the research space for this specific necessity called proof of vote (POV). The architecture of PoV is based by the governmental candidates on their credibility between the core nodes and other nodes in the blockchain within a distributed environment (Li et al., 2020).

In addition, separate smart contracts for registering voters and candidates are considered to be out of scope due to the contextual limit and feasibility of the time and resources given. This could be later added on in future work. In order to make this a fully implemented voting system, the registration of voters and parties is essential. The voter is an essential component of a voting system since it is the second stage in the system within the identity verification phase to keep track of the person’s casted vote. This also serves as a control mechanism since it prevents unregistered voters from participating in the election by stopping them from casting a ballot (Alvi et al., 2022). Although parties are similar to voters, parties must provide additional information, for example, candidate names, region, and seat numbers all require a separate smart contract. The creation of a candidate contract would add the information of a candidate onto the blockchain.

In regards to the limitations of online voting systems as a whole is directly attributed to the deployment and implementation process of digital signature schemes and other factors that correlate with privacy, unreusability, eligibility, fairness, soundness, and completeness that can be referenced in the Related Works.

8. Conclusion

In this paper, we utilise blockchain and fabcar to answer the question *how blockchain technology can be leveraged to regain trust in democratic elections*. Implementing blockchain technology is a feasible solution for regaining trust in democratic elections but there are a few considerations when using this: a fully decentralised approach for wide-scale democratic elections is not feasible since there must be a private network for legal purposes and it is best to utilise a different consensus algorithm catered towards a voting system. The voting office, political parties, and voters use an API to interface with the blockchain and execute the specified applications. Parties and voters can be registered with the use of smart contracts, but elections can also be held and results can be queried. The suggested solution assures security, anonymity, and integrity by utilising blockchain. By storing voter information as a hash in the blockchain, this technique ensures voter anonymity.

References

Alvi, S. T., Uddin, M. N., Islam, L., & Ahamed, S. (2022). DVTChain: A blockchain-based decentralized mechanism to ensure the security of digital voting system voting system. *Journal of King Saud University-Computer and Information Sciences*, *34*(9), 6855-6871.

Ayed, A. B. (2017). A conceptual secure blockchain-based electronic voting system. *International Journal of Network Security & Its Applications*, *9*(3), 01-09.

Czepluch, J. S., Lollike, N. Z., & Malone, S. O. (2015). The use of block chain technology in different application domains. *The IT University of Copenhagen, Copenhagen*.

Fujioka, A., Okamoto, T., & Ohta, K. (1992, December). A practical secret voting scheme for large scale elections. In *International Workshop on the Theory and Application of Cryptographic Techniques* (pp. 244-251). Springer, Berlin, Heidelberg.

Gailly, N., Jovanovic, P., Ford, B., Lukasiewicz, J., & Gammar, L. (2018). Agora: bringing our voting systems into the 21st century.

Hsiao, J. H., Tso, R., Chen, C. M., & Wu, M. E. (2018). Decentralized E-voting systems based on the blockchain technology. In *International Conference on Ubiquitous Information Technologies and Applications, International Conference on Computer Science and its Applications* (pp. 305-309). Springer, Singapore.

Khan, S., Arshad, A., Mushtaq, G., Khalique, A., & Husein, T. (2020). Implementation of decentralized blockchain e-voting. *EAI Endorsed Transactions on Smart Cities*, *4*(10).

Lee, K., James, J. I., Ejeta, T. G., & Kim, H. J. (2016). Electronic voting service using block-chain. Journal of Digital Forensics, Security and Law, 11(2), 8.

Li, C., Xiao, J., Dai, X., & Jin, H. (2021). AMVchain: authority management mechanism on blockchain-based voting systems. *Peer-to-peer Networking and Applications*, *14*(5), 2801-2812.

Li, K., Li, H., Wang, H., An, H., Lu, P., Yi, P., & Zhu, F. (2020). PoV: An Efficient Voting-Based Consensus Algorithm for Consortium Blockchains. *Frontiers in Blockchain*, *3*:11. https://doi.org/10.3389/fbloc.2020.00011

McCorry, P., Shahandashti, S. F., & Hao, F. (2017, April). A smart contract for boardroom voting with maximum voter privacy. In International conference on financial cryptography and data security (pp. 357-375). Springer, Cham.

Pennycook, G., & Rand, D. G. (2021). Examining false beliefs about voter fraud in the wake of the 2020 Presidential Election. *The Harvard Kennedy School Misinformation Review*.

S. T. Alvi, M. N. Uddin & L. Islam, "Digital Voting: A Blockchain-based E-Voting System using Biohash and Smart Contract," *2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT)*, 2020, pp. 228-233, doi: 10.1109/ICSSIT48917.2020.9214250.

Yavuz, E., Koç, A. K., Çabuk, U. C., & Dalkılıç, G. (2018, March). Towards secure e-voting using ethereum blockchain. In *2018 6th International Symposium on Digital Forensic and Security (ISDFS)* (pp. 1-7). IEEE.

Zhao, Z., & Chan, T. H. H. (2015, December). How to vote privately using bitcoin. In *International Conference on Information and Communications Security* (pp. 82-96). Springer, Cham.

Zimmermann, F., & Kohring, M. (2020). Mistrust, disinforming news, and vote choice: A panel survey on the origins and consequences of believing disinformation in the 2017 German parliamentary election. *Political Communication*, *37*(2), 215-237.