electronics

MDPI

Article

Improving Election Integrity: Blockchain and Byzantine Generals Problem Theory in Vote Systems

Patrick Mwansa * and Boniface Kabaso

Department of Information Technology, Faculty of Informatics and Design, District Six Campus, Cape Peninsula University of Technology, Cape Town 7925, South Africa; kabasob@cput.ac.za
* Correspondence: 212300482@mycput.ac.za; Tel.: +27-63-554-0916

Abstract: In the digital age, maintaining election integrity is critical, especially in Africa, where the security of electronic elections is often questioned. This study presents a blockchain-based vote counting and validation (BBVV) system developed using a mixed methods approach that combines stakeholder questionnaires to capture system specification and randomized historical election data analysis, following the Design Science Research strategy. Using the theory of the Byzantine General Problem, the BBVV protocol is proposed, which provides an accurate local count of votes at polling stations before national aggregation. The system was tested with randomized historical election data on the Algorand blockchain TestNet and confirmed that a local consensus on the vote count could be reached before it is added to the national tally on the blockchain. Our results show that in the cases where consensus was reached, this was the instance in only about 5% of the voting scenarios, with only 10% of the total vote being considered valid due to the strict consensus requirements. In addition, significant discrepancies were found between officials, with no consensus reached in 95% of cases which was due to the rogue values generated by a randomized dataset. The performance of the BBVV system was evaluated using transaction metrics, saturation, throughput, traffic, and latency to assess its efficiency, scalability, and reliability. The results suggest that blockchain technology can significantly improve the integrity of elections by ensuring a transparent, secure, and accurate vote-counting process. Future work will focus on improving the adaptability and scalability of the BBVV system for different electoral situations.

Keywords: blockchain; e-voting; Byzantine; consensus algorithm; Algorand; TestNet

Citation: Mwansa, P.; Kabaso, B. Improving Election Integrity: Blockchain and Byzantine Generals Problem Theory in Vote Systems. *Electronics* **2024**, *13*, 1853. https://doi.org/10.3390/electronics13101853

Academic Editor: Mehdi Sookhak

Received: 20 March 2024
Revised: 20 April 2024
Accepted: 2 May 2024
Published: 9 May 2024

1. Introduction

Elections are the cornerstone of democratic governance. They serve as the medium through which citizens express their preferences and elect their representatives. However, the integrity of elections in many democratic African countries is a cause for concern, mainly due to inconsistencies and ambiguities in the counting of votes. Such inconsistencies often lead to disputes and mistrust among stakeholders and undermine the essence of the democratic process. Numerous studies have highlighted the challenges African states face in ensuring transparent and trustworthy elections, with a focus on the vote-counting phase [1,2].

With the advent of technology, there has been increased interest in the use of digital solutions to address the above challenges. Blockchain technology, known for its decentralized and immutable nature, has shown promise when it comes to improving transparency and trust in various sectors, including elections [3,4].

Recent research has investigated the potential of blockchain for election management. Initial results show that it can ensure a transparent and tamper-proof election process [5,6]. Blockchain technology, characterized by its revolutionary attributes of decentralization, transparency, and immutability, has gained acceptance in numerous sectors, including finance, supply chain, and healthcare [7]. It holds particular promise in the area of electronic

voting (e-voting), where traditional systems have repeatedly struggled with issues of trust, transparency, and security [8].

The blockchain's ability to record transactions in a tamper-proof manner makes it an ideal candidate for ensuring the integrity of the vote count. Furthermore, applying the Byzantine Generals Problem as a theoretical framework to solve a social problem, such as reaching a consensus on the actual vote count at each polling station before that vote count is recorded on the blockchain for national aggregation, increases the integrity and accuracy of the election results.

The electoral process involves several different stages, including canvassing for votes, voter registration, voting, and the subsequent counting, recording, and announcement of results. However, the critical stage of vote counting and validation poses a major challenge, especially when it comes to ensuring the accuracy and trustworthiness of recorded votes. Conventional methods of vote counting are prone to human error, manipulation, and lack of transparency, which can undermine public confidence in election results.

This research focuses on overcoming vote counting and validation challenges by proposing the blockchain-based vote counting and validation (BBVV) protocol underpinned by the principle of the Byzantine General Problem (BGP) into the vote counting and validation phase. The aim is to reach a consensus between the poll workers in the polling stations who are in charge of entering the physically counted votes into the blockchain network. The vote count is entered at the edge of the network, where the BBVV protocol takes effect and automatically runs to achieve the required consensus. A trustworthy record of the vote count in the blockchain requires the agreement of more than two-thirds (over 67%) of the poll workers to enter the same vote count. This approach seeks to improve the accuracy, transparency, and integrity of elections by enabling each polling station to validate its vote count as part of the national totals on the blockchain, shown in Figure 1. In addition, this study evaluates the performance, scalability, capacity, and reliability of the blockchain-based vote validation (BBVV) artifact through transaction metrics, saturation analysis, transaction throughput, traffic analysis, and latency assessments.

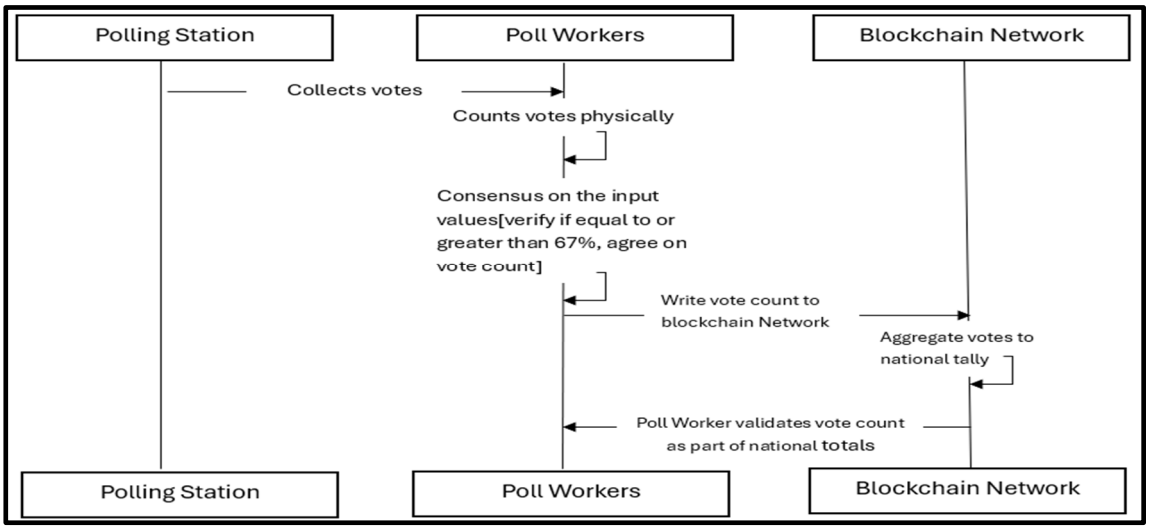

Figure 1. Vote counting, recording, and validation.

In this paper, the term Electoral Proof of Stake (EPoS) is used to refer to the collective roles of poll workers, election observers, and election officials.

2. Theoretical Framework

The Byzantine General Problem (BGP) serves as a fundamental concept in the development of consensus algorithms that are critical to blockchain technology, especially in applications such as blockchain-based vote counting and validation (BBVV) artifacts. The BGP illustrates the difficulties associated with achieving consensus in distributed systems

with potentially treacherous components. This is similar to ensuring trust in the vote counting and validation process in elections [9]. An underlying theoretical paradigm in distributed computing is the Byzantine General Problem [10]. It describes the difficulty in reaching agreement amongst a variety of organizations, particularly when some of these entities, "like generals in a Byzantine army," act treacherously by disseminating inaccurate or misleading information. Ultimately, the issue is how to create a framework where compliant generals can come to a consensus despite the traitors' cunning tactics. This issue emphasizes the intricacy of distributed systems as well as the value of dependability and trust in cooperative settings.

Kuo et al. [11] contribute to this area by proposing a fair Byzantine agreement protocol that addresses the fairness and performance issues in blockchain consensus. Their work is particularly relevant to BBVV as it ensures that each participant's value has an equal probability of being selected, which is essential for trust in voting processes. The protocol they propose is responsive and partition-proof. It tolerates up to one-third corruption, meaning it can maintain security even if the network is partitioned, and it can resume normal operation once the partitioning is resolved. In the case of the BBVV artifact, this is applied synonymously to require two-thirds approval under the Electoral Proof of Stake (EPoS) to achieve consensus in recording the correct vote count on the blockchain so that one-third could be malicious.

In addition, the work of [12] on the Practical Byzantine Fault Tolerance (PBFT) protocol with repairable voting nodes provides insights into the reliability and performance of blockchain systems. Their analysis using a multi-dimensional Markov process and the first-passage time method provides a framework for understanding the throughput, availability, and reliability of PBFT-based blockchain systems. This analysis guided the development of BBVV artifacts by ensuring that the system remains functional and fair even when nodes fail and recover, reflecting the dynamic nature of real-world voting systems.

Figure 2 shows a graphical representation of the Byzantine General Problem, in which several generals, represented as nodes, use messages to coordinate a joint decision. The diagram shows the direct exchange of messages between some generals, which represents ideal, non-deceptive communication. However, the introduction of disloyal generals complicates this scenario. These untrustworthy figures send deceptive or contradictory messages, which are labeled "traitor messages" in the diagram. The main challenge is that the loyal generals must overcome these deceptive messages to reach a unanimous decision, which is depicted as a "consensus among the loyal generals". This image effectively conveys the key challenge of balancing trust and deception to achieve unified decision-making.

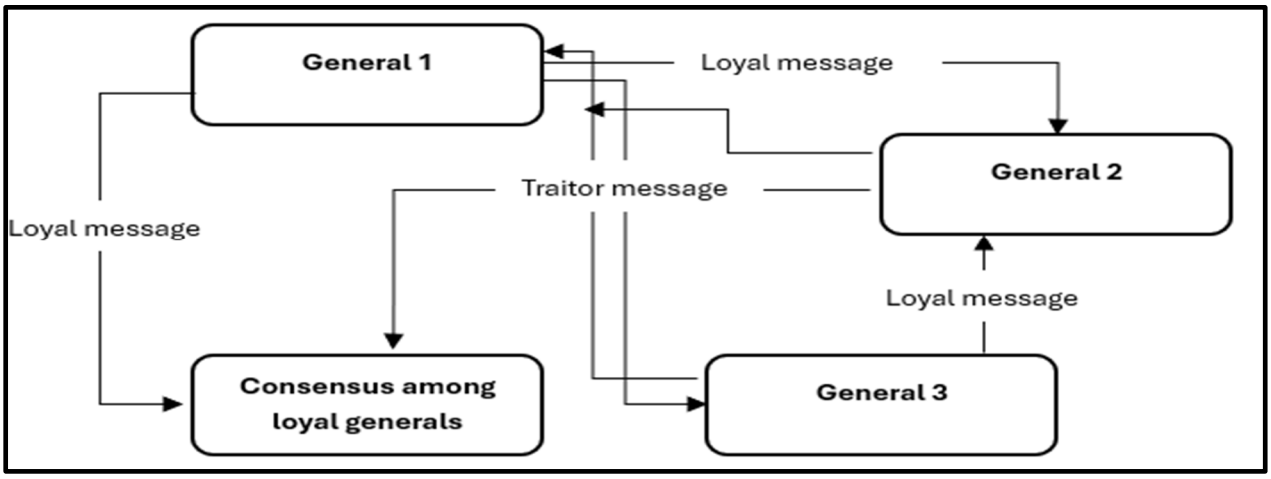

Figure 2. The Byzantine consensus.

In summary, the theoretical framework established by BGP, with the advances in fair, responsive, and partition-resistant Byzantine agreement protocols, provides a solid

foundation for the development of a BBVV artifact. By leveraging these concepts, a BBVV artifact was created that ensures a trustworthy and reliable vote counting and validation process in elections. This was achieved by developing a BBVV protocol based on the BGP. This protocol allows the EPoS in a polling station to reach a consensus on the actual vote count to be recorded on the blockchain in order to aggregate the votes at a national level.

3. Conceptual Structure

Election data are managed via Algorand's blockchain platform, which is known for its efficiency and speed, especially with its Layer 1 smart contracts. Figure 3 shows the structure of the BBVV implementation.

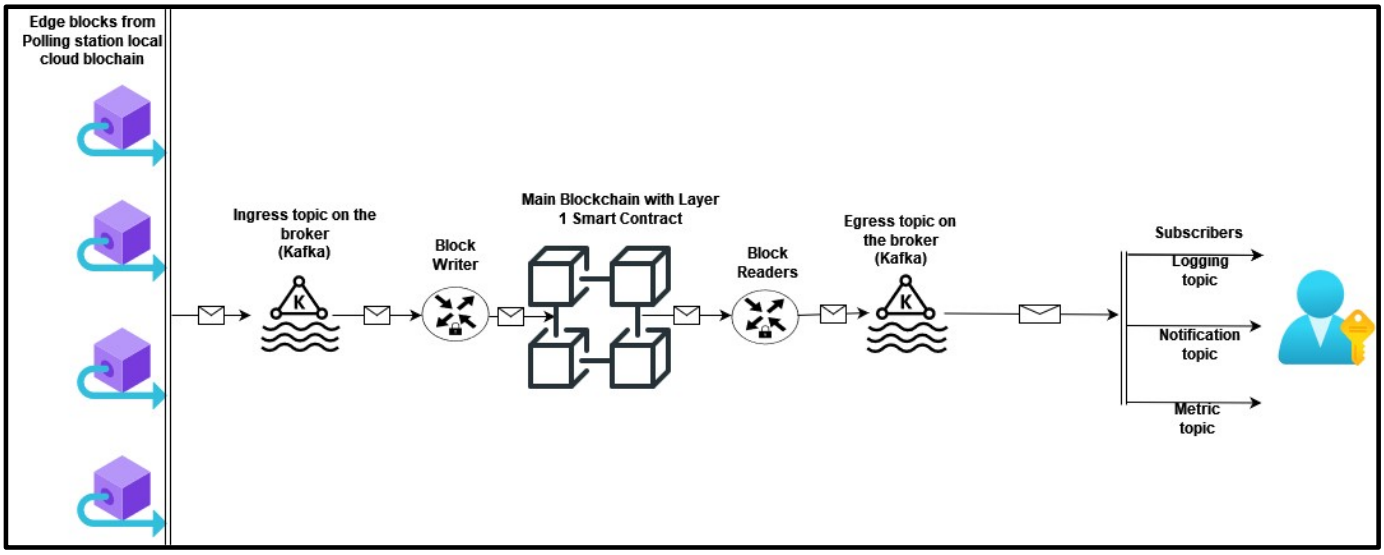

Figure 3. The BBVV overall structure.

3.1. Transferring Edge Blocks via Kafka

Edge blocks: These blocks, located at the local level of each polling place's blockchain, store the final vote count.

3.2. Kafka as an Ingress Message Broker

Kafka acts as an entry point for these edge blocks and effectively manages the incoming data. It queues the data from the various polling stations, ensures that the system is not overloaded, and maintains an orderly flow of data to the main blockchain.

3.3. Layer 1 Smart Contracts on Algorand

As soon as the vote count reaches the Algorand blockchain, Layer 1 smart contracts process the data. Algorand is particularly advantageous for this purpose as it can process transactions quickly and efficiently thanks to its high throughput and low latency. This fast processing is crucial for election scenarios where timely results are important. The smart contracts at this level automatically aggregate vote counts from different locations to provide an overall nationwide result in a much shorter timeframe.

3.4. Aggregated Data Management and Storage

National count: once processed by Algorand's smart contracts, the aggregated vote count is securely stored on the blockchain. This record is immutable and tamper-proof, providing a reliable and transparent record of all votes cast.

3.5. Controlled Release by Block Readers

Block readers: These entities or systems within the blockchain network are responsible for verifying the summarized vote counts. They determine the appropriate time to release

the results to the public and ensure that all procedural checks are met before the data are published.

3.6. Egress Message Broker for Distribution of Data

Release to subscribers: Once released by the block readers, an egress message broker manages the distribution of the election results to the various subscribers. This step ensures a coordinated release, prevents premature publication, and ensures that all subscribers receive the information at the same time.

The Implementation of the BBVV is outlined in Figure 4 where,

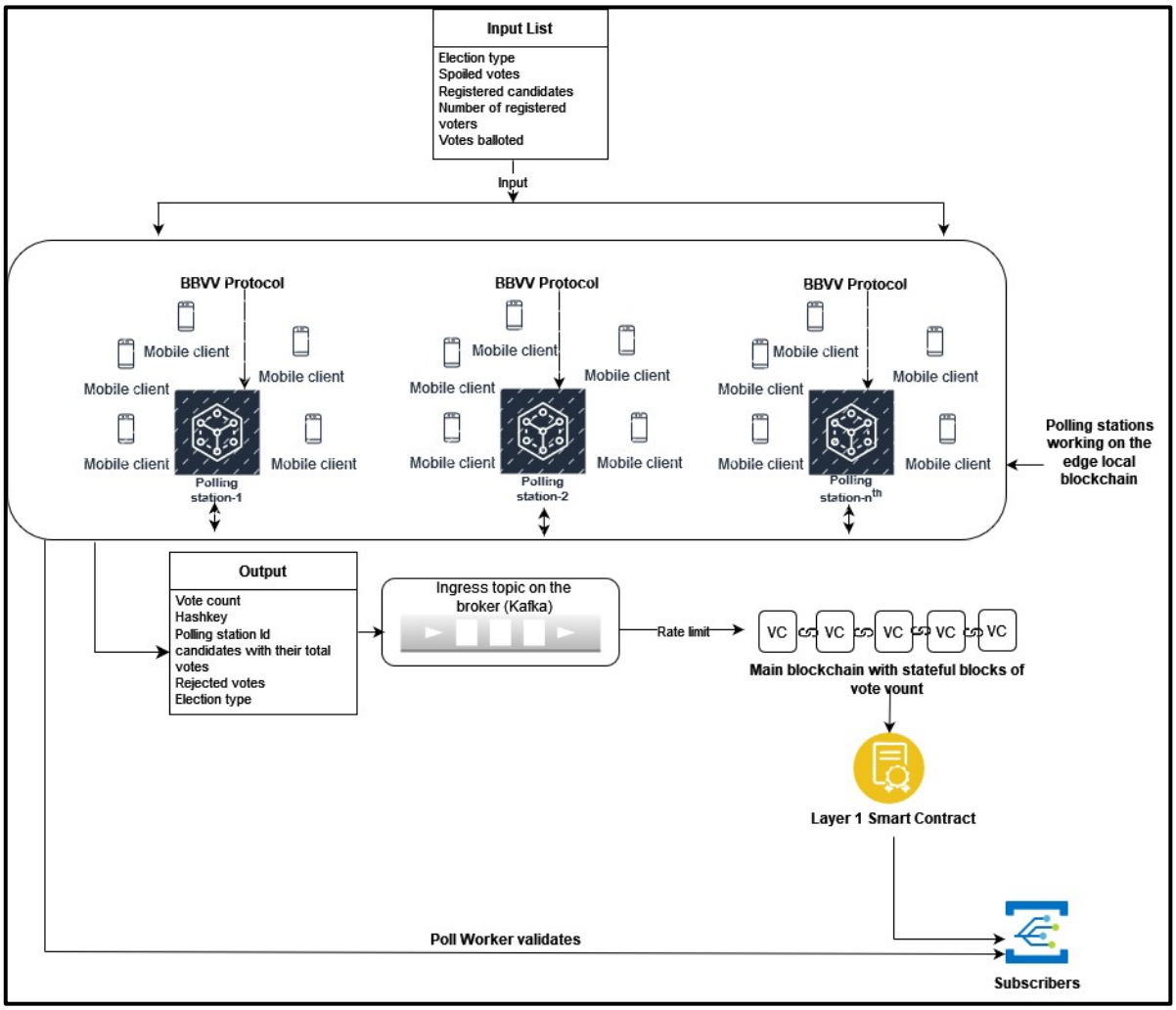

Figure 4. The BBVV implementation.

Local blockchain storage: Each local polling station maintains a blockchain in which the votes are recorded as transactions. The last block in the local blockchain, the so-called edge block, contains important data such as the hash key and the total number of votes. This hash key serves as a unique identifier that ensures data integrity between the blocks and across the entire network.

Integration of the Kafka message blocker: Once voting is complete, the data are transferred from the edge blocks to a Kafka system, the message blocker. Kafka is a distributed streaming platform that can process large amounts of data. It queues these blockchain blocks and manages the data flow so that the system is not overloaded. Kafka is configured to forward the blocks to the next stage of the process at a specific speed, ensuring a steady and manageable stream of data.

Cloud blockchain synchronization: The blocks released by Kafka are then forwarded to a cloud-based blockchain. This secondary blockchain serves as a centralized ledger where the votes from multiple local blockchains in different polling stations are merged. This centralization is essential for creating a nationwide tally and ensures that all data remains consistent and secure.

Smart contract execution on Layer 1: As soon as the blocks arrive on the cloud blockchain, a smart contract is automatically triggered. This smart contract is designed to calculate the total number of votes from the incoming data. Smart contracts are self-executing contracts where the terms of the agreement are written directly into the code. In this case, the total number of votes is calculated automatically as soon as the required data are received.

Distribution of results to subscribers: Once the smart contract has calculated the total number of votes, this sum is sent to various subscribers. The subscribers can be media, government agencies, or other authorized entities interested in the election results. This distribution is handled via the blockchain network, which ensures that all subscribers receive the same tamper-proof data at the same time.

Verification by polling stations: To further increase security and trust in the election process, each polling station can independently verify the vote count contained in the national totals. For this purpose, they use a combination of public and private cryptographic keys. The private key is unique to each polling station and is used to confirm the vote totals, while the public key allows others on the network to verify that the data come from a legitimate source and matches the national totals.

4. Layer 1 Smart Contract Implementation

Overall, the equations ensure proper recording of votes across time intervals, proper aggregation between polling stations, and a continuous record of the election period without overlaps or gaps. This is critical to maintaining the integrity and verifiability of election results.

4.1. Definitions

$C(\mathrm{T}, \mathrm{P})$ Vote count for candidate Y from polling station P received at time T.

$S(\mathrm{T_b}, \mathrm{T_c})$—Total votes for candidate Y from a set of polling stations received between the beginning of time $(\mathrm{T_b})$ and end of time $(\mathrm{T_c})$, where $(\mathrm{T_c} - \mathrm{T_b}) = 2$ h. Where the time can be changed to suit the time an election vote counting period must run.

ST—Total votes for candidate Y accumulated over various time intervals $(\mathrm{T_b}, \mathrm{T_c})$ spanning a total period of $\mathrm{X_1}$ h or however long an election runs.

1. Relationship between C and S.

To accumulate the votes for candidate Y from multiple polling stations over a time interval from $\mathrm{T_b}$ to $\mathrm{T_c}$, we consider all polling stations P and all relevant timestamps T within the interval $[\mathrm{T_b}, \mathrm{T_c}]$:

$$S(\mathrm{T_b}, \mathrm{T_c}) = \sum \mathrm{p} \sum_{\mathrm{T}=\mathrm{T_b}}^{\mathrm{T_c}} C(T, P) \tag{1}$$

This equation in (1) sums up all votes $C(T, P)$ from each polling station P during the specified interval $[\mathrm{T_b}, \mathrm{T_c}]$.

2. Relationship between S and ST.

Given that ST in (2) is the total number of votes counted over a series of intervals across a total period of $\mathrm{X_1}$ h, where $\mathrm{X_1}$ is the number of hours it takes an election to be conducted assuming n such intervals:

$$\mathrm{ST} = \sum_{\mathrm{i}=1}^{\mathrm{n}} S(\mathrm{T_{bi}}, \mathrm{T_{ci}}) \tag{2}$$

where $\mathrm{T_{bi}}, \mathrm{T_{ci}} = 2$ h for each interval i, and the series of intervals cumulatively spans $\mathrm{X_1}$ h.

3. Validation of consistency across intervals.

To validate that the intervals properly cover the X_1 h period without overlap or gaps, we can establish the following invariant in (3):

$$T_{bi+1} = T_{ci} \text{ for i} = 1 \text{ to n} - 1 \quad (3)$$

This ensures that each interval begins immediately after the previous one ends, with no overlap or gap between them.

4. Coverage and continuity over X_1 Hours.

Ensure the first interval begins at the start of the X_1h period and the last interval ends precisely at the X_1 h mark: This we can change as in polling closes, or all counting should be carried out, and all coverage carried out, this is shown in (4).

$$T_{b1} = \text{Start time}$$

$$T_{cn} = \text{Start Time} + X_i \text{ hours} \quad (4)$$

To ensure that the vote counts from individual polling stations are verifiable in the final totals through cryptographic means, such as hashing or digital signatures, we incorporate cryptographic hash functions or signatures into the mathematical model. This addition helps to validate that a specific polling station's data were included in the overall count.

4.2. Cryptographic Enhancement of the Model

1. Introduction of cryptographic hashes and signatures.

Let H represent a cryptographic hash function.

Let Sig (X, K_p) represent a digital signature of data X with the private key K_p of polling station P. This could be the block hash.

2. Incorporating hash into vote count.

Define C(T, P) not only as the vote count but also include a hash or signature that certifies its authenticity: C(T,P) = (count,Sign(count,KP)) Here, the count is the actual number of votes recorded at polling station P at time T, and Sign(count,KP) is its digital signature or block chain hash.

3. Aggregation with verification.

When aggregating these counts into the total $S(T_b, T_c)$, the process would also involve verifying the signatures to ensure data integrity:

$S(T_b, T_c) = \sum p \sum_{T=T_b}^{T_c} \text{verify}\big(C(T,P), K_p\big)$ Here, verify $\big(C(T,P), K_p\big)$ checks the signature of the count from polling station P to confirm it was indeed issued by P.

4. Cumulative verification for total votes ST.

The total ST is calculated by summing up all verified S intervals: $ST = \sum_{i=1}^{n} S(T_{bi}, T_{ci})$.

The integrity of each interval S is ensured by the verification of all included signatures.

5. Providing proof of inclusion.

To prove that the results from a specific polling station P have been included in the total, one would need to provide:

The signed vote counts Sign(count, K_p).

A chain of verified totals from S to ST showing the inclusion of P's counts.

This is facilitated by using the Merkle trees of blockchain or similar cryptographic structures, where each node is a hash of its children, providing a verifiable path from each individual entry to the root (in aggregate).

5. Related Works

The integrity of electoral systems is a fundamental aspect of democratic governance, and the emergence of blockchain technology has opened new possibilities for improving the security and reliability of electronic elections. The decentralization, immutability, and

transparency of blockchain are particularly well suited to addressing the vulnerabilities of traditional voting mechanisms, such as susceptibility to fraud and coercion, as well as the challenges of ensuring privacy and accessibility. A look at existing blockchain solutions for voting systems reveals a variety of approaches that aim to overcome these problems.

Onur and Yurdakul [13] have proposed ElectAnon, a protocol that prioritizes voter anonymity through zero-knowledge proofs and increases robustness by decentralizing authority control with timed machines. This approach not only addresses privacy concerns but also provides a scalable solution that significantly reduces operational costs, as evidenced by lower gas consumption compared to previous systems. Similarly, Ref. [14] developed SHARVOT, which uses Shamir's secret sharing and a circle shuffle technique to ensure the confidentiality and anonymity of votes. This secret share-based voting system utilizes the blockchain's ability to maintain a transparent and irrevocable record of votes.

Wang et al. [15] introduced an insecure and collusion-proof voting consensus mechanism on the blockchain. Their mechanism focuses on reducing the side effects of candidate uncertainty, thereby reducing false voting. They also introduced an incentive-compatible scoring rule to assess the trustworthiness of voting, with the aim of motivating voters to report true beliefs about candidates.

Mishra et al. [16] proposed an anonymous voting system using a quantum-based blockchain. Their work combines the advantages of blockchain with quantum resources, such as quantum random number generators and quantum key distribution. The proposed system is designed to be verifiable and can be implemented with currently available technology.

Balilo Jr. et al. [17] proposed an electronic voting system (EVS) using unique one-time password table sequence pattern authentication. Their work aimed to overcome the challenges associated with traditional voting methods, such as ballot forgery and coercion, by using the security mechanisms embedded in the EVS.

Eldridge examined the development of electronic voting systems for Australian federal elections [18]. His work emphasized the need for a system that is secure, accurate, and understandable to the average voter. His study also analyzed the iVote electronic voting system used in the 2017 Western Australian state election and highlighted potential security risks posed by cloud-based distributed denial-of-service measures.

Spanos and Kantzavelou [19] presented EtherVote, a secure electronic voting system that uses the Ethereum blockchain network. Their proposal focuses on identifying eligible citizens and aims to improve security and privacy and reduce election costs by eliminating the need for central government servers or databases.

Blessing et al. [20] conducted a security investigation and analysis of postal voting systems, focusing in particular on the electronic systems used in this procedure. Their findings revealed vulnerabilities in online voter registration systems that could allow attackers to alter or prevent a voter's registration. In addition, they pointed to privacy concerns related to vote-tracking systems.

The work of [3] presents a fully decentralized e-voting system that uses smart contracts to increase security and maintain voter privacy. Their system aims to establish a transparent and tamper-proof voting mechanism that minimizes the role of intermediaries and thus reduces the potential for voter fraud. In addition, ref. [21] introduced SBvote, a scalable, self-tuning voting protocol that can be customized for large-scale elections. The protocol is designed to process a large number of voters and is limited only by the capacity of the underlying blockchain platform. This scalability is significant for the adoption of blockchain in larger electoral contexts, such as national elections. The integration of blockchain technology into electoral systems has been sought to mitigate the risks associated with traditional voting methods and reap the benefits of digital transformation. However, this integration is not without its challenges. The literature identifies several key issues that need to be resolved to ensure the successful implementation of blockchain in electoral systems.

Another challenge is the scalability of blockchain systems to handle the volume of transactions involved in elections. Faour [22] provides a comprehensive comparison between current election systems and analyses their structure and the drawbacks that

should be considered for future improvements. Faour points out the limitations of current blockchain platforms such as Ethereum, which can only process a limited number of votes per minute, raising concerns about the feasibility of blockchain for large-scale elections.

The security of blockchain voting systems is also a cause for concern, particularly with regard to possible attacks by quantum computers. Mishra et al. [16] propose an anonymous voting system with quantum-assisted blockchain to improve the security features of blockchain with quantum resources. This approach aims to fulfill the requirements of a good voting system while being auditable and implementable with current technology. In addition, the existing infrastructure for conducting elections with electronic voting machines (EVMs) has numerous loopholes that could be exploited to cast false votes or distort the results. Mukherjee et al. [23] propose a blockchain-based e-voting system that eliminates these security risks and preserves voter anonymity. Their prototype, developed on the Ethereum platform, demonstrates the power of the system and its potential to enable a more reliable and fairer voting process.

Lastly, the time it takes to count the votes and the overall efficiency of the voting process are also important. Bulut et al. [24] suggest that blockchain can significantly reduce the waiting time for election results and improve the security and data integrity of votes. They emphasize that the protection of voters' privacy and the transparency of the election process are important requirements that their proposed system ensures. While blockchain offers a promising way to reform voting systems, there are still significant challenges to overcome in terms of privacy, scalability, security, and efficiency. The literature suggests that ongoing research and development is crucial to overcoming these challenges and realizing the full potential of blockchain in electoral systems.

The literature shows that blockchain technology holds great promise for reforming electronic voting systems. The analyzed blockchain solutions are designed to protect voter privacy, ensure the integrity of the voting process, and offer scalability. However, implementing these systems on a larger scale still requires further research to overcome the limitations of current technology and ensure that these systems are trustworthy and can be used in elections around the world. The references to the work of Onur and Yurdakul, Bartolucci et al., Sadia et al., Spanos and Kantzavelou, and Stančíková and Homoliak provide a comprehensive overview of the state of blockchain in electronic elections and lay the groundwork for future progress in this area.

6. Methodology

BBVV uses the Algorand blockchain platform, which is known for its efficiency, scalability and cost-effectiveness on its transaction fees. This system uses an architecture featuring poll workers, in this case called Electoral Proof of Stake (EPoS), at a polling station, who input vote counts, and a validator consensus algorithm, called the BBVV protocol, verifying these entries at the edge of the network. A stateful smart contract, written in Algorand's Transaction Execution Approval Language (PyTeal), manages this voting protocol. It restricts the submission of EPoS votes to those that are authenticated and authorized. Each EPoS interacts with the blockchain via the Pera Wallet, a secure blockchain Wallet that facilitates identity verification and transaction management on the Algorand network. This integration ensures that each submission can be accurately traced back to its polling station, confirming its legitimacy before and after it is aggregated at a national level.

EPoS submits encrypted vote counts through secure transactions via their Pera Wallets. A transaction consists of a validated block containing a vote count. These submissions are temporarily stored in a pending state within the smart contract. When a vote count is submitted, the smart contract triggers the BBVV protocol for the new submission to be verified. The smart contract, which runs at Layer 1 of the Algorand, is programmed to calculate whether submissions reach the required two-thirds majority (67%) consensus among EPoS. Reaching this threshold confirms the vote count's validity, which is then permanently stored in the national aggregation block. If consensus is not reached, the vote

count is rejected and discarded. This approach not only utilizes the security features of the blockchain and cryptographic authentication but also integrates the Pera Wallet to ensure the traceability and validation of each vote. This method increases the integrity of the system and provides a secure, transparent, and verifiable record of each vote count as part of the national count.

This study used a mixed methods approach to develop and evaluate a blockchain-based vote counting and validation system. The Design Science Research (DSR) methodology underpins our research strategy and ensures a thorough and systematic development of the technological solution. We utilize both qualitative and quantitative techniques to achieve our research objectives. As part of the qualitative research, questionnaires were used to identify the system requirements of election stakeholders, which helped in the design and development of the BBVV artifact. After considering the requirements gathered, the Byzantine Generals Problem was used as an underpinning theoretical framework to propose the BBVV protocol. Historical election results were randomized and used as quantitative data to assess the performance of the artifact. Particular attention was paid to maintaining the reliability and validity of the study, recognizing and addressing potential limitations and challenges. This was carried out by randomly selecting African countries with a mature democracy of 27 years and above. The DSR of the build and evaluate underpinned the process of developing the artifact through to its implementation.

In the DSR, Firstly, the project collected data and requirements from Electoral Proof of Stake (EPoS) and other selected stakeholders. Secondly, the proposed BBVV protocol consensus algorithm based on Byzantine theory was applied to authenticate and record legitimate votes on the edge network and later consolidate them on the blockchain. This process, secured by cryptographic keys, allows EPoS to verify their votes at the national count, which increases confidence in the accuracy of the vote. Thirdly, the accuracy of the output and the scalability of the system were tested in different environments. Lastly, the artifact was compared with current voting systems.

7. Proposed BBVV Protocol

In this study, we propose a protocol designed to streamline the voting process via the implementation of blockchain technology. This is achieved with the application of the Byzantine General's Problem Theory as an underpinning theoretical framework. The steps involved in executing the protocol are as follows:

- Initialization:

P: This is the number of the polling station. Each polling station is assigned a unique identifier called P. This is important in order to be able to distinguish between different polling stations.

- Authentication:

Auth(E): This function represents the authentication process of the Electoral Proof of Stake (EPoS), which is labeled EE. The function returns 1 if the EPoS has been successfully authenticated and 0 if authentication has failed. This step is important to ensure that only authorized persons can participate in the vote.

- Creation and allocation of Cryptotally:

CryptoTally(E): This function allows an authenticated EPoS to write to the blockchain. An EPoS right to write to the blockchain is only created if Auth(E) returns the value 1, indicating successful authentication. The function contains important tallied votes data, such as the total number of all counted votes and the current number of votes for each candidate.

- Initialization of the counted votes writing process:

X: This variable represents the total number of counted votes in the election.

Vi: These variables represent the counted votes each candidate has received. This is part of the setup process where the initial counted vote writing to blockchain parameters are set.

- Write blockchain:

WriteBlockchain (E, V1, V2,. . ., Vn): This function symbolizes the process by which the poll worker/ EPoS writes the voting data to the blockchain. This includes entering information about candidates, their party names, and party IDs.

- Consensus and validation:

Consensus (n, N): This function checks whether a consensus has been reached on the vote count. It returns 1 if at least 67% (the majority) of the poll workers / EPoS are of the opinion that the vote count is correct, where n stands for the number of officials or agents who agree and N for the total number of officials or agents present.

- Termination:

Close (C, E): This function represents the conclusion of the vote count writing process, which depends on the consensus result CC. If a consensus is reached, the vote count is confirmed and transferred to the blockchain.

- Validation and completion:

Validate(E): This function allows a polling station to verify that its vote count has been added or counted correctly in the total national vote aggregation. This step is crucial to ensure the integrity and transparency of the election process. Algorithm 1 shows a concise algorithmic structure of the BBVV protocol.

The BBVV Protocol

Algorithm 1. Algorithmic structure of The BBVV Protocol

Start: *Initialization:*
Let P be the polling station number, uniquely identifying each station.
Authentication:
Define a function Auth(E) Where E represents Electoral Proof of Stack, 1 if authentication is successful, 0 otherwise.

$$f(x) = \begin{cases} 1, & if\ EPoS\ is\ authenticated \\ 0, & otherwise \end{cases}$$

CryptoTally assignment:
Define a $CryptoTally(E)$ *that generates the right to write for authenticated EPoS.*
CryptoTally(E)=Auth(E) x right to write.
Cryptotally contains information like total counted votes to be written (y); counted votes for each candidate (a, b, c. . .. .).
Counted vote writing process initialization:
Let X be the total number of counted votes to be written.
Let Vi be the counted votes received by candidate i.
Consensus and validation:
Define a consensus function, consensus (n, N), where n is the total number of agreeing officials. N the total number of officials at the polling station.

$$Consensus(n, N) = \begin{cases} 1 \begin{cases} if = \frac{100n}{N} > 67 \\ \\ otherwise \end{cases} \\ 0 \end{cases}$$

Finalization:
Define a function Finalize (C, E), where C is the consensus result and E is the EPoS.
Finalize (C, E) = C x WriteBlockchain (E, V1, V2. . .. . .Vn).
Validation:
Define a function Validate(E) for each polling station to verify their counted votes as part of the national totals.
Stop.

8. BBVV System Architecture Design

The BBVV artifact, which is an election collation system, is based on a client-server architecture paradigm. The client-side or front-end uses the capabilities of Next.js, an outstanding framework built on top of React. The server-side element consists of a smart contract carefully developed using PyTeal by Algorand. To enhance the security of authentication and transaction signatures, the system is seamlessly integrated with Pera Wallet.

8.1. Primary Modules

- Client-side interface: Next.js;
- Server-side logic: PyTeal Smart Contract;
- Authentication mechanism: Integration with Pera Wallet.

8.2. Operational Workflow

- End users access the system interface via standard web browsers.
- Data request and transmission are carried out through the interaction of the interface with the backend smart contract.

To enhance security, Pera Wallet provides a mechanism for users to authenticate and digitally sign transactions. Figure 5 shows the integrated components.

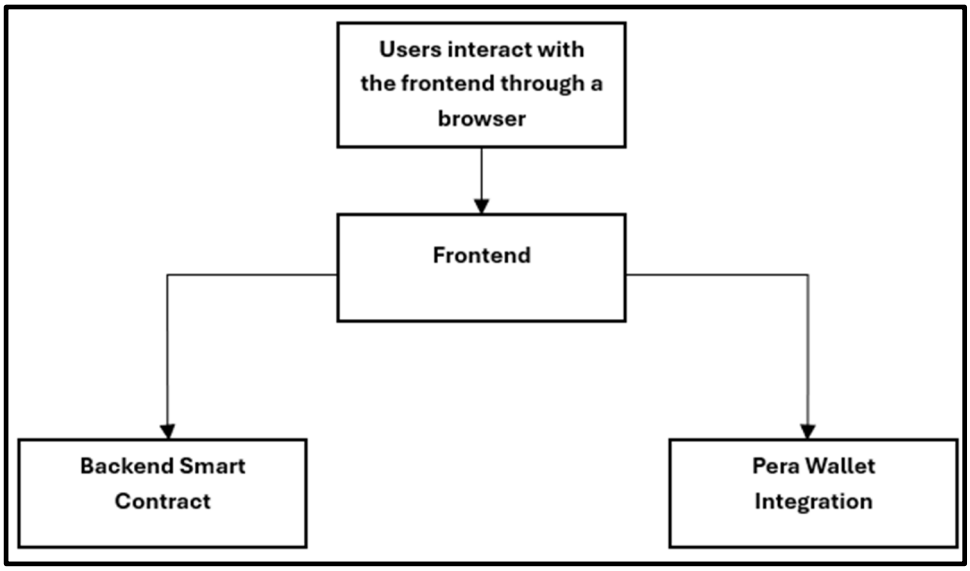

Figure 5. Integrated components.

8.3. Technological Stack Employed

- Client-side development: Next.js (Based on React);
- Server-side logic: Algorand PyTeal Smart Contract;
- Authentication mechanism: Pera Wallet;
- Versioning control: Git;
- Deployment mechanisms: Vercel (for the client side), with the smart contract commissioned on the TestNet iteration of the Algorand.

8.4. Functional Overview of the System

The BBVV is designed to streamline the collation and monitoring of election results. At its core, it uses a blockchain-anchored smart contract to ensure the integrity and secure management of election records. The client-side interface is not only intuitive but also provides users with a comprehensive portal to interact with the backend. The integration of Pera Wallet underlines the security framework, especially during the authentication and digital signing processes.

A. Distinctive Features of the Artifact

- Immutable data retention: election records, including results, find a secure repository on the blockchain thanks to the PyTeal Smart Contract, which ensures inviolability and enhanced security.
- Synchronous data reflection: The client-side interface can provide synchronous updates that reflect the collection and validation of election results in real time.
- Enhanced user identity verification: Pera Wallet integration increases security and provides users with a strengthened authentication process.
- Secure data transfer: Pera Wallet integration gives users the ability to add digital signatures to transactions, increasing data integrity during transmission.
- Comprehensive audit functions: The design of the blockchain ensures a comprehensive, tamper-proof log of all transaction activities and enables transparent and traceable audit trails.

B. The BBVV on Algorand

The BBVV uses non-relay and relay nodes. In this case, the non-relay nodes are implemented on the edge of the network of a polling station, since non-relay nodes are participating nodes, they were used to reach consensus on the vote count. The agreed vote count was then written to the Archival and indexed relay node containing the main blockchain ledger. A "full" node in a blockchain usually stores the whole ledger, comprising all the transactions in each block. The archival nodes in Algorand serve the same purpose and store all of the ledger information [25]. This solution leverages the usage of internet resources on edge only by EPoS to write the physically counted vote count to the blockchain. This allows all polling stations to verify if their final vote count was included in the final national tally of the vote count results. Figure 6 illustrates this architecture of the proposed blockchain vote-counting artifact on the Algorand platform. This platform creates security in that the vote cannot be altered and allows verification to ascertain if the vote was counted in the national tally.

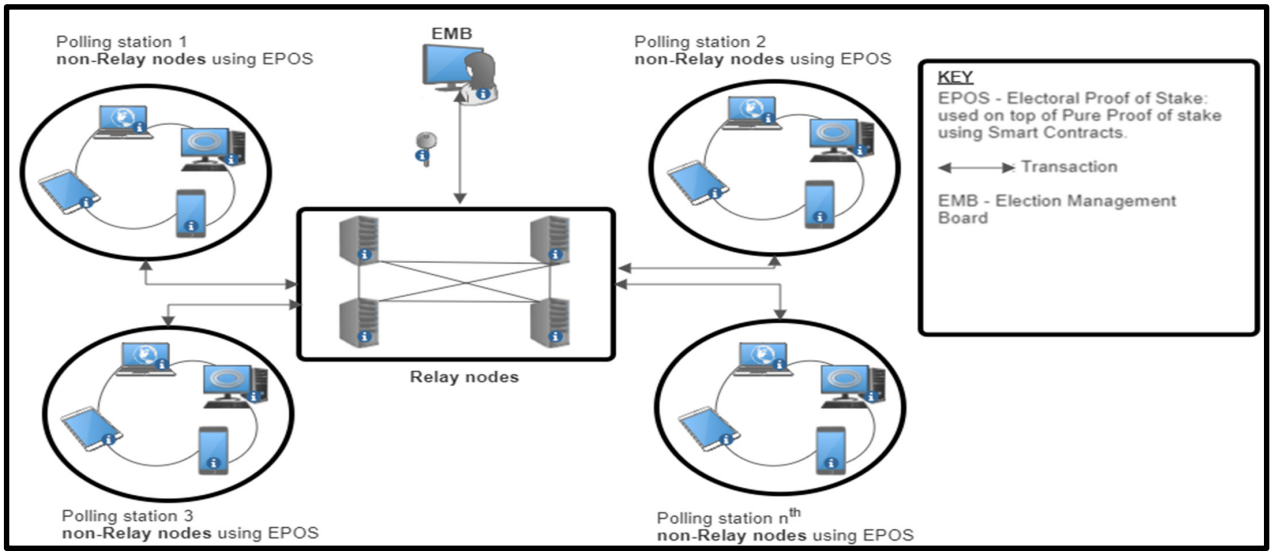

Figure 6. The BBVV on Algorand platform.

C. Vote Validation with Pera Wallet

The integration of Pera Wallet into a blockchain-based vote counting and validation (BBVV) represents a significant step forward in ensuring transparent, secure, and trustworthy election processes.

Below, you will learn how these components have been integrated and connected:

i. Leveraging Edge Computing:

- Decentralized processing: edge computing enables the decentralized processing of votes. This reduces latency and dependency on centralized servers and makes the system more resilient and scalable.
- Local storage and management of keys: Edge nodes have been used for local storage and management of keys, increasing security and reducing the risk of key compromise.

ii. Pera Wallet integration for validation:

- Wallet integration: EPoS can use Pera Wallet to interact with the BBVV system. This includes writing vote counts or performing administrative tasks.
- Transaction Signing: Pera Wallet allows users to securely sign blockchain transactions, ensuring that vote counts are written by legitimate EPoS.
- Verification of transactions: Election officials can use Pera Wallet to verify transactions on the Algorand blockchain to ensure the integrity of the vote count.

iii. This ensures security, transparency, and trust:

- End-to-end verification: from writing vote count to vote count tallying at the national level, every step is verifiable. EPoS can verify their written vote count on the blockchain, and election officials can check the entire process.
- Immutable record: The blockchain provides an immutable record of all vote counts, preventing tampering and ensuring the integrity of the vote counting and validation process.
- Real-time verification: The use of edge computing enables real-time verification of the vote counting and validation process, increasing transparency and trust.

iv. User interface and accessibility:

- Accessible interface for writing vote counts: a user-friendly interface is critical. EPoS should be able to write their vote count easily, and Pera Wallet integration is intuitive and straightforward.
- Feedback and confirmations: EPoS receive instant feedback and confirmation once their vote count has been recorded on the blockchain, enhancing user experience and trust.

The diagram in Figure 7 shows a simplified overview of BBVV, highlighting the role of Pera Wallet in validating vote counts. Edge computing processes the vote counts and manages the keys, increasing the security and efficiency of the system by decentralizing these functions. Pera Wallet facilitates EPoS interaction with the system and allows EPoS to securely verify and validate their vote counts via the blockchain. An additional focus is on Pera Wallet's special role in validating vote counts to ensure the integrity of the vote writing process. The system is designed to be secure and trustworthy, as demonstrated by the emphasis on security and trust, and it provides an easy-to-use interface to improve accessibility. Finally, the voting system ensures a secure and transparent vote counting and validation experience for all users, with Pera Wallet playing a significant role in validating each vote on the blockchain.

In the BBVV (blockchain-based voting counting and validation) system, the blockchain architecture uses different roles for nodes, transactions, blocks, and the ledger to ensure the integrity and security of elections. The nodes are divided into non-relay and relay nodes. The non-relay nodes are located in the polling stations and are primarily used for local vote count recording at the edge of the network and then transmitting these data to the relay nodes. The relay nodes, which include the archive nodes, maintain a comprehensive ledger that contains all transaction records and ensures the integrity of the blockchain. Transactions are defined in this system as secured actions to record vote counts, which are verified by digital signatures enabled by the integration of the Pera Wallet. Each block encapsulates a batch of these verified transactions, which are cryptographically sealed and sequentially linked to ensure the integrity of the data. The ledger, which is maintained on the Algorand blockchain, serves as an immutable and tamper-proof record of all transactions and promotes a transparent and secure reconciliation process. The

network uses both non-relay and relay nodes to optimize the use of resources at polling stations and ensure that all votes are accurately reflected in the national count, improving both the security and auditability of the election process.

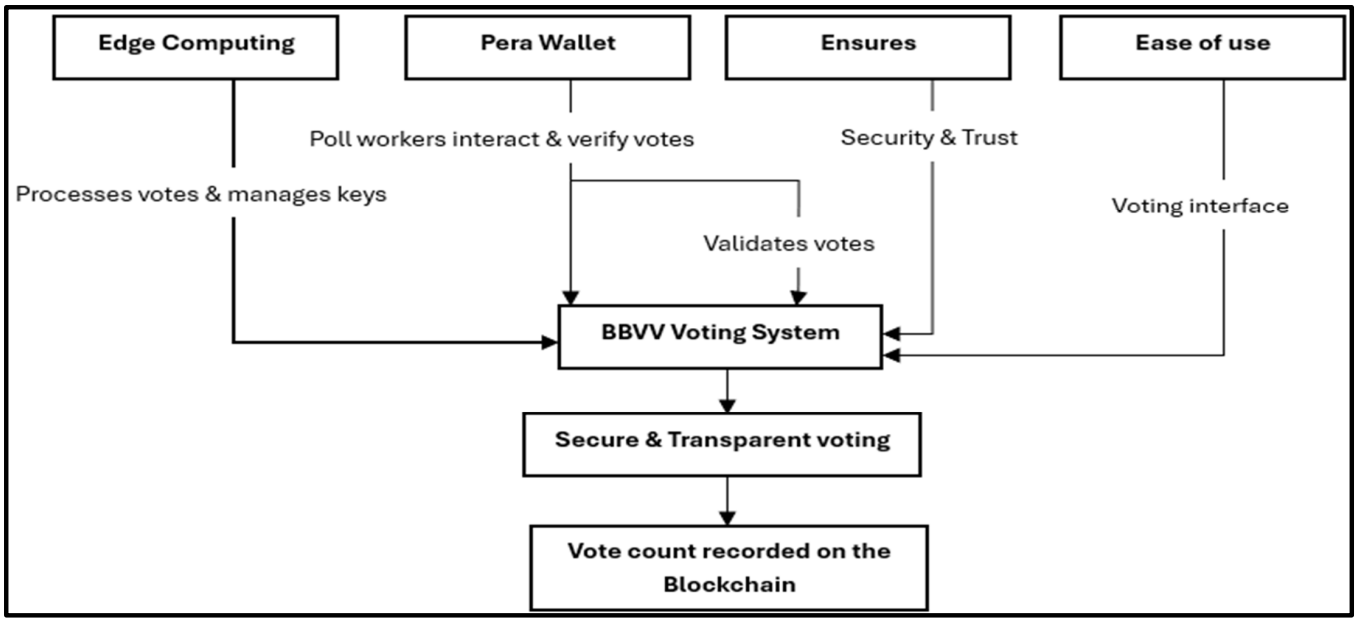

Figure 7. Overview of BBVV with Pera Wallet validation.

9. Results

The data analysis and visualization presented provide valuable insights into various aspects of a blockchain-based voting system and offer a comprehensive understanding of data trends and results. The data include information on consensus reached, transaction performance, traffic patterns, and election-related statistics. These insights can help decision-makers, network operators, and stakeholders make informed decisions, optimize system performance, and evaluate the efficiency of the election process. In the evaluation carried out, a random number of polling agents were introduced to input the same vote count, symbolizing the small 'n' in the formula 100 n/N = 2/3 majority (67% and above), keeping 'N' constant.

i. Consensus reached and not reached

The graph in Figure 8 shows a bar chart. The red bar represents No (consensus not reached), and the blue bar represents Yes (consensus reached). The above analysis shows that a larger percentage of the vote count did not reach a consensus.

Labeling of the X-axis ("consensus"): This label indicates the categories plotted on the X-axis, i.e., the different types of consensuses.

Y-axis label ("Number"): The label on the y-axis indicates that the number of occurrences is measured.

Interpretation:

- This plot is a bar chart that shows the distribution of different consensus outcomes.
- It helps visualize how many times each type of consensus outcome (e.g., "Yes" or "No") has been reached in the data.
- By observing the height of the bars, you can quickly determine the frequency or count of each consensus outcome.
- The colors differentiate between different types of consensus outcomes. In this case, red and blue bars represent different consensus results, such as "Consensus Reached—Yes" and "Consensus Reached—No."

Given the above interpretation and the bar chart, it shows that only about 5% of the officials arrived at a consensus level the remaining 95% did not reach a consensus.

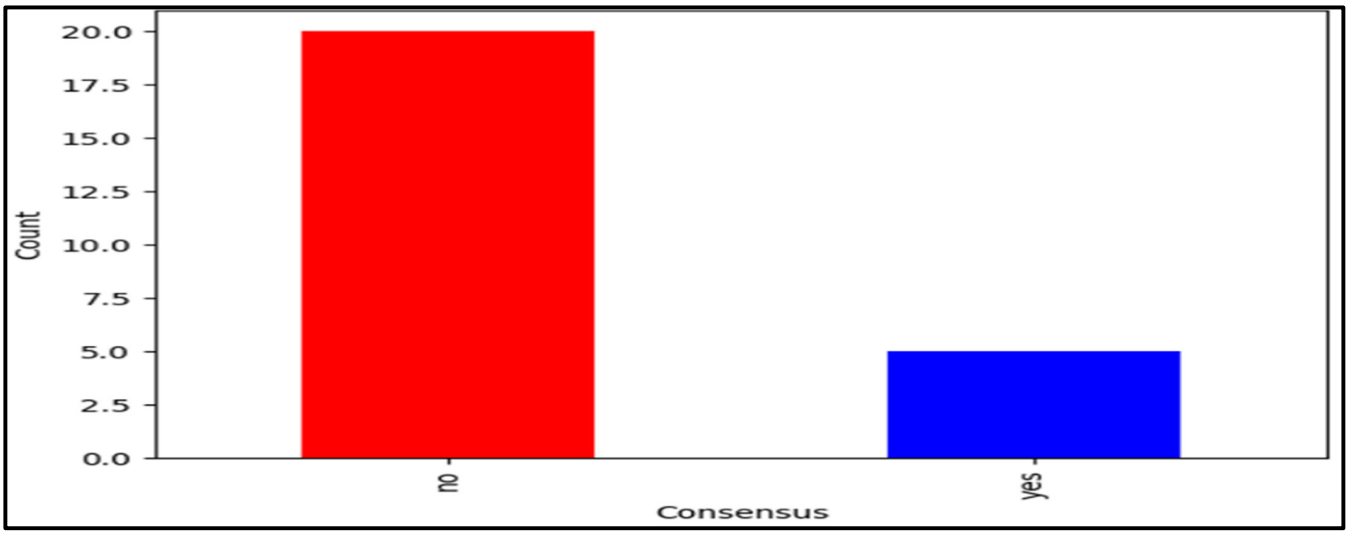

Figure 8. Consensus reached.

ii. Actual data compared to the aggregation of consensus reached

In Figure 9, the bar on the left, labeled 'Total Vote Count,' represents the total vote count for all data, irrespective of whether 'Consensus Reached' is 'Yes' or 'Not'. The bar on the right, labeled 'Total Vote Count (Consensus Reached Yes), represents the total vote count, considering only the rows where 'Consensus Reached' is 'Yes.' The plot allows you to visually compare these two categories of vote counts. It is a straightforward way to see how the total vote count changes when 'Consensus Reached' is 'Yes' and when it is not. The color-coding (blue and green) helps distinguish between the two categories.

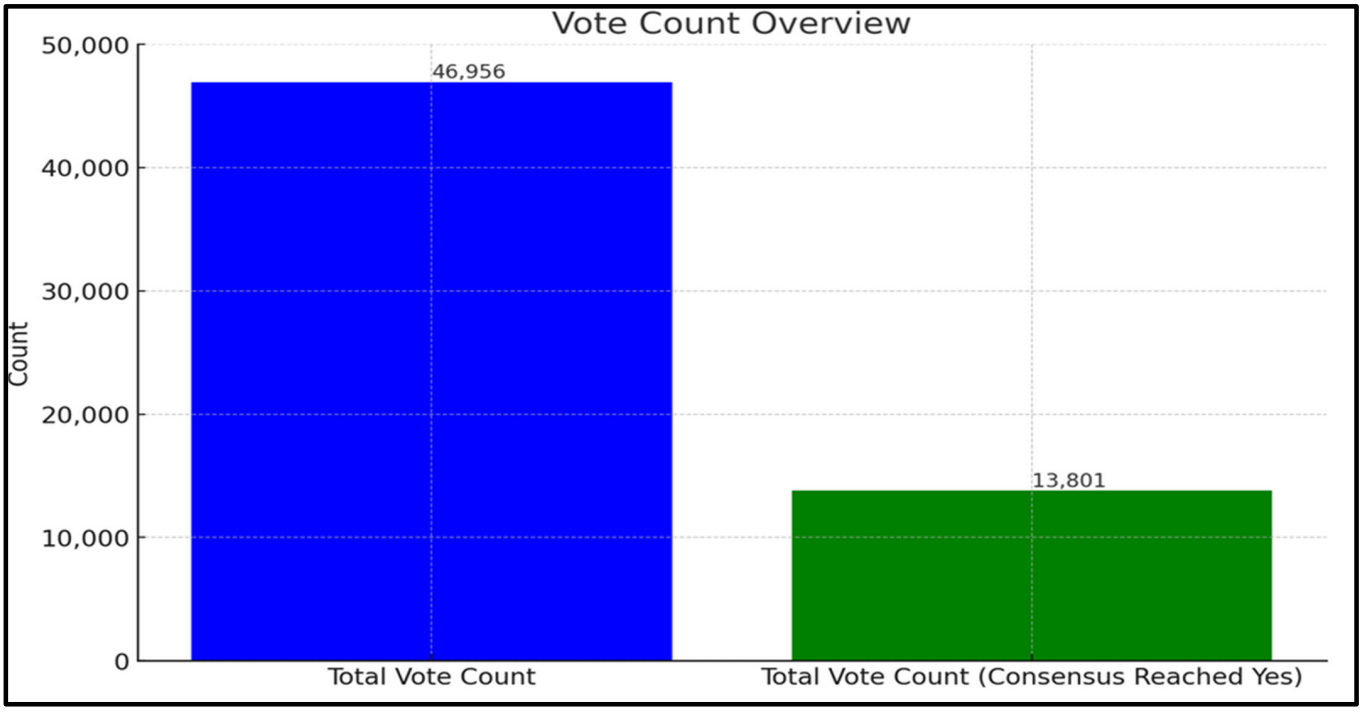

Figure 9. Comparative analysis of actual vote count and consensus vote count.

This information is useful for understanding the impact of 'Consensus Reached' on the total vote count. Figure 5 shows that the total number of votes counted is greater than the aggregate consensus vote count. Only about 10% of the vote count submitted will be taken into consideration as those were the vote counts that reached consensus.

iii. Officials (EPoS) are in agreement compared with total officials at polling stations.

The officials in agreement (n) vs. total number of officials (N) were also visualized as indicated in Figure 10, where:

Y-axis (count): The y-axis represents the count, which measures the number of officials in agreement (n) and the total number of officials (N).

X-axis label (S/N): The label on the y-axis specifies that the count is being measured.

Legend: The legend in the plot explains the color code for the bars. The green bars represent "Officials in Agreement," while the blue bars represent "Total Number of Officials (N)."

Interpretation:

- This plot provides a visual comparison between the count of officials who agree and the total number of officials.
- By observing the height of the bars, it can be determined whether most officials agree or if there is a significant disagreement on the vote count captured at the polling station.
- The plot is useful for decision-makers or officials to quickly grasp the level of consensus or disagreement among a group of officials.

If the green bars (officials in agreement) are close in height to the blue bars (total number of officials), it indicates a high level of agreement. Conversely, if the green bars are significantly shorter, it suggests a lower level of agreement.

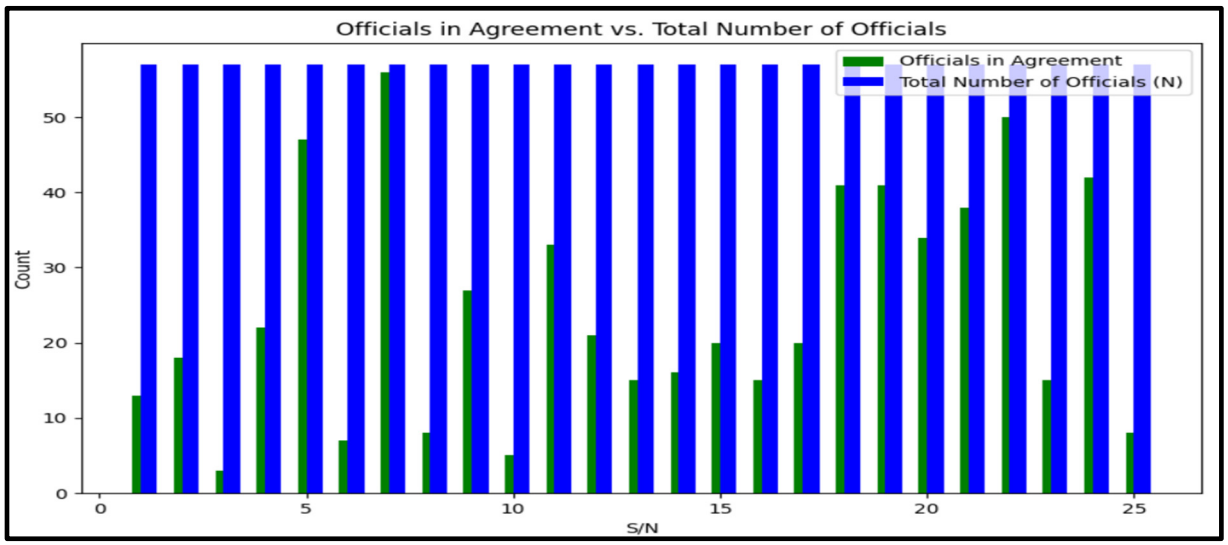

Figure 10. Officials in agreement vs. the total number of officials.

In summary, this plot is a visual tool for officials to assess and understand the degree of consensus or agreement among a group of officials in a clear and concise manner. The above plot shows a significant level of disagreement between the officials. This means little level of consensus was reached; however, this was caused by the randomized data that were introduced in the actual data.

iv. Transaction Performance Metric Analysis

The graph in Figure 11 visualizes the transaction confirmation time over different confirmed rounds.

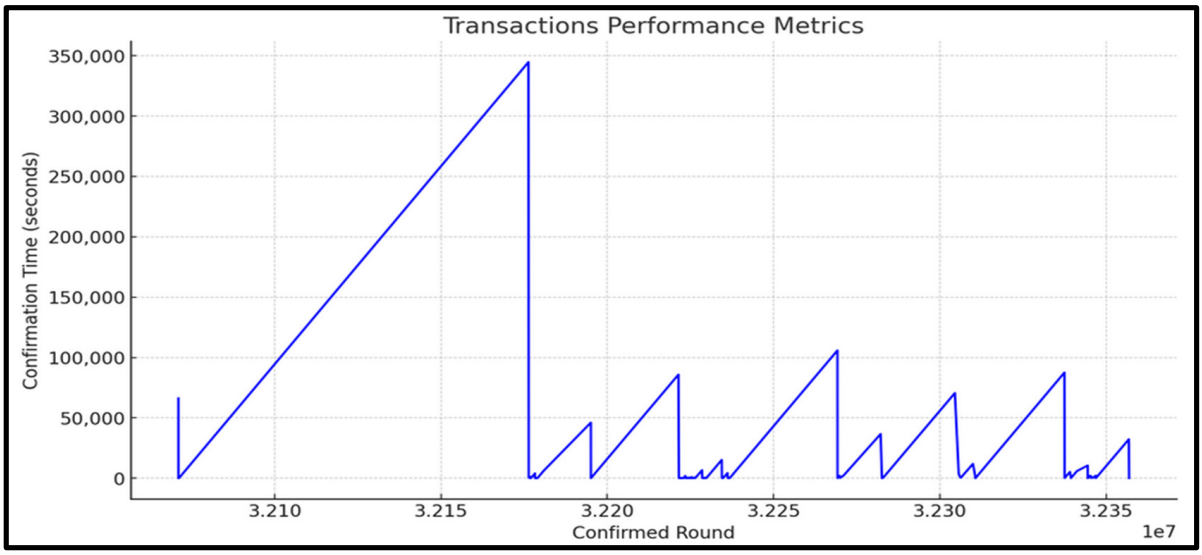

Figure 11. Transaction metrics.

Interpretation:

The plot allows you to observe how the confirmation time for transactions varies over different rounds. You can look for patterns, spikes, or fluctuations in confirmation times. Sudden peaks may indicate delays in transaction processing, while valleys represent quicker confirmations. There was a delay in the transaction at point 35,000 s, which was confirmed in 3.20 confirmed rounds.

v. Transaction Throughput Over Rounds Analysis

The illustrated plot in Figure 12 visualizes transaction throughput over different confirmed rounds.

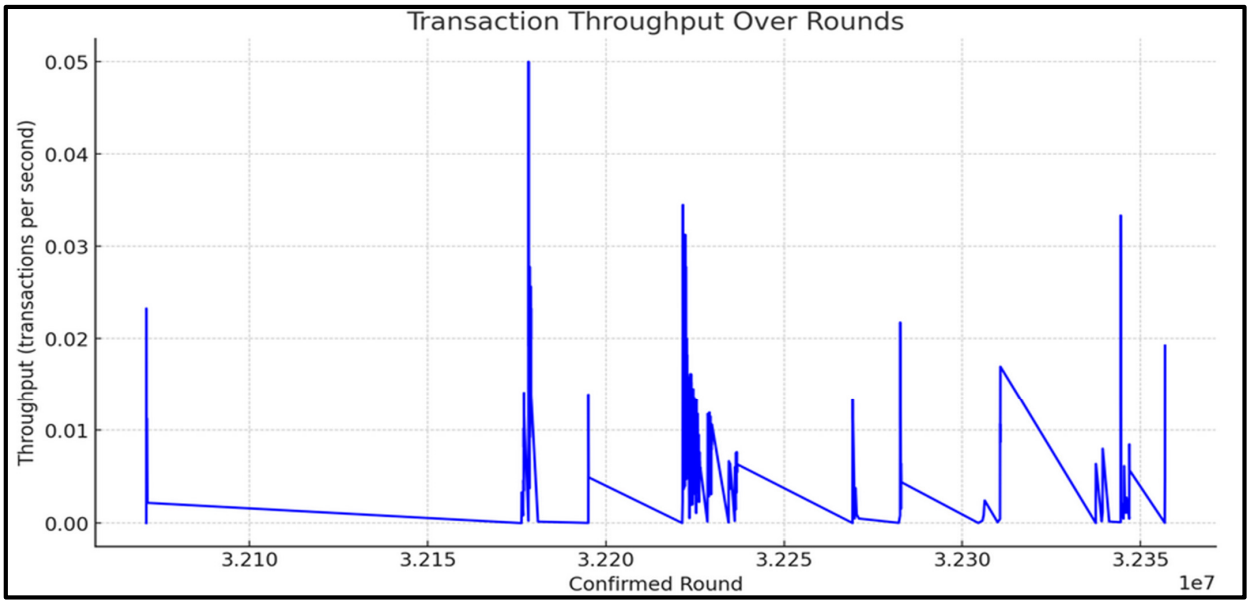

Figure 12. Transaction throughput.

Interpretation:

The graph in Figure 12 helps to understand the capacity of the system to process transactions. It shows how many transactions were confirmed per second during different rounds. Higher peak values indicate better throughput, while lower values may suggest

congestion or reduced processing capacity. The confirmed rounds at 3.20, 3.222, and 3.23, respectively, had a higher peak value, indicating better throughput.

vi. Saturation Analysis

The graph in Figure 13 shows a line graph where each point on the line corresponds to a specific timestamp (time) and its associated transaction fee. The points are marked with circular markers ("o") connected by lines ("-"). This visualization method allows you to track changes in transaction fees over time.

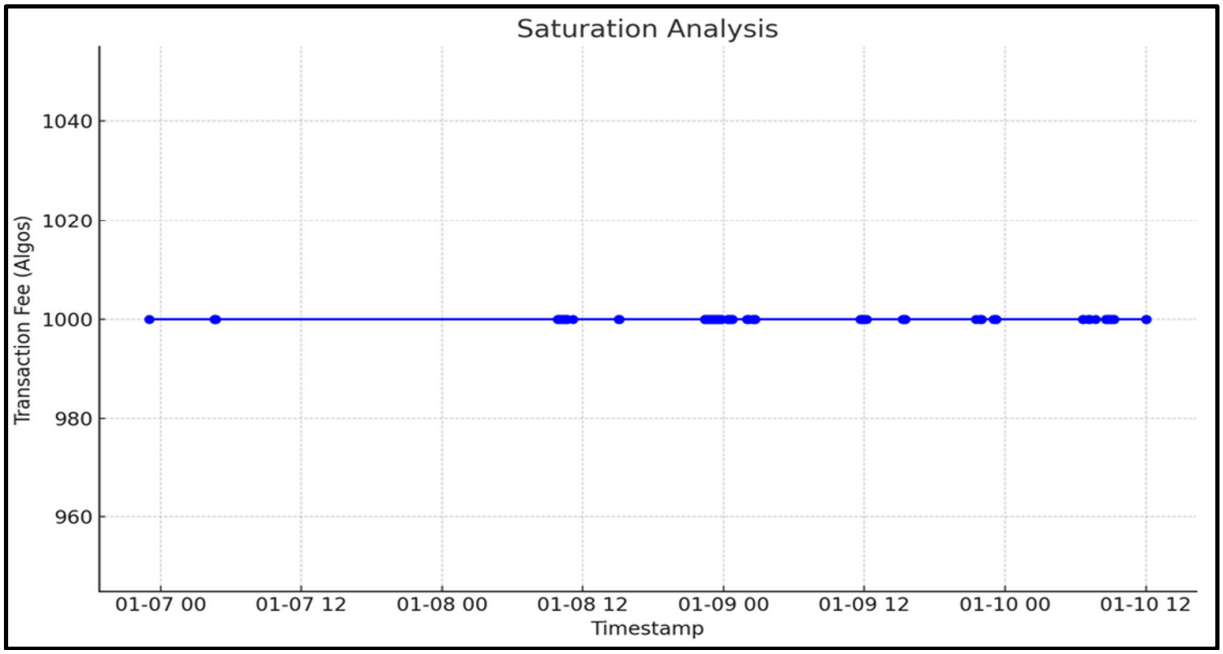

Figure 13. Saturation analysis.

X-axis (timestamp): The x-axis represents time in the form of timestamps. It shows when the transactions were confirmed. This axis allows you to track the progression of time.

Y-axis (transaction fee in Algos): The y-axis represents the transaction fee in Algos. It quantifies the cost associated with each transaction. Transaction fees are typically used to incentivize network nodes to process and confirm transactions.

Interpretation:

- The plot provides an overview of how transaction fees change over time. It can help you identify trends and patterns in transaction fees on the blockchain network.
- Rising transaction fees might indicate increased demand for network resources, potentially suggesting network congestion.
- Falling transaction fees may indicate reduced demand or improved network efficiency.
- Sudden spikes in transaction fees could be linked to particular events, such as a surge in network usage or the introduction of new applications or assets on the blockchain.
- A consistent flat line could suggest stability in the network with relatively constant transaction fees.

The saturation analysis plot in Figure 14 shows a consistent flat transaction fee across different timestamps and transactions. This suggests the stability of the BBVV artifact on the Algorand network. Understanding how transaction fees change over time is essential for blockchain users, developers, and network operators to make informed decisions and adapt to changing conditions on the network. The visualization can also be useful for forecasting and optimizing transaction costs.

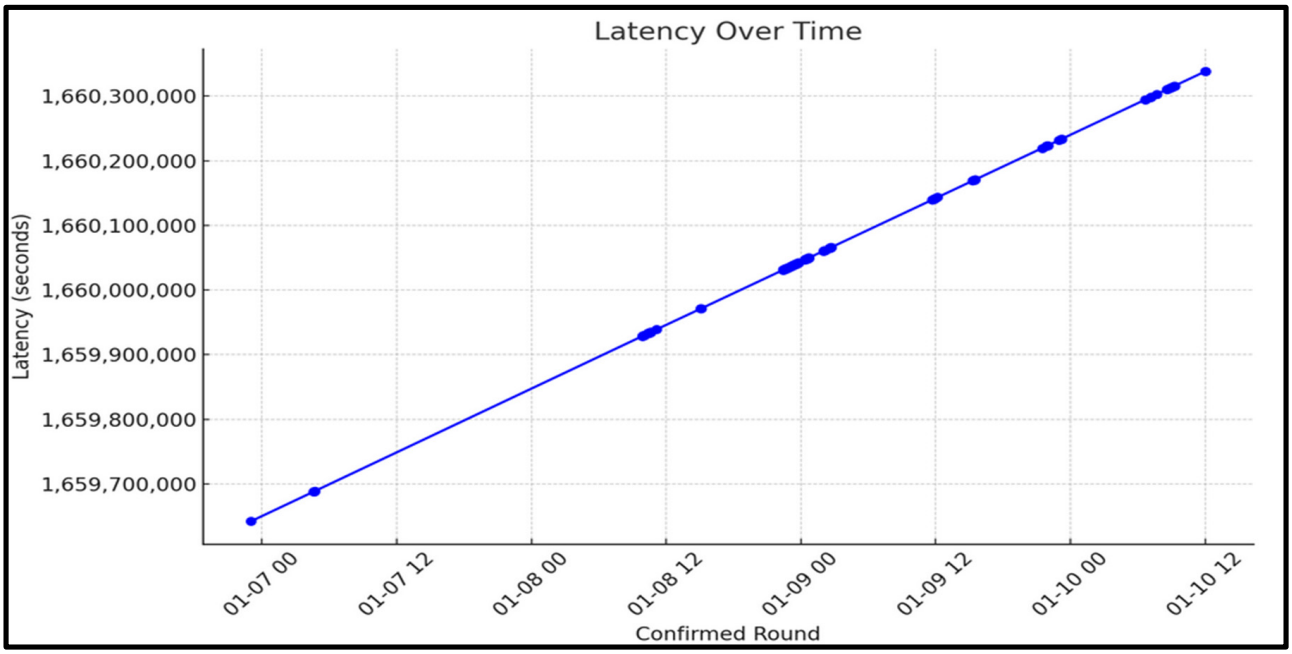

Figure 14. Latency analysis.

vii. Latency Analysis

The graph provided in Figure 13 helps in understanding the latency in the confirmation of transactions over a period. The plot is a line graph, with each data point represented as a circular marker ("o") connected by lines ("-"). This visualization method allows you to track changes in latency over time.

Interpretation:

- The plot provides insights into the latency experienced by transactions on the blockchain network.
- An upward trend in latency suggests that transaction confirmation times are increasing, which might indicate network congestion or increased demand.
- A downward trend in latency indicates decreasing confirmation times, potentially due to network optimization or reduced demand.
- Spikes in latency might be linked to specific events or congestion periods when transactions are taking longer to confirm.
- Consistent, stable latency indicates that the network is maintaining a relatively constant confirmation time.
- Fluctuations in latency can reveal patterns and help users and developers understand the performance of the blockchain network at different times.

This plot is valuable for assessing the efficiency and responsiveness of our artifact (BBVV) on the Algorand blockchain network in processing transactions. Monitoring and analyzing latency trends can assist in making informed decisions about when to submit transactions to achieve desired confirmation times and to identify periods of network stress or congestion. The plot indicates an upward trend in latency, which suggests that confirmation times are increasing, which might indicate network congestion or increased demand.

The majority of block confirmations occur within a relatively short period of time. In particular, the median time interval between block confirmations is 90 s, indicating that the blockchain processes transactions efficiently under normal operating conditions. Furthermore, the 75th percentile of time intervals is approximately 185 s, meaning that 75% of blocks are confirmed within approximately 3 min of the previous block. These intervals reflect a high level of efficiency in the blockchain network, as blocks are confirmed consistently and without significant delays for the majority of transactions. This efficiency

indicates a well-functioning system that is able to process transactions in a timely manner, which is significant for user confidence and the smooth operation of blockchain applications.

viii. Traffic Analysis

This type of analysis is useful for understanding transaction behavior and identifying trends or anomalies in the dataset over time. It can be helpful for monitoring network activity, identifying peak usage times, or analyzing the impact of specific events on transaction traffic. Figure 15 counts the number of transactions in each round and plots the results as a line chart.

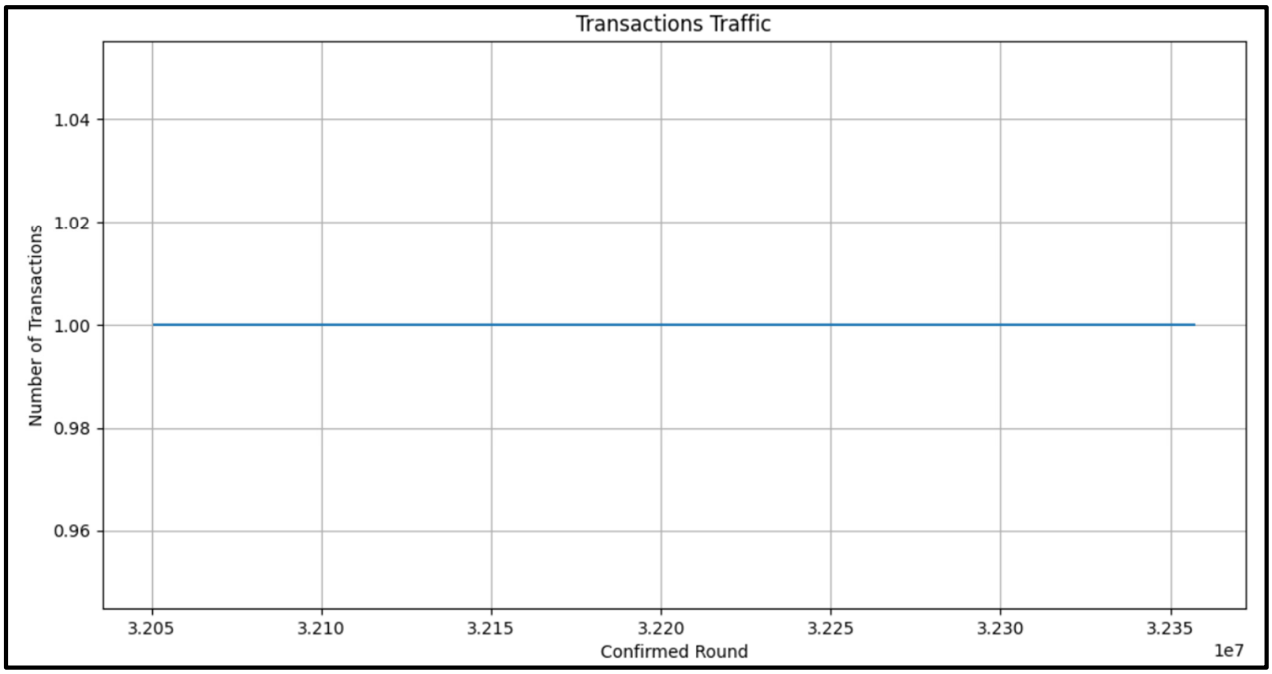

Figure 15. Traffic analysis.

Here is an interpretation of the plot:

X-axis (confirmed round): This represents the "confirmed round" of the transactions, which appears to be a measure of time or sequence of events. As the confirmed round increases, it indicates the progression of time or the order in which transactions were confirmed.

Y-axis (number of transactions): This axis represents the number of transactions that were confirmed in each round. It measures the intensity of transaction activity during each round.

Interpretation:

The plot in Figure 11 shows how the number of transactions varies over time (confirmed rounds). You can see patterns, spikes, or fluctuations in transaction activity. For example, if there are sudden peaks in the graph, it suggests moments of high transaction activity, while flat regions indicate periods with lower transaction volumes. The above graph shows flat regions, which indicate prolonged moments of low transactions.

Grid lines: The grid lines help in reading the values more accurately and are present in both the X and Y axes.

The evaluation of the BBVV artifact has been carefully conducted, including a thorough evaluation in terms of performance, saturation, traffic analysis, and transaction throughput. The front-end of this system is based on a client-server architecture model that integrates Next.js with a smart contract developed by Algorand with PyTeal at the back end. As security is critical, the system includes Pera Wallet for robust authentication and advanced transaction signatures. This configuration allows users to interact with the front-end via browsers, exchange data with the smart contract, and utilize Pera Wallet for

superior security for both authentication and transactions. The comprehensive evaluation of the system, which focuses on performance, ability to handle high traffic and peak loads (saturation), and traffic analysis to optimize data flow and transaction throughput efficiency, ensures that the BBVV artifact not only meets its design and functional criteria but also adheres to the highest standards of reliability and trustworthiness that are essential for modern voting systems.

10. Discussion

A. Practical Implications of Findings

i. Trust and Governance:

The observed divergence in the counting of votes and the inability of a significant proportion to reach consensus raises concerns about the governance of the network. There is a potential risk of dishonest activities, such as vote rigging. This points to the need for tighter monitoring and possibly improved security measures to ensure the integrity of the voting process.

ii. Network efficiency:

Insights into transaction performance, particularly observed delays and spikes, suggest that the network may face challenges in handling large transaction volumes, especially at peak times. This requires technology upgrades or optimizations to improve the network's processing capacity and reduce bottlenecks.

iii. Stability and predictability:

While the constant trend in transaction fees indicates stability, it also serves as a reminder for network administrators to remain proactive. Ensuring predictable transaction costs is critical to user satisfaction, and any change, no matter how small could disrupt this stability. This means that continuous monitoring and a willingness to implement adaptive measures are required.

iv. Latency and scalability:

Increasing network latency is a clear sign of potential congestion problems. This could lead to lower user confidence and transaction efficiency. To counter this, it may be necessary to explore advanced technological solutions, such as sharding or Layer 2 solutions, to ensure that the network remains scalable and responsive.

v. Strategic planning:

Insights from traffic analysis, such as understanding periods of low activity and peak periods, can support strategic decisions. For example, network maintenance or upgrades can be scheduled during periods of low activity to minimize inconvenience to users. In addition, resource allocation at times of high traffic can ensure network resilience and efficiency.

vi. Transparency and credibility:

Detailed analysis of election-related data underscores the importance of transparency in the electoral process. The availability of such comprehensive data can enhance the confidence of network participants and observers. This suggests that maintaining transparency and providing detailed data should be a priority for any blockchain-based election system.

In summary, this data analysis and visualization provides a comprehensive overview of blockchain-based election data. It sheds light on consensus results, transaction performance, traffic patterns, and election statistics. Overall, the Algorand blockchain is well suited for this research as the transaction fee is only 0.001 algo and remains the same regardless of network congestion. Furthermore, a more accurate consensus has been achieved as the election results submitted by the different polling stations are publicly available. These insights are invaluable for optimizing system performance, understanding transaction dynamics, and improving the integrity of the electoral process. Stakeholders, officials,

and network operators can use these insights to make data-driven decisions and continuously improve the blockchain-based election system. It highlights the importance of data analytics in ensuring transparency, efficiency, and trust in the electoral process within a blockchain network.

B. Design Science Research (DSR) in Action

As described, the development of the BBVV artifact follows the Design Science Research (DSR) approach, a problem-solving process that involves the creation and evaluation of innovative artifacts. The DSR approach typically involves identifying a problem, developing an artifact as a solution, and evaluating the effectiveness of the artifact. Here, you can see how the development of the BBVV artifact is in line with the DSR approach:

i. Problem Identification and Motivation (Relevance Cycle)

The first phase of the DSR approach is about understanding the problem area. For the BBVV artifact, this was achieved by examining the perceptions and expectations of election stakeholders in African countries. The thematic analysis revealed challenges such as poor network connections, inadequate staff training, and corruption, which justified the need for a new system.

ii. Objectives of a Solution (Rigor Cycle)

This study then defined the objectives for a solution, which included ensuring accuracy, speed, efficiency, transparency, and security in the voting process. The system also needed to be resilient to network issues, litigation, and corruption while encouraging active stakeholder participation and compliance with electoral rules.

iii. Design and Development (Design Cycle)

In the design and development phase, the BBVV artifact was conceived with a clear system architecture. The BBVV artifact was designed using a client-server model, using Next.js for the client-side interface, PyTeal for the server-side smart contract logic on Algorand, and Pera Wallet for secure authentication and transaction signatures. In this phase, primary modules and an operational workflow were created detailing user interactions with the system via web browsers, data requests, and transfers.

iv. Artifact Description

The technological stack used and the functional overview of the system were described in detail, emphasizing special features such as immutable data storage, synchronous data reflection, improved user identity verification, secure data transmission, and comprehensive audit functions. This description meets the DSR's requirement for a clear and detailed presentation of artifacts.

v. Demonstration and Evaluation (Design Cycle)

While the demonstration and experimental evaluation of the BBVV protocol were set to be conducted in this paper, the design and development phase laid the groundwork for these future steps. The system's architecture and operational workflow were established to demonstrate the artifact's capabilities in a controlled environment.

vi. Communication (Relevance, Rigor, and Design Cycles)

The final phase of the Design Science Research (DSR) approach is the communication of the problem, the artifact, and its utility to an academic and practitioner audience. This is carried out by disseminating the knowledge gained, the methods used, and the implications of the artifact's design. The conclusion of this study and subsequent publications tie back to the original objectives and challenges and summarize how the design and development of the BBVV artifact addresses the identified problems and contributes to the field of blockchain-based voting systems.

The research underlying the BBVV artifact has been successfully communicated in other academic publications and conference presentations, demonstrating the relevance

and rigor of the work undertaken. These efforts ensure that the solution is not only theoretically sound but also practically relevant, with a clear path to empirical testing and validation in the real world. The publications serve as a bridge to industry practitioners, providing a comprehensive overview of the state of the art in blockchain-based voting systems and emphasizing the practical implications of the research. They highlight the potential impact on future electoral processes and the improvement of democratic practices through technology, demonstrating the contribution of the BBVV artifact to both academic discourse and practical application.

C. The Byzantine Generals Problem in Action

The application of the theory of the Byzantine Generals Problem (BGP) in a blockchain-based voting system serves as a pertinent illustration of the consensus challenges in distributed networks. This is illustrated by the BBVV protocol that allows participants in a polling station to collectively agree on the final vote, which is documented in the blockchain, thus facilitating the collation of votes at the national level.

i. Purpose of the Test

The tests described aim to verify the ability of the blockchain system to reach a consensus on the vote count, which is a practical application of BGP theory. The scenarios tested demonstrate the resilience of the system to dishonest reporting, as consensus requires a supermajority to ensure that the final vote count is accurate and accepted by the majority of election officials, thus reflecting the true will of the voters. To recap, applying the theory of the Byzantine Generals Problem to blockchain-based voting systems provides a framework for understanding how distributed consensus can be achieved in an environment where participants do not necessarily trust each other. The practical implementation of this theory through blockchain technology ensures that the integrity of the voting process is maintained and that the final vote count accurately and verifiably reflects the collective decision of the voters.

ii. Application of the Theory

In the context of blockchain-based voting, the "generals" are analogous to the EPoS at each polling station, and the "city" is the correct vote count that must be agreed upon. The blockchain serves as a communication channel through which the generals send their plans (vote count) to each other. The smart contract on the blockchain is designed to record the vote count only when a consensus of 67% is reached, similar to how the generals must agree on a common plan of action.

iii. The Byzantine Generals Problem Theory Applied

(a) Trust and Consensus

The BGP theory emphasizes the problem of trust between parties who must agree on a single value (in this case, the vote count). The role of the blockchain is to create a trustless environment in which consensus can be reached without the parties having to trust each other, as the integrity of the vote count is guaranteed by the immutable ledger of the blockchain.

(b) Tolerance of Malicious Actors

The theory's requirement that consensus can be reached even if some participants are malicious (up to a third) is reflected in the voting system's requirement of 67% consensus. This ensures that, even if some electoral officials are dishonest, they cannot influence the total number of votes as long as the majority (more than two-thirds) are honest.

(c) Cryptography and Digital Signatures

The use of digital signatures in BGP theory is reflected in the blockchain voting system through the use of Pera Wallet and smart contracts. These digital signatures ensure that once a vote count has been entered, it cannot be altered, and the identity of the poll worker entering the data can be verified.

D. Comparative Analysis of the Findings of the Literature

The results of this study on the BBVV artifact on the Algorand network, particularly in relation to voting inconsistency and transaction performance, can be critically analyzed in light of the existing literature on blockchain technology and voting systems. The concerns about possible dishonest manipulation of vote counting identified in the research are directly addressed in the literature [13,14]. These studies emphasize the importance of voter anonymity and security through zero-knowledge proofs, decentralization and Shamir's secret sharing and suggest potential mitigation strategies for the risks highlighted in the Algorand network.

Furthermore, the observed transaction delays and bottlenecks in the Algorand network coincide with the scalability challenges highlighted by [19,21]. They emphasize scalable solutions such as EtherVote and SBvote that could identify strategies to improve the processing capacity issues identified in this study. The analysis of transaction fees and network latency aligns with concerns raised about the efficiency and integrity of blockchain-based systems, as noted by [22,24]. These studies suggest that maintaining a stable and efficient network is essential for user trust, which is also emphasized by research on the Algorand network.

Regarding the integrity and transparency of elections, the need to analyze election data in detail is supported by the emphasis on transparent and tamper-proof systems in the literature. The SHARVOT protocol by [14] and the Ethereum-based prototype by [23] emphasize the importance of such systems that can increase trust and credibility in blockchain-based elections and address some of the concerns raised in this study.

This study does not explicitly mention quantum security, but this emerging threat is addressed by [16], suggesting that the integration of quantum-resistant functions into the Algorand network may be an important future consideration. Furthermore, the delicate balance between transparency in vote counting and the protection of voter privacy addressed in this study is a much-discussed challenge in the literature. This challenge is to maintain transparency and fairness while ensuring security, as [14] emphasizes.

Lastly, the results of this study on the BBVV artifact in the Algorand network are consistent with the broader challenges and solutions discussed in the literature. Emphasizing voter anonymity, scalability, transparency, and security in blockchain-based voting systems is critical to addressing these challenges. Ongoing research and technological advances in this area provide valuable insights into potential strategies for improving the performance and reliability of blockchain networks such as Algorand in electoral contexts.

11. Conclusions

Through rigorous experiments with the BBVV protocol to recap, the integration of blockchain technology into the electoral process, as demonstrated in this study, provides a robust solution to the challenges of consensus building and maintaining the integrity of the vote count. Applying the theory of the Byzantine Generals Problem to blockchain-based voting systems ensures a trustworthy environment in which consensus can be reached even in the presence of potentially dishonest participants. The data analysis and visualization performed on the Algorand blockchain illustrate the effectiveness of this approach, showing clear consensus results, consistent transaction performance, and recognizable traffic patterns and voting statistics. The low and stable transaction fee on the Algorand platform emphasizes its suitability for processing election data, even under changing network conditions. The transparency created by making election results from different polling stations publicly available on the blockchain has led to a more accurate consensus, which is critical for the legitimacy of the electoral process. These findings are not only theoretical in nature but also provide practical insights that can be used by stakeholders, election authorities, and network operators to improve the performance of the system and increase user trust. Ultimately, this study highlights the central role of data analytics in enhancing transparency, efficiency, and trust in blockchain-based voting systems and marks a significant step forward in the modernization of democratic processes.

The first major contribution is the implementation of a Layer 1 smart contract on the Algorand platform, which improves the efficiency and scalability of the system through fast, secure processing and aggregation of votes. In addition, the integration of the Byzantine General Problem as a theoretical framework strengthens the ability of the BBVV protocol to reach consensus under difficult conditions and maintain the integrity of the vote. This study also demonstrates how blockchain technology supports the integrity of elections through decentralization, immutability, and transparency and protects elections from fraud and manipulation while promoting voter privacy and security through cryptographic techniques.

12. Limitation and Future Work

This research specifically addresses the vote counting and validation phases of elections, where it seeks to improve accuracy and trustworthiness using blockchain technology. However, several limitations complicate its application. Firstly, there are scalability issues. Blockchain may not be able to efficiently handle the high demands of large national elections due to inherent processing limitations. Secondly, the integration of blockchain into existing electoral systems poses significant technical and logistical challenges that require extensive adaptations to the new processes. In addition, the different legal and regulatory frameworks in different countries create a complex environment for the introduction of a widely accepted blockchain-based voting system. These limitations highlight the complexity of implementing blockchain in the context of vote counting and validation alone and emphasize the need for comprehensive solutions that address these multi-layered challenges.

Future work will look at refining the BBVV protocol and explore the potential for scaling beyond the current parameters of polling agents and stations. As the integration of technology and stakeholder engagement has proven critical, further research will focus on improving the user interface of the Election Collation System and expanding its compatibility with emerging blockchain technologies. It will also focus on exploring more advanced authentication measures, building on the foundation created by Pera Wallet, to ensure greater security and trust in the system. The overall goal is to strengthen the legitimacy and transparency of the system while optimizing its operational efficiency.

Author Contributions: This work was carried out by P.M. under the supervision of B.K. as part of the Ph.D. in Informatics thesis work. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The supporting data analysis and Algorand PyTeal Smart Contract Code for this research can be found on GitHub with the following links: https://github.com/gconnect/election-data-analysis (accessed on 18 March 2024), https://algo-election.vercel.app (accessed on 18 March 2024), https://github.dev/gconnect/Algo-Election/blob/main/backend/app.py (accessed on 18 March 2024), https://github.dev/gconnect/Algo-Election/blob/main/utils/Interact.ts (accessed on 18 March 2024).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Akwei, B.; Machar, B.A.; Mnyandu, P. 'Debris' of Coups D'état: Electoral Democracy, Election Violence, Political Vigilantism, and Elections Securitizations in Africa. *South Asian Res. J. Humanit. Soc. Sci.* **2023**, *5*, 65–75. [CrossRef]
2. Achieng, M.; Ruhode, E. The Adoption and Challenges of Electronic Voting Technologies within the South African Context. *Int. J. Manag. Inf. Technol.* **2013**, *5*, 1–12. [CrossRef]
3. Kazi, S.; Md, M.; Kumar, P.; Anik, I. Blockchain Based Secured E-voting by Using the Assistance of Smart Contract. *arXiv* **2019**, arXiv:1910.13635.
4. Benabdallah, A.; Audras, A.; Coudert, L.; El Madhoun, N.; Badra, M. Analysis of Blockchain Solutions for E-Voting: A Systematic Literature Review. *IEEE Access* **2022**, *10*, 70746–70759. [CrossRef]
5. Damle, S.; Gujar, S.; Moti, M.H. FASTEN: Fair and secure distributed voting using smart contracts. In Proceedings of the 2021 IEEE International Conference on Blockchain and Cryptocurrency (ICBC), Sydney, Australia, 3–6 May 2021. [CrossRef]
6. Singh, J.; Rastogi, U.; Goel, Y.; Gupta, B. Blockchain-based decentralized voting system security Perspective: Safe and secure for digital voting system. *J. Pharm. Negat. Results* **2023**, *13*, 2022.

7. Namasudra, S.; Deka, G.C.; Johri, P.; Hosseinpour, M.; Gandomi, A.H. The Revolution of Blockchain: State-of-the-Art and Research Challenges. *Arch. Comput. Methods Eng.* **2021**, *28*, 1497–1515. [CrossRef]
8. Jones, D.; Simons, B. Broken Ballots: Will Your Vote Count in the Electronic Age? 2012. Available online: https://ci.nii.ac.jp/ncid/BB10317321 (accessed on 22 August 2023).
9. Bashir, I. *Mastering Blockchain: A Deep Dive into Distributed Ledgers, Consensus Protocols, Smart Contracts, DApps, Cryptocur-Rencies, Ethereum, and More*, 3rd ed.; Packt Publishing: Birmingham, UK, 2020.
10. Abraham, I.; Devadas, S.; Dolev, D.; Nayak, K.; Ren, L. Efficient Synchronous Byzantine Consensus. Available online: http://arxiv.org/abs/1704.02397 (accessed on 6 October 2023).
11. Kuo, P.C.; Chung, H.; Chao, T.W.; Cheng, C.M. Fair byzantine agreements for blockchains. *IEEE Access* **2020**, *8*, 70746–70761. [CrossRef]
12. Chang, Y.-X.; Wang, Q.; Li, Q.-L.; Ma, Y. Performance and Reliability Analysis for Practical Byzantine Fault Tolerance with Repairable Voting Nodes. *arXiv* **2023**, arXiv:2306.10960. [CrossRef]
13. Onur, C.; Yurdakul, A. ElectAnon: A Blockchain-based, Anonymous, Robust, and Scalable Ranked-choice Voting Protocol. *Distrib. Ledger Technol. Res. Pract.* **2023**, *2*, 1–25. [CrossRef]
14. Bartolucci, S.; Bernat, P.; Joseph, D. SHARVOT: Secret SHARe-based VOTing on the blockchain. In Proceedings of the 1st International Workshop on Emerging Trends in Software Engineering for Blockchain, Gothenburg, Sweden, 27 May–3 June 2018. [CrossRef]
15. Wang, S.; Qu, X.; Hu, Q.; Wang, X.; Cheng, X. An Uncertainty- and Collusion-Proof Voting Consensus Mechanism in Blockchain. *IEEE/ACM Trans. Netw.* **2023**, *31*, 2376–2388. [CrossRef]
16. Mishra, S.; Thapliyal, K.; Rewanth, S.K.; Parakh, A.; Pathak, A. Anonymous voting scheme using quantum assisted blockchain. *arXiv* **2022**, arXiv:2206.03182.
17. Balilo, B., Jr. A Unique One-Time Password Table Sequence Pattern Authentication: Application to Bicol University Union of Federated Faculty Association, Inc. (BUUFFAI) eVoting System. *Int. J. Comput. Sci. Res.* **2017**, *1*, 1–10. [CrossRef]
18. Eldridge, M. A Trustworthy Electronic Voting System for Australian Federal Elections. *arXiv* **2018**, arXiv:1805.02202. [CrossRef]
19. Spanos, A.; Kantzavelou, I. A Blockchain-Based Electronic Voting System: EtherVote. *arXiv* **2023**, arXiv:2307.10726. [CrossRef]
20. Blessing, J.; Gomez, J.; Patiño, M.; Nguyen, T. Security Survey and Analysis of Vote-by-Mail Systems. May 2020. Available online: http://arxiv.org/abs/2005.08427 (accessed on 6 October 2023).
21. Stančíková, I.; Homoliak, I. SBvote: Scalable Self-Tallying Blockchain-Based Voting. In Proceedings of the 38th ACM/SIGAPP Symposium on Applied Computing, Tallinn, Estonia, 27–31 March 2023; pp. 203–211. [CrossRef]
22. Faour, N. Transparent Voting Platform Based on Permissioned Blockchain. *arXiv* **2018**, arXiv:1802.10134. Available online: http://arxiv.org/abs/1802.10134 (accessed on 11 November 2023).
23. Mukherjee, A.; Majumdar, S.; Kolya, A.K.; Nandi, S. A Privacy-Preserving Blockchain-based E-voting System. *arXiv* **2023**, arXiv:2307.08412. Available online: http://arxiv.org/abs/2307.08412 (accessed on 6 October 2023).
24. Bulut, R.; Kantarci, A.; Keskin, S.; Bahtiyar, S. Blockchain-Based Electronic Voting System for Elections in Turkey. In Proceedings of the 2019 4th International Conference on Computer Science and Engineering (UBMK), Samsun, Turkey, 11–15 September 2019; pp. 183–188. [CrossRef]
25. Huré-Maclaurin, L. Scalable System for Indexing and Providing Access to Verifiable Blockchain Transaction Data. Harvard University. 2020. Available online: https://search.proquest.com/openview/00a49c40c83cada42ec1a0d8db0a91e5/1?pq-origsite=gscholar&cbl=18750&diss=y (accessed on 24 September 2022).