Investigating performance constraints for blockchain based secure e-voting system

Investigating Performance Constraints for Blockchain Based Secure e-Voting System

Kashif Mehboob Khan[a], Junaid Arshad[b], Muhammad Mubashir Khan[c]

[a]*Department of Software Engineering, NED University of Engineering & Technology Karachi, Pakistan*
[b]*School of Computing and Engineering, University of West London, UK*
[c]*Department of Computer Science & IT, NED University of Engineering & Technology Karachi, Pakistan*

Abstract

Voting is one of the fundamental pillars of modern democracy. Continuous efforts have been made to strengthen the processes and methods involved to achieve verifiable, transparent voting systems. In recent years, blockchain has been increasingly used to address multi-dimensional challenges across widespread application domains including healthcare, finance and e-voting. However, achieving an efficient solution via use of blockchain requires consideration of a range of factors such as block generation rate, transaction speed, and block size which have a profound role in determining the overall performance of the solution. Current research into this aspect of blockchain is focused on Bitcoin with the objective to achieve comparable performance as of existing online payment systems such as VISA. However, there exists a gap in literature with respect to investigating performance constraints for wider application domains. In this paper, we present our efforts to address this gap by presenting a detailed study into performance and scalability constraints for an e-voting system. Specifically, we conducted rigorous experimentation with permissioned and permissionless blockchain settings across different scenarios with respect to voting population, block size, block generation rate and transaction speed. The experiments highlighted interesting observations with respect to the impact of these parameters on the overall efficiency and scalability of the e-voting model including trade-offs between different parameters as well as security and performance.

Keywords: Blockchain, e-Government, Blockchain Scalability, e-Voting, Electronic Voting, Performance Constraints

1. Introduction

Voting is one of the fundamental characteristics of human democracy with continuous efforts made throughout human history to improve the processes, mechanisms and methods involved to conduct voting in a verifiable, transparent and accessible manner. The advent of Information and Communication Technologies (ICT), has brought renewed emphasis on using ICT to facilitate voting processes primarily to enable transparency and accessibility. Since its first use as punched-card ballots in 1960s, e-voting systems have achieved remarkable progress via the internet technologies [1] with a range of terminologies used such as *e-voting* (using a machine in a polling station), *digital voting* (use of electronic devices such as voting machines) and *i-voting* (using web browser to cast vote). The focus of our research is on using advancements in ICT via electronic machines in polling stations as part of a public vote (commonly called General Election in the UK) with the view of investigating challenges in this respect and potential solutions to address them.

Blockchain has attracted significant attention with prominent applications across finance [2], healthcare [3] and supply chain management systems [4]. A Blockchain resembles a data structure which maintains and shares all the transactions being executed through its genesis. It is primarily a distributed decentralized database that maintains a complete list of constantly germinating and growing data records secured from unauthorized manipulation, tampering and revision. Blockchain allows every user to connect to the network, send new transactions to it, verify transactions and create new blocks [5, 6, 7]. Each block is assigned a cryptographic hash (which may also be treated as a fingerprint of the block) that remains valid as long as the data in the block is not altered. If any changes are made in the block, the cryptographic hash would change immediately indicating a change in the data which may be due to a malicious activity. Therefore, due to its strong foundations in cryptography, blockchain has been increasingly used to mitigate against unauthorized transactions across various domains [7, 8, 9].

Electronic voting (e-voting) is one of the emerging applications of blockchain whereby researchers aim to leverage benefits such as integrity, anonymity and non-repudiation which are critical for a voting application. Use of blockchain to facilitate e-voting applications has received significant attention recently with efforts such as [10, 11, 12, 13] leveraging blockchain technology to achieve secure and verifiable voting. Specifically, blockchain enables creating a pool of voters by generating unique physical addresses which can be used to represent the voter population and the candidates. A single voting token may be issued to every voter corresponding to their unique address thereby

representing a unique voter. Similarly, some addresses are reserved to identify candidates which may then be used to transfer voting token from voters addresses. The voting transaction generated by the voter contains the transfer of voting token to his desired candidate as a receivers address. This transaction becomes part of the main blockchain when confirmed into a block by one of the designated miners. The connected node to the seed node then takes part in the data synchronization process to update the public ledger of blockchain along with its local copy. This process enables achieving strong non-repudiation for the e-voting application as the record of the transaction becomes immutable due to consensus protocol acknowledged by all miners.

Whilst existing research has focused on leveraging benefits of blockchain such as those highlighted above, efforts with respect to in-depth investigation into challenges surrounding scalability of blockchain based solutions are primarily limited to cryptocurrencies such as Bitcoin. For instance, [14] is one of leading efforts to investigate scalability constraints of Bitcoin where authors investigated practical limitations in achieving performance level expected with respect to attributes including time required to mine a transaction into the block of the longest blockchain, processing rate for number of transactions, time required to download and run a full copy of blockchain from scratch in order to participate in the process of transaction validation, costs incurred due to purchasing, maintaining and working of resources (such as electricity utilization). The motivation for this and other similar research is the comparison with mainstream online payment system such as VISA. Specifically, VISA can process up to 2000 transactions per second whereas Bitcoin system is limited to 7 transactions per second primarily due to limits on block generation rate and transaction processing speed [15, 16, 17].

However, wider applications of blockchain mentioned earlier use platforms such as Ethereum [18] and Multichain [19] that are fundamentally different from Bitcoin with respect to factors such as block generation rate, consensus algorithm, transaction process rate and block size. These parameters have profound role in determining scalability of a blockchain based solution. Furthermore, the throughput of a blockchain based system not only depends upon factors such as the capacity of infrastructure such as hashing power and memory but also on the type of transactions used within an application. Additionally, in scenarios such as e-voting in public domain, large number of transactions are expected to be recorded in blockchain concurrently. Therefore, if the rate of incoming transactions to the unconfirmed pool of transactions does not match to rate of confirmation of transactions to the blocks by the miners, it can result in significant performance overhead as well as delays in transaction confirmation time. Consequently, an in-depth investigation is required to identify and assess challenges with respect to scalability for wider application domains such as the one considered in this paper.

In view of the above challenges, we conducted a thorough investigation into the challenges surrounding scalability of blockchain based applications in general and blockchain based e-voting applications in particular. Our investigation includes experimentation with a blockchain test-bed hosting an e-voting application focusing on variety of scenarios across *permissioned* and *permissionless* blockchain settings. Permissionless setting simulates environments with smaller number of users adopting a public blockchain with experimentation varying with respect to level of difficulty to mine a block, the block-size and average block creation time. These experiments reveal significant trade-off between transaction block size, block generation rate, and transaction processing speed. Furthermore, scenarios with permissioned blockchain aim to simulate e-voting environments with large number of voters such as public voting systems. The experimentation involves evaluation of the system with respect to the volume of transactions as well as the number of remote client voting machines operating from different network locations to observe impact of such constraints on the overall performance of a blockchain based system.

We present the findings from our investigation in this paper with the aim to aid research community to understand the caveats with respect to achieving scalable blockchain based solutions as well as expanding new horizons in this respect. Consequently, through our investigation, we are able to provide a rigorous assessment of the capability of the underlying blockchain fabric and identify potential trade-offs required to achieve desired level of performance. Specifically, the paper makes the following contributions:

- A thorough investigation is performed aimed at identifying and highlighting significance of parameters such as block size, block generation rate and transaction processing speed to achieve scalable solutions using blockchain technology.
- Rigorous experimentation is conducted which aims to identify and highlight performance constraints for both *permissioned* and *permissionless* blockchain settings which can potentially aid researchers in wide range of application domains.
- A novel blockchain based e-voting system is presented which investigates the capabilities of blockchain technology to achieve e-voting for permissioned and permissionless voting models.

Rest of the paper is organized as follows: Section 2 presents a background for e-voting systems providing important context to cutting edge research within this domain. Section 3 presents a summary of existing work with respect to scalability research within blockchain identifying the gap addressed by this research. Section 4 presents an overview of the proposed e-voting system followed by details of implementation and experimentation in section

5 which also includes details of different scenarios and an analysis of results observed. Section 6 concludes the paper.

2. Electronic Voting

Electronic voting has been an area of research focus for many years by using computing machines and equipment to cast votes and produce high quality and precise results in accordance with the sentiments of the participating voters. Various attempts have been adopted in practice to support election process through the use of ICT. One of the first uses of technology involved casting vote on paper which were scanned and tallied at every polling cell on a central server [6, 20, 21]. Direct Recording Electronic (DRE) voting systems were adopted later and have attracted significant success in encouraging voters to use this technology. In the case of DRE, voters begin their journey by going to their polling place and get their token to vote where they utilize their token at the voting terminal to vote for their preferred candidate. When the candidate selection procedure is completed, DRE systems present the final selection to the voter before ballot casting is completed [22].

More recently, Hao et al. [21] proposed a two round protocol that computes the tally in two rounds without using a private channel or a trusted third party. The protocol is efficient in terms of computation and bandwidth consumption but is neither robust nor fair in certain conditions [22]. Dalia et al. [22] proposed a protocol to improve the robustness and fairness of the two round protocol. Shahandashti et al. [23] proposed E2E verifiable voting system named DRE-ip (DRE-i with enhanced privacy), that overcomes limitations of DRE-i [24] i.e. instead of pre-computing ciphertexts, DRE-ip encrypts the vote on the fly during voting process. DRE-ip achieves E2E verifiability without TAs, but at the same time provides a significantly stronger privacy guarantee than DRE-i. In [25], end-to-end verifiability is achieved through the Mixnet protocol [26] that recovers the plaintext ballot in an unlinkable manner by randomizing the ciphertext through a chain of mix servers. These approaches perform well for end-to-end verifiability without compromising k0the privacy of voters.

An e-voting system may be broadly classified into two types facilitating; i) remote user interaction and ii) voting station engagement.

Remote user interaction: This model facilitates a voter to vote from any remote location utilizing a pre-verified identity such as a card or login credentials. Consequently, a person can vote from their home using their smartphone or computer to register to vote using a software or a web application. This voting model relies on cutting edge technological advancements to provide secure and easy method to use alternative to traditional voting system thereby helping to reduce human errors. Furthermore, voters can be provided a feedback that their vote is counted thereby improving the trust of the voter as well as verifiability and non-repudiation. A number of initiatives in this regard have been adopted indicating challenges with respect to security and correctness of the vote such as those identified by [27, 28].

Voting station engagement: This e-voting model includes scenarios where a voter can vote through a polling station where electronic voting systems such as DRE are deployed to facilitate the voting process. Typically, such machines are connected to the Internet to aid the overall voting process. A major advantage of DRE machines is the significant reduction in the cost as it avoids using ballot papers. A number of initiatives have been adopted by different governments across the world to implement such e-voting model such as Estonian i-vote system [29] and Norwegian i-voting [30]. However a number of security challenges were identified to achieve transparent and verifiable execution of these systems such as those highlighted in [31].

More recently, distributed ledger technologies such as blockchain have been used to achieve e-voting systems primarily due to their advantages in terms of end-to-end verifiability. With properties such as anonymity, privacy protection and non-repudiation, blockchain is a very attractive alternative to contemporary e-voting systems and has been used both for boardroom voting as well as public voting (focus of this paper). Within this context, McCorry et al. [32] proposed one of the early efforts for implementation of decentralized and self-tallying internet voting protocol over Ethereum blockchain using openvote [24] e-voting approach as their baseline. Khan et al. in [33] proposed a more recent secure digital voting system using blockchain with Prt Voter [34] as the underlying voting approach. Blockchain enhances the level of transparency as compared to traditional electronic voting system, primarily due to its decentralized nature. It bears the tendency to preclude many conventional ways of fraud which are very common in the manual or traditional electronic voting system. Furthermore, it is computationally infeasible to change any of the blocks, due to the highly complex formal problem solving required thereby significantly reducing the risk of vote manipulation.

Although blockchain introduces a range of benefits to e-voting domain, a number of challenges remain as observed in attempts to conduct blockchain-based e-voting in New South Wales [27, 28] and Estonia [31]. The challenges encountered by these voting models within the context of implementation with blockchain are essentially different primarily due to their inherent characteristics such as voting population, voting methods and logistic considerations. For instance, performance scaling to a large population is not a significant concern in boardroom voting where typical population is less than 20 voters [32], however this is a significant challenge for public voting model as the system is expected to take into account large number of voters with concurrent processing. The volume of voting population at the blockchain layer is significant as it has consequences on factors such as number and characteris-

tics of miners, number and characteristics of validators and mining process.

3. Related Works

With respect to existing work related to investigating performance and scalability of blockchain, most of the efforts are focused at cryptocurrency applications of blockchain such as Bitcoin. A primary reason for this is the comparison with existing online payment systems such as VISA which have the capability to process 2000 transactions per second whereas a Bitcoin system can only process 7 transactions per second. A fundamental cause of Bitcoin's comparatively low performance is due to the block generation rate (currently limited to 10 minutes) and transaction processing speed. Therefore, existing efforts have emphasized enhancing transaction speed of Bitcoin to improve its overall scalability thereby facilitating its wider adoption.

One of the leading works in this respect is presented by Croman et al. [14] where authors investigated practical limitations in achieving desired performance level with respect to attributes such as time required to mine a transaction into the block of the longest blockchain, processing rate for number of transactions, time required to download and run a full copy of blockchain from scratch in order to participate in the process of transaction validation, costs incurred due to purchasing, maintaining and working of resources (such as electricity utilization). The investigation included taking into account the performance metrics such as block size, transaction processing speed and block generation rate for Bitcoin. In order to assess the practical limitations, experimentation with varying settings with respect to transaction processing speed were conducted monitoring effective throughput of the Bitcoin network. The work presented also acknowledges impact of a scalable blockchain on its security i.e. improving scalability of blockchain can contribute towards limiting opportunities for forking, transaction malleability attack, and double spending thereby improving overall security of the network.

A fundamental concept within Bitcoin network is the reward for miners for correctly mining transactions leading to scenarios where greedy miners can affect the overall transaction processing within the network by prioritizing transactions with higher transaction fee. In this context, Karame et al. [35] and Zheng et al. [17] presented efforts to study the behaviour of mining nodes where greedy nodes can skew the mining process towards centralization by attempting to utilise their computing power to gain financial rewards. Specifically, [35] also highlighted interesting relationship between scalability and security through network propagation delays. Due to such delays and utilising Bitcoin network bandwidth, an attacker can cause disruption to the blockchain by attempting to stop new blocks to the victim.

Block size is another important factor in determining the transaction speed within a blockchain network. The block size within Bitcoin blockchain is currently fixed to 1 MB with maximum 7 transactions allowed per second with an average one hour required to confirm a transaction. In order to increase the transaction speed, one approach can be to increase the size of the block as proposed by Segregated Witness [36]. However as identified by Zheng et al. [17], if block size is high, it is very likely that the generation of new blocks will be very slow potentially leading to forking. This challenge is also highlighted by Zheng et al. in [16] proposing that if the size of the block is increased to gather more transactions in the block, this may lead the network towards centralization as the number of participants with capabilities to store such volume of data will be less. Consequently, authors highlight the non-trivial challenge to achieve a trade-off between transactions processing rate and network latency thereby achieving scalability without compromising security of the blockchain.

Furthermore, Xu et al. [15] proposed the implementation of GHOST and Pair-wise ledgers such as R3s Corda24 however there are significant concerns in their adoption such as availability of data due to the absence of globally replicated database. In this respect, authors proposed increasing the number of nodes to run full copy of decentralized and trust-less blockchain network in Bitcoin application to achieve security.

With respect to on-chain solutions to improve scalability of blockchain, Vukoli et al. [37] and Bano et al. [38] presented a study into the feasibility of available solutions. Specifically, Vukoli et al. [37] presented a comparison between two different approaches to achieve consensus among nodes i.e. Proof of Work (PoW) and Byzantine Fault Tolerance (BFT) replication targeting their scalability constraints. Focusing on Bitcoin blockchain, authors highlighted block creation rate and block size as key parameters to improve scalability of bitcoin. An interesting observation from this effort is that PoW based blockchains are more scalable in terms of number of nodes by compromising performance unlike BFT based blockchains, however BFT performs better for smaller number of nodes. Although the effort identifies new horizons, it requires discussion on how varying block sizes and block creation rate may reveal strengths and weaknesses of blockchain in general or with respect to an application in particular. Furthermore, Bano et al. [38] focused on the design based techniques of blockchain proposed to address the challenge of scalability including Bitcoin-NG [39] and Practical BFT (PBFT) and Sharding transactions.

Among other efforts, Kim et al. [40] define scalability as a function of three dimensional parameters; the transaction processing speed, the associated fee, and the volume for repository of the chain in case of a cryptocurrency application. In particular, authors have categorized solutions to improve scalability into classes i.e. on-chain, off-chain and side-chain. On-chain solutions include efforts such as Big Block which increases the block size as adopted by Bitcoin Unlimited to attain the maximum transmission limit reducing cost of the transmission. Although, a block can

contain more transactions however it can increase propagation delay due to high volume of block size which causes forking. Other on-chain methods discussed include Merkelized Abstract Syntax Tree (MAST) [41], recommended in Bitcoins BIP-114 and Segregated Witness (SegWit) [36]. Off-Chain approaches discussed include Lightning network where a channel is established between two addresses for exchange of transaction that uses multi-sign address to create stake for both sender and receiver. Channel creation is an on-chain activity and requires a transaction fee at main blockchain while exchange of transaction is off-chain and is not maintained at main blockchain so there is neither a transaction fee nor a noticeable waiting time. Side-Chain is another technique for addressing scalability issue where exchange occurs across blockchain so that different functionalities across blockchains may be brought together to the blockchain in focus. For instance, using one blockchain based cryptocurrency into another blockchain based cryptocurrency to use its features. Using this approach, a Bitcoin can make use of smart contracts on Ethereum blockchain whereby transactions are valid through such contracts.

Although most of the research into scalability of blockchain is focused on Bitcoin, Mattias et al. [42] present an effort focused on the applications of blockchain technology in the areas other than cryptocurrency. The authors compare performance of a blockchain network through the spectrum of security vulnerabilities, the impact of scalability and how these are related to decentralized nature of blockchain.

The research presented in this paper is focused on the challenge of investigating performance constraints with respect to scalability of a blockchain. Within our research our focus is on wider application domains with a special use case of e-voting and therefore not limited to cryptocurrencies such as Bitcoin. The research is aided by rigorous experimentation to highlight observations which are important in achieving a scalable solution. In order to observe the constraints of scalability, we performed experiments and evaluation with respect to number of attributes of blockchain scalability including block size, block generation rate, the impact of Proof of Work and non-Proof of Work based blockchains (public versus private blockchains), on-chain and off-chain data storage and fetching of the processed data over the network involving network constraints. Our study included large number of transactions submitted parallel from multiple clients aiming to explore relationship between the capacity of the block to accommodate number of transactions and the rate at which transactions are processed.

4. A Blockchain based e-Voting System

We have developed a blockchain based e-voting system for public voting which is also presented at its initial stage in [33] and illustrated in Fig. 1. In this paper, we present an overview of our model including interactions between different entities involved to provide context to the experimentation and analysis. We use our e-voting system as an example scenario to identify and analyse scalability considerations within a blockchain based system.

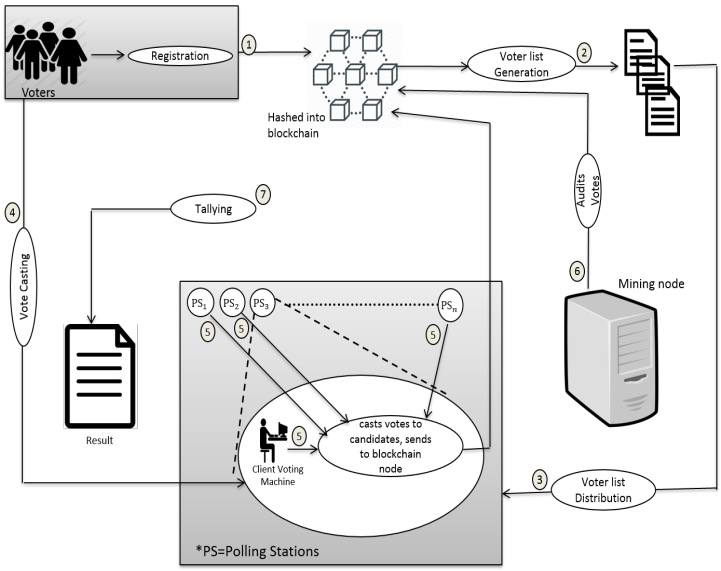

Figure 1: A blockchain based e-voting architecture

As presented in Fig. 1, the electoral process requires certain tasks to be carried out before the voting can be conducted including offline generation of voter addresses, generation of candidate addresses, entitling the voters and candidates to participate in the election process (as per their specified roles) at designated locations. These addresses are then used to cast vote from voters to candidate. The offline activities also involve generation of voter list and its distribution to individual voting machines which may specifically be designated for casting votes by the voters to their desired candidates. As with the real-life voting systems, the proposed permissioned environment only allows participants with valid rights such as voter, candidates or miners. For instance, for a voter, rights are granted to receive a voting token from vote issuing authority which is consequently sent to their desired candidate. Similarly, a candidate should be allowed to receive the voting tokens to their address from voters addresses.

The status of the voting process relies upon a pool of trusted miners which are responsible for either accepting a voting transaction by adding it to their newly created block or can simply reject a transaction preventing it to become part of the blockchain ledger. In the core polling day activity, the voters, using the voting machine (located at the polling station), may cast their vote. The voting status (whether a vote has been cast successfully or not) remains unconfirmed until a miner from the mining group confirms it and updates the ledger of main blockchain. In this way, an immutable collection of voting records start to deposit and link in the form of chains of block through their hashes which may easily be used for counting and tracking of votes.

Our proposed e-voting system is successfully deployed as an e-voting application using Multichain as the blockchain fabric. The proposed e-voting system achieves the objective of maintaining secrecy of voting population, verifying voter entitlement and confidentiality of the vote while facilitating user in the process of casting vote. The overall voting process is illustrated in Fig 2.

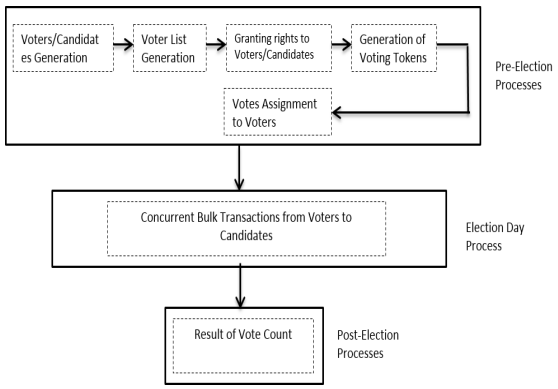

Figure 2: High-level illustration of the voting process

The voting process starts with the registration of voters as unique hashes. In terms of blockchain, these are addresses on blockchain that will be used by these voters to transfer their votes to the candidates. Each voter address is assigned a voter token which represents a vote in the system with a voter address only allocated one token to avoid duplicate voting. Since the domain of our research application is voting, the asset here refers to a vote. Therefore, after creating votes as assets on our private blockchain, these voting assets are then allocated to an address which will act as an authority to transfer the right of vote (by performing a transaction of moving voting asset) to the eligible candidates. Using these voter addresses, a list of registered voters is generated which is used to segregate voters into different polling stations. Therefore, each polling station will have its own list of voters (list of valid blockchain addresses) which will be used by client voting machines at each polling station. In this way, as with the current public voting systems in modern world democracies, only a voter registered to vote at a specific polling station will be able to cast their voter using these client voting machines at the polling station.

Furthermore, similar to the contemporary public voting systems, vote casting activity will be carried out in parallel across all designated polling stations simulating scenarios where polling stations will have designated e-voting machines to facilitate voters. In terms of blockchain, the operation of vote casting is achieved through a blockchain transaction whereby the voter token is transferred from a valid voter address to a valid candidate address. In order to achieve verifiability, a vote can is not successfully cast unless it is confirmed by one of miners and acknowledged by all the miners (consensus) to maintain an auditable and tampered proof list of voting records. Furthermore, when all the voters are able to cast their votes, the records from blockchain can be tallied to produce the result of the vote count against each candidate.

5. Implementation and Experimentation

In order to evaluate our voting model with respect to both permissionless and permissioned access of blockchain network, an experimental setup was established as a test bench. The setup consisted of miners, full blockchain node containing list of candidates and voters, and a client for submitting voting transactions to the blockchain node. The blockchain was implemented using an open source blockchain platform Multichain using its Alpha 4 version released in August, 2018. We selected this platform as it includes new features for handling streams of data which we plan to use along with voting transactions in the future. The machines for creating blockchain node and client zone were running Windows 10 operating systems with Intel Core i7-7500 CPU processors. This set up was extended for permissioned blockchain by introducing an additional connected full node to the master blockchain seed node and introducing number of Java based remote clients to the blockchain network facilitating submitting votes in large numbers from various clients. These Java based remote clients are capable of communicating to the blockchain node by accessing Multichains Application Programming Interface (API) through JSON based Remote Procedure Call (RPC) client to submit their voting transactions into the blockchain after being verified by the miners.

The objective of investigation presented in this paper is to understand the impact of blockchain engine and its associated attributes on e-voting architecture presented in the previous section with the view to identify bottlenecks to achieve optimum performance level. In order to achieve this objective, a thorough investigation was conducted by implementing three different scenarios each corresponding to different voting model settings based on the number of voters, candidates and the type of client (local vs remote) involved. The first two scenarios were implemented using a permissionless blockchain with a population of 10 voters where all participants had the privilege to mine the votes into the blockchain. However, third scenario was implemented as a permissioned blockchain with designated nodes as miners and validators. This scenario represents a public voting model where designated bodies are responsible for conducting fair voting process.

Although the experimentation includes both permissioned and permissionless blockchain settings, the parameters monitored through these experiments were: maximum number of voting transactions a block can contain, average number of voting transactions a block currently can host, voting transaction size, maximum transaction processing speed, and the current operational transaction processing speed of the system. For instance, if the blockchain generation rate is not compatible with the overall systems desired output it may lead to serious performance degradation. For instance, if block generation rate is too high, there

Platform	Hardware Specifications		
	Processor	Memory	Page file
Windows 10 Pro 64-bit (10.0, Build 10586)	Intel(R) Core (TM) i3-4005u CPU @ 1.70GHZ (4 CPUs)	4096MB RAM	5586MB used 1887MB available

Figure 3: Blockchain node specification for permissionless setting

Scenario	Blockchain Parameters				
	Difficulty	No. of miners	Block generation Rate	Maximum Allowable Block size(MB)	No. of Bits Required for Proof of work
A	00001526	10	15	1	16
B	1.00000000	10	60	1000	32

Figure 4: Initial parameter values for permissionless setting

may exist a situation where significant number of transactions keep waiting in the pool of unconfirmed transactions which can ultimately overflow the volume of memory pool of node. This situation may cause the blockchain to either respond slowly or become non-responsive. Similarly, in a situation where the blocks are being generated very quickly, it may affect the blockchain significantly by mining empty blocks without any transactions. This phenomena is very common in private blockchain and therefore an important challenge to be addressed.

Furthermore, different settings were experimented with respect to varying values for block size, block creation rate to identify and highlight significant trade-offs between block size, block generation rate and total number of voting transactions per second. Finally, in each of these scenarios, transactions were carried out to transfer a vote from a voters addresses to the candidate addresses. Based on the output data of these scenarios, response of the blockchain was recorded and analysed in the form of graphs analysing transaction mining time and waiting time of a transaction in the pool.

5.1. Experiments with permissionless setting

Our first setup involved a permissionless blockchain setting where all participants had the role of a voter and miner. For this setting, we created a population of 10 voters with their wallet addresses which represents a voting scenario with small number of voters such as boardroom. The experiments for this setting have been carried out on setup explained in Fig 3. Within this setup, we conducted experiments with varying configuration for blockchain including parameters such as block size, block generation rate and the difficulty of the Proof of Work. Once the chain was created, it initialized the blockchain with immutable parameters as detailed in Fig 4 which also presents the blockchain configuration for two scenarios for permissionless setting.

S. No.	Voter Hash	Candidate Hash	Transaction Hash
01	1Zq8GjEd5NUnXqqQjsnKFzuj5v92fvt4qekfsQ	1SS6H6G7aU96oMerxDu752FYAsn4ZaKVHZVyNR	b9c8432b858c47d23ac44a44a64ca0c44cbc11c5b5ef8216cef9934af651ef7d
02	1Nz4aWjykexrxgeJq2xRLixrmj4hU3oKaHtyCD	1SS6H6G7aU96oMerxDu752FYAsn4ZaKVHZVyNR	8de26636e4b0c035b55cd82cea119d11e24321c7b771779720b69bd74a925b6b
03	16ckT5efpKVzu6fM7d6TvU5Uae7fgAxqv5Q2TK	1SS6H6G7aU96oMerxDu752FYAsn4ZaKVHZVyNR	7a763820bb758ac0d060da5be8c7598e43a495e608f654e4f3e1a3fe8d9e4b3d
04	1bPRUUm8EeYA1cGLX5585RNAdWhxj3uyahQ1z6	1SS6H6G7aU96oMerxDu752FYAsn4ZaKVHZVyNR	1ee2338ae825b3b78c4a21492ea23f8e03cb5723d3bd3216dbde6bb39f3e78ed
05	1DtQn2g3MCNLtHZxGhUj67Kc4SWTeXi5jTeeCy	1SS6H6G7aU96oMerxDu752FYAsn4ZaKVHZVyNR	91674fdb026e0a8fb0e407ea2ee69eb4123377ed9a0ee58c93d9e50db0fe2f09

Figure 5: Voter, Candidate and Transaction hashes for initial experiments

Our aim is to investigate the impact of parameters such as transactions processing per second, transaction size, maximum and average number of transactions which can be processed, and average block size on key performance indicators. These parameters are critical in identifying constraints of blockchain process execution with respect to scalability and consequently on the application running on blockchain. If the block generation rate is not compatible with the overall systems desired output, serious performance degradation can be experienced. For instance, if block generation rate is too high, there may exist a situation in which too many transactions are kept waiting in the pool of unconfirmed transactions leading to overflow of the volume of memory pool of node. This situation may cause the blockchain to either respond slowly or even become unresponsive at all. Similarly, in a situation where the block generation rate is very small, blocks are being generated very quickly, which may affect the blockchain by engaging the blockchain into mining empty block even without having a transaction. Through our experience, this phenomena is commonly witnessed in private blockchain. Therefore, an optimum level is required to be set to keep a matching condition of equilibrium between the rate of incoming transactions and the rate at which the transactions are mined into the new blocks.

Scenario A: For the first scenario within permissionless blockchain setting, the time interval for adding new blocks to blockchain was set at 15 seconds i.e. at a regular interval of 15 seconds, unconfirmed transactions will be processed, packed and appended to the existing chain of blocks in the form of a new block. In case of this experiment, the miner has been configured to compute on average 2^{16} transactions to add its block. Additionally, the block size is set to 1MB for this experiment. In order to make the blockchain up and running, there were a total of 60 blocks at the start of the blockchain. A sample set of voter, candidate and transaction hashes for this experiment are presented in Fig 5.

Scenario B: Following on from the initial experiments presented in scenario A, we conducted further experiments using larger block size, increased block generation rate and the required amount of bits in Proof of Work was extended to 32 bits. This implies that on average the miner required 2^{32} hashes to mine a block. These variations in the values of the parameters significantly affected the overall behaviour of the system. Fig 4 shows the parameter values for the experimentation for scenario B.

Let the arrival time of a transaction Tx to be submitted to unconfirmed pool is represented as *Tau*, the time taken by node for adding the transaction to its local wallet is represented as *Tnw*, total time taken for confirma-

```
{
    "balance" : {
        "amount" : 0.00000000,
        "assets" : [
            {
                "name" : "VotingAsset",
                "assetref" : "20769-267-53562",
                "qty" : -1.00000000
            }
        ]
    },
    "myaddresses" : [
        "1Zq8GjEd5NuNXqqQjsnKFzuj5v92fvt4qekfsQ"
    ],
    "addresses" : [
        "1SS6H6G7aU96oMerxDu752FYAsn4ZaKVHZVyNR"
    ],
    "permissions" : [
    ],
    "data" : [
    ],
    "confirmations" : 21588,
    "blockhash" : "0000ef50ccfd64745a4a95e239e9f2feea57a45f89518dcf797d9655660b90bf",
    "blockindex" : 1,
    "blocktime" : 1542300661,
    "txid" : "b9c8432b858c47d23ac44a44a64ca0c44cbc11c5b5ef8216cef9934af651ef7d",
    "valid" : true,
    "time" : 1542300652,
    "timereceived" : 1542300652
```


Figure 6: A transaction from voter to candidate

Transaction No.	Transaction ID	Taw=Tnw (time stamp)	Tad (time stamp)	Tctn	Bs
01	b9c8432b858c47d23ac44a44a64ca0c44cbc11c5b5ef8216cef9934af651ef7d	1542300652	1542300661	9	516
02	8de26636e4b0c035b55cd82cea119d11e24321c7b771779720b69bd74a925b6b	1542300713	1542300724	11	517
03	7a763820bb758ac0d060da5be8c7598e43a495e608f654e4f3e1a3fe8d9e4b3d	1542300750	1542300761	11	516
04	1ee2338ae825b3b78c4a21492ea23f8e03cb5723d3bd3216dbde6bb39f3e78ed	1542300785	1542300787	02	517
05	91674fdb026e0a8fb0e407ea2ee69eb4123377ed9a0ee58c93d9e50db0fe2f09	1542300994	1542301002	08	516

Figure 7: Sample voting transaction details for scenario A

tion of transaction from unconfirmed pool to enter into blockchain is *Tctn* and the block size is represented by *Bs*. In this respect, Fig 7 presents sample voting transaction details for scenario A whereas Fig 8 presents final candidate votes for scenario A. Similarly, Fig 9 presents sample voting transaction details and Fig 10 presents the final candidate votes for scenario B respectively. Furthermore, Fig 6 shows the blockchain output for a transaction from voter to candidate i.e. simulating casting of a vote whereas Fig 15 shows blockchain output for specification of a block containing a transaction. Finally, typical voting asset within the proposed system is presented in Fig 11.

5.1.1. Analysis and discussion for permissionless setting

Scenario A: Fig. 12.A shows that new transactions are arriving at different timestamps but are added immediately to the nodes wallet upon arrival. Therefore, as evident from this figure, there is no delay in the transition of its journey from pool to blockchain. In order to study this behaviour in the context of throughput and blockchain performance (transaction processing speed, maximum number of transactions per block etc.), it may be considered that the transaction movement from pool to nodes wallet is not a major factor of consideration for the initial

S.No	Candidate Hash	Asset Name	Asset Reference	Qty
01	1SS6H6G7aU96oMerxDu752FYAsn4ZaKVHZVyNR	Voting Asset	20769-267-53562	6
02	1YetjPS8i6km12QavaMX2PJkRU1LuBmcYcG541	Voting Asset	20769-267-53562	4

Figure 8: Final candidate votes in their wallet for scenario A

T No.	Tau=Tnw (time stamp)	Tad (time stamp)	Tctn	Bs
01	1543813800	1543817161	3361	5468
02	1543813851	1543817161	3310	5468
03	1543813898	1543817161	3263	5468
04	1543813938	1543817161	3223	5468
05	1543814081	1543817161	3080	5468

Figure 9: Sample voting transaction details for scenario B

S.No	Candidate Hash	Asset Name	Asset Reference	Qty
01	1PxXxADdtsrG25cGBHqD8UugaTWbiPZYSULiWS	Voting Asset	3-265-63893	6
02	1WJkwH6ZYQnXBivdnGjoSqeC58AbsmhpAx5gLs	Voting Asset	3-265-63893	4

Figure 10: Final candidate votes in their wallet for scenario B

experiments. However, there are some other (although indirect) factors such as pool size and number of dependent transactions which may affect behaviour of this scenario in blockchain. According to the official documentation of Multichain blockchain, there is no fixed barrier in limiting the capacity of the pool but nodes may have constraints in picking up transactions from other nodes. Another thing to ponder here is about the dependent transactions such as multiple transactions from the same node address which are dependent on each other. In this case, there is a possibility that the subsequent dependent transactions start to arrive earlier than its previous transaction on which it depends. Consequently, in this case, the earlier arrived transactions cannot be confirmed until its previous transaction gets into the block delaying the transaction.

Fig 12.B represents the arrival of voting transactions at different timestamps into the nodes wallet. These transactions were picked up by the miners to begin the race of mining those transactions first. Therefore the transaction processing depends primarily upon the Proof of Work which is performed by the miner while keeping the average block generation rate constant. An observation from Fig 12.B is that it is not necessarily important that the transactions which arrive earlier to the pool, take shorter time to add to the block even though when the difficulty is kept constant among all the transactions.


```
    "genesis-nbits" : 536936447,
    "genesis-nonce" : 504,
    "genesis-pubkey-hash" : "424c725a189254acc999ee5402f69de2dfd8e408",
    "genesis-hash" : "0033384d52a48642dc7f0ac1f2bfa5610beb48714405c58290f99d0b645c6c65",
    "chain-params-hash" : "072b3e22fcd7f85f1f28307f2332a24a4fa1618d9dd79e0ff2cbd3a8680f716f"
}

::\multichain-windows-2.0-alpha-4>
::\multichain-windows-2.0-alpha-4>
::\multichain-windows-2.0-alpha-4>multichain-cli.exe PermissionedVotingChain listassets
{"method":"listassets","params":[],"id":"98948026-1550821332","chain_name":"PermissionedVotingChain"}

[
    {
        "name" : "VotingAsset",
        "issuetxid" : "b724ae37a752dd216d3a59f2a64227d58fe911f9e93b6ef95e4625c94d0aa50a",
        "assetref" : "60-265-9399",
        "multiple" : 1,
        "units" : 1,
        "open" : false,
        "restrict" : {
            "send" : false,
            "receive" : false
        },
        "details" : {
        },
        "issueqty" : 200000,
        "issueraw" : 200000,
        "subscribed" : false
    }
]
```


Figure 11: Sample voting asset within blockchain

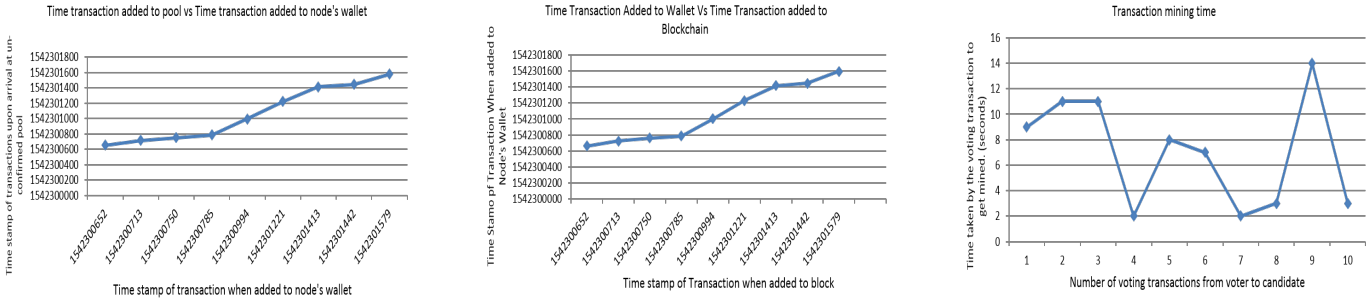

Figure 12: A) Time to add unconfirmed transactions from pool to local wallet, B) Time taken by miners to confirm transactions into the block C) Transaction Mining Time

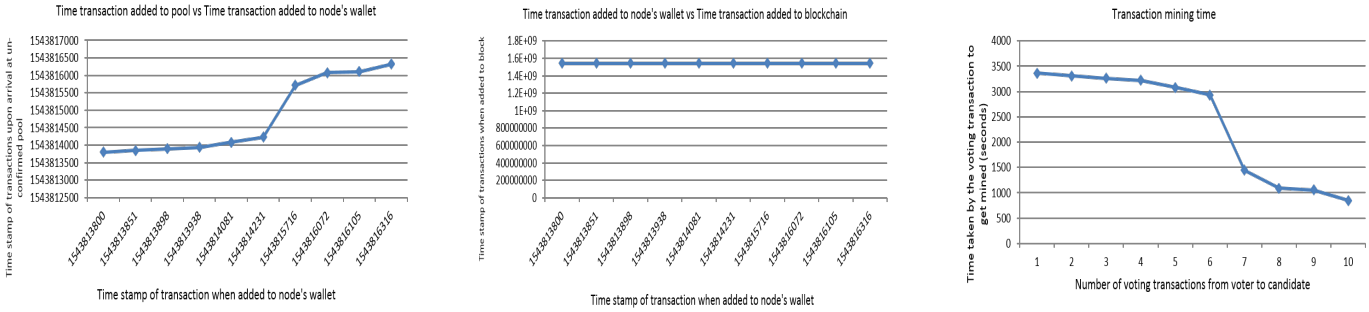

Figure 13: A) Time to add unconfirmed transactions from pool to local wallet, B) Time taken by miners to confirm transactions into the block C) Transaction Mining Time

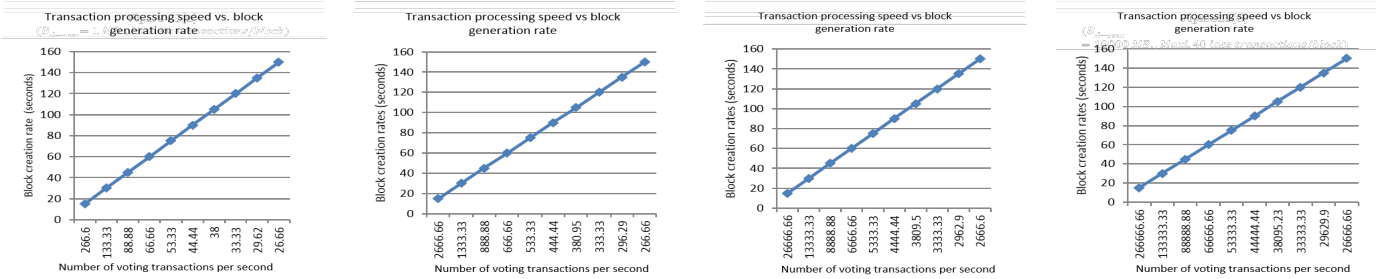

Figure 14: A) B_{d-max}= 1MB, Max.4k transactions/block, B) B_{d-max}= 10MB, Max.40k transactions/block, C) B_{d-max}= 100MB, Max.400k transactions/block), D) B_{d-max}= 1000MB, Max.4000k transactions/block


```
{
    "hash" : "0000ef50ccfd64745a4a95e239e9f2feea57a45f89518dcf797d9655660b90bf",
    "miner" : "1Zq8GjEd5NUnXqqQjsnKFzuj5v92fvt4qekfsQ",
    "confirmations" : 21610,
    "size" : 515,
    "height" : 26644,
    "version" : 3,
    "merkleroot" : "353578f9bcc257372e486753d90abd3c9b7fe203b6d98f6f30b2c3c6af19c977",
    "tx" : [
        "75938e2bc7eff456e4d49671115046c51015229caae6cbf6eaae44c400076e52",
        "b9c8432b858c47d23ac44a44a64ca0c44cbc11c5b5ef8216cef9934af651ef7d"
    ],
    "time" : 1542300661,
    "nonce" : 42200,
    "bits" : "1f00ffff",
    "difficulty" : 0.00001526,
    "chainwork" : "0000000000000000000000000000000000000000000000000000000000068156815",
    "previousblockhash" : "00001bfa52453a54f12ab952eaae6fd649bc3510ca2924454e19884b8a4142bd",
    "nextblockhash" : "0000b02836e70925bbceb31bfbb4cdf1439b3041729c000a96bab1cf6f74a0e4"
}
```


Figure 15: Specification of block containing transaction

In order to evaluate the throughput of the system, transaction processing speed can be one of the vital factors to consider. In this context, Fig 12.C demonstrates time consumed by each voting transaction in the experiment. Since we have set our target block time to 15 seconds, this means that each block will be generated at an interval of 15 seconds. This interval is an average time of block generation which means that if a block is mined much earlier than the target block time then the rest of the block will be mined at a rate to restore the average rate of block generation. The difference in transaction delays is due to the time taken by the Proof of Work. The difficulty level has been kept constant throughout all the experiments so that it may not affect our dataset and we can build a direct relationship between number of transaction and its processing time to check the scalability of the system. The total amount of time taken by a transaction to be published publicly also depends upon its arrival time (in case of our Multichain platform). For instance, in our experiment where maximum target block time is 15 seconds, a transaction arrives into a wallet and a block is about to be

published within 3 or 4 seconds to keep the average time of 15 seconds, the transaction (if it does not depend upon any unconfirmed transaction) will most likely be published in the main blockchain within those 3 or 4 seconds, provided the block size does not exceed the maximum size limit. The situation may be different if the same transaction arrives just moments after the latest block was added.

Scenario B: As is evident from Fig 13.A, the transactions move instantly to nodes wallet upon arrival. An interesting observation here is that even if the transactions are issued with delay (as it is the case of transaction number), it has no impact on the time it takes to move the transaction into wallet. Furthermore, in Fig 13.B, it can be observed that the transactions are being confirmed to the same block due to increased block generation rate which were issued subsequently and also were independent of each other.

In order to understand the overall outcomes, we consider the Fig 13.B in the context of Fig 13.C. The above graph shows that the transactions arriving late to system are unexpectedly being mined earlier than the previous transaction even though these transactions do not depend upon each other. We believe primary reason for this occurrence is that the new transactions are arriving at a time when there is some time left for the block to add. Another interesting point here to note is that not only the block can contain more transactions but also that in this setting the work required by the miner is increased significantly due to Proof of Work been increased to 32 bits.

5.1.2. Scalability analysis for permissionless setting

In order to assess the scalability of the proposed e-voting system, we first use mathematical formulation to evaluate the ability of the proposed system to scale. We represent size of a voting transaction to be represented by T_s, block generation rate by B_t, maximum amount of data which can be held by the block $B_{(d-max)}$, and average block size by $B_{(d-avg)}$. In order to calculate the maximum number of transactions that can be accommodated by a block in our system $T_{(s-max)}$ per block and the average number of transaction $T_{(s-avg)}$ per block, we use standard metrics defined by Multichain platform. Therefore the size of the transaction T_s may be obtained by taking arithmetic sum of all the hexadecimal characters and dividing the sum by 2 [43]. For scenario A, the transaction size is 250B and with the maximum block size set to 1MB, it can support up to 4000 transactions in a block. However, an interesting observation is that with average block size as 516B, the system currently contains an average number of two transactions per block indicating that the block size is not used to its full capacity. We believe the reason behind small number of transactions per block is twofold; i) due to the small (10) voter population of the current system, and ii) due to the short block generation rate i.e. 15sec. Therefore the setup is not able to fully utilize the capacity of a block resulting in smaller blocks. The above result can also be confirmed from the blockchain output in Fig 15. However, as indicated above, the system has ability to scale to a significantly larger population.

The second parameter we are interested in is the transaction processing speed of our voting system. Let us assume the number of transactions per second to be represented by T_n. Mathematically, T_n can be written as

$$T_n = B_{(d-max)}/T_s/B_t \tag{1}$$

Substituting the values for block size, transaction size and block generation rate, we get;

$$T_n = 1000000/250/15$$

Therefore, $T_n = 266.6$ transactions per second.

Through the above calculation, we conclude that the proposed voting model is capable of supporting transaction speed of upto sixteen thousand voting transactions per minute which is envisioned to enable the system to perform effectively for small and medium sized voting environments. However, as identified in the earlier experiment, this represents capacity of the block and the actual number of transactions processed may also vary depending upon the block generation rate.

Similar to above, for scenario B, transaction size is 250B whereas maximum block size $B_{(d-max)}$ is increased to 1GB to assess the impact of larger block size on the transaction processing speed. Although the larger block size has the capacity to accommodate up to 4000000 transactions for each block however our experimentation has revealed the average block size $B_{(d-avg)}$ to be 5468B. Interestingly, this represents a 10-times increase compared to the average block size for scenario A and therefore indicates a directly proportional relationship between the maximum block size and average block size. Furthermore, B_t has been set to 60 seconds and number of required bits for Proof of Work has been set to 32 bits increasing the difficulty level. However, average transactions per block in this case are recorded to be 21 which is again a 10-times increase on the scenario A. Analysing the values of $T_{(s-max/block)}$ and $T_{(s-avg/block)}$, it can be seen that by increasing the block size the number of transactions that can be handled by a block has increased however increasing the block generation rate and difficulty level of Proof of Work did not have significant adverse impact. We present graphical analysis of this relationship in Fig 14. Therefore, we conclude that block generation rate and block size affect the rate at which transactions arrive to pool and the rate at which these transaction are confirmed to the block.

5.2. Experimentation with permissioned setting

For this phase of experimentation, the setup consisted of miners, full blockchain node containing list of candidates and voters, and a client for submitting voting transactions to the blockchain node. These machines for creating blockchain node and client zone are running Windows 10 operating systems over intel Core i7-7500 CPU processors. Additionally, we introduced a full node connected

to the master blockchain seed node and a number of Java based remote clients to the blockchain network to facilitate large number of voters on various clients. The setup is summarized in Fig 16.

The permissioned blockchain was created with 100,000 voters and designated nodes as miners and observers to simulate a conventional public voting environment. The parameters monitored through these experiments were; maximum number of voting transactions a block can contain, average number of voting transactions a block can host, voting transaction size, maximum transaction processing speed, and the operational transaction processing speed of the system.

Furthermore, the blockchain setting for these experiments is non-Proof of Work and instead uses *mining diversity* and *mining turnover* parameters provided by Multichain platform to achieve consensus. Specifically, *mining diversity* defines the proportion of mining power which is necessary to run blockchain properly. For these experiments, mining diversity is set to 0.3. Therefore the minimum number of miners required to run the blockchain successfully will be obtained by multiplying the value of mining diversity to the number of miners and then round the output to the closest integer. In our case, this value is found to be value is 3 as the total number of miners which are involved in running the public voting model is 10 as it is shown in Fig. 16. This implies that at least three out of ten miners will have to actively participate in the process of adding a new block to the transaction. Additionally, *mining turnover* sets up the scheduling of miners to get their turns in rotation using a round robin scheme. The value of mining turnover lies between 0 and 1 with a value of 1 implying that every miner will be looking to add a block to the blockchain which may create forks and potentially unnecessary computational tasks. Similarly if the value is set to 0, then a default round robin scheduling algorithm will be followed. For the current experimentation settings, the value has been set to 0.5 to achieve a balance between these two extremes. Consequently, Proof of Work is not required by the miners to get their turn and add the proposed block into the blockchain.

Within this setup, voter rights are granted to receive a single voting token from vote issuing authority and send this voting token to their desired candidate. Similarly, a candidate should be allowed to receive the voting tokens to their address which are expected to be received from voters' addresses. The process in this setup relies upon a pool of trusted miners responsible to either accepting a voting transaction by adding it to their newly created block or reject a transaction restricting it to become a part of the ledger.

The experimentation was divided into three different cases based on varying number of concurrent clients. Specifically, we evaluated our e-voting system with one, two and seven concurrent clients where these clients were located on a remote machine thereby taking into account network bandwidth and delay. As with the real-life public voting system, the proposed system is implemented as a permissioned and controlled environment so that only designated nodes are able to construct and commit voting transactions simulating the role of a polling station in real-life systems. Fig 17 and Fig 18.A present specifications of blockchain master node used for these experiments whereas Fig 18.B presents a sample voting asset within the proposed e-voting system.

Platform	Hardware Specifications		
	Processor	Memory	Page file
Windows 10 Pro 64-bit (10.0, Build 10586)	Intel(R) Core (TM) i7-7500U CPU @ 2.70GHZ (4 CPUs)	8192MB RAM	5586 MB used 15444 MB available

Figure 16: System specification for e-voting architecture

Platform	Blockchain Parameters				
	Mining Diversity	No. of miners	Block generation Rate	Maximum Allowable Block size(MB)	Mining Turnover
Windows	0.3	10	15	8.3	0.5

Figure 17: Blockchain setup for experimentation


```
{
    "chain-protocol" : "multichain",
    "chain-description" : "MultiChain PermissionedVotingChain",
    "root-stream-name" : "root",
    "root-stream-open" : true,
    "chain-is-testnet" : false,
    "target-block-time" : 15,
    "maximum-block-size" : 8388608,
    "maximum-chunk-size" : 1048576,
    "maximum-chunk-count" : 1024,
    "default-network-port" : 6743,
    "default-rpc-port" : 6742,
    "anyone-can-connect" : false,
    "anyone-can-send" : false,
    "anyone-can-receive" : false,
    "anyone-can-receive-empty" : true,
    "anyone-can-create" : false,
    "anyone-can-issue" : false,
    "anyone-can-mine" : false,
    "anyone-can-activate" : false,
    "anyone-can-admin" : false,
    "support-miner-precheck" : true,
    "allow-arbitrary-outputs" : false,
    "allow-p2sh-outputs" : true,
    "allow-multisig-outputs" : true,
    "setup-first-blocks" : 60,
    "mining-diversity" : 0.300000000001,
    "admin-consensus-upgrade" : 0.500000000001,
    "admin-consensus-admin" : 0.500000000001,
    "admin-consensus-activate" : 0.500000000001,
    "admin-consensus-mine" : 0.500000000001,
    "admin-consensus-create" : 0,
```


Figure 18: Experimentation specification for permissioned blockchain

5.2.1. Scalability analysis for permissioned setting

Our experimentation with permissioned blockchain involved three different settings varying based on the number of concurrent clients involved i.e. one, two and seven concurrent clients to conduct voting transactions. We present analysis of the results obtained below.

Experiments with one client: Fig. 19 demonstrates peak performance in terms of transaction processing speed when nine thousand voting transactions were cast from one client to the blockchain master node. At this stage, the system was operating at a frequency of 23.01 transactions per second i.e. in this case, a single voting transaction took 0.043 seconds to be mined. In order to understand its implications with respect to scalability of blockchain, block size and its generation rate have a profound role. For instance, if a transaction arrives shortly before a block is about to mine, it is likely that it will be mined to the block

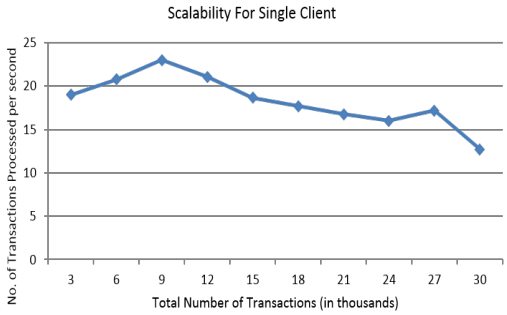

Figure 19: Tx processing speed vs number of Tx for one client

which is about to be added to the longest blockchain. Similarly, if the transaction arrives after the latest block has been added to the main blockchain, this transaction will have to wait until a miner mines it to the next block after the average block generation time (fifteen seconds in this case). The impact of block generation rate can increase when the transactions are constructed and committed from multiple remote concurrent clients with network latency also a factor. However, the general trend shown in Fig 19 is that the proposed e-voting system shows no limitations with respect to scalability in this case. Even in the worst case, when transaction speed is 12.71 transactions per second, the e-voting system looks to be in a very stable state and hence shows strength with respect to scalability. Another important observation is that this experimentation was conducted using a non-Proof of Work based permissioned blockchain where we have a pool of trusted and designated miners. Therefore the system suits real mode of transaction validation in this context and it does not introduce disadvantage of over utilization of electric and computational power.

Experiments with two concurrent clients: The second set of experiments involved two concurrent voting clients with one local client and the other interacting with the e-voting application remotely through a programmable interface. Since the additional remote client is concurrently running on the network, the network latency has a considerable impact as can be observed from the graph in Fig 20 which represents the average transaction execution time. Both clients started with the same number of voters and their voting transactions were being processed by the blockchain node with almost same frequency. Later on, due to potential network latency, transactions started to arrive late from the remote client. In this case, empty blocks are mined by the blockchain (Multichain and many other private blockchain platforms use this mechanism to keep the blockchain live while operating at a pre-set block generation rate). This happens in case if no transaction is being sent from any client (we did not observe this to be the case for our experiments). Another interesting possibility can be that the mining nodes picked up transactions which were coming from the first block, in that case the transaction hashes for first client transactions should be present more in the block which caused delay for processing transactions of the remote client. By examining the data at the blockchain, it was obvious that the miner at that moment picked up mostly the transaction from first client while the second client transactions were supposed to gather at the pool of unconfirmed transactions.

Experiments with seven concurrent clients: The third setting attempts to execute a public voting model simulating a real world scenario where overall 222,000 transactions were sent from seven different voting clients running concurrently across different remote locations. This scenario introduced a number of factors including network latency, block generation rate, block size, increased workload on blockchain master and connected node that caused variation in response time.
As is evident from Fig 21, a fluctuation can be observed in the processing speed of voting transactions. Furthermore, as in the previous scenarios, this speed is calculated by considering maximum execution time which is taken by an individual client. The execution time of individual client to process same number of transactions is presented in Fig 21.A. Here, due to network latency and bulk transaction load on blockchain node, there may arrive a situation where bulk transactions which are generated by seven different clients may increase memory pool size of blockchain node in such a way that the incoming flux of transaction does not synchronize effectively with the transaction mining process resulting in a mismatch between these two processes. Therefore, execution time of some clients is greater than the others. However, the system demonstrates to be scalable as the average time taken by an individual transaction is around 0.10 second which should be healthy for a voting process to carry on.

Summarizing the evaluation we have conducted with permissioned blockchain, we have observed that with the increase in number of concurrent clients, the workload for the blockchain master node increases which may result in delays in transactions committed to the blockchain. Fig 22 shows a comparison of systems throughput in terms of number of transactions which are processed by an individual client when conducted for case 1, 2, and 3. Comparing case 1 with 2 and 3 in Fig 22, it can be inferred that executing multiple clients in parallel over the network has a noticeable impact on the performance of blockchain. We have also observed the trade-off represented by block generation rate and block size and their significance to achieve a scalable solution. Furthermore, an interesting observation is the delay caused due to network connectivity. For instance, analysing the case where six thousand voting transactions were executed from a client, the transaction processing speed is almost similar in case 2 and 3 however when the same number of transactions are executed in case 1, the transactions processing speed is almost double which we believe is due to the absence of network delay.

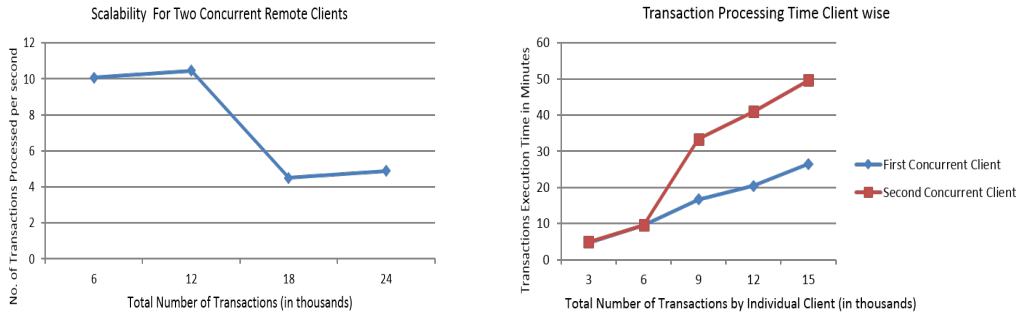

Figure 20: Experimentation with two clients: A) Avg. Tx processing speed, B) Tx processing time

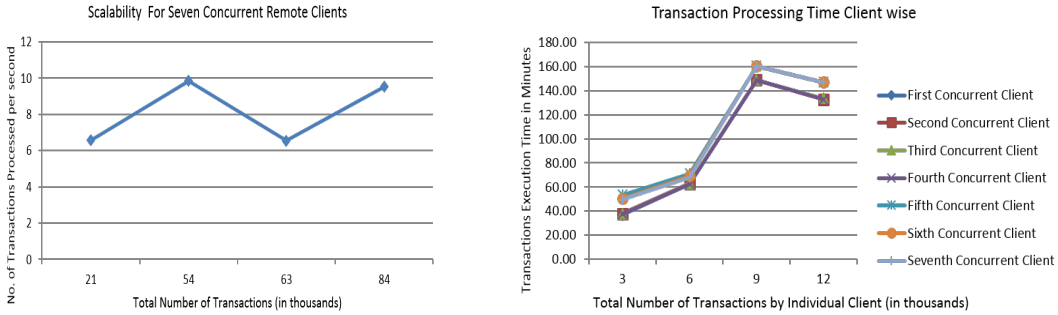

Figure 21: Experimentation with seven remote clients: A) Avg. Tx processing speed, B) Tx processing time

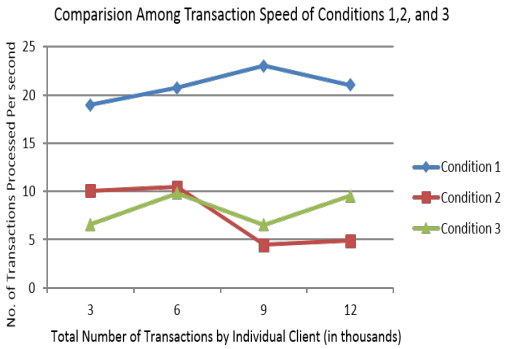

Figure 22: Comparative summary of Tx speed for case 1, 2 and 3

6. Conclusion

Blockchain is a disruptive technology and has attracted significant attention with prominent applications across diverse application domains including supply chain management, gaming, healthcare, real estate and finance. Electronic or e-voting is one of the emerging applications of blockchain where researchers have proposed to leverage blockchain capabilities to achieve integrity, anonymity and non-repudiation which are critical for a voting application. Current research into scalability and performance of blockchain is focused on Bitcoin with the objective to achieve comparable performance as of existing online payment systems such as VISA. However, there exists a gap in literature with respect to investigating performance constraints for wider application domains. In this paper, we have presented our efforts to address this gap by conducting an in-depth investigation of parameters which have a profound role in achieving scalable solutions using blockchain. Specifically, we investigate the role of block generation rate, block size, transaction processing rate and transaction size with respect to scalability of blockchain based solution. Our experimentation results highlight a trade-off between these parameters and identify avenues to explore for further research.

References

[1] J. Gobel, H. Keeler, A. Krzesinski, and P. Taylor, "Bitcoin blockchaindynamics: The selfish-mine strategy in the presence of propagation delay," *Performance Evaluation*, vol. 104, pp. 23 – 41, 2016. [Online]. Available: https://bit.ly/2IuarKL

[2] P. Treleaven, R. Gendal Brown, and D. Yang, "Blockchain technology in finance," *Computer*, vol. 50, no. 9, pp. 14–17, 2017.

[3] M. Mettler, "Blockchain technology in healthcare: The revolution starts here," in *2016 IEEE 18th International Conference on e-Health Networking, Applications and Services (Healthcom)*, Sep. 2016, pp. 1–3.

[4] F. Tian, "An agri-food supply chain traceability system for china based on rfid amp; blockchain technology," in *2016 13th International Conference on Service Systems and Service Management (ICSSSM)*, June 2016, pp. 1–6.

[5] M. Rosenfeld, "Analysis of hashrate-based double spending," *CoRR*, vol. abs/1402.2009, 2014. [Online]. Available: http://arxiv.org/abs/1402.2009

[6] J. P. J. S. Kadam, M., "Double spending prevention in bitcoins network," *International Journal of Computer Engineering and Applications*, 2015.

[7] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," *Cryptography Mailing list at https://metzdowd.com*, 03 2009.

[8] D. Kraft, "Difficulty control for blockchain-based consensus systems," *Peer-to-Peer Networking and Applications*, vol. 9, 04 2015.

[9] B. J. F. E. M. A. Narayanan, A. and S. Gold, *Bitcoin and Cryptocurrency Technologies,*. Princeton, 2015.

[10] D. Khoury, E. Kfoury, A. Kassem, and H. Harb, "Decentralized voting platform based on ethereum blockchain," 11 2018, pp. 1–6.

[11] N. Kshetri and J. Voas, "Blockchain-enabled e-voting," *IEEE Software*, vol. 35, no. 04, pp. 95–99, jul 2018.

[12] X. Sun, Q. Wang, P. Kulicki, and M. Sopek, "A simple voting protocol on quantum blockchain," *International Journal of Theoretical Physics*, 10 2018.

[13] B. Shahzad and J. Crowcroft, "Trustworthy electronic voting using adjusted blockchain technology," *IEEE Access*, vol. 7, pp. 24 477–24 488, 2019.

[14] K. Croman, C. Decker, I. Eyal, A. E. Gencer, A. Juels, A. Kosba, A. Miller, P. Saxena, E. Shi, E. Gün Sirer, D. Song, and R. Wattenhofer, "On scaling decentralized blockchains," in *Financial Cryptography and Data Security*, J. Clark, S. Meiklejohn, P. Y. Ryan, D. Wallach, M. Brenner, and K. Rohloff, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 106–125.

[15] X. Xu, I. Weber, M. Staples, L. Zhu, J. Bosch, L. Bass, C. Pautasso, and P. Rimba, "A taxonomy of blockchain-based systems for architecture design," in *2017 IEEE International Conference on Software Architecture (ICSA)*, April 2017, pp. 243–252.

[16] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "An overview of blockchain technology: Architecture, consensus, and future trends," in *2017 IEEE International Congress on Big Data (BigData Congress)*, June 2017, pp. 557–564.

[17] Z. Zheng, S. Xie, H.-N. Dai, X. Chen, and H. Wang, "Blockchain challenges and opportunities: A survey," *International Journal of Web and Grid Services*, vol. 14, p. 352, 10 2018.

[18] E. Project. Blockchain app platform. [Online]. Available: https://www.ethereum.org/

[19] Multichain. Open platform for blockchain applications. [Online]. Available: www.multichain.com

[20] Rockwell, "Bitcongress process for block voting and law," 2019.

[21] F. Hao, R. P. Y. A., and P. Zielinski, "Anonymous voting by two-round public discussion," *IET Information Security*, vol. 4, no. 2, pp. 62–67, 2010.

[22] K. Dalia, R. Ben, P. Y. A, and H. Feng, "A fair and robust voting system by broadcast," in *5th International Conference on E-voting*, 2012.

[23] S. Shahandashti and F. Hao, "Dre-ip: A verifiable e-voting scheme without tallying authorities," vol. 9879, 09 2016, pp. 223–240.

[24] D. Chaum, A. Essex, R. Carback, J. Clark, S. Popoveniuc, A. Sherman, and P. Vora, "Scantegrity: End-to-end voter-verifiable optical- scan voting," *IEEE Security Privacy*, vol. 6, no. 3, pp. 40–46, 2008.

[25] D. Chaum, "Secret-ballot receipts: True voter-verifiable elections," *IEEE Security Privacy*, vol. 2, no. 1, pp. 38–47, 2004.

[26] ——, "Untraceable electronic mail, return addresses, and digital pseudonyms," *COMMUNICATIONS OF THE ACM*, vol. 24, 1981.

[27] R. Gor and V. Teague, "Submission to parliament of victoria electoral matters committee inquiry into electronic voting," Tech. Rep., 2016, https://bit.ly/2IpUSmP.

[28] R. Pearce, "Nsws evoting system under fire," 2016.

[29] E. M. of Foreign Affairs, "Estonian internet voting system," 2019.

[30] H. Baldersheim and J. Saglie, "Internet voting in norway 2011: Democratic and organisational experiences," in *The 4th International Conference on Democracy as Idea and Practice*, 2013.

[31] A. Barnes, C. Brake, and T. Perry, "Digital voting with the use of blockchain technology," 2016, the Economist Competition on Blockchain based e-voting. [Online]. Available: https://www.economist.com/sites/default/files/plymouth.pdf

[32] P. McCorry, S. Shahandashti, and F. Hao, *A Smart Contract for Boardroom Voting with Maximum Voter Privacy*, 01 2017, pp. 357–375.

[33] K. M. Khan, J. Arshad, and M. M. Khan, "Secure digital voting system based on blockchain technology," *Int. J. Electron. Gov. Res.*, vol. 14, no. 1, pp. 53–62, Jan. 2018. [Online]. Available: https://doi.org/10.4018/IJEGR.2018010103

[34] P. Ryan, "Pret a voter with paillier encryption," *Mathematical and Computer Modelling*, vol. 48, no. 9, pp. 1646 – 1662, 2008, mathematical Modeling of Voting Systems and Elections: Theory and Applications. [Online]. Available: http://tiny.cc/4eac5y

[35] G. Karame, "On the security and scalability of bitcoin's blockchain," in *The 2016 ACM SIGSAC Conference*, 10 2016, pp. 1861–1862.

[36] SegWit, 2019. [Online]. Available: https://segwit.org/

[37] M. Vukolić, "The quest for scalable blockchain fabric: Proof-of-work vs. bft replication," in *Open Problems in Network Security*, J. Camenisch and D. Kesdoğan, Eds. Cham: Springer International Publishing, 2016, pp. 112–125.

[38] M. A.-B. Shehar Bano and G. Danezis, "The road to scalable blockchain designs," *;Login:*, vol. 42, no. 4, pp. 31–42, 2017.

[39] I. Eyal, A. E. Gencer, E. G. Sirer, and R. V. Renesse, "Bitcoin-ng: A scalable blockchain protocol," in *13th USENIX Symposium on Networked Systems Design and Implementation (NSDI 16)*. Santa Clara, CA: USENIX Association, 2016, pp. 45–59. [Online]. Available: https://www.usenix.org/conference/nsdi16/technical-sessions/presentation/eyal

[40] S. Kim, Y. Kwon, and S. Cho, "A survey of scalability solutions on blockchain," 10 2018, pp. 1204–1207.

[41] J. Rubin and M. Naik, "Merkelized abstract syntax trees," 2014.

[42] M. SCHERER, "Performance and scalability of blockchain networks and smart contracts," Masters thesis, Umee University, Department of Computing Science, 2017.

[43] M. C. T. Size, 2019. [Online]. Available: https://www.multichain.com/qa/6551/how-to-calculate-transaction-size