Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 167 (2020) 2506–2515

Procedia Computer Science

www.elsevier.com/locate/procedia

International Conference on Computational Intelligence and Data Science (ICCIDS 2019)

Blockchain Ethereum Clients Performance Analysis Considering E-Voting Application

Praveen M Dhulavvagol [a,*], Vijayakumar H Bhajantri [a], S G Totad [a]

[a,] *School of Computer Science and Engineering. KLE Technological University, Hubballi, 580031, India*

Abstract

Blockchain technology is evolving and revolutionizing the IT industry with better security, efficiency, and resilience. Blockchain technology is being used in many applications majorly in cryptocurrencies and bitcoin applications. Verified transactions which make a block and group of such transactions or blocks are immutable making the blockchain more secured and reliable. Blockchain achieves decentralization of power, trust, and secured of being hacked, which solves major problems or issues with the current systems. Ethereum, the most widely used blockchain platform because of its unlimited block size. Many complex problems with smart contracts can be implemented with Ethereum and the eradication of third party organizations interfering in transactions helps solving the issues of financial crisis and it is easy to implement compared to other blockchain technologies. There are certain limitations/issues in processing large number of transactions due to lack of speed in processing the transactions. Ethereum Blockchain code will be executed by different clients with varying speed and the performance level will be different. The goal of this paper is to understand Ethereum transactions and perform the comparative analysis of Geth and Parity ethereum clients on the private blockchain. In this paper, a private blockchain network is setup where the nodes will share the data among peer nodes or blocks within the network. Using this network setup a democracy voting application is developed which makes use of the blockchain to store and process the data, smart contracts are deployed to execute the transactions. Performance analysis of the two most popular ethereum clients Geth and Parity is carried out considering time, consistency and scalability parameters. Results interpret that the overall transactions are 91% on average faster in parity client as compared to Geth client.

Keywords: Proof-of-work; Proof-of-stake; Ethereum; consensus ; Geth; Parity; Blockchain;

1. Introduction

Blockchain is a decentralized system where all the participating parties are peers nodes, the transactions of these nodes will be carried out without any third party allies. Blockchain uses consensus algorithms like proof of stake (PoS) and proof of work (PoW) to validate the transactions depending upon the agreement made by all peers.

* Corresponding author: Praveen M Dhulavvagol. Tel.: +91-9964266154
E-mail address: praveen.md@kletech.ac.in

1877-0509
10.1016/j.procs.2020.03.303

The PoS and PoW protocol provide more secure and reliable transaction operation between the peer nodes in the blockchain network. With the conventional single node data processing operation the owner entity needs to maintain all the copy of data or transactions. Consequently, the owner entity controls the data transfer and management activity without intervention of other client nodes. With the advancement of distributed ledger technology the conventional approach has radically changed the data processing operations in distributed architecture where multiple nodes will have one copy of the data and these nodes are individually allowed to contribute in the data processing operation. In distributed architecture the main challenge is to ensure that all the nodes agree on the common rule or truth i.e. the validation of ledger. Any changes done by single node need to be broadcasted to all the peer nodes of the network. This mechanism where all the peer nodes of the blockchain network will be arriving to the common truth this process is known as consensus.

The openness and immutability properties of blockchain network make it highly decentralized and scalable. Therefore blockchain can be used to store sensitive and confidential information. Blockchain technology has evolved rapidly over the past few years and many new and innovative solutions have been proposed to solve large problems, one of the largest problems a blockchain technology face is scalability. With every node being privy to every transaction, there is a natural limit to transaction processing. Lots of temporary solutions such as increasing the block size seem obvious, but are not sustainable for long term network demands [2].

The main objectives addressed in this paper are:

- Set up of Ethereum multi-node blockchain in a private network
- Implementation of E- Voting application using smart contract.
- Perform transaction operation read, write on multiple nodes.
- Perform the comparative analysis of Parity and Geth ethereum clients considering consistency, time and scalability parameters.

1.1. Public vs. Private Blockchain Network

Public blockchain network consists of ‘N’ number of multiple nodes, any user can join the network and perform transaction operations and can participate in consensus process, so this type of network is completely trust-less. Public blockchain network are completely decentralized and nodes must be properly synchronized and if the block chain is very big then it takes huge amount of time and energy to perform the operations [4]. In case of private blockchain user or nodes need to take permission to join the blockchain network then only the nodes can read the current state of blockchain. Compared to public blockchain private blockchain are much faster, safe and efficient, in this case all the permissions are carried out centralized so there is no decentralization [5].

1.2. E-Voting

E-Voting mechanism mainly involves two functionalities casting and counting of votes which are most widely discusses in both academic and commercial world. In order to have secure e-voting then the following properties should be considered and must hold good [4].

- Fairness: Voting results should not be declared in prior without the completion of voting process. This will assure that the remaining voters will not be influenced to vote.
- Eligibility: Ensure that only eligible voters must be allowed to cast their vote.
- Privacy: Once the voter cast his vote, the details of the casted vote should not be revealed to any other users.
- Verifiability: This property helps the voters to verify whether the casted vote has been counted or not. There are two types of verifiability: individual and universal verifiability. Individual verifiability checks weather the casted votes of the individuals are selected for counting process or not. And in case of universal verifiability any user can verify the election outcomes once it is published.
- Forgiveness: Ability of the voter to alter or modify voters vote after the casting.

Table 1. Overview of commercial blockchain e-voting protocols

	Protocols		
Properties	Bit congress	Follow my vote	TIVI
Fairness	NO	NO	NO
Eligibility	NO	YES	YES(Unclear how)
Privacy	YES	YES	YES
Individual-Verifiability	YES	YES	YES
Universal-Verifiability	YES	YES	YES
Forgiveness	NO	YES(Unclear how)	YES(Unclear how)

The table 1 provides an overview of the most popular commercial remote e-voting protocols like bitcongress, follow my vote and TIVI. However most of these protocols lack documentation and the internal workings [6].

In summary, Major contributions of the proposed work are:

- Design and development of democracy contract application using ethereum blockchain technology which accomplishes all the functional requirements of fundamental E-voting, Providing decentralization and maximum control of the entire process to the users or voters.
- Understanding the implementation challenges and different platforms and its advantages and limitations.
- Performance analysis of Geth and Parity ethereum clients.

The section 2 provides details of related work and basic blockchain architecture. Section 3 describes the proposed methodology, set-up of private blockchain and Ethereum Geth & Parity client details. Section 4 deals with the implementation details of the proposed work with process flow diagrams and algorithms. Section 5 presents the results and discussion along with the Performance Analysis of Geth and Parity Clients. Section 6 describes the conclusion and the future enhancement of the proposed work.

2. Related Work

Blockchain is the technology that can change the perspectives of the people in the current era, it can make high impact on the current traditional way of transactions, investments, security, scalability and the efficient use of the resources would be possible with this technology. The blockchain is made up of distributed ledger technology where each and every user who is part of the blockchain can manage the ledger according to the transactions that takes place .There are mainly three types of blockchain public, private and consortium blockchain. In public the ledger is visible to all the users of the blockchain where as in private only specific people in the organization are allowed to add the transactions. Consortium blockchain is such where group of organizations are allowed to verify and add transactions but the ledger can be open or restricted to some groups [3].

A blockchain consists of blocks, a block is a recent transaction that had happened and verified with a hash code so to maintain the consistency in uniqueness, such group of transactions make blocks and they linked such that they are immutable. Blockchain is immutable because of majorly two reasons, all the transaction are stored in blocks using a hash key , the hash key links the previous block to the next block. If the data is tampered then the hash key value will be different and the peer nodes will be able to recognize the data tampering operation. Blockchain maintains a public ledger which is shared among all the nodes of the network. Any changes made in ledger will be updated and synchronized with all peer nodes of the network. To hamper a transaction one has to tackle all the blocks of the blockchain network which require unimaginable amount of resources to do so [8].

Blockchain technology has huge advancement in the field of research, educational and developing by making use of the platform to innovate, adapt and make a world better place out of it. The workflow of blockchain architecture is shown in fig 1.

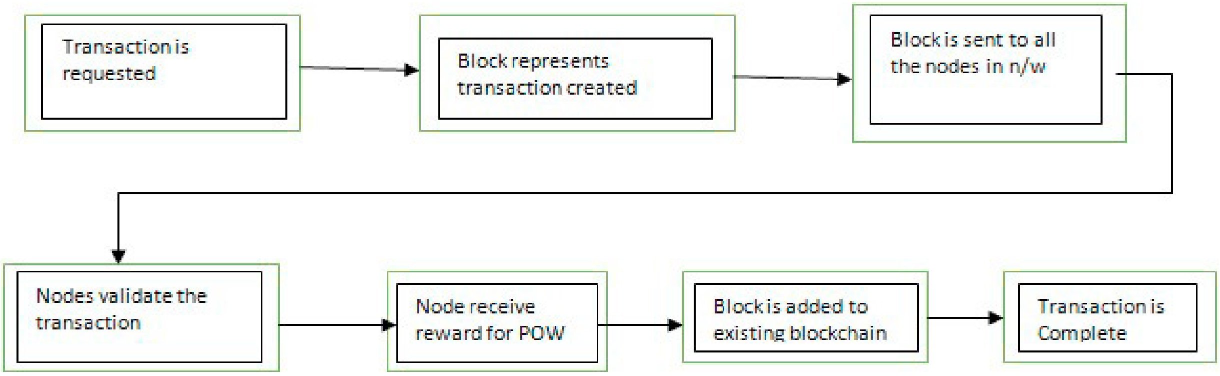

Fig 1. Work flow of blockchain architecture

Blockchain technology provides several platforms such as ethereum, hyperlredger, bitcoin, multichain and so on. Hyperledger, ethereum are most widely used because of open source and it supports for different use cases. The fig 1 shows the blockchain architecture working process in the form of a digital wallet. A block in blockchain mainly consists of data, the hash value of block and also the hash value of previous block. Data stored in each block depends on the type of blockchain technology. For example in Bitcoin, the block stores data about the sender, amount of coins and the receiver information [1, 2]. The hash key is generated using hashing algorithms (SHA 256) this hash key helps in easily identifying each block in the blockchain structure. Once a block is created a hash key is assigned to it, any changes done for block will intern effect the hash value. The final and important component within the block is the hash value of previous block, using this value we build a chain of blocks and it will be the main element for blockchain architecture security. If any block gets corrupted and attempts to provoke the blocks to change then all the other blocks which are connected to a chain will carry incorrect information and whole blockchain system will be invalid. It is also possible to adjust all the blocks using proof-of-work protocol. This will allow user to slow down the new block creation process .In bitcoin it approximately takes 10 min to find the required proof-of-work and to add new block to the chain this task is performed by miners. Miners need to store the transaction fees information received from the block which they have verified as reward. A new user node joining the peer to peer network will get the full information of the system. If any new block is created that it is being sent to all the nodes in the blockchain system, each node verify and validates the information. If the information is correct then the block is added to the local blockchain in each node. All nodes which are within the blockchain architecture creates a consensus protocol which holds the set of rules information , if all the nodes agree to this then all the blocks will be secured[4].

3. Proposed Methodology

Democracy contract mainly consists of owner who will be acting as administrator or owner. The owner can add or delete voting members to the organization. All the members who are participating in democracy contract can prepare a proposal, the proposal will be in the form of ethereum transaction. Any member of the organization can broadcast or execute the contract. The other members can cast their vote in support or against the proposal. If all the predetermined members have casted their vote then the proposal can be executed. The contract will verify and count the casted votes and if the numbers of votes are more, then the contract will execute the given transaction.
The functional requirements of the proposed system are as follows:

- Owner shall be able to initialize minimum quorum for proposals, and the debate time and margin of votes for majority.
- Owner shall be able to add members eligible for quoting proposals and voting.
- Owner shall be able to transfer ownership of the contracts.
- Member shall be able to quote proposal and cast vote for or against the proposal.
- Finally the system should be able to execute the proposal.

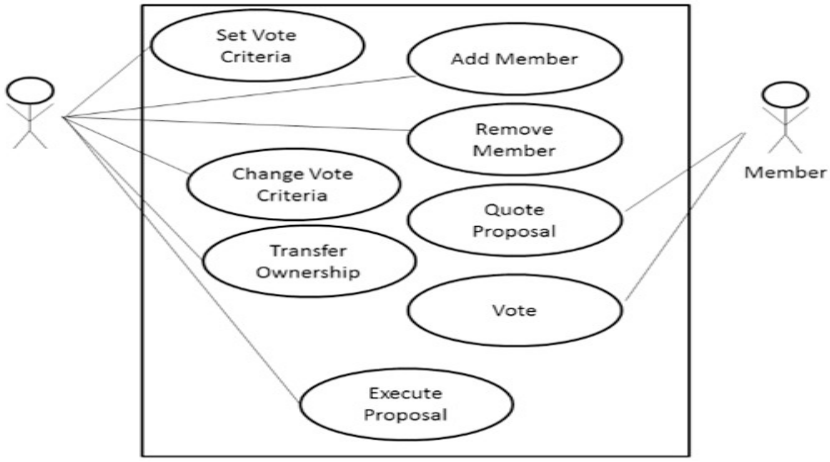

Fig 2. Democracy contract use case diagram

The fig 3 shows the core components of Blockchain architecture and its working mechanism. The main core components of blockchain architecture are:

- Node: Computer or user is considered as node maintains the independent copy of digital ledger.
- Transaction: A small unit of building block which used in the blockchain architecture.
- Block: A block is a data structure used for maintaining the transactions which are broadcasted to all the nodes of network.
- Chain: A sequence of blocks.

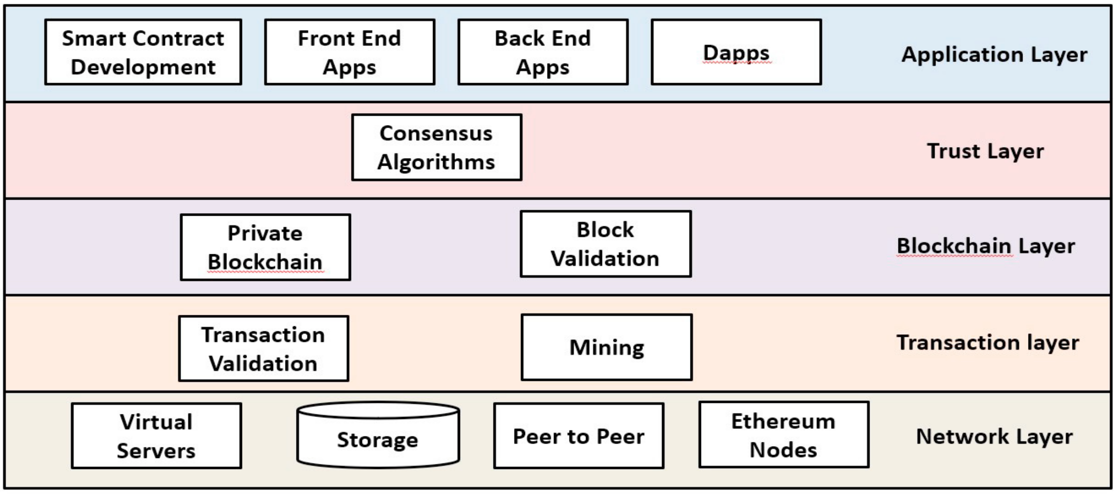

Fig.3. Blockchain layered architecture

3.1. Proposed System Model (Ethereum- Private Blockchain)

The fig 4 shows the overall system model of the application, it mainly consists of Remix IDE, EVM's and Web3.js components. Remix is an IDE and browser based complier which is used to build and debug Ethereum contracts using solidity language. Web3.js mainly used to interact with the local or remote nodes using RPC port through HTTP protocol also it consists of collection of library flies. EVM is a decentralized virtual machine; these virtual machines can execute the scripts using interconnection of public nodes.

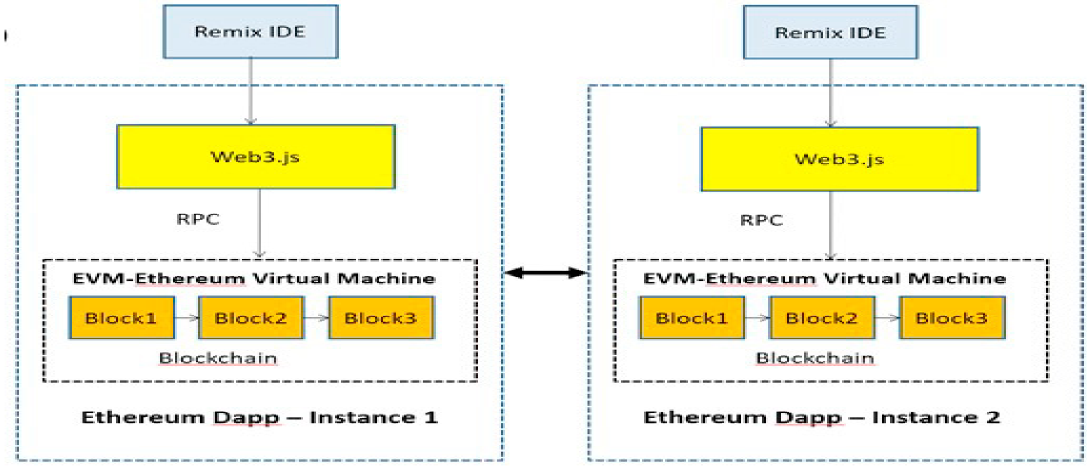

Fig 4. System model diagram

3.2. Ethereum Clients

In blockchain architecture if any new node need to be added to the peer-to-peer network, then the first step is to run Ethereum client, this code will help to parse and verify blockchain transactions. Ethereum provides many clients which are developed by different programming language. The two most widely used clients are Geth and Parity. Geth is Ethereum first client and it provides mining option by its own. On the other hand parity is Ethereum latest developed client which is basically developed to enhance the efficiency [5]. It provides faster sync operations, it doesn't have mining option same as Geth but it provides JSON-RPC support so that it can integrate with any of the miner which uses PoW (Proof of Work). The table 2 shows the Geth & Parity clients.

Table 2: Ethereum Geth & Parity client features

	Programming	**Database**	**State**	**JavaScript console**	**Mining**
Geth	Go-Lang	Level DB	Heap	Yes	Yes
Parity	Rust	Rock sdb	Stack	Not directly but can run Geth attach or it can use Node.js CLI console	No

4. Implementation

The fig 5 shows the process flow diagram of democracy system, which mainly consists of three variables minimum quorum for proposal, minutes for debate and margin of votes for majority. Values of the variables are set initially before voting. Add Member function is used to add members for voting for any proposal as well as quote proposals. When any proposal is proposed by a member, the other members of the network are given rights to either vote for or against the proposal. After certain fixed time, the numbers of votes are counted. Based on the number of votes, the proposal is either accepted or rejected. The accepted proposal is executed and added in the blockhe democracy application mainly consists of three contracts (classes): owned, token Recipient and Election. The contract Election inherits properties of contracts owned and token Recipient. Each of these contracts has a set of functions and different properties.

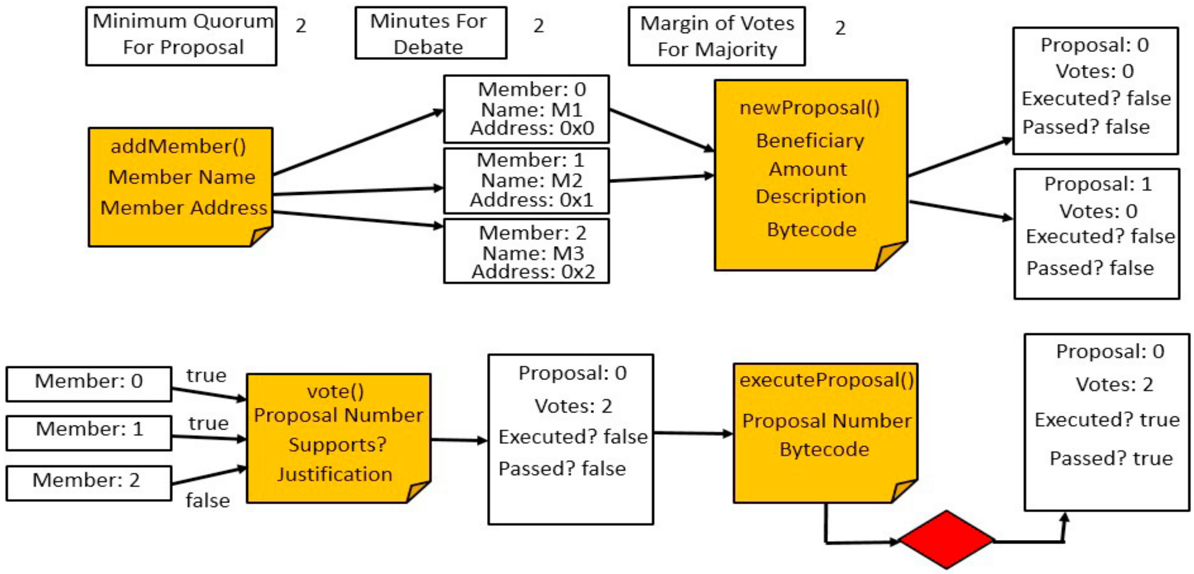

Fig. 5 Process flow diagram of democracy contract system

4.1. Setting up parameters for initial block

Algorithm/Code Snippet

```
{
  "nonce": "0x0000000000000042",
  "timestamp": "0x00",
  "parentHash": "0x0000000000000000000000000000000000000000000000000000000000000000",
  "extraData": "0x00",
  "gasLimit": "0x8000000",
  "difficulty": "0x400",
  "mixhash": "0x0000000000000000000000000000000000000000000000000000000000000000",
  "coinbase": "0x3333333333333333333333333333333333333333",
  "alloc": {},
  "config": {}
}
```

4.2. Contract owned

Algorithm/Code Snippet

Step 1: Constructor ()
Step 2: Set owner to address of user who deploys the contract
Step 3: Modifier only Owner ()
Step 4: Require that user using this contract is owner of contract
Step 5: Function transfer Ownership (new Owner) only Owner
Step 6: Assign owner to new Owner

4.3. Contract token recipient and contract election

Algorithm/Code Snippet

Constructor (minimum Quorum, minutes for debate, margin of votes for majority)
Call
Change Voting Rules (minimum Quorum, minutes for debate, margin of votes for majority)
Call addMember (address (0),””)

```
Call addMember (owner,'founder')
Function addMember (targetMember, memberName) OnlyOwner
    Add targetMember to members array with memberName
Function removeMember (targetMember)
    OnlyOwner
Remove targetMember from members array
Function change Voting Rules (minimum Quorum, minutes for debate, margin of votes for majority) only Owner
    Assign values for minimum Quorum, minutes for debate, margin of votes for majority
Function vote (proposalNumber, supportProposal, justificationText)
     If (msg.sender has not voted)
          Set voted to true for msg.sender
          Increment the number of votes
Function executeProposal (proposalNumber, transactionByteCode)
 If (proposal not executed && time>minExecutionDate &&numberofvotes>=   MinimumQuorumForProposals)
     Set execute of proposal to true
       Else
     Set executed of proposal to false
```

5. Results and Discussions

The experiment and performance analysis is carried out using Go-ethereum, Geth v1.4.18 with the parity release of version v1.6.0. Initially a private testnet for ethereum Geth and parity need to be setup by defining a genesis block and adding the peer nodes to the miner network. In case of ethereum manually need to set the difficulty variable in genesis block so that the miners do not diverge in large networks. Next for parity we need to initialize the step-duration variable to 1. The confirmation-length variable is set to 5 seconds in both ethereum and parity clients. The experiments were carried out on a 10-node commodity cluster. Each node has 1.5GHz CPU, 8GB RAM, 1TB hard drive, with Ubuntu 16.04, the nodes were connected via 1GB switch.
Performance metrics considered to evaluate the blockchain networks:

- Throughput: The number of transactions executed per second considering the workload of the multiple clients and threads per client of the blockchain network.
- Latency: The delay in executing the transactions successfully i.e. the response time.
- Scalability: Able to increase the number of concurrent nodes and workloads in the blockchain networks. Scalability is an important factor which will affect the throughput and latency.

5.1. Block creation vs Time

All major processes in blockchain voting are faster compared to traditional voting system and are within 13 second duration as per the literature study. Same number of blocks are created for transactions in each vital process meaning the system is consistent since the system supports multi node structure(Tested upto 10 systems) across multiple systems that means the system is highly scalable. Hashed encryption and decentralised storage makes data tampering difficult and hence increasing security of the system

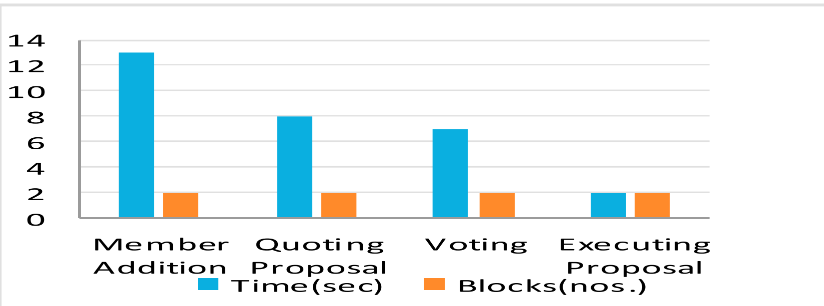

Fig.6 Number of blocks vs. time

6. Performance Analysis of Geth and Parity Clients

The table 3 interprets the performance analysis of Geth and Parity considering the speed with number of transactions executed by the system.

Table 3: Comparative analysis of Geth and Parity clients considering increasing speed

Num of Transactions	Geth	Parity	% of increasing speed in parity
1000	3.14	1.84	85.1
2000	6.00	3.42	88.14
3000	10.01	6.12	84.92
4000	12.15	7.21	89.96
5000	15.86	9.47	90.12

The Fig.7 shows the processing time considered by both clients Geth and Parity. From table 3 and fig 5 we can interpret that parity client is 85% faster in executing the transactions as compared to Geth with transactions ranging from 1000 to 5000.

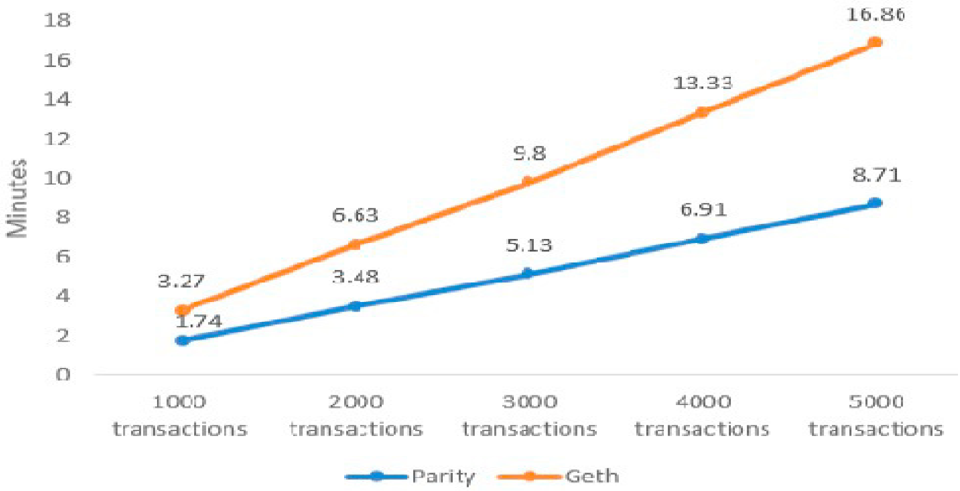

Fig 7. Geth & Parity clients processing time

7. Conclusion

In this paper, a proof-of-concept application is developed to improvise the E-Voting process in a completely decentralised manner using a blockchain technology. Smart contracts were implemented in order to achieve peer to peer network security using a solidity programming language. The implementation of consensus algorithm used in the blockchain helped to determine the effect of smart contract. Performance analysis of the two most popular ethereum clients Geth and Parity is carried out considering time, consistency and scalability parameters. Results interpret that the overall transactions are 91% on average faster in parity client as compared to Geth client. Apart from performance the other two important measures are security and safety are the challenging issues in blockchain system. There is a large scope for research with distributed processing based on blockchain technology, specifically with proof-of-work and proof-of-stake protocols in developing IoT based applications.

References

[1] D. Vujicic, D. Jagodic and S. Randic, "Blockchain technology, bitcoin, and Ethereum: A brief overview," 2018 17th International Symposium INFOTEH- JAHORINA (INFOTEH), East Sarajevo, 2018, pp. 1-6. doi: 10.1109/INFOTEH.2018.8345547

[2] Vishnu Prasad Ranganthan, Ram Dantu, Aditya Paul, Paula Mears, Kirill Morozov, "A Decentralized Marketplace Application on the Ethereum Blockchain", Collaboration and Internet Computing (CIC) 2018 IEEE 4th International Conference on, pp. 90-97, 2018.

[3] R. Skudnov, "Bitcoin Clients," Turku University of Applied Sciences, Turku, 2012.

[4] Maximilian Wöhrer, Uwe Zdun, "Smart Contracts: Security Patterns in the Ethereum Ecosystem and Solidity", University of Vienna. Wikipedia, "List of controversial elections," 20 September 2016. [Online]. Available: https://en.wikipedia.org/wiki/List_of_controversial_elections. [Accessed 27 September 2016].
[5] P. Noizat, "Blockchain Electronic Vote," in handbook of digital Currency, Paris, Elsevier Inc., 2015, pp. 453-461.
[6] N. N. Pokrovskaia, "Tax, financial and social regulatory mechanisms within the knowledge-driven economy. Blockchain algorithms and fog computing for the efficient regulation," 2017 XX IEEE International Conference on Soft Computing and Measurements (SCM), St. Petersburg, 2017, pp. 709-712.
[7] Zhentian Liu, Jing Liu, "Formal Verification of Blockchain Smart Contract Based on Colored Petri Net Models", Computer Software and Applications Conference (COMPSAC) 2019 IEEE 43rd Annual, vol. 2, pp. 555-560, 2019
[8] Ittai Abraham, Dahlia Malkhi, Kartik Nayak, Ling Ren, and Alexander Spiegelman. Solida: A blockchain protocol based on recon_gurable byzantine consensus. In Proceedings of the 21st International Conference on Principles of Distributed Systems, OPODIS '17, Lisboa, Portugal, 2017.
[9] SG Totad, RB Geeta, CR Prasanna, NK Santhosh, PV Reddy ." Scaling data mining algorithms to large and distributed datasets" International Journal of Database Management Systems (IJDMS), Vol.2, No.4, November 2010.
[10] Koc et al., "Towards Secure E-Voting Using Ethereum Blockchain", 2018 IEEE 6th International Symposium on Digital Forensic and Security.
[11] Suporn Pongnumkul, Chaiyaphum Siripanpornchana, Suttipong Thajchayapong, "Performance Analysis of Private Blockchain", 26th International Conference on Computer Communication and Networks (ICCCN), 2017.
[12] Yue Hao, Yi Li, Xinghua Dong, Li Fang, Ping Chen, "Performance Analysis of Consensus Algorithm in Private Blockchain", Intelligent Vehicles Symposium (IV) 2018 IEEE, pp. 280-285, 2018.
[13] Dhulavvagol P.M., Totad S.G., Sourabh S. (2019) Performance Analysis of Job Scheduling Algorithms on Hadoop Multi-cluster Environment. In: Sridhar V., Padma M., Rao K. (eds) Emerging Research in Electronics, Computer Science and Technology. Lecture Notes in Electrical Engineering, vol 545. Springer, Singapore
[14] T. T. A. Dinh, J. Wang, G. Chen, R. Liu, B. C. Ooi, K.-L. Tan, Blockbench: A framework for analyzing private blockchains
[15] Ethereum project, [online] Available: https://Ethereum.org/.
[16] Weichao Gao, William G. Hatcher, Wei Yu, "A Survey of Blockchain: Techniques Applications and Challenges", Computer Communication and Networks (ICCCN) 2018 27th International Conference on, pp. 1-11, 2018.
[17] T. T. A. Dinh, J. Wang, G. Chen, R. Liu, B. C. Ooi, K.-L. Tan, Blockbench: A framework for analyzing private blockchains.
[18] V. Buterin, Ethereum white paper: a next generation smart contract & decentralized application platform, 2013, [online]Available:http://www.the-blockchain.com/docs/Ethereumwhitepaper a_nextgeneration_smart_contract and decentralized application platform-vitalik-buterin.ndf.
[19] Niranjan C Kundur1, Praveen M Dhulavvagol2, Prasad." Recommendation System Based on Content Filtering for Specific Commodity" International Journal of Latest Technology in Engineering, Management & Applied Science (IJLTEMAS)Volume V, Issue VII, July 2016 | ISSN 2278-2540
[20] E. Kokoris-Kogias, P. Jovanovic, N. Gailly, I. Kho_,L. Gasser, and B. Ford. Enhancing bitcoin security and performance with strong consistency via collective signing. In USENIX Security, 2016.
[21] Dhulavvagol P.M., Shet R., Nashipudi P., Meti A.S., Ganiger R. (2018) Smart Helmet with Cloud GPS GSM Technology for Accident and Alcohol Detection. In: Nagabhushan T., Aradhya V., Jagadeesh P., Shukla S., M.L. C. (eds) Cognitive Computing and Information Processing. CCIP 2017. Communications in Computer and Information Science, vol 801. Springer, Singapore
[22] L. Luu, V. Narayanan, C. Zhang, K. Baweija, S. Gilbert, and P. Saxena. A secure sharding protocol for open block chains. In CCS, 2016.
[23] K. Croman, C. Decker, I. Eyal, A. E. Gencer, A. Juels,A. Kosba, A. Miller, P. Saxena, E. Shi, and E. G un. On scaling decentralized blockchains. In Proc. 3rd Workshop on Bitcoin and Blockchain Research, 2016.
[24] J. Bonneau, A. Miller, J. Clark, A. Narayanan, J. A.Kroll, and E. W. Felten. Sok: Research perspectives and challenges for bitcoin and crypto-currencies. In 2015 IEEE Symposium on Security and Privacy, pages 104{121. IEEE, 2015.
[25] G. Zyskind, O. Nathan, and A. S. Pentland, ``Decentralizing privacy: Using blockchain to protect personal data," in Proc. Secur. Privacy Work- shops (SPW), May 2015, pp. 180_184.
[26] Dhulavvagol P.M., Ankita K.R., Sohan G., Ganiger R. (2018) An Enhanced Water Pipeline Monitoring System in Remote Areas Using Flow Rate and Vibration Sensors. In: Nagabhushan T., Aradhya V., Jagadeesh P., Shukla S., M.L. C. (eds) Cognitive Computing and Information Processing. CCIP 2017. Communications in Computer and Information Science, vol 801. Springer, Singapore
[27] Bajpai, P. 2017. Bitcoin vs Ethereum Driven by Different Purposes. Investopedia. Accessed 15 November 2017 https://www.investopedia.com/articles/investing/031416/bitcoin-vs-ethereum-driven-different-purposes.asp.
[28] E. Kokoris-Kogias, P. Jovanovic, N. Gailly, I. Khoffi, L. Gasser, and B. Ford. Enhancing Bitcoin Security and Performance with Strong Consistency via Collective Signing. In Proceedings of the 25th USENIX Conference on Security Symposium, 2016.
[29] R. Pass, C. Tech, and L. Seeman. Analysis of the Blockchain Protocol in Asynchronous Networks. Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT), 2017.
[30] A. Kiayias, A. Russell, B. David, and R. Oliynykov. Ouroboros: A Provably Secure Proof-of-Stake Blockchain Protocol. Cryptology ePrint Archive, Report 2016/889, 2016
[31] Dhulavvagol, Praveen & Desai, Abhilash & Ganiger, Renuka. (2017). Vehical Tracking and Speed Estimation of Moving Vehicles for Traffic Surveillance Applications. 373-377. 10.1109/CTCEEC.2017.8455043.
[32] Zhang Peng, Walker Michael, White Jules, C. Schmidt Douglas, "Metrics for Assessing Blockchain-based Healthcare Decentralized Apps", IEEE, 2018.
[33] Joshi Y., Totad S.G., Geeta R.B., Prasad Reddy P.V.G.D. (2018) Mobile Agent-Based Frequent Pattern Mining for Distributed Databases. In: Bhalla S., Bhateja V., Chandavale A., Hiwale A., Satapathy S. (eds) Intelligent Computing and Information and Communication. Advances in Intelligent Systems and Computing, vol 673. Springer, Singapore