Survey on Private Blockchain Consensus Algorithms

Mr. Sunny Pahlajani
Department of Computer Engineering, College of Engineering.
Pune, India
pahlajanish17.comp@coep.ac.in

Mr. Avinash Kshirsagar
Department of Computer Engineering, College of Engineering.
Pune, India
ark.comp@coep.ac.in

Dr. Vinod Pachghare
Department of Computer Engineering, College of Engineering.
Pune, India
vkp.comp@coep.ac.in

***Abstract*— A blockchain is a distributed ledger of records called as blocks. These blocks are linked using cryptographic hash. Each block contains a hash of the previous block, a timestamp, and transaction data. Consensus layer is the main layer in Blockchain Architecture, in which consensus protocol is configured to decide how new block is added in blockchain. Consensus algorithm solves the problem of trust in blockchain. Consensus algorithms can be classified into two classes. The first class is voting-based consensus, which requires nodes in the blockchain network to broadcast their results of mining a new block or transaction, before appending the block to blockchain. The second class is proof-based consensus, which requires the nodes joining the blockchain network to solve and mathematical puzzle to show that they are more eligible than the others to do the appending or mining work. Performance of blockchain can be increased with the use of suitable consensus algorithm. However, theory and data support for the selecting suitable consensus in private blockchain is very limited. This paper contributes theory and data used for selecting suitable consensus algorithm and would help researchers for further exploring of consensus in private blockchain environment.**

Keywords—Proof-based consensus; Voting-based consensus; Byzantine fault tolerant consensus; Crash fault tolerant consensus.

I. INTRODUCTION

Blockchain is a distributed database existing on multiple nodes of the network at the same time in sync. It is a constantly growing ledger as new 'blocks', are added to it. Each block contains a timestamp and a link to the previous block, so they actually form a chain like structure tracking back to the root i,e. genesis block. No single node maintains the database, rather each node has a copy of database. Old blocks of transactions are preserved and new blocks are added to the ledger irreversibly, making it impossible to manipulate previously recorded transactions thus it leads to immutable ledger.

All blocks are encoded uniquely, so every node can verify the data however just a node who possesses a unique cryptographic key (private) can add another block to a specific chain. As long as you remain the only node who knows the key (private), nobody can control your exchanges. Likewise, cryptography is utilized to ensure synchronization of duplicates of the blockchain on every node in the network.

You can consider blockchain as an advanced restorative record: each record is a block which has a mark expressing the date and time when the record was entered. The medicinal history is critical for conclusion and treatment purposes, so neither the specialist nor the patient ought to have the capacity to alter the records effectively made. In any case, the specialist claims a private key that enables him to make new records, and the patient possesses a public key that enables him to get to the records whenever required. This technique makes the information both open and secure.

In this way, blockchain is autonomous, straightforward, and secure. The upsides of such a distributed ledger are self-evident: being it cost and risk reduction, information security, or exchanges straightforwardness, organisations from most domains can clearly profit by this new innovation.

The thought itself isn't new, however: it was initially illustrated in a 1976 research paper "New Directions in Cryptography" [1], yet for quite a while it was viewed as confused and not explored. Be that as it may, in 2008 an obscure individual or group of individuals known by the pen name Nakamoto presented (Bitcoin) - a less complex usage of blockchain innovation as a digital currency [2]. Bitcoin ended up one of the main computerized monetary standards to utilize distributed innovation to encourage instant payments. On account of blockchain, it likewise turned into the principal computerized cash to tackle the issue of twofold spending. All the more essentially, Bitcoin is run by absolutely nobody and is immutable.

The stability, security, and dispersed nature of Bitcoin made it one of the quickest developing resources: this year alone, its cost has ascended by over 500%. Thus, the developing notoriety of Bitcoin pulled in thoughtfulness regarding its fundamental innovation. Presently many new tech businesses are racing to deliver the Next Big Thing depending on blockchain, and an ever-increasing number of individuals working in finance and different areas start to perceive the down to practical benefits past the Bitcoin publicity. Over worldwide supply chains, financial administrations, medicinal services, governments and numerous different ventures, trailblazers are investigating approaches to utilize blockchain to upset and change conventional plans of action. Numerous industry chiefs have just accomplished noteworthy business benefits, including more prominent straightforwardness, upgraded security, enhanced

detectability, expanded proficiency and speed of exchanges, and diminished expenses.

II. CONSENSUS

Consensus algorithms are a resolution process for a group of nodes, where nodes of the group build and bolster the choice that works best for them. It's a type of decision where nodes need to help the greater part choice, regardless of individual choice. In basic terms, it's only a technique to reach a conclusion inside a group. Envision a group of ten nodes that need to settle on a choice about a task that benefits them all. All of them can propose a thought, however the dominant part will be agreeable to the one that encourages them the most. Others need to manage this choice in any case. Presently envision a similar thing with a large number of nodes. Wouldn't that definitely make it way increasingly troublesome? Consensus calculations don't just concur with the dominant part cast a ballot, however it additionally consents to one that benefits every one of them. It's dependably a success for the system. Blockchain consensus models are strategies to make uniformity and decency in the online world. The consensus frameworks utilized for this understanding is known as a consensus hypothesis. Consensus algorithms are of two types proof-based and voting-based. They are explained in below sections.

2.1 Proof-based Consensus Algorithm

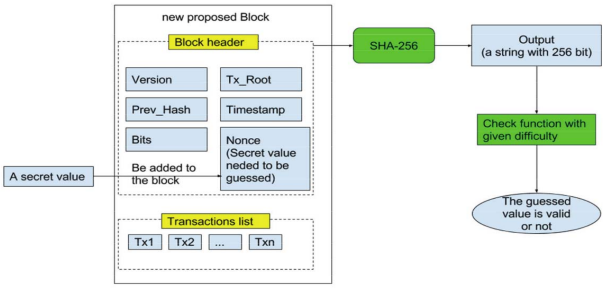
In Proof based consensus algorithm, nodes joining the network need to solve a cryptographic problem to get the right of appending the block. In public blockchain, nodes get rewards after appending the block to the blockchain. The first version of proof-based concerns algorithm is proof of work proposed by Santoshi Nakamoto [2]. Till date there are different versions of proof-based consensus as below-

1. Proof of Work – solving an mathematical puzzle.
2. Proof of Stake – having more stake in blockchain
3. Proof of Elapsed time – some timeout is set with scheduling.
4. Proof of Luck – random selection.
5. Proof of Space – bigger size hard-disk is required.

And their hybrid versions i.e. combination of any of the above consensus types.

1. Proof of Work example

As shown in the below figure a secret value is to be calculated by the node/miner. After guessing a value called nonce, hashing is done with the help of SHA 256 and output is checked with the given difficulty level. If the hash value is accepted then the nonce is also correctly guessed, else the whole process is repeated i.e. guessing another nonce value until the correct one is found. After finding out the correct nonce value the nodes are permitted to broadcast their blocks hash value and the nonce value to other nodes in the network for verifying and appending it to their own ledger. On positive side proof of work is highly secured and on the negative side it requires high computation power. Due to its drawbacks Proof of Stake and many more proof based consensus algorithms were evolved.



Proof of Work [3]

2.2 Voting Consensus Algorithm

Voting consensus algorithm are preferable in private blockchain, where the nodes are known. This is the fundamental distinction contrasted with Proof based consensus calculations, where nodes are regularly allowed to join and pull back from the checking system. In vote base consensus it requires to exchange the results in the network before appending the block to the blockchain. If a peer wants to append a block to its chain, a check is to be made that at least x (x is the threshold set) peers agree on it. If there are f failed nodes, then for a decision f plus one should be operable. So, voting consensus is broadly classified as below.

A. Byzantine – Nodes are crashed and unsettle

B. Crash – Nodes are crashed

III. BYZANTINE CONSENSUS

3.1 Hyperledger

Hyperledger was started in 2016 as a project of Linux Foundation and presently have more than 50 active members. Hyperledger fabric is an application of distributed ledger platform. It is used for running smart contracts, modular architecture which allows for pluggable implementation. Hyperledger fabric protocol that is distributed ledger protocol is run by the peers [5].
There are two types peers in hyper-ledger fabric

- Validating peer - These are the peers that deploy the consensus algorithm and validate the transaction
- Non-validating peer - These peers act as a proxy that helps in connecting clients for validating peers. They can be used to verify the transaction but not for execution.

The validating peers execute a consensus protocol named Byzantine Fault Tolerance (BFT) and there are 3 types of transactions to operate

Deploy transaction - Deploy the smart code on the peers.
Query transaction - Returns the entry of the transaction

Invoke transaction - Executes deployed transaction and returns the results.

For validation of the transactions, where there are X number of nodes, atleast two thirds of the validating nodes should properly execute the chain code deployed on the node for proper results in BFT consensus. For Practical Byzantine Fault Tolerance (PBFT) consensus, smart code transactions should be deterministic else it would lead to inconsistent state of nodes. A simpler implementation is done in SIEVE [4] protocol, where in non-deterministic smart code transactions are filtered out initially.

Features of current fabric release are –

- Private blockchain with deterministic results
- Pluggable consensus protocol
- Certificate authorities for security certificates, transaction certificates, enrolment certificates.
- Node.js SDK to interact with the fabric.

Hyperledger fabric implements private ledger and contains security infrastructure. All the transactions authorisation and authentication are done through private key certificates. Certificate Authority (CA) issues enrolment certificate to the nodes who want to participate in the network. Similarly, transaction certificate is issued for submitting and transaction needs to be availed. More fine-grained security for transactions is planned for next release [6].

3.2. Corda

Corda is a version of blockchain which specifically focuses on finance domain. Corda supports smart contracts which are executed automatically once a criterion is matched and can be controlled as per need of application. Finance data is linked by smart contract to its logic which is legally enforceable.

A digital document agreement between the two parties and its states is called state object in corda. State object is visible only to the involved parties unlike in blockchain where it is public. To ensure this kind of security hashing is used to identify parties and their data. All this state objects are immutable and constitutes a ledger [7].

All the modifications in corda are done with the help of transactions, which expects a state object as input and returns another object as output. Main characteristics of consensus in Corda [7] are –

- Transaction validity
- Transaction uniqueness

Validity check for proper output and the contract code ran successful or not and uniqueness checks that no other same transaction has been executed.

Corda provides tools to achieve consensus, they are listed as follows -

- Smart contract logic to check the valid transactions
- Time-stamping and unique services to sequence the transactions
- A simpler process to write complex protocols between parties

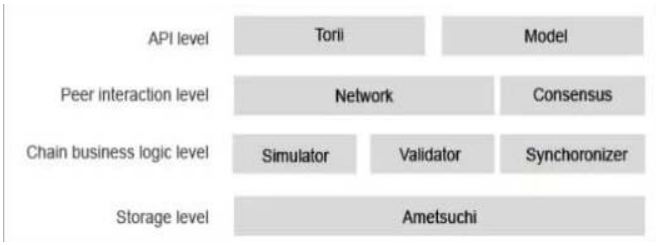
3.3. Iroha with Sumeragi

Hyperledger iroha is a framework hosted by the Linux Foundation. Hyperledger iroha is simple to use, it provides C++ driven special support to mobile application. Iroha is designed in such a way that it can be easily integrated with another distributed ledger technologies. It can be used as a repository of reusable components [8,9].

Goals of Iroha

- C++ environment.
- Mobile and web support.
- API support and integration with other frameworks.

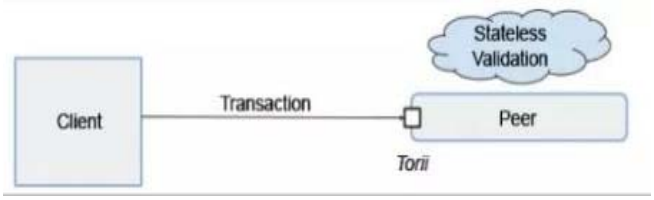
Below is the hyperledger iroha architecture [8] with sumeragi consensus algorithm. Features of hyperledger iroha is its interoperability, an open source library which fulfils the needs of developers and demands of business. The architecture is a layered view of different components starting from API, peer interaction, chain business logic and storage.



Hyperledger Iroha Architecture [19]

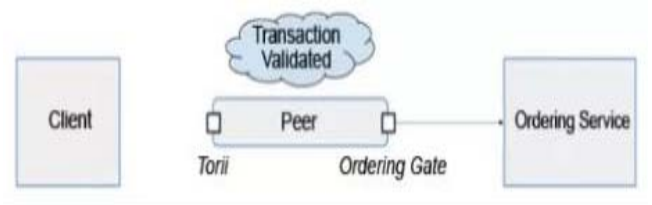
Steps in Transaction flow of a hyperledger iroha are as follows –

Torii Gate accepts the transactions composed by client, in turn it forwards the transaction to the specific peer for stateless validation.



Transaction flow: Step 1 [19]

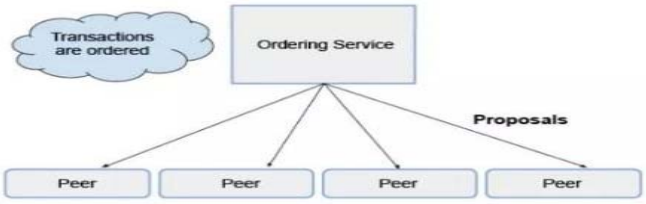
Transaction is first routed to the ordering gate after validation. Ordering Gate creates a control plan for connection to ordering service.



Transaction flow: Step 2 [19]

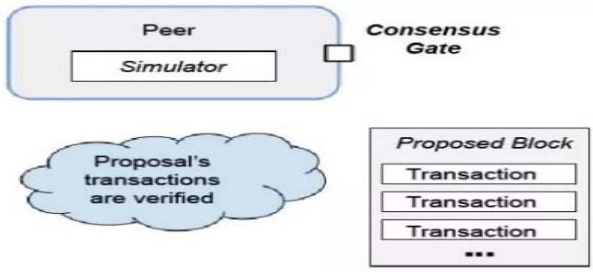
Ordering service performs 1.Transactions sequencing 2.Routing transactions to the peers as a proposals. Ordering service routes a unsigned block known as proposal as shown

in below. Proposals are routed only when certain amount of transactions are collected or after certain time out, due to which sharing of empty proposals are prohibited.



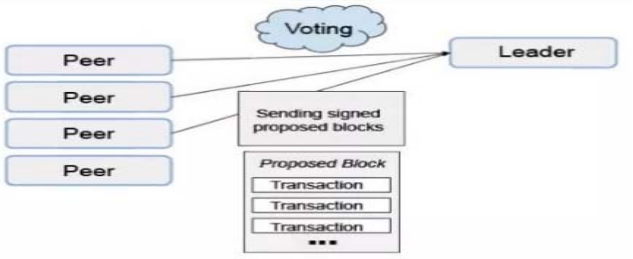
Transaction flow: Step 3 [19]

In simulator block stateful validation is performed by nodes and compose a block of verified transactions, which is then routed to consensus gate, for YAC execution.



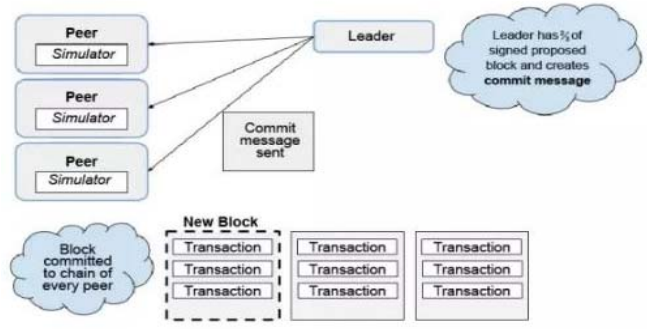
Transaction flow: Step 4 [19]

A leader is elected from the set of peers using Yet Another Consensus (YAC) consensus logic, then vote is casted by nodes by forwarding its proposed block to leader along with its signature.



Transaction flow: Step 5 [19]

A block is committed only when leader receives majority signed proposed blocks (threshold set, here its 2/3rd) from nodes, then leader broadcast commit message suggesting that block should be committed as next block in the chain of each node.



Transaction flow: Step 6 [19]

3.4. Ripple

Ripple consensus algorithm was developed to solve three main problems that distributed payment system face, they are correctness, agreement and utility. There are multiple consensus available in the market to address byzantine generals problem but none could solve the above three problems stated [10].

Ripple protocol components

1. Server - It runs ripple server software and helps in consensus process.
2. Ledger - Ledger is used for book-keeping of transactions and gets updated as and when transactions those are successfully committed.
3. Last Closed Ledger - Last successfully modified copy of the ledger.
4. Open Ledger - Local copy of a node containing transactions that are not gone through the process of consensus, once they are gone through the consensus process the open ledger would turn into the last closed ledger.
5. Proposer - multiple servers proposed transactions to be included in the consensus round, but only the proposals submitted by the servers UNL list would be accepted.
6. Unique Node List (UNL) - Each server has a unique node list of servers which helps in the process of consensus.

Ripple consensus algorithm runs every specific interval on all the nodes. On successful completion of consensus open ledger becomes the closed ledger and is being maintained by all the nodes in the network.

Ripple consensus algorithm run in rounds [11, 12]

1. All the valid transactions are taken up by the servers that were not previously committed in the close ledger as candidate set.
2. All the servers then combine the candidate sets of the servers on its UNL and starts voting.
3. Transactions receiving more than minimum percentage of votes is moved to the next round rest are removed or start back from 1st round.
4. Consensus need to be met in the final round 80% of the UNL, leading the transactions from open to last close ledger.

With low latency ripple consensus algorithm has done a major contribution in distributed payment systems that were impractical with the earlier consensus algorithm.

3.5. Stellar

Stellar consensus protocol based on completely new model Federated Byzantine Agreement (FBA). FBA is a chain like structure, where in each node has a set of nodes that it considers important. It waits for greater part of other important nodes to agree on any transaction before considering the transaction is committed. Those other important nodes not agree on transaction until the nodes they know also agrees on it, likewise the majority of the network accepts the transaction making it impractical for any attack. After all this the transaction is said to be committed by all the peers. SCP is an asynchronous protocol which guarantees the consensus even in the case of node failure.

SCP caterers the properties favourable for any consensus as decentralized control, low latency flexible trust and asymptotic security.

SCP can further be classified as [13] –

1. Nomination protocol
2. Ballot protocol

In nomination protocol key values are generated for each slot, after sufficient running same key values are generated at each node. However there are two major challenges. Firstly, there is no coordination between the nodes and second faulty nodes can reset multiple times nomination process. After then nodes predict the nomination protocol has completed execution, it starts ballot protocol. It uses federated voting to abort or commit ballot. Abort ballot is discarded and the stuck ballot is moved to ballot with higher votes. So all the stuck ballots are moved to higher ballots decision.

SCP provides guarantee to be free from stuck state but does not provide security if proper quorums slice is not selected. A quorum is a set of nodes, and quorum slice is its subset which helps the node for agreement process.

IV. Crash Consensus

4.1 Raft

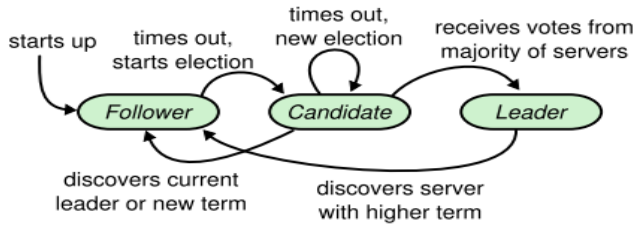
Raft based consensus mechanism is used in private blockchain where byzantine failure is not required, faster block creation and transaction finality. It creates blocks on demand and manages replicated log. It communicates using RPC calls[14,15].

In Raft, there are three types of peers [15,16]

1. Follower

2. Candidate

3. Leader



Lifecycle of peers in Raft Consensus [20]

Leader is the single node in the Raft consensus for the entire network through which all log entries must be recorded. While, the followers are passive and candidate is used to elect a new leader.

RPC calls

1. Request vote for voting a leader.
2. Append entries for replicated log entries initiated by leader
3. Install snapshot for sending replica of leader log to the followers.

Client interaction

Clients send the request to the leader. When the client first start-up, it connects to any server randomly. If the connected Server is not leader it rejects the request of client and provide the information related to the most recent leader. If the leader crash, client request would timeout and again the client starts sending request to another server. Heartbeat mechanism is used for checking the state of leader.

4.2 Chain with Federated consensus

Federated consensus protocol is used by the Chain protocol for consensus. Blocks are accepted only when they have been signed by specific quorum of blocked signers. It checks for M-of-N multi signature rule, where n is the number of blocked signers, and m is the number of signatures considered for a block to be accepted[17]. M and N values can be manipulated in accordance today business demands and its security.

As and when the new block signers join and leave the Federation, the signatures are added or removed. Public keys of the block signers are passed as arguments to the program. The program verifies with respect to the public keys validation of signatures of the hash of new block.

Consensus Algorithm

Each peer has a set of trusted peers for block signing, consensus is to be reached only to m out of n block signers thus reducing the complexity to reach the concerns throughout the network. There is single block generator coordination between block signers due to which consensus reach efficiently.

Block generator received transaction from the network filters out invalid one and aggregates them into blocks and route them block signers for validation. Block signer verifies the block forwarded by generator and signs it if found appropriate.

After the block is signed by majority of block signers it publishes the block in the network. Sale the nodes, validates it and append to their local ledger.

V. Conclusion

This paper has summarized some of the most important blockchain consensus protocols, used in private blockchain. We have discussed different voting based consensus used in private blockchain, so that it will be helpful to choose the consensus as per the business need which has direct impact on its performance. The overview of consensus protocols and their properties would be helpful in future study by researchers. Further research can be done with implementation of different consensus comparison with varying the number of loads and peers and evaluate it with some benchmark to get the actual performance indicator of the consensus used.

REFERENCES

[1] Whitfield Diffie and Martin E. Hellman, "New Directions in Cryptography" IEEE Transactions On Information Theory, Vol. It-22, No. 6, November 1976.

[2] S. Nakamoto, "Bitcoin: a peer-to-peer electronic cash system," 2008 [Online]. Available: https://bitcoin.org/bitcoin.pdf.

[3] "Blockchain Data structure," [Online]. Available: https://vitalflux.com/blockchain-linked-list-like-data-structure

[4] C. Cachin, S. Schubert, and M. Vukolic. Non-determinism in Byzantine fault-tolerant replication. ´e-print, arXiv:1603.07351 [cs.DC], 2016. URL: http://arxiv.org/abs/1603.07351

[5] Hyperledger [Online]. Available: http://hyperledger.org/.

[6] Hyperledger fabric [Online]. Available: https://github.com/hyperledger/fabric.

[7] Richard Gendal Brown, James Carlyle, Ian Grigg, Mike Hearn, Corda: An Introduction, August, 2016.

[8] Hyperledger iroha [Online]. Available: https://github.com/hyperledger/iroha.

[9] Sumeragi [Online]. Available: https://github.com / hyperledger / iroha/wiki/Sumeragi.

[10] D. Schwartz, N. Youngs, and A. Britto, "The Ripple protocol consensus algorithm," 2014 [Online].Available: https://ripple.com/files/ripple_consensus_whitepaper.pdf.

[11] F. Armknecht, G. O. Karame, A. Mandal, F. Youssef, and E. Zenner, "Ripple: overview and outlook," in Trust and Trustworthy Computing. Cham, Switzerland: Springer, 2015, pp. 163-180.

[12] Ripple [Online]. Available: from https://ripple.com/.

[13] David Mazieres, The Stellar Consensus Protocol: A Federated Model for Internet-level Consensus , February 25,2016.

[14] L. Lamport, "Paxos made simple," ACM SIGACT News, vol. 32, no. 4, pp. 18-25, 2014.

[15] D. Ongaro and J. K. Ousterhout, "In search of an understandable consensus algorithm ," in Proceedings of 2014 USENIX Annual Technical Conference, Philadelphia, PA, 2014, pp.305-319

[16] Raft-based consensus for Ethereum/Quorum [Online]. Available: https://github.com/jpmorganchase/quorum/blob/master/raft/doc.md

[17] Federated Consensus [Online]. Available: https:// chain.com /docs /1.2/ protocol/ papers/federated-consensus.

[18] Giang-Truong Nguyen, Kyungbaek Kim, "A Survey about Consensus Algorithms Used in Blockchain" Journal of Information Processing Signal, February 2018, p.7

[19] "Hyperledger iroha architecture and its transaction flow," [Online]. Available: https://opensourceforu.com/2018/08/hyperledger-iroha-a-blockchain-framework-for-mobile-apps/

[20] "Lifecycle of peers in Raft Consensus," [Online]. Available: https://raft.github.io/raft.pdf