BlockfinBFT Review by Mousebelt Labs

Youssef Alaa, MouseBelt

BlockfinBFT is the consensus protocol designed for the STORE chain to replace trust in a leaderless and decentralized manner while allowing efficient parallelization of block creation and validation.

As the name suggests the protocol is another variation of the famous Byzantine Fault Tolerant (BFT) systems. It can guarantee authentication, non-repudiation, and integrity as long as the number of faulty nodes is less than a third of the number of honest nodes ($3f + 1 \leq N$). Also, the protocol proposes strong cryptographic primitives by using the Ed25519 public key signature system.

BlockfinBFT proposes an asynchronous network with **2 types of nodes:** Block miners and Cloud miners.

Block Miners	Cloud Miners
- Validate Transactions	- Assemble transactions into blocks
- Pass validated transactions to cloud miners	- Pass blocks to block miners for final verification
- DDOS/Spam Prevention	- Compute state of chain to determine transaction reverts
- Approve Assembled Blocks from Cloud Miners	

Block miners do not maintain p2p connections with each other for consensus. Instead, each Block Miner connects to a number of Cloud Miners randomly during the consensus process. Users submit transactions to Block Miners through a RPC API interface. Then, Block Miners validate the transactions to determine if they are valid. Invalid transactions are dropped, but valid transactions are batched and submitted to Cloud Miners. Cloud Miners build the blocks based on the input of all Block Miners, then write to a shared database, and share back the block hash for retrieval to block miners to the block miners to be validated.

Cloud miners are the “block assemblers” and “ledger keepers”, their number is relatively small compared to Block miners, which enhances the network’s performance.
Since the block creation and validation processes are isolated, it requires > ⅓ Block Miners to collude with > ⅓ Cloud Miners to destabilize the consensus process. Cloud miners accept transactions from multiple block miners, then form them into blocks.

Cloud miners will attempt to form transactions into blocks depending on the available blocks. STOREs blocks are sized dynamically - some additional space is available for users who opt to pay “surge fees”, so users with latency-sensitive use cases can pay an additional amount to have their transaction validated first.

Cloud miners take the union of sets of transactions submitted from block miners, ordering by each block miner's reported wall clock time. When at least ⅔ of cloud miners have signed off on a particular set of transactions in a block, it is passed back to block miners for approval.

One of the key benefits of BlockfinBFT is its leaderless nature. Most BFT consensus algorithms that we know of, are leader-based, as they assign an elected leader to build a block of transactions, broadcast it and wait for others' acceptance.

BlockfinBFT defines a new concept called **Block Assembly**. Cloud miners create "empty blocks" and sign them to guarantee integrity of the sequence of numbers. This approach also avoids chain forks, as it doesn't follow the concept of "longest chain wins". Rather, there is an agreed upon sequence of blocks/miners who can submit transactions through the empty blocks. Only this chain is valid, rather than one of many possible chains.

Next, the cloud miners assemble the incoming transaction batches from block miners into these empty blocks. In this process, transaction order is not very important to block miners unless it is from the same account. As there is no leader to establish order of transactions within a block, order is instead determined by the wall clock of the submitting block miner. Transactions are then batched in a 2 step propose-agree process, where at least ⅔+1 of the total cloud miners must agree on the contents of a block.

After the transactions are batched and ordered into a block, the block is signed by each cloud miner. Finally, these assembled blocks are sent back to block miners for validation.

This process allows parallelization and pipelining, resulting in higher throughput. A good thing to note here also is that since block assembly can be done in parallel. Cloud miners can be assembling block n+1 while waiting for block miners to validate block n.
With this structure in place, the consensus in BlockFinBFT is reduced to agreeing which transaction batches are included in which empty block, among all Cloud Miners.

Since BlockfinBFT is designed to be asynchronous, the Block time is not predefined, but it is a function of the rate at which the STORE network receives transactions, If the STORE network receives no transactions, nothing happens, and that makes the STORE chain favor "consistency over liveness".

Why is this Significant?

- STORE's approach of pipelining can give performance benefits in some workflows. This addresses some issues in blockchain scalability seen in many traditional blockchains.
- Splitting roles in transaction validation helps divide effort and potentially scale throughput. Some blockchains are starting to adopt a two-tier approach, STORE's solution here could dramatically improve scalability.
- Most blockchains are concerned with the next block, and transactions vying for space in it. Between block assembly and economics around transaction urgency, STORE offers a new approach.
- This approach avoids chain forks on sequence numbers/longest chain