LiLaCC project

A blockchain is a distributed data structure which stores blocks of data. The blocks form a tree rooted at the first (genesis) block, and each block is linked by a pointer to its predecessor block in the blockchain. The tree is almost linear with a few short side branches.

The blockchain is maintained by a community of participants which each execute a distributed consensus algorithm. This algorithm, if executed correctly by a majority of the participants, ensures that the blockchains at each participant are identical (in consensus) for most of the time.

Consensus cannot be observed since such measurements would require an observer with instantaneous access to the state of the blockchain at each of the participants. The absence of instantaneous global state access is precisely the reason for the existence of distributed consensus.

Part 1. Public blockchains place no restriction on which participants (nodes) take part in the distributed consensus algorithm. These participants compete with each other to add the next block to the blockchain. This competition consumes resources and the winner of the competition is rewarded for the resources it consumed in winning the competition. The amount of resources consumed by the many participants in the competition is very large.

Permissioned blockchains allow a relatively small number of approved participants to take part in the distributed consensus algorithm. These participants compete with each other to elect a leader who adds a block to the blockchain. After a delay the participants elect a new leader to add the next block to the blockchain. The leader is elected by a so-called Byzantine Fault Tolerant process. Sometimes the BFT process does not identify a leader and, after a delay, the BFT election process restarts.

We will study the properties of the distributed consensus process as used in the permissioned Hyperledger blockchain. Hyperledger currently has several variants each using a different BFT algorithm. We will develop a discrete event simulator to evaluate the performance of the Hyperledger variants. We will develop a simulation model of a custom BFT algorithm designed with improved scalability properties.

Part 2. Consensus occurs when the blockchains at all the participants are identical: each blockchain at each participant consists of a single main branch and the leaf blocks of all the blockchains are identical.

When blocks are discovered frequently and/or when the block communication delays are lengthy, the blockchain can split and multiple branches are formed. The distributed consensus algorithm will preferentially append blocks to one of these branches and delete (prune) the other branches, allowing consensus to emerge.

When many homogeneous participants compete and when the block discovery interval is of the same order as the block communication interval, the blockchain splits into many branches. The blockchain is out of consensus for long periods of time, although it will with certainty return to consensus, but only for a very short time.

Another form of consensus, so-called *weak* consensus (joint work with colleagues from the Department of Mathematics and Statistics, University of Melbourne, Australia) is observed in simulations of public blockchains where the blockchain operates under conditions where consensus is rare.

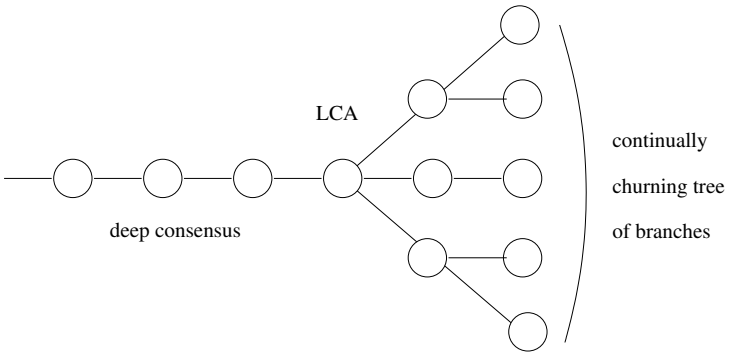

Figure 1: Weak consensus: the blockchain is in consensus from the genesis block to the Last Common Ancestor (LCA) block.

Under these circumstances, simulations show that the blockchain remains in consensus up to the Last Common Ancestor block. A tree of branches descends from the LCA block. The consensus algorithm will eventually prune the tree to a single branch, consensus occurs, and the LCA advances to the single leaf block which is identical among all the blockchains. This consensus is short-lived and a tree quickly emerges from the leaf block. However, the blockchain from the genesis block to the LCA is (tentatively) immutable. The blocks in this prefix of the blockchain can be tentatively confirmed and this is what we call weak consensus.

We propose to simulate the occurrence of weak consensus in public blockchains and in permissioned blockchains. We will study the dynamics of weak consensus and establish how distant a block must be from the LCA before the block can be regarded as confirmed.

Weak consensus may allow blockchains to function reliably under conditions where they were previously regarded as ineffective because no long-lived main branch emerged.