Author Sudhir Upadhyay

Blockchain Consensus Algorithm: Mechanisms in a digitized world

The history of consensus

Of late, in the field of computer science, the term 'consensus' has been mostly associated in the context of blockchain, and it is believed that the consensus algorithm was introduced because of blockchain. However, contrary to conventional wisdom, consensus in computer science has existed for over three decades. The concept of consensus in distributed systems has been in discussion since the 1980s - much prior to the existence of blockchain.[1]

Today, consensus algorithms continue to be an active area of research, and new algorithms are frequently designed and developed. There are different types of consensus algorithms, and each is suitable for different implementations of blockchains. Here are a few well known algorithms:

- Proof-Of-Work (PoW) — Bitcoin, solving mathematical puzzles, high energy consumption
- Proof-Of-Stake (PoS) — Network participants put in a stake to be part consensus
- Byzantine Fault Tolerant (BFT) — Protection against bad actors (e.g., Istanbul[2])
- Crash Fault Tolerant (CFT) — Protection against Node Crashes (e.g., RAFT)
- LibraBFT — A variant of BFT above (e.g., HotStuff)

The current Liink Network uses CFT based RAFT consensus algorithm, JPM Coin uses BFT based Istanbul, and Meta's Libra used a BFT based algorithm called HotStuff.

Defining consensus algorithms

What does it really mean to have a consensus in a decentralized world? Why do we have so many consensus algorithms? Why can we not reach a consensus on consensus algorithms?

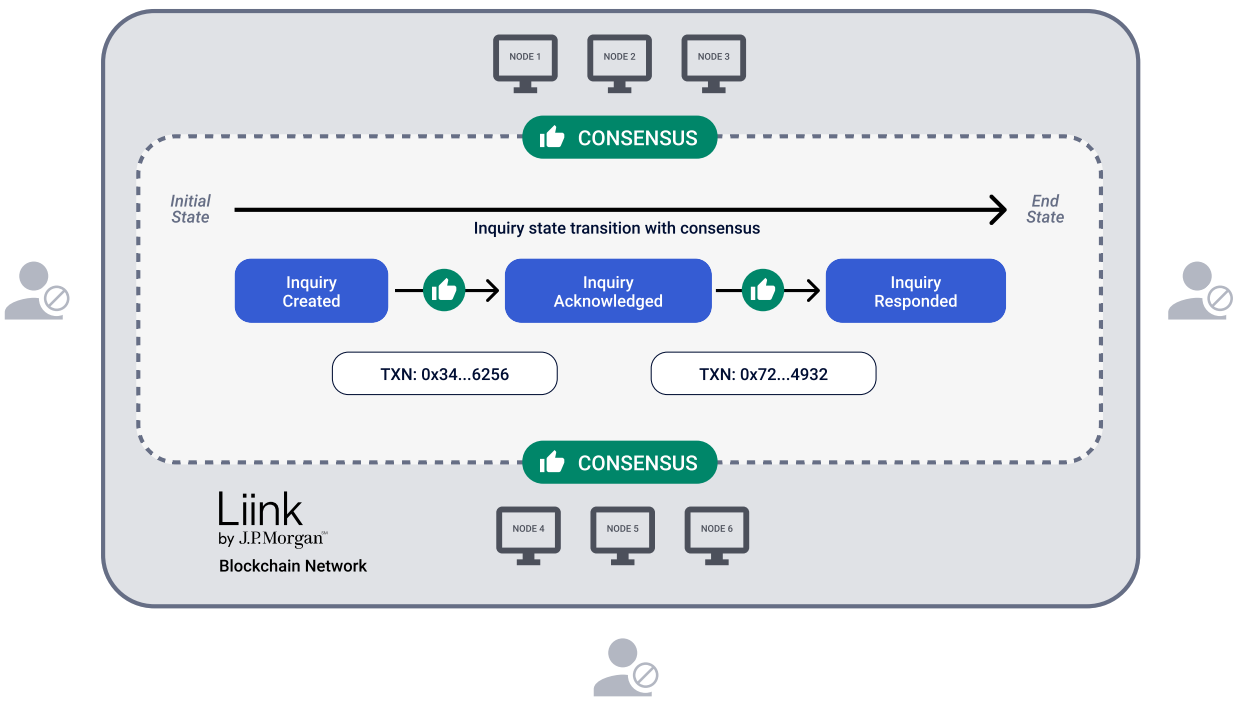
An intuitive definition would suggest that two or more individuals agree on a specific item, e.g., if they agree on the shape of an object to be a 'rectangle, they have reached consensus'. As the object changes its shape to, say a square or a circle, humans can utilize their senses to identify and agree on the new shape. However, in the world of computers, machines rely on the consensus algorithms to determine the changes in the state and a set of rules that facilitate enabling those changes.

Those rules in the blockchain world are known as smart contracts – a set of rules or logic that when executed independently and without any external intervention, change the state of an object in the blockchain network from its current state to a new state. Since the rules (smart contracts) are clearly defined and well known, one would think, it should be easy for computers to reach a consensus following those rules. However, in reality, it is extremely difficult to achieve consensus because computers do not have a common time, are geographically distributed, can turn rogue, be disconnected, may be compromised, or slow to respond — just to name a few basic reasons. Each new algorithm attempts to solve these challenges in its own way.

Applicability for Liink

How does this apply to our current work in Liink? For that, let's consider inquiries in Confirm, a Liink application. As one node creates a Confirm inquiry, an initial state is created on the chain. It then privately notifies all participants that are a part of this inquiry about this new state. In order to preserve the privacy of the participants and their data, Liink maintains two states — "Private" and "Public". This is in contrast to a general blockchain where there is only one state and all nodes are publicly notified. This also implies that on Liink, we have two consensus — one for private and another for public. A detailed implementation of private/public consensus is described in this white paper.

Each state of the inquiry (creation, acknowledgment, response, etc.) is agreed upon by participants of that inquiry based on a set of rules (Inquiry Smart Contract). Further, this update is also recorded on the Liink network as a unique transaction reference number (txn: 0x34…6256, txn: 0x72…4932 and so forth) that can be retrieved in case of non-repudiation.



Inquiry flow in Liink

Consensus algorithms govern how the state of the system within a blockchain network changes. This overview should provide a general idea about the role of consensus algorithms in blockchains and its applicability in Liink applications.

If you are interested in working with blockchain technologies, you can find current opportunities at Onyx here.

1 - https://medium.com/avalancheavax/history-of-consensus-protocols-a-short-thread-6402a140d84d
2 - https://arxiv.org/abs/2002.03613