Consensus in Distributed Systems

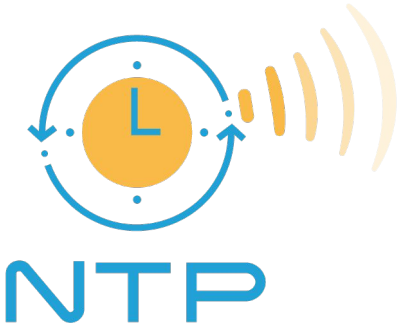
Vladyslav Nekriach

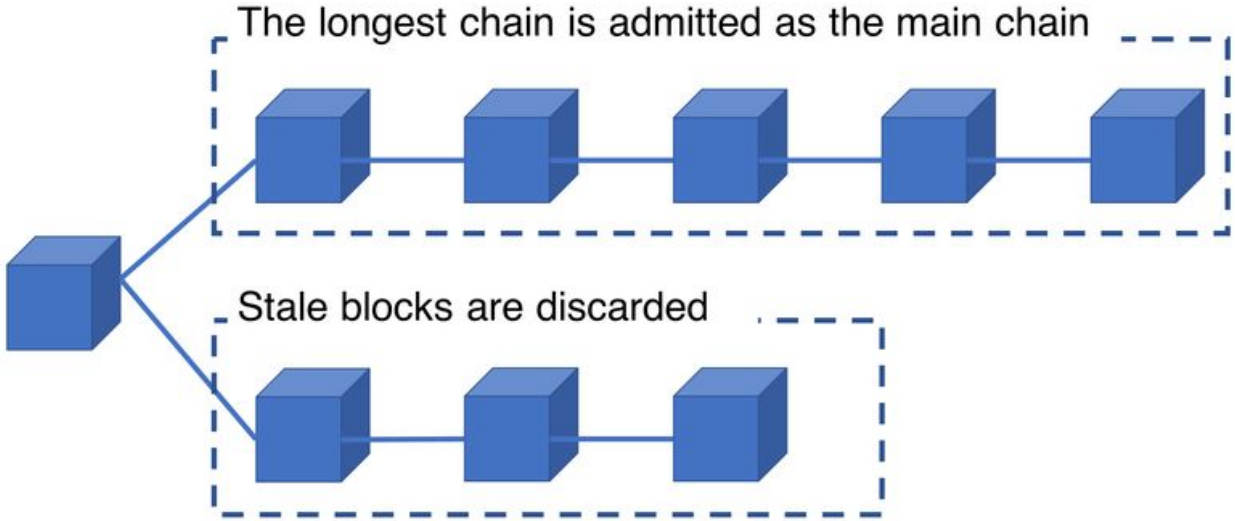
What is Consensus?

- Consensus: agreeing on something
- Example: choosing a restaurant when eating out with friends
- Focus: **asynchronous Byzantine consensus**

Real-World Applications

- Clock synchronization
- Blockchain - state of the blockchain database





Motivation and Goal

- There are many **different** algorithms for reaching consensus, but the conditions **necessary** for reaching consensus have not been rigorously studied
- Possible outcomes: simplifying and saving energy

Previous Work

- H. Attiya, J. Welch - Distributed Computing. Fundamentals, Simulations and Advanced Topics, 1st edition
- Lower bounds for running time / upper bounds for number of faulty processes
- Most of the algorithms use broadcasts - running time / memory used is huge

Current Study

- Finding bottlenecks in existing algorithms / lowering upper bounds for faulty processes
- Most of the algorithms use broadcasts - running time / memory used grows larger than O(n^2) - can we improve?
- Can we make a more fault-tolerant algorithm?

Work in Progress: Insights

- Consensus seems easy, but sometimes even simple constraints make it impossible to achieve (e.g. impossibility result for Async Byzantine model)
- Agents **can not** agree on a value right away, so there must be time for information exchange
- We **can simulate** less malicious models with more severe ones

Next Steps

- Cutting-edge papers haven’t been examined yet
- Can we create an algorithm that does not rely on broadcasts at all?
- Techniques for improving fault-tolerance will be reviewed