Radoslaw Krzeski
Summary of Topic VIII – Alternative Consensus

Algorand: Scaling Byzantine Agreements for Cryptocurrencies

The paper begins by describing a new cryptocurrency called Algorand that can confirm transactions within the minute. User have never divergent views of transactions in this system, even with malicious nodes. It argues that other currencies risk forks and take longer time to consent. It uses a new Byzantine Agreement protocol on the next set of transactions along with VRF (Verifiable Random Functions) to confirm them. User privately check if they can participate in the next transactions and include proof of their selection. After agreements users to not keep any private state apart from their keys. In this paper, the consensus system is tested on the equivalent of 500.000 users, achieves transactions quickly and 125x throughput, and little penalty for scaling more users.

Introduction:
Algorand tries to avoid three challenges; Sybil attacks (numerous pseudonyms), it must scale to millions of users and resistant to DoS-attacks.
The first is mitigated by assigning a weight to each user, where consensus is reached when 2/3 of users are honest. Weight is based on the money in their account, and by this the paper assumes honest users it seems.
Consensus by committee is used to achieve scalability by choosing a committee – random selection of users -to run each step of the protocol. Users are chosen based on their weight (with risk of committee members being subject to attacks)- All other users observe. The committee also select users in a private and non-interactive way. Every user can determine if they are in the committee trough a computation of VRF and their private keys and public information. If one is chosen the system returns a short string to prove membership. Can be included in network messages.
Once the committee members send a message, they can only do so once, avoiding attack risk.
New members for the BA process are then chosen.

The paper mentions Bitcoin-NG uses the Nakamoto consensus to elect a leader, then have him publish blocks of transactions, improving latency on transaction in Bitcoin. Trees and DAGs (directed acyclic graph) are suggested as possible improvement to this protocol as well.
Another keyword is hybrid consensus. Algorand avoid this by using the weight options, where attackers cannot break consensus when they control less than 2/3 of the currency. All user agrees on an ordered log of transactions. Algorand then achieves the goals of safety and liveness.
Strong synchrony is achieved by all user receiving messages within a known time bound. Safety is achieved with “weak synchrony” – can be asynchronous for a long but bounded period. Recovery protocol kicks off to ensure the same “time-stamping”.

Each user has a public key. System has a chain. Payment between public keys. The system builds chain in asynchronous rounds. Block structure and pointing looks the same as Bitcoin. Something called the Gossip Protocol is used to submit new transactions and messages. Never broadcasts the same twice, and check the weight of peers to avoid pollution attacks.
BA executes, chooses users and produced new agreed-upon block.
Two kinds of consensus – Final and tentative. Final is reached when broadcasted and everyone agrees. BA is periodically invoked to reach the final consensus.
User execute cryptographic sortation to determine if they are selected. They are assigned priority and proof. The sortation is based on a publicly known random called seed.

Gossip messages can be two things; one contains priorities and proofs; the other the whole block (proposers sortition hash, and proof). Users can also agree on empty block, if after a certain time BA is activated with proposed empty block, and other users do the same. Estimated at around 5 seconds waiting time in between rounds.

In each step of the BA process, committees members cast a vote for some value, and all user count the votes.
Other users are passive observers of all information apart from private keys. It takes a context ctx, which takes the state of the ledger, a round number and the proposed block from the highest priority bidder. Algorand checks if its valid. The context also consists of seed, user weights, and the last agreed-upon block. For efficiency hashes are voted on.The reduction procedure can convert the consensus process to an arbitrarty value.

The BA also has two values that describe the committee size and number of votes each has. This is in order to control the 2/3 consensus control. To deploy the system a genesis block must be created and provided to all users in addition to cryptographic sortition seed. Users are then given certification on previously processed blocks to get access to the chain, to verify sortition proofs and validating safety.

At the end, the paper states the lack of incentives might be downside for a successful implementation of the system. There is a high cost of joining and power required to fetch all data. Faking certificates and keys is also a security risk.

Bitcoin-NG: A Scalable Blockchain Protocol
Bitcoin-NG stands for Next Generation blockchain protocol with numerous improvements so that it scales well. Robust to endure extreme churn. The system is only limited by the capacity of the nodes, latency and propagation time of the network. Block size and Block interval have usually been the culprits that limit the scalability of the traditional Bitcoin.

Bitcoin-NG avoids these issue by decoupling the links to the chain into two planes; leader election and transaction serialization. Time is divided into something called epochs, where a single leader is in charge. Leaders are chosen randomly and infrequently. Leaders are entitled to serialize transactions unilaterally. The system itself is based on a set of nodes, and the peer-to peer network. - Each node can poll a random oracle as a random bit source. It uses similar puzzle system as Bitcoin, trying to hit the target hash.
Each node has a limited amount of mining power (avoiding issues with mining pools), measured by the amount of puzzles it can try per second. Proof-of work is required too.
The system has a few consensus limiters in the form of a few variables; termination, agreement and validity.

The system also proposes two new types of blocks; key blocks for leader election and microblocks that contain ledger entries. They contain information closely resembling that of the original chain blocks. They key block contains a public key used in subsequent microblocks. Keys must be mined. When mined, a leader a can then generate microblocks with a predefined maximum interval of creation. Timestamping defends from swamping attacks.

Leaders are compensated for their efforts, by each key block, and by each entry to the ledger. Ledger entry is split by the current (40% and subsequent leader (60%). These funds are locked for 100 key blocks to avoid non-mergeable transactions following a fork.

Microblocks do not require mining and are therefore splitting the required machine power needed. Poison transactions are use in entries to the ledger to maintain the first block in the pruned branch as a proof of fraud.

The blockchain also provides user with incentives in the two ways it can split. Heaviest Chain Extension, where users choose the chain with heaviest proof-of work. The second is by Longest Chain Extension.
Overall, the protocol provides a framework for splitting the work on the chain ,and reducing impact on system latency, fairness and mining power utilization

References:

Algorand: Scaling Byzantine Agreements for Cryptocurrencies
by Gilad, Yossi and Hemo, Rotem and Micali, Silvio and Vlachos, Georgios and
Zeldovich, Nickolai
http://delivery.acm.org/10.1145/3140000/3132757/p51-gilad.pdf?ip=84.213.39.36&id=3132757&acc=OA&key=4D4702B0C3E38B35%2E4D4702B0C3E38B35%2E4D4702B0C3E38B35%2EFD1C447234F14652&__acm__=1522684916_2bac57eb38258a7959ec644702e932c3

Bitcoin-NG: A Scalable Blockchain Protocol
by Ittay Eyal, Adem Efe Gencer, Emin Gün Sirer, and Robbert van Renesse
https://www.usenix.org/node/194907