Stateless Distributed Ledgers

François Bonnet, Quentin Bramas, Xavier Défago

Stateless Distributed Ledgers*

François Bonnet[1], Quentin Bramas[2], and Xavier Défago[1]

[1] School of Computing, Tokyo Institute of Technology, Japan
{bonnet,defago}@c.titech.ac.jp
[2] ICUBE, University of Strasbourg, CNRS, France
bramas@unistra.fr *(corresponding author)*

Abstract. In public distributed ledger technologies (DLTs), such as Blockchains, nodes can join and leave the network at any time. A major challenge occurs when a new node joining the network wants to retrieve the current state of the ledger. Indeed, that node may receive conflicting information from honest and Byzantine nodes, making it difficult to identify the current state.
In this paper, we are interested in protocols that are *stateless*, i.e., a new joining node should be able to retrieve the current state of the ledger just using a fixed amount of data that characterizes the ledger (such as the genesis block in Bitcoin).
We define three variants of stateless DLTs: weak, strong, and probabilistic. Then, we analyze this property for DLTs using different types of consensus.

Keywords: Distributed Ledger Technology · Blockchain · Consensus.

1 Introduction

Distributed Ledger Technologies (DLTs) are usually partitioned depending on the type of consensus used to order incoming transactions. Here, we consider the three most used classes of technologies: Byzantine agreement, Proof of Work, and Proof of Stake.

Byzantine agreement protocols [7] are used to maintain consistent states replicated over multiple servers. It can tolerate crash or Byzantine faults, up to a number that depends on the synchrony assumption of the communications. Such protocols are executed by known servers in a fixed network environment, a setting called *permissioned*. They can easily be used by nodes to maintain a ledger of transactions. Every insertion in the ledger is the result of a consensus among participating nodes.

Blockchains based on *Proof of Work (PoW)* are the first distributed ledger technologies that work in an environment where nodes can join and leave and any node can participate to the protocol. Nodes are elected randomly proportionally to their computational power, and an elected node can append transactions to the ledger. All current PoW Blockchains, including Bitcoin, work with synchronous communications, and assume correct nodes to have strictly more computational power than Byzantine nodes.

Blockchains based on *Proof of Stake (PoS)* are similar to the PoW based ones, but nodes are elected proportionally to the amount of tokens they own in the blockchain itself. In protocols based on PoS, a well-known concern is called *long-range* attacks [2], where a group of nodes create an alternative chain extending an old block. This is made possible because block generation is not computationally heavy, and if a node can extend a block at a given time, nothing prevents it from extending the same block in a different way at a later time. This attack becomes even worse when the nodes owning a majority of tokens at a previous time, do not have stake at the current time. Performing such attacks could be appealing as they have nothing to lose. This problem is known as posterior corruption [1].

Existing Proof of Stake based protocols such as SnowWhite [1], Algorand [4], Ouroboros [6], have identified such risks. The main solution proposed is to have some sort of checkpointing

* This research is partly supported by Japan Science and Technology Agency (JST) OPERA Grant Number JY280149.

mechanism to avoid considering past majorities of stake holders. A variant of this attack is called stake-bleeding attacks [3]. It uses other mechanisms such as block rewards and transaction fees to allow even a past minority of stake holders to execute long-range attacks.

Contributions. In this paper we aim at defining a general model to capture the main difference between various kind of Distributed Ledger Technologies (DLTs). Our model is abstract enough to be independent of the implementation and capture only the main mechanisms of the DLTs. Then, we focus on one property that we call the *Stateless Property*. We define this property in our general DLT model and show whether existing technologies satisfies it or not. In particular the fact that Proof of Stake based DLTs are not Stateless implies the existence of the long-range attack vulnerability.

We believe that our model could be use independently to capture other properties and compare technologies in an abstract way.

2 Model

We consider that time is discrete and at each time $t \in \mathbb{N}$, $\mathcal{N}_t$ represents the set of nodes in the network. We consider that communication is instantaneous and there is a communication link between any two nodes in the network at any given time. We also assume that each node is identified, and is able to securely sign messages.

A distributed ledger is a data structure with an *"append"* function. It is maintained by a set of processing nodes. The network receives events and the nodes react to the events according to the distributed ledger protocol. Each time the ledger is updated, a new time instant begins. Formally, a DLT is characterized by its initial state I and a state transition function σ that takes a current state S_t, the events E_t, and the network $\mathcal{N}_t$ containing all the nodes that are online at least once before the next "append". Then σ returns a new state S_{t+1} when the "append" function is called at time $t+1$. A state can be seen as a sequence of "append" and we write $S \preccurlyeq S'$ when state S is a prefix of state S', *i.e.*, S' can be obtained from S by appending data. The state S^{-k} denotes the truncated state S where the last k occurrences of "append" of S are omitted.

Given a DLT (I, σ), a sequence of $\mathcal{N} = (\mathcal{N}_t)_{t\in\mathbb{N}}$ of networks and a sequence $E = (E_t)_{t\in\mathbb{N}}$ of events, we can construct the sequence of states $States(I, \sigma, \mathcal{N}, E) = (S_t)_{t\in\mathbb{N}}$ in the following way $S_0 = I$ and $\forall t \in \mathbb{N}, S_{t+1} = \sigma(S_t, \mathcal{N}_t, E_t)$.

Stateless DLT. When a new node joins the network, it obtains the current state from the other nodes in the network. Informally, we say a DLT is stateless if a new joining node is able to deduce what is the current state of the DLT from the information received from the current network and by knowing the initial state.

At a given time t, each node $u \in \mathcal{N}_t$ has a local state $LS(u)$. For a correct node, the local state is exactly the current state S_t (communications are supposed instantaneous so all correct nodes agree on the current state at any time). For Byzantine nodes, the local state is constructed by an adversary. The set of pairs $(u, LS(u))$ for all nodes u in N_t is denoted $\mathbb{S}_t$ *i.e.*, $\mathbb{S}_t = \{(u, LS(u)) \mid u \in \mathcal{N}_t\}$.

Definition 1 (Weakly Stateless DLT). *A DLT is weakly stateless if there exists a function f such that $f(I, \mathbb{S}_t) = S_t$.*

Definition 2 (Strongly Stateless DLT). *A DLT is strongly stateless if there exists a function f such that $f(I, \mathbb{S}_t) = S_t$ and, for any subset $A \subset \mathbb{S}_t$, $f(I, A) = S_t$ or $\perp$.*

Definition 3 (Probabilistically Stateless DLT). *A DLT is probabilistically stateless if there exists a function f such that $\forall k, t, t' \in \mathbb{N}$, with $k \leq t \leq t'$, $f(I, \mathbb{S}_t)^{-k} \preccurlyeq S_{t'}$ with probability greater than $1 - O(e^{-ck})$ for some constant $c > 0$.*

3 Examples of Stateless DLTs

Byzantine Agreement Protocols It is well-known that, in a fixed network C of known nodes where communication is synchronous, consensus is possible and can tolerate up to $\frac{(|C|-1)}{2}$ Byzantine nodes [7].

We denote by σ_{BA} the transition function of a Byzantine agreement protocol among the nodes in C. σ_{BA} represents the fact that, at time t, the nodes in $C \subset \mathcal{N}_t$ perform a Byzantine agreement to order the transactions received in E_t and update the state S_t accordingly to obtain S_{t+1}. In the Byzantine agreement protocol, we consider that the state contains the information about the set C of nodes participating in the consensus protocol.

Interestingly, when a majority of nodes in C are correct, any node u outside C can ask the nodes in C for the current state. Then, the current state is the one received by a majority of nodes in C, and is guaranteed to be correct. From this we deduce the following theorem:

Theorem 1. *For a set of nodes C and an initial state I (which contains the information of C), if $\forall t \in \mathbb{N}$, $C \subset \mathcal{N}_t$ and at most $\frac{(|C|-1)}{2}$ nodes in C are Byzantine, then the DLT (I, σ_{BA}) is strongly stateless.*

Proof. Assuming at most $\frac{(|C|-1)}{2}$ nodes are Byzantine, for any sequence $\mathcal{N}$ and E, all the correct nodes in C agree on all the state $S_t \in States(I, \sigma_{BA}, \mathcal{N}, E)$ for any $t \in \mathbb{N}$. The function f returns the local state that appears in at least $\frac{|C|+1}{2}$ pairs associated with a node in C. If no such state exists (if the set of local states is only a subset of $\mathbb{S}$), f returns $\perp$. Formally, we have $f(I, A) = S$ if $|\{u \in C | (u, S) \in A\}| > \frac{|C|}{2}$, and $f(I, A) = \perp$ otherwise. □

Proof of Work Blockchains In PoW Blockchains, there are no assumptions about the nodes in the network, except that at any time t, in $\mathcal{N}_t$, the computational power of the correct nodes is strictly greater than the computational power of the Byzantine nodes. Then, the transition function σ_{PoW} applies an ordering on the transactions received in E_t (decided by some node in the network) and appends the block of transactions to state S_t resulting in S_{t+1}. In addition, from state S_t, one can compute the proof of work performed until time t, denoted $PoW(S_t)$. So $PoW(S_{t+1})$ is the sum of $PoW(S_t)$ and the proof of work corresponding to the last "append". For the initial state I, $PoW(I) = 0$.

Theorem 2. *For an initial state I, if for all t the computational power of correct nodes in $\mathcal{N}_t$ is strictly greater than the computational power of Byzantine nodes in $\mathcal{N}_t$, then the DLT (I, σ_{PoW}) is probabilistically stateless.*

Proof. Let f be the function returning the local state that maximizes the Proof of Work, $f(I, \mathbb{S}_t) = \mathbf{argmax}_S(\{PoW(S) | \exists u, (u, S) \in \mathbb{S}_t\})$. Such a local state could have been generated by an adversary. If k denotes the number of blocks we have to truncate to obtain a prefix of the correct state S_t, then the probability decreases exponentially fast when k tends to infinity. Indeed, let p_t, resp. q_t, be the computational power of the correct nodes, resp. of the adversary, at time t. By assumption, $\forall t, p_t > q_t$. Let $\lambda_t = \max_{t' \leq t}(p_{t'} q_{t'})$.

From [5] (Th.2), we deduce that, at a given time t, the probability that an adversary rewrites the last k blocks is in $O(e^{-cz})$ with $c = \log(1/(4\lambda_t)) > 0$. □

4 Impossibility of Stateless Proof of Stake Blockchains

In PoS Blockchains, the consensus at a given time t is possible assuming the nodes owning a majority of the tokens are correct. So we can assume that the state transition function σ_{PoS} is performed by those correct nodes, owning a majority of the tokens, creating a new state S_{t+1} from state S_t and incoming events E_t. However nothing prevents the nodes in $\mathcal{N}_0$ to create an alternative state after time 0.

Theorem 3. *Even if, at each time, all the nodes owning tokens are correct, the PoS DLT is not weakly stateless.*

Proof. To simplify, assume that, at some time $t > 0$, all nodes in $\mathcal{N}_0$ are owned by the adversary. Indeed, there is no assumption on the correctness of $\mathcal{N}_0$ after time 0. Then, the adversary can simulate an execution of the DLT in a sequence of networks $\mathcal{N}'_t$ where each node in $\mathcal{N}'_t$ is owned by the adversary and such that there is a bijection m mapping any nodes in $u \in \mathcal{N}_t$ to a malicious node $m(u) \in \mathcal{N}'_t$, for all $t \in \mathbb{N}$ ($m(u) = u$ if u is already malicious). Hence, the sequences of networks $(\mathcal{N}'_t)_t$ and $(\mathcal{N}_t)_t$ differ only in the addresses that identify nodes. The adversary can execute the same DLT using the same sequence of events $(E_t)_t$ but in the malicious sequence of networks $\mathcal{N}' = (\mathcal{N}'_t)_t$, which gives a different sequence of states $States(I, \sigma_{PoS}, \mathcal{N}', E) \neq States(I, \sigma_{PoS}, \mathcal{N}, E)$.

At time t, the adversary can connect all nodes $\mathcal{N}'_t$ to the network so that the set of local states is $\mathbb{S}_t = \{(u, LS(u)) | u \in \mathcal{N}_t\} \cup \{(u, LS(u)) | u \in \mathcal{N}'_t\}$. The set is symmetric as half of the local states contain S_t and the other half contain S'_t and both states differ only in the addresses used to identify nodes.

$States(I, \sigma_{PoS}, \mathcal{N}', E)$ is a valid sequence of states if the sequence of networks was $\mathcal{N}'$ and if the adversary creates symmetrically the state S_t using the sequence of networks $\mathcal{N}_t$. A function f should answer S_t in the first case and S'_t in the second case, with exactly the same input, which is a contradiction. □

One way to make the PoS DLT stateless is to assume that any set of nodes owning a majority of the token at a given time t are correct at any time $t' > t$. This is a very strong assumption as for instance, it gives the same kind of trust in the initial set of nodes as in the Byzantine agreement protocol. Indeed, if the nodes in $\mathcal{N}_0$ are still connected at time t then, they act as a trusted committee. We could even remove the proof of stake entirely and only rely on standard Byzantine agreement between nodes in $\mathcal{N}_0$. In the case where nodes in $\mathcal{N}_0$ go offline, they can select replacement nodes (using any method including election or simply one-to-one replacement), and include a signed message that define their choice in the state so that any joining node could identify the current set of trusted nodes. In future work, we plan to identify other sufficient conditions for the existence of Stateless DLTs.

References

1. Iddo Bentov, Rafael Pass, and Elaine Shi. Snow white: Provably secure proofs of stake. *IACR Cryptology ePrint Archive*, 2016(919), 2016.
2. Evangelos Deirmentzoglou, Georgios Papakyriakopoulos, and Constantinos Patsakis. A survey on long-range attacks for proof of stake protocols. *IEEE Access*, 7:28712–28725, 2019.
3. Peter Gaži, Aggelos Kiayias, and Alexander Russell. Stake-bleeding attacks on proof-of-stake blockchains. In *2018 Crypto Valley Conference on Blockchain Technology (CVCBT)*, pages 85–92. IEEE, 2018.
4. Yossi Gilad, Rotem Hemo, Silvio Micali, Georgios Vlachos, and Nickolai Zeldovich. Algorand: Scaling byzantine agreements for cryptocurrencies. In *Proceedings of the 26th Symposium on Operating Systems Principles*, pages 51–68, 2017.
5. Cyril Grunspan and Ricardo Pérez-Marco. Double spend races. *International Journal of Theoretical and Applied Finance*, 21(08):1850053, 2018.
6. Aggelos Kiayias, Alexander Russell, Bernardo David, and Roman Oliynykov. Ouroboros: A provably secure proof-of-stake blockchain protocol. In *Annual International Cryptology Conference*, pages 357–388. Springer, 2017.
7. Michel Raynal. *Distributed algorithms for message-passing systems*, volume 500. Springer, 2013.