Solana: DFuture?

Ruben Teimas

Universidade de Évora, Évora, Portugal
January 13, 2022
m47753@alunos.uevora.pt

Abstract: This paper makes a revision on blockchain concepts and its appearance. It also looks at some of the most well known blockchains, like *Bitcoin* and *Ethereum*, where they thrived and failed and how *Solana* fixed those flaws, focusing on the consensus mechanism. At the end is explored what *Web 3.0* is and how *Solana* can make it happen.

Keywords: Blockchain · Cryptography · Ethereum · Smart Contracts · Solana · Web 3.0

1 Introduction

In a time where we try to automate everything as much as possible some tasks, such as money transactions, remain captive of the systems developed dozens of years ago.

If *Person A* wanted to make a transaction to *Person B* she would have to transfer the amount of money to a broker (either a bank or a platform like *Paypal*) and the broker would forward the money to *Person B*. This presents a problem: centralization.

By using a broker we're introducing an entity that does nothing more than serving as a bridge. For that reason, by the end of the *2000*s, an anonymous person (who identifies itself as *Satoshi Nakamoto*) published an article [7] where he present a peer-to-peer (*P2P*) electronic cash system.

P2P protocols were already very popular at the time, specially for file sharing, with *BitTorrent* and *Napster* being widely used around the world. However, these protocols did not assure the token (or file in these cases) *Node A* sends to *Node B* did not remain with *Node A*, which would eventually lead to a *double-spending problem*. The solution proposed by Satoshi solved this problem.

The author stated that "what is needed is an electronic payment system based on cryptographic proof instead of trust, allowing any two willing parties to transact directly with each other without the need for a trusted third party" [7].

Bitcoin was born and so was *Blockchain*.

2 Blockchain

Blockchain can be defined as a distributed storage, distributed over a network of peers that ensure the consistency of the chain. Blocks of information, validated by the network of peers as trustworthy, are then recorded in the storage.

Each block holds a list of transactions functioning like a public ledger, and a reference (hash) to the previous block. If there is any attempt to change data in the chain, the change is detected by the network - invalidates the hashes - and the attacked chain is replaced by a valid one.

Is almost impossible to attack all the network, so the chain remains valid. Figure 1 illustrates an example of a blockchain architecture.

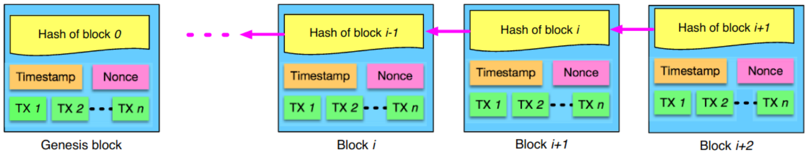


Fig. 1. General blockchain architecture [10]

The example structure of the block [10] is composed by the block header and by the block body. The block header contains the following information:

- Block version: indicates which set of block validation rules to follow.
- Parent block hash: 256-bit hash value that points to the previous block.
- Merkle tree root hash: the hash value of all the transactions in the block.
- Timestamp: current timestamp as seconds since 1970-01-01T00:00 UTC.
- nBits: current hashing target in a compact format.
- Nonce: a 4-byte field, which usually starts with 0 and increases for every hash calculation.

The block body is composed of a transaction counter and the transactions. The number of transactions that a block can hold is limited by the block size.

3 Blockchain applications

Although *Bitcoin* was a revolutionary system it was developed solely to operate as a peer to peer digital currency. Nonetheless, some years after being released, *Vitalik Buterin*, who was part of *Bitcoin* community at the time, released an white paper [5] which would revolutionize blockchain.

Vitalik believed the community was not approaching blockchain the right way and, therefore, not getting the maximum potential out of it. He thought of blockchain as a platform to build and deploy different types off applications rather than an application itself.

Those applications could tackle specific tasks in many different areas such Finances, Social Services, Health... Some of those tasks are presented in Figure 2.

With that vision in mind, and due to *Bitcoin* restrictions, *Vitalik* and a small core team developed Ethereum - A *Next-Generation Smart Contract and Decentralized Application Platform*.

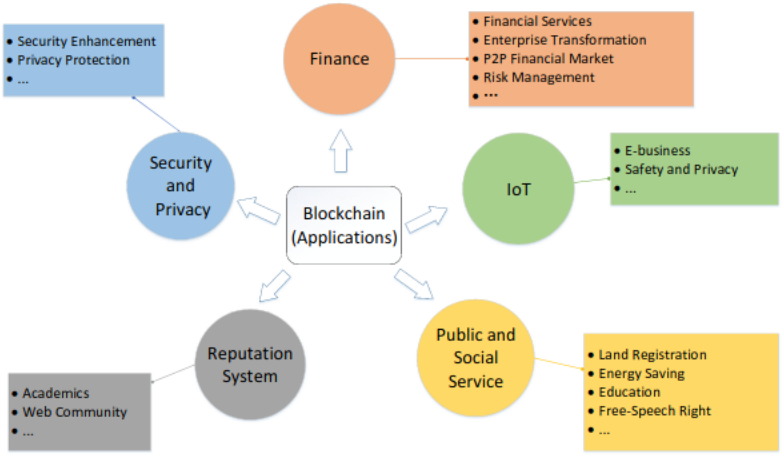


Fig. 2. Possible blockchain applications [10]

4 Ethereum

Ethereum is an open software platform based on blockchain technology that enables developers to build and deploy decentralized applications (*dApps*).

Aiming at being a general purpose blockchain it introduced new concepts and mechanisms which added much more freedom than any other blockchain had at the time.

4.1 Smart Contracts

One of features that made *Ethereum* stand out was the Smart Contract. It can be simply translated to a bunch of code that is executed in the blockchain when specific conditions are met.

Whereas in *Bitcoin* we could only demand *Bob* to send a coin to *Alice*, in Ethereum (with smart contracts) we can demand *Bob* to send an *ether* (the *Ethereum* currency) to *Alice* if the date is *1st of January* and *Alice*'s balance is less than 100 *ether*.

In the end, this type of contracts is what powers the *dApps*.

4.2 dApps

A decentralized application refers to an application that is built on top of blockchain technology.

They run on a blockchain and benefit from all of its properties like immutability, security, tamper resistance and zero downtime. Essentially any service could be turned into a decentralized application.

4.3 Running out of GAS

Despite being innovative, *Ethereum* is far from perfect. One of problems regularly pointed out is the blockchain's scalabilty. [4]

Just like *Bitcoin*, *Ethereum* consensus mechanism is *Proof of Work* (*POW*). *POW* requires, in a few words, miners to solve complex mathematical puzzles in order to validate transactions. Due to the puzzle's high complexity a lot of computational power is needed, which consumes a lot of energy.

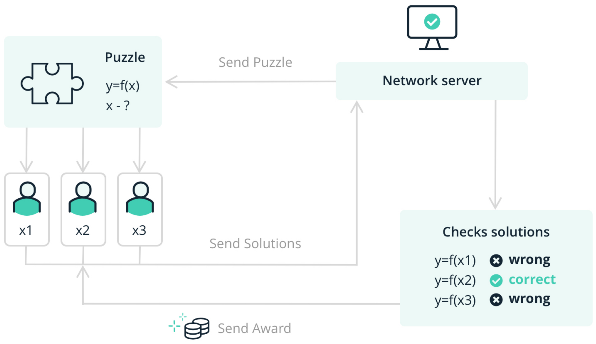


Fig. 3. Proof of Work illustration. [3]

This is very bad for the environment (another problem pointed) but also very expensive. To compensate these expenses the users must pay fees (usually called *GAS*).

At the begging of 2021 *GAS* fee were very high, which is correlated with high demand, and more demand on a blockchain is a good thing, or so it would seem. [6] They are also correlated with high transaction times, which is itself a problem. Even though it has been reported that a newer version of *Ethereum* (*Eth2*) [1] is in development, while it's not available and with higher and higher *GAS* fees users have started migrating to other blockchains such as ***Solana***.

5 Solana

When *Ethereum* was introduced, the development team introduced the ***scalability trilemma***. It stated that a blockchain can have, at most, 2 edges of a triangle. The triangle is represented at Figure 4.

What the *Ethereum* developers did not explain was that, even thougth *Ethereum* fails on *scalability*, the trilemma was not proved mathematically, hence we cannot be sure about it.

Recent blockchains claim to present solutions to that problem. One of those blockchains is *Solana* and the way it tries to solve the trilemma is centered on its

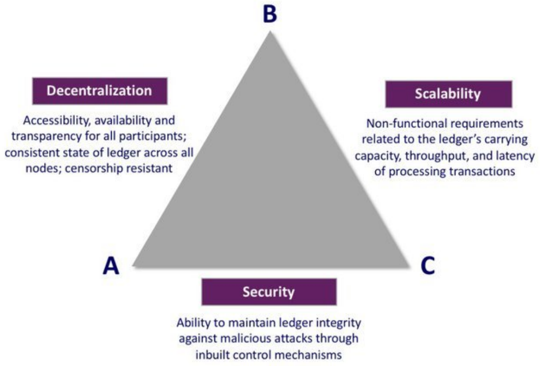

B
Decentralization
Accessibility, availability and transparency for all participants; consistent state of ledger across all nodes; censorship resistant
Scalability
Non-functional requirements related to the ledger's carrying capacity, throughput, and latency of processing transactions
A
C
Security
Ability to maintain ledger integrity against malicious attacks through inbuilt control mechanisms

Fig. 4. Blockchain scalability trilemma. [8]

consensus system. While *Ethereum* uses *PoW* as a consensus mechanism, *Solana* uses an hybrid consensus mechanism by combining *Proof of History* (*PoH*) and *Proof of Stake* (*PoS*).

In most blockchains, each node in the network usually relies on their own local clock without knowledge of any other participants clocks in the network. The lack of a trusted source of time means that when a message timestamp is used to accept or reject a message, there is no guarantee that every other participant in the network will make the exact same choice. *PoH* is designed to create a ledger with verifiable passage of time (duration between events and message ordering). This ways, every node in the network will be able to rely on the recorded passage of time in the ledger without trust.

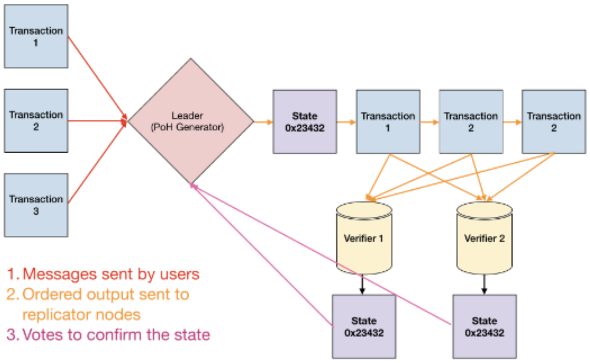


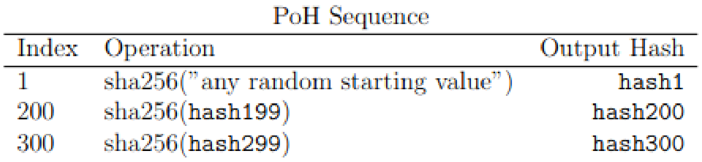
Fig. 5. Solana network flowchart. [9]

As shown in Figure 5, at any given time a system node is designated as Leader to generate a Proof of History sequence, providing the network global read consistency and a verifiable passage of time. The Leader sequences user messages and orders them such that they can be efficiently processed by other nodes in the system, maximizing throughput. It executes the transactions on the current state and publishes the transactions and a signature of the final state to the replications nodes called Verifiers. Verifiers execute the same transactions on their copies of the state, and publish their computed signatures of the state as confirmations. The published confirmations serve as votes for the consensus algorithm.

5.1 Proof of History

Proof of History is a sequence of computation that can provide a way to cryptographically verify passage of time between two events.

It works the following way: with a cryptographic hash function, whose output cannot be predicted without running the function (e.g. sha256), run the function from some random starting value and take its output and pass it as the input into the same function again. Record the number of times the function has been called and the output at each call.



PoH Sequence

Index	Operation	Output Hash
1	sha256("any random starting value")	`hash1`
200	sha256(`hash199`)	`hash200`
300	sha256(`hash299`)	`hash300`

Fig. 6. Iteration example. [9]

This set of hashes can only be computed sequentially by a single computer thread because there is no way to predict what the hash at index n will be without actually running the algorithm, starting at 0, n times.

This allows the verifiers not to worry about timestamps and internal clocks, which means there is no need to communicate in that regard, speeding up the process.

We can also use this mechanism to timestamp external actions. For instance, if a picture is taken and we want to store its metadata in the blockchain we simply append the *sha256* (in example) of the metadata to the hash of the previous computation and pass it as input to the hash function. This will give us a chronological order of external events.

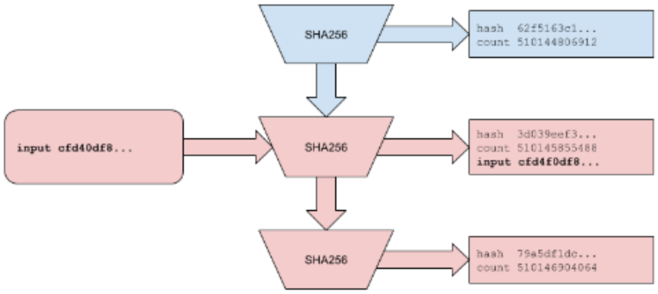


Fig. 7. Inserting data into Proof of History. [9]

While the creation of the hashes can only be performed by a single threaded core, the sequence can be verified correct by a multi core computer in significantly less time than it took to generate it.

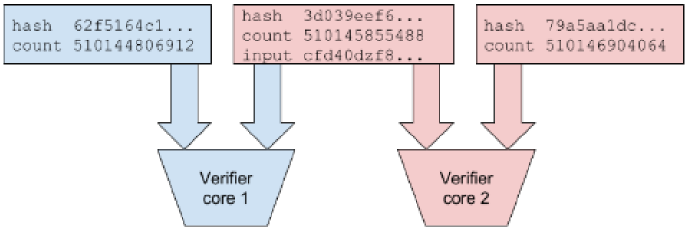


Fig. 8. Proof of History verification. [9]

In the example in Figure 9, each core is able to verify each slice of the sequence in parallel. Since all input strings are recorded into the output, with the counter and state that they are appended to, the verifiers can replicate each slice in parallel. The red colored hashes indicate that the sequence was modified by a data insertion.

Its possible to synchronize multiple Proof of History generators by mixing the sequence state from each generator to each other generator, and thus achieve horizontal scaling of the Proof of History generator. This scaling is done without sharding. The output of both generators is necessary to reconstruct the full order of events in the system.

Given generators A and B, A receives a data packet from B (hash1b), which contains the last state from Generator B, and the last state generator B observed from Generator A. The next state hash in Generator A then depends on the state from Generator B, so we can derive that hash1b happened sometime before hash3a. This property can be transitive, so if three generators are synchronized through a single common generator $A \leftrightarrow B \leftrightarrow C$, we can trace the dependency between A and C even though they were not synchronized directly.

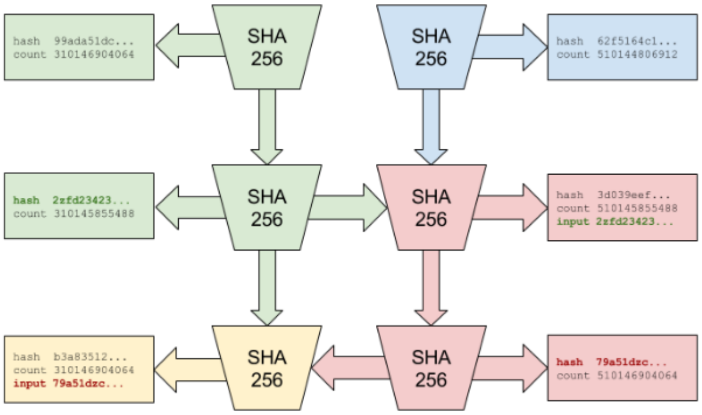


Fig. 9. 2 generators synchronising. [9]

5.2 Proof of Stake

This specific instance of Proof of Stake is designed for quick confirmation of the current sequence produced by the Proof of History generator, for voting and selecting the next Proof of History generator, and for punishing any misbehaving validators. This algorithm depends on messages eventually arriving to all participating nodes within a certain timeout.

6 Web 3.0

The concept of Web 3.0 has been around for a while, but it wasn't until until the last few years that it started gaining traction.

It refers to decentralized apps that run on the blockchain. These are apps that allow anyone to participate without monetising their personal data. In a decentralised web, each participant holds a secret key and can then use it to identify each other.

By having the applications placed on the blockchain rather than on centralized servers we also assure that the availability of said applications won't go down and affect the life of thousands og people.

These characteristics make the web a much more transparent, available and personal space.

Even thought this is a very utopic vision and we might be far from it, multiple *dApps* have been developed in *Solana* ecosystem. Some of the most well known are *Audius* (who recently announced a partnership with *TikTok*) and the integration in *Brave* browser.

7 Conclusion

In the end, the future is always uncertain but *Solana* has, without a doubt, been gaining traction and it might be one of the agents to power the new age of web.

However, that envisionment of the web seems to be yet far from happening, at least in the next few years, and if *Solana* really wants to be part of it it sure needs to fix some of its most recently availability problems [2].

Bibliography

[1] The eth2 upgrades. https://ethereum.org/en/eth2/ Visited: 10/01/2022.

[2] Solana labs ceo denies that solana went down after a ddos attack (updated). https://cryptonews.com/news/solana-repotedly-went-down-again-after-ddos-attack.html Visited: 11/01/2022.

[3] What is proof-of-work. https://www.ledger.com/academy/blockchain/what-is-proof-of-work Visited: 06/01/2022.

[4] Ryan Browne. Ethereum: What is it and how is it different from bitcoin?, May 2021. https://www.cnbc.com/2021/05/10/ethereum-what-is-it-and-how-is-it-different-to-bitcoin.html (Visited: 06/01/2022).

[5] Vitalik Buterin et al. Ethereum white paper: a next generation smart contract & decentralized application platform. *First version*, 53, 2014.

[6] Travis Hoium. Gas fees could be ethereum's kryptonite, Dec 2021. https://www.fool.com/investing/2021/12/15/gas-fees-are-ethereums-kryptonite/ Visited: 06/01/2022.

[7] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. *Cryptography Mailing list at https://metzdowd.com*, 03 2009.

[8] Hong Wan, Kejun Li, and Yining Huang. Blockchain: A review from the perspective of operations researchers. 01 2021.

[9] Anatoly Yakovenko. Solana: A new architecture for a high performance blockchain v0.8.13. *First version*, 2018.

[10] Zibin Zheng, Shaoan Xie, Hong-Ning Dai, Xiangping Chen, and Huaimin Wang. Blockchain challenges and opportunities: A survey. *International Journal of Web and Grid Services*, 14:352, 10 2018.