Comparison of TON, Solana and Ethereum 2.0

April 7, 2022

Abstract

New blockchain projects, such as Solana and Ethereum 2.0, have appeared since the original TON Whitepaper [1] was written in 2017. In this text, we compare TON to some of these newer projects.

1 Formal comparison

It is natural to make a formal comparison based on the classification of blockchain projects in sections **2.8** and **2.9** of the original TON Whitepaper. What can be said about Solana and Ethereum 2.0 in this context?

1.1 General comparison guidelines

Recall that we classify blockchain projects based on the following criteria, explained in more detail in section **2.8** of TON Whitepaper [1]:

- Single-blockchain / Multi-blockchain projects
- Consensus algorithm: PoW (proof-of-work) / PoS (proof-of-stake)
- For PoS projects, the exact consensus algorithm (such as dPOS or BFT)
- Support for arbitrary (Turing-complete) smart contracts

For multi-blockchain projects, we have to consider further issues:

- Type and rules of member blockchains: homogeneous, heterogeneous, mixed
- Presence of a masterchain
- Native support for sharding, static and dynamic sharding
- Interaction between blockchains: loosely-coupled / tightly-coupled

In addition to that, a simplified classification of blockchain projects was presented in **2.8.15** of TON Whitepaper [1], and a table comprising the basic properties of most popular blockchain projects was given at the beginning of section **2.9**.

2 Solana

2.1 An overview of Solana

Solana [2] is a somewhat unusual project for the 2020's: it is a single-blockchain project optimized for very fast execution of specialized transactions. In this respect, it is similar to the BitShares project [9] (developed in 2013–2014), a predecessor of EOS [8] (developed in 2016–2018). However, instead of dPOS, a variant of PBFT [10] called Tower Consensus [3] is used. Solana claims to generate one block every second or even faster; however, this comes at a certain price since the next block is generated before the previous is finalized (to quote from an official blog post [4], "Unlike PBFT, Tower Consensus prefers liveness over consistency"). This can lead to creation of short-lived forks. The finalization of a block in a real-life situation when the validators are distributed in different locations around the world would require several round trips (PBFT in the optimistic case essentially is a three-phase commit protocol), and therefore would require several seconds at best. The explanation from the official documents seems to imply that a block is usually finalized after 16 voting rounds, with each round expected to last approximately 400ms; this means a 6.4 second finalization time.

We might say that Tower Consensus, while formally being a variant of PBFT, is better compared with dPOS consensus protocols, which provide shorter block generation times at the expense of longer block finalization times.

Another interesting feature of Solana is that it is heavily optimized towards the execution of very simple predefined transactions which do not alter account data, with the possible exception of its balance. This allows for a massively parallelized execution and verification of transactions. In this respect, Solana resembles BitShares, a predecessor of EOS which used dPOS (with similarly short block generation times and long block finalization times) and was optimized towards the massive execution of very simple predefined transactions. Apart from that, Solana is designed in such a way that the verification of the correct order of transactions may be sped up thousandfold on a high-end GPU compared to the time required to generate these transactions.

Ultimately, Solana claims to be able to perform up to 700,000 simple transactions per second (the actual number according to [11] is 65,000 rather than 700,000), provided they do not change account state and do not require much data, and provided the total state of all accounts fits into computer RAM. Again, this is much in line with what BitShares [9] promised several years before. A major difference is that, in contrast to BitShares, Solana does provide support for transaction types not predefined by the blockchain software; for this, a variant of virtual machine called Berkley Packet Filter is employed, and pre-compiled programs for this machine may be uploaded into the Solana blockchain and then referenced in transactions [12][13]. Therefore, Solana formally *is* Turing complete; however, the performance metrics usually quoted are related only to very simple predefined transactions, and only in the situation when all the data of all accounts fits into RAM, so we feel that a comparison to BitShares is still valid.

To summarize, Solana is an "alternative third generation blockchain project" in the terminology of **2.8.15** in the TON Whitepaper [1], ultimately very similar to BitShares [9], a predecessor of EOS [8], but with further optimizations. It is formally Turing complete, but actually only capable of performing a large amount of very simple transactions of several predefined types, or a much smaller amount of more general transactions; it claims to be able to generate more than one block per second on average and perform 700,000 simple transactions per second after a future hardware upgrade (the actual number appears to be 65,000 rather than 700,000 [11]). It is an inherently unscalable specialized single-blockchain project, no support for sharding or different workchains is provided or possible without a complete redesign (we refer to **2.8.16** of TON Whitepaper [1] for an explanation of why such a redesign is very hard to perform at a later stage). In this respect it is a step back from EOS [8].

In contrast, TON allows for instantly deployable smart-contracts of any complexity, a higher level of security due to a consensus mechanism with shorter transaction and block finality times, and perhaps most importantly, dynamic sharding. The latter automatically scales the blockchain into more and more shardchains as the load increases, providing a level of scalability unfeasible for any single-blockchain architecture, such as used in Solana.

It is natural that the success of Solana's predecessors, other single-blockchain or loosely coupled multi-blockchain projects without sharding support such as EOS, appeared to be spectacular in the early stages, but proved to be short-lived as such concepts inevitably hit inherent limitations that negatively impact their scalability and stability in later stages. An early indication that this might be the case for Solana as well is the breakdown of the Solana blockchain [5] in September 2021, when it was effectively stalled for 17 hours after an unexpected surge of transactions that "created a memory overflow, which caused many validators to crash, forcing the network to slow down and eventually stall", to quote from an official document describing this malfunction. This makes us question the future performance of Solana on real-life transactions, as opposed to specially crafted very simple transactions involving only a small number of distinct accounts and performed in a very specific test environment with hundreds of powerful validator servers located in one datacenter or in several nearby datacenters. TON appears to be much more robust in this respect.

2.2 Metaphor: Solana is a super steam locomotive

Solana is an interesting example of a venerable engineering approach with well-known inherent limitations pushed to its extremes. As such, it reminds us of several similar stories in the history of technology, which we would like to relate at this point.

One, of course, is the world speed record of 203 km/h achieved in 1938 by a British LNER Class A4 4468 Mallard steam locomotive. It did not reach these average speeds during regular passenger service, where it was rather running at 150 km/h. However, at that point it held the world speed record for all locomotives, steam, diesel or electric. Nonetheless, this was a technological dead end, and all later high-speed trains, such as Japan's Shinkansen, France's TGV or Germany's ICE, were multiple-unit electric trains. It is interesting to note that all modern high-speed trains are electric and multiple-unit, meaning that there is an engine or even more than one engine in each carriage, as opposed to traditional trains pulled by a steam locomotive. We see the power of sharding in action. And we see why it was obvious even in 1938 that the future belongs to electric trains: electric engines can be easily scaled and distributed along the whole train, while the steam engine technology cannot be scaled in this fashion.

The second technological story that comes to mind is that of Intel's Pentium 4 CPUs at the beginning of 2000's. Intel promised to gradually increase the clock frequency of these processors up to 10 GHz in several years, and achieve unprecedented performance levels. In practice, Pentium 4 often ran real-life code slower than the previous-generation Pentium 3 with a formally lower clock frequency, and Intel was unable to deliver the clock frequency growth it originally promised after reaching the boundary of 4 GHz. After that, Intel completely rethought its CPU development roadmap and essentially reverted to the Pentium 3 architecture (rebranded as Intel Xeon or Intel Core 2) with lower clock speeds, but with more and more CPU cores installed in one physical device. This approach proved to be more scalable and durable, and now we can buy 64-core processors if we want to. Again, the approach based on making one computing core faster and faster foundered, and the multi-core approach (which can be likened to multiple-unit trains and to sharding in blockchains) turned out to be viable.

The third technological story is that of supercomputers, such as Cray, which were popular in 1970s and 1980s, but were later superseded by clusters consisting of thousands of commodity CPUs (usually server versions of Intel and AMD CPUs). Nowadays, all of the top 100 supercomputers are clusters of commodity CPUs. Again, "sharding" or "multiple-unit system" won over the super-optimization of a single-unit system.

We would like to conclude our exploration of technological history by likening Solana to a super steam locomotive, which exploits all possible technological optimizations of a venerable technological paradigm, but is ultimately unscalable and a technological dead end. We may praise and admire the ingenuity employed in designing and running such technological marvels; but they are technological dead ends nonetheless.

3 Ethereum 2.0

The comparison of TON to Ethereum 2.0 is somewhat complicated by the fact that the development and deployment of Ethereum 2.0 is still incomplete as of 2022. Let us describe what seems to be known at this moment [6]–[7].

The transition to Ethereum 2.0 is to be performed in several stages. First, a new Beacon blockchain [6] (similar in its role to a masterchain in the terminology of the original TON Whitepaper) is to be deployed. This Beacon blockchain will employ an original PoS consensus algorithm called Casper. Its main purpose is to register the states (hashes of last blocks) of up to 64 shardchains (auxiliary blockchains). The proposed PoS algorithm is unusual in that it would involve and even require a very large number of participating validators (at least 16,384), each staking a small amount of Ethers (32 Ethers each). These validators essentially are usual Ethereum nodes that only have to stake 32 Ethers; no specific communication between these nodes is required apart from the usual Ethereum network gossip related to block and mempool propagation. In this respect, Ethereum 2.0 appears to be unusually "democratic" (almost all other PoS blockchain projects are rather "oligopolic", where tens or at most hundreds of validators are involved at a given instant of time in actually creating the blocks). However, this comes at a price: block finality time appears to be around 10–15 minutes, both for the Beacon blockchain as well as for the 64 shardchains. In other words, one would have to wait for 10–15 minutes just to be sure that their transaction was indeed finalized.

The second stage of the supposed transition would consist in transforming the existing Ethereum 1.0 (PoW) blockchain into one of the 64 shardchains (say, shardchain zero) tied to the new Beacon blockchain. After that, the PoW consensus mechanism will be disabled and Ethereum will continue as a PoS blockchain.

Finally, the third stage would consist in the creation of 63 other shardchains [7]. In this way Ethereum would consist of 64 shardchains, one of which will be the old Ethereum 1.0 blockchain, and a Beacon blockchain, which is the masterchain primarily dedicated to staking, slashing (punishing misbehaving validators), achieving consensus and registering hashes of shardchain blocks.

Unfortunately, it is not clear at this stage what the exact capabilities of the new 63 shardchains will be and how the shardchains would interact with each other. Without this information, we cannot truly complete our classification of a multi-blockchain system. However, if messaging between shardchains is ever introduced, one would have to wait for 10–15 minutes until the finalization of the shardchain block originating a message before that message can be processed in another shardchain. This seems to be the reason why shardchain interaction is not considered at this point. Furthermore, the additional shards currently are not supposed to be able to run EVM smart contracts at all (though there are some indications that this can be reconsidered in the future) [7]. Instead, they are supposed to be used as additional data storage in a distributed ledger. They will not be used to run arbitrary smart contracts; instead, their preferred use is finalization of off-chain or layer-2 blockchain computations (similar to payment channels or Lightning Network for Bitcoin), possibly similar to those previously proposed by the Plasma project (discussed in **2.9.10** of the original TON Whitepaper).

In this way Ethereum 2.0 seems to completely avoid the whole issue of shardchain interaction, passing messages between smart contracts residing in different shardchains and so on. Instead, future users of Ethereum are expected to run all their transactions in unrelated sidechains and use Ethereum 2.0 shardchains for the finalization of the final state of these sidechains. It is in this sense that Ethereum 2.0 claims to be able to scale from the current 15 transactions per second to tens of thousands of transactions per second. We think that such claims are misleading because different kinds of transactions with different consistency and finality guarantees are being compared. The current 15 transactions per second are the usual on-chain Turing-complete EVM smart contract executions; tens of thousands of "transactions" promised in the possibly not-so-distant future are something completely different, likely to be very specialized transactions with limited sets of participants that become universally visible only much later than they are generated. One might also compare this to Bitcoin perfomance with and without Lightning Network. However, in that case one should also be allowed to quote TON performance including "transactions" inside all potentially possible payment channels and payment networks bound to smart contracts residing in TON blockchain shardchains. Therefore, if we accept the claim that Ethereum 2.0 would be able to perform tens of thousands of "transactions" per second (actually meaning sidechain and payment channel transactions), then by that definition TON would be able to perform billions of such "transactions" per second.

To summarize, Ethereum 2.0 seems to sidestep the really complicated problem of shardchain interaction, which cannot be solved without completely rethinking the EVM and smart contract interaction model (we refer to **2.8.16** of the original TON Whitepaper [1] for a more detailed explanation), and to augment the original Ethereum blockchain with 63 additional shardchains (with 10–15 minute finalization time) good only for storing the finalized states of sidechains and payment channels [7]. This is a somewhat defeatist approach. One would expect something more ambitious from the second-oldest major blockchain project in the world, which was the first to introduce Turing-complete smart contracts!

In the form it is currently envisioned and tested, Ethereum 2.0 doesn't aim to achieve the level of speed and versatility that has been already reached by the existing implementations of TON.

4 Conclusion

TON Blockchain was originally envisioned and described five years ago, in 2017. Its whitepaper [1] carefully explained why the design choices made by TON appear necessary for building a truly scalable blockchain project, capable of handling millions of transactions per second in the future without any essential changes involving the logic of its smart contracts and their interaction. That was the reason why TON was singled out as the only fifth-generation blockchain project at the time.

Since then, five years have passed and new blockchain projects have emerged. One would expect them to overcome the limitations of all the older blockchain projects that were discussed in the TON Whitepaper, and possibly suggest some novel ways of blockchain development. Instead, we see the reemergence of blockchains based on ideas which were outdated even in 2017. One such project is Solana, designed starting in 2019, which is an inherently unscalable "alternative third generation project" in the terminology of TON Whitepaper, comparable to the BitShares project from 2013, a predecessor of EOS. If the reader is frustrated by these repeated comparisons of Solana to a seemingly obscure project from 2013 that claimed to deliver similar performance, it may be for a good reason: if we can use the past to predict the future to some extent, we might predict that Solana would be equally obscure nine years after its inception — in 2028. Furthermore, adding sharding to Solana later to overcome its inherent inscalability will be virtually impossible, for reasons explained in the original TON Whitepaper.

Another blockchain development that seems disappointing to us is Ethereum

2.0, which essentially undoes the main achievement of Ethereum — Turing-complete smart contracts, and claims that they are not especially useful after all. On the other hand, Ethereum 2.0 is a very good illustration of the general principle mentioned above in connection with Solana: you cannot retrofit sharding and scalability into a blockchain project originally designed without these issues in mind.

As of 2022, TON Blockchain remains one of the few truly scalable blockchain projects. As such, it still is the most advanced blockchain project ("fifth generation" in terms of the original whitepaper), capable of performing millions and, if becomes necessary in the future, tens of millions of true Turing-complete smart contract transactions per second, requiring only minor internal changes. It aged surprisingly well in the five years since its inception, still remaining at the cutting edge of general-purpose blockchain technology.

Since 2017, the efficiency of the architectural approach proposed in the TON Whitepaper has been further validated by the demonstratedly high performance of various testnets and mainnets based on the implementations of the TON technology developed in the last few years.

References

[1] *TON Whitepaper*, 2017, currently available at `https://ton-blockchain.github.io/docs/ton.pdf`

[2] *Solana: A new architecture for a high performance blockchain v0.8.13*, `https://solana.com/solana-whitepaper.pdf`

[3] *Tower BFT: Solana's High Performance Implementation of PBFT*, 17.07.2019, `https://medium.com/solana-labs/tower-bft-solanas-high-performance-implementation-of-pbft-464725911e79`

[4] *8 Innovations that Make Solana the First Web-Scale Blockchain*, 29.07.2019, `https://solana.com/news/8-innovations-that-make-solana-the-first-web-scale-blockchain`

[5] *Explained: How a system outage disrupted Solana's massive rally*, 21.09.2021, `https://www.cnbctv18.com/cryptocurrency/solana-sol-token-solana-outage-cryptocurrency-10822571.htm`

[6] *The Beacon Chain Ethereum 2.0 explainer you need to read first*, 23.05.2020, https://ethos.dev/beacon-chain/

[7] *Ethereum upgrades: Shard chains*, https://ethereum.org/en/upgrades/shard-chains/

[8] *EOS.IO technical white paper*, https://github.com/EOSIO/Documentation/blob/master/TechnicalWhitePaper.md

[9] S. Larimer, *The history of BitShares*, 2013, https://docs.bitshares.org/bitshares/history.html

[10] M. Castro, B. Liskov, et al., *Practical byzantine fault tolerance*, Proceedings of the Third Symposium on Operating Systems Design and Implementation (1999), p. 173; available at http://pmg.csail.mit.edu/papers/osdi99.pdf

[11] Crtomir Ipavec, *Solana*, 15.06.2021, https://medium.com/crypto-articles-randomly/solana-9c432a1b84a8

[12] https://docs.solana.com/developing/on-chain-programs/overview

[13] https://docs.solana.com/developing/programming-model/transactions