Vilnius university
Institute of Data Science and
Digital Technologies
LITHUANIA

INFORMATICS ENGINEERING (T007)

ANALYSIS OF THE SCALABILITY SOLUTIONS FOR PROOF-OF-WORK BASED BLOCKCHAINS

Rytis Bieliauskas

October 2020

Technical Report DMSTI-DS-T007-20-01

VU Institute of Data Science and Digital Technologies, Akademijos str. 4, Vilnius
LT-08412, Lithuania
www.mii.lt

Abstract

In this paper, we explain the need for scalability of proof-of-work blockchains and review the methods to it, by analyzing the amount of transactions per second increase, privacy implications, influence to decentralization of the network, and current status of available or in-development scalability solutions. We then analyze and review existing articles regarding this subject.

Keywords: Blockchain, Bitcoin, proof-of-work, scalability, Lightning Network, Segregated Witness.

I. Introduction:

Blockchain is one of the most important innovations in recent times. In the last decade, blockchain progressed from being a domain and interest of cryptographers and computer scientists, to mainstream technology used in various industries for a wide array of purposes. While the usage of the blockchain technology varies, the main purpose why it's used is payments and financial transactions. For proof-of-work blockchains to become a viable payment method used worldwide by billions of people, it has to be able to support a huge number of transactions per second.

According to the Federal Reserve Bank of Atlanta, US consumers make an average of 70 transactions per month. [1] There are around 5.85 billion people over the age of 15 in the world. [2] [3] [4] Assuming all of them would want to make the same amount of transactions per month as the average US consumer, this adds up to a need to be able to process around 150,000 transactions per second, for consumer payments alone.

Scalability is dependent on several factors: maximum throughput – the maximum rate of how many transactions can be confirmed by the network, latency – how quickly transactions are confirmed,, cost per confirmed transaction – the cost in USD for confirming a single transaction, [5], bootstrap time – the time it takes to launch and synchronize a new node [6].

In this article, we will review the existing articles regarding scalability of proof-of-work blockchains. We then describe various scalability methods on a technological level and compare them according to several criteria. The main contribution of this article is focusing on how different scalability methods would affect the decentralization, privacy, and security of proof-of-work blockchain, and what improvements in regards of increased number of transactions per second processed they bring.

II. Articles about scalability of proof-of-work blockchains:

There have been many articles reviewing and detailing the scalability problem, and different methods to solve it. However, most of the currently existing articles either review only one or few of the scaling methods, or do not include in-depth and accurate information regarding them. Some also explore scalability methods which make the blockchain no longer proof-of-work or no longer decentralized. In the table below, we give a score for each article based on whether they review a particular scaling method and how in-depth and accurate information in the article is.

Regarding increasing the number of transactions per second being able to be processed by the network, "A Comprehensive Study on the Scalability Challenges of the Blockchain Technology", E. Ademi et al. only mentions increasing the block size – as a direct increase, and as part of Segregated Witness. It's important to note, that blockchain scalability in general is not limited to an increased number of transactions being able to be processed, but also to various other parameters, however, they are outside the scope of our article.

“On scaling decentralized blockchains (A position paper)”, K. Croman *et al.*, also mainly explores other aspects of scalability, which do not relate to increasing the number of transactions being processed, and only shortly discusses scalability methods related to that, such as sharding.

While “The Road to Scalable Blockchain Designs Functional Components of a Blockchain”, S. Bano et al. discusses relevant methods more intentionally, it does so quite superficially and only talks about a few methods, such as sharding and sidechains.

“Public blockchains scalability: An examination of sharding and segregated witness”, A. Singh et al. is a much more thorough and technical article, detailing various scalability methods, comparing them to each other, and going into detailed explorations of how specific scalability methods would affect existing blockchains in general, as well as specific parts of blockchains. This article explores the majority in scalability methods, including blocksize increase, transaction size reduction, sharding, and sidechains.

State channels and payment channels are explored in great detail in C. Buckland, S. Bakshi, K. Wüst, and A. Miller, “You Sank My Battleship! A Case Study to Evaluate State Channels as a Scaling Solution for Cryptocurrencies,” P. McCorry et al. The article explores using state and payment channels to create a game of battleship using the Ethereum blockchain, going into great detail of potential costs, advantages, and disadvantages of such an implementation.

“Privacy in bitcoin transactions: New challenges from blockchain scalability solutions”, J. Herrera-Joancomartí et al. explores privacy in proof-of-work blockchains, specifically Bitcoin, and how privacy would be affected by scaling Bitcoin by increasing it’s blocksize, vs using off-chain methods such as state channels and payment channels.

In “Anonymous Multi-Hop Locks for Blockchain Scalability and Interoperability”, G. Malavolta et al. explore how off-chain channel networks could be used to improve scalability, their effects on the privacy of the users, and how such networks would interoperate with each other. The article, while not extremely long, is very detailed and technically oriented“.

“A Survey of Scalability Solutions on Blockchain”, S. Kim et al. reviews all the previously mentioned scalability methods: sharding, transaction size reduction, blocksize increase, state and payment channels, and sidechains. This is the most thorough article, which explores all the aforementioned scalability methods, compares them by various attributes, such as how they affect privacy, security, transaction throughput. While the article explores all scalability methods, state channels, payment channels, and sidechains, are the main focus of the article.

Review of articles according to information thoroughness and correctness regarding specific scaling methods.

The possible scores are as follows:

1 – No information, or only a mention of the name of the scaling method.
2 – A short description of the scaling method.
3 – A detailed explanation of the scaling method.

4 – In-depth explanation of the scaling method with no inaccuracies.

	Increasing the block size	Reducing the transaction size	Sharding	State channels	Payment channels	Sidechains
[6]	3	1	1	1	1	1
[13]	1	1	3	1	1	1
[14]	1	1	2	1	1	2
[15]	3	4	4	1	1	2
[16]	1	1	1	4	4	1
[27]	4	1	1	3	4	1
[26]	1	1	1	4	4	2
[24]	2	2	2	4	4	4

Accuracy and completeness of scalability method information in articles

III. Scalability methods:

In this chapter we explain the technological workings of existing and planned scalability methods, review current and historical attempts to implement them, and compare different scalability methods based on how they affect decentralization, privacy, and security of a blockchain on which they are (or would be) implemented.

Increasing the block size

Originally Bitcoin's block size did not have an explicit limit. A limit of 1 MB was introduced in 2010, by Satoshi Nakamoto. [7] [8] On 2015 December 21st, on BIP141 proposal called Segregated Witness was officially formulated, which removed the 1 MB block size limit, and replaced it with a 4 MB block weight limit, with block weight being calculated by multiplying block size in bytes with the original transaction serialization without any witness-related data by three and adding block size in bytes with transactions serialized as described in BIP144. [9] [10] This results in a practical block size limit of around 2 MB if all transactions are the new Segregated Witness type, as explained by Jimmy Song in his book "Programming Bitcoin". [11] Segregated Witness was activated on 2017 August 24th. [12]

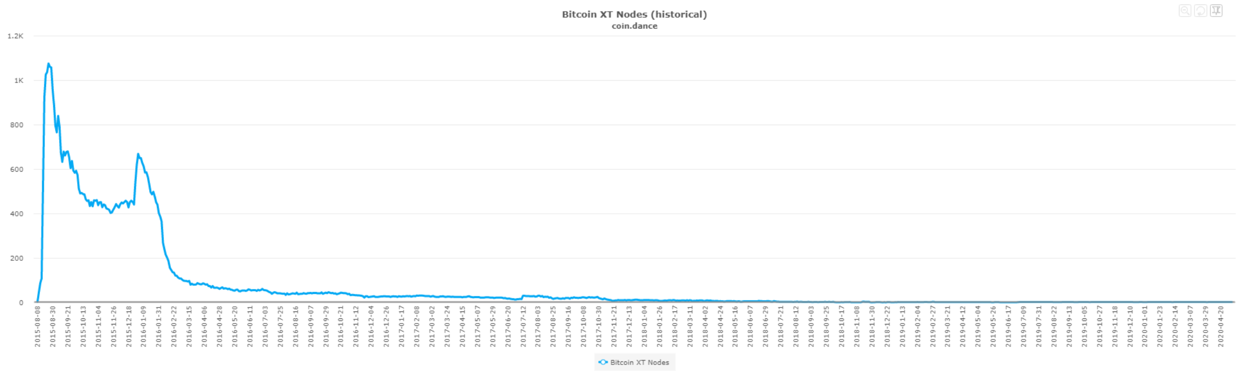
Some of the articles regarding Bitcoin and other proof-of-work cryptocurrencies scalability make the claim that the block size limit in Bitcoin is still 1 MB, either because they were written before the 1 MB limit was removed, or, some that were written after the 1 MB limit removal, because they take the data from outdates sources. More importantly, related to that, some articles state that Bitcoin can process 7 transactions per second, which is a number based on the block size limit of 1 MB. [13]; [14]; [15]; [16].

This is the most straightforward method of scaling a proof-or-work blockchain. It has been a contentious issue and source for years long debates in the Bitcoin community, whether the Bitcoin block size limit, which was 1 MB at the time, should be raised. With 1 MB block size limit and the blocks being almost always full, my node, running on Raspberry Pi 3 Model B+ (1.4GHz 64-bit quad-core processor, 1GB RAM, Gigabit Ethernet over USB 2.0, default settings) would use up around 1.8 TB of bandwidth per month. Raising the limit to 2 MB, assuming the blocks are always or almost always full, would at least double the bandwidth requirement, and according to

[17], increasing bandwidth requirements by more than a factor of 1.7, would lead to decreased level of decentralization.

Historically, there were several attempts to force the original Bitcoin to increase block size limit, or to create a hard fork of Bitcoin with an increased block size limit.

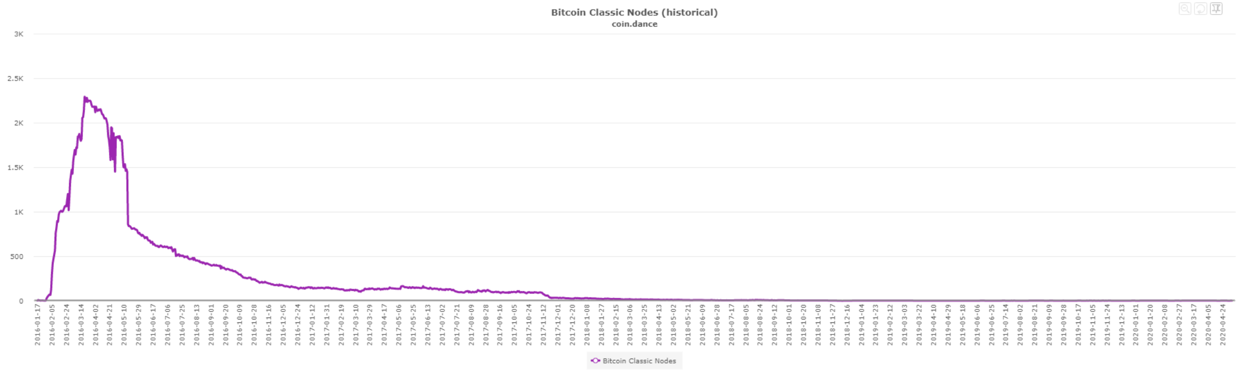
The first notable attempt was called Bitcoin XT, which was a Bitcoin node implementation that would have forked an increased block size limit of 8 MB (from 1 MB), if and when a 75% hash rate threshold was met. This was later changed to 2 MB on 2016 January 28th. The threshold was never met, and the main developers of the project announced they are abandoning it on 2019 May 8th. The latest commit to Bitcoin XT codebase was on 2018 December 19th. As of 2020 May 2nd, there were a total of 3 listening (accepting inbound connections) Bitcoin XT nodes running on the Bitcoin network.



Number of Bitcoin XT nodes (historical chart) [18]

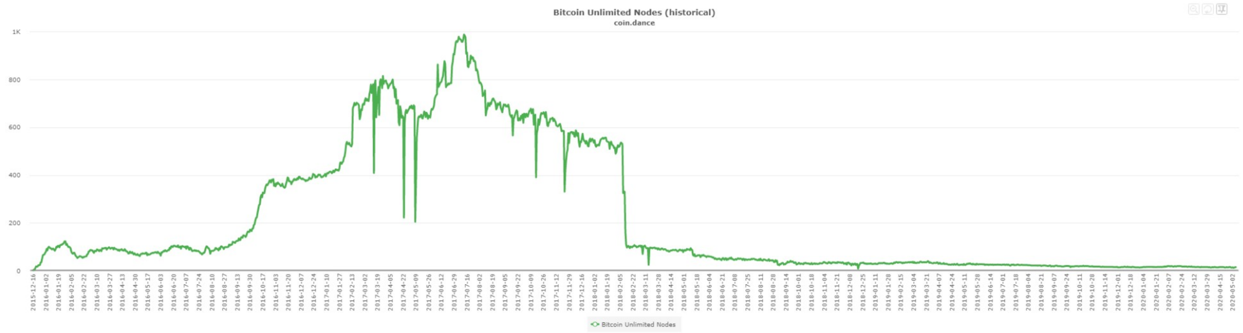
There were several similar unsuccessful attempts afterwards.

Another notable attempt was Bitcoin Classic, which promoted an increase of block size limit to 2 MB, and later discontinued. The latest commit to Bitcoin Classic codebase was on 2017 October 18th. As of 2020 May 2nd, there were a total of 2 listening Bitcoin Classic nodes running on the Bitcoin network).



Number of Bitcoin Classic nodes (historical chart) [19]

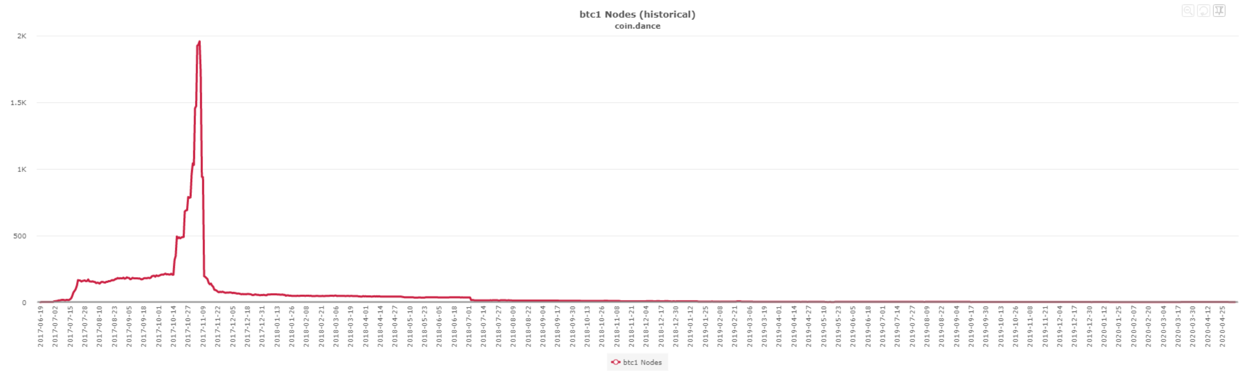
The first successful attempt was Bitcoin Unlimited, a Bitcoin node implementation, which ultimately forked from the Bitcoin network on 2017 August 1st and became known as Bitcoin Cash. It initially increased the block size limit from 1 MB to 8 MB, which was further increased to 32 MB on 2018 May 15th. However, it failed to influence the block size limit on the Bitcoin network. As of 2020 May 2nd, there were a total of 15 listening Bitcoin Unlimited nodes running on the Bitcoin network).



Number of Bitcoin Unlimited nodes (historical chart) [20]

A few weeks after the Bitcoin Cash fork, on 2017 August 24th, a technological update called Segregated Witness was activated on the original Bitcoin blockchain. It removed the 1 MB block size limit and replaced it by a 4 MB block weight limit. In practice, this resulted in around 1.7 MB size limit for blocks, and therefore, an increase to support of around 12 transactions per second.

After Segregated Witness activated, there was an additional attempt to increase the newly introduced block weight limit to 8 MB. This attempt was known as SegWit2x, and used a Bitcoin node implementation called btc1. The latest commit to btc1 codebase happened on 2017 July 21th. As of 2020 May 2nd, there were a total of 4 listening btc1 nodes running on the Bitcoin network.



Number of btc1 nodes (historical chart) [21]

Processing transactions off-chain can be done either in a centralized or decentralized manner. Centralization would negate the whole point of blockchain – decentralization of control – therefore it is not an acceptable method.

An interesting thing to observe in the graphs of all those failed attempts is the very quick rise, and even more quicker – almost instantaneous – drop of large numbers of nodes, which gives suspicion that in each of those situations, all the nodes that dropped off the network at the exact same time were run by the same person or group of people, perhaps to artificially manipulate the appearance of support for that particular implementation.

Reducing the transaction size

Compressing public keys. Bitcoin originally used 65-byte uncompressed public keys to identify the owner of a set of bitcoins. On 2012 March 30, a way to compress the public keys to 33 bytes was introduced in the most popular Bitcoin node implementation now called Bitcoin Core. This change allowed to reduce a typical transaction (1 input, 2 outputs) from about 258 bytes to 226 bytes.

Batching transactions. Each Bitcoin transaction must include some information which does not depend on the amount of inputs and outputs in the transaction. As Bitcoin allows adding virtually unlimited numbers of outputs to a transaction, it takes less block space to send one transaction to two people at the same time by adding an additional output, than to send two separate transactions. The amount of bytes saved is at least 79 bytes for each additional output instead of a separate transaction.

Signature aggregation using Schnorr signatures. Schnorr signatures allow making only one signature for the whole transaction, instead of making a separate signature for each input. This can be used even if inputs come from different addresses.

Sharding

Sharding is a technique used to partition distributed databases, such as Dynamo, MongoDB, MySQL, and BigTable, which can be used on blockchains as well. [13] It has been proposed to be used on the Ethereum blockchain. [22] As Ethereum works currently, each full node validates all transactions and stores all information about all transactions ever occurred. This gives a high amount of security and decentralization, at the cost of scaling. Blockchain sharding is a method the entire state of the whole network is split into independent pieces of state and transaction history (called “shards”), and specific nodes validate and store transactions only for specific shards. [15] This allows the amount of transactions being able to be processed by the network to be increased linearly, depending on how many shards the network is split into. Participants belonging to the same shard can transact as before, and a separate protocol is used for communication between shards. It is expected, that in a traditional UTXO model (like the one used in Bitcoin), more than 90% of transactions would happen between different shards. For an account/balance model (like the one used in Ethereum), this number can still reach 90% if the number of shards is more than 64. This approach introduces complexity and additional attack vectors, one of them being an ability to take over a shard, by controlling a relatively low number of nodes. This can be prevented by assigning nodes to particular shards randomly. [23]

State channels and payment channels

State channels are a technology which allows users to transact outside the blockchain (off-chain) and only synchronize (settle) the final state of their transactions to the main blockchain. State channels support not only payment type transactions, but also general “state updates”, which enables various types of non-financial transactions to be performed off-chain, while still keeping the security provided by the blockchain.

Payment channels are a subset of state channels, allowing only transactions where cryptocurrency is transferred from one address to another to be sent, therefore reducing complexity and possible attack vectors. There are several types of payment channels available [16]

Different types of payment channels

Spillman-style payment channels. Implemented in BitcoinJ, Spillman-style payment channels require one transaction to open the channel and one transaction to close it. These payment channels are unidirectional, meaning that only user A would be able to send funds to user B, but not vice versa, unless a second channel is opened. Spillman-style payment channels also have a specific expiration date, and the received of the funds must close the channel before it expires.

CLTV-style payment channels. Similar to Spillman-style payment channels, in the way that CLTV-style payment channels are also unidirectional and have a specific expiration date. However, CLTV-style payment channels have a technological improvement (described in BIP65) which makes them resistant to transaction malleability problems, which are inherent in Spillman-style payment channels. [24]

Decker-Wattenhofer duplex payment channels. A duplex payment channel is composed of two unidirectional payment channels, where each unidirectional channel is in essence a Spillman-style payment channel, but these channels use relative lock time (nSequence) instead of nLockTime. While Decker-Wattenhofer duplex payment channels do not have an expiration time, they have to be “reset” each time one of the unidirectional channels comprising it uses up all balance, and the number of “resets” is limited.

Decker-Russell-Osuntokun eltoo payment channels. This method of creating payment channels does not require punishment like Poon-Dryja payment channels do, but it requires a new feature to be implemented in the main blockchain. These channels are bidirectional and while the amount of transactions being able to be sent using the channel before it needs to be closed and reopened is limited, the limit is exceptionally high (about 1 billion).

Poon-Dryja payment channels. These are bidirectional payment channels with no time limit, and no usage limit, which implement a punishment system for any participant trying to cheat. If they are “caught” by another participant, they risk losing all of their funds in the channel. Poon-Dryja payment channels are used in the currently most popular off-chain scaling solution – the Lightning Network. [17]

The Lightning Network is an upgrade for the Bitcoin network (or any other compatible cryptocurrency network), which uses payment channels to enable users of Bitcoin to perform transactions between each other, without immediately settling them on the Bitcoin blockchain, and without having to trust any third-party to do that. The Lightning Network enables Bitcoin to scale far beyond traditional payment networks capabilities, by allowing basically unlimited number of transactions per second in Bitcoin to be performed. In addition, Lightning Network transactions can be trusted immediately (in less than 1 second), which is much faster compared to on-blockchain Bitcoin transactions, which require (on average) 5 – 15 minutes to become trusted (irreversible). The Lightning Network could not exist without the main Bitcoin blockchain, which acts as a settlement layer, and the ultimate security guarantee for any (both Lightning Network and on-blockchain) Bitcoin transactions. The Lightning Network enables using payment channels in the following way.

User A opens a payment channel with user B on the Lightning Network. One on-blockchain Bitcoin transaction is required to open the payment channel between these two users. While the payment channel is open, user A and user B can send an unlimited number of transactions to each other (there may be practical limitations of internet bandwidth throughput and payment channel balance at a particular moment, but fundamentally the number of transactions that can be sent between two users is not limited). If user B has a payment channel open with user C, user A can make a payment to user C as well, even if user A themselves do not have a direct payment channel to user C. This can be extended to include unlimited number of users. At any time, the payment channel can be closed by both users agreeing to close the payment channel. One on-blockchain Bitcoin transaction is required to close the payment channel between the two users. When the payment channel is closed, all payments made from user A to user B (and vice versa) are settled on the Bitcoin blockchain. The Lightning Network includes mechanisms to allow users to close the payment channel unilaterally, in case the other party becomes unresponsive.

The Lightning Network also includes mechanisms to prevent cheating (double-spending). All of this is achieved in a completely decentralized way, without introducing any central authorities, trusted third-parties, etc. This is done by forming transactions exchanged between participants in such a way, that if one party tried to publish an older version, the other party would be able to mathematically prove they have a newer version of the transaction, and that would allow them to take **all** the funds in that payment channel. This highly disincentivizes attempts to cheat, as failure would mean the dishonest party would lose all of their funds in the payment channel. [25]

The Lighting Network has other benefits as well, like improving privacy of users, enabling micro-transactions (the smallest amount which can be sent on the Bitcoin network is 0.00000001 BTC; the smallest amount which can be sent on the Lightning Network is 0.000000000001 BTC).

The Lighting Network also enables decentralized cryptocurrency swapping (trading) between different compatible blockchains (for example Bitcoin and Litecoin), i.e. it enables users to trade cryptocurrencies without requiring an exchange platform. [26]

Scaling method	**Scaling solution type**	**Decentralization**	**Transactions per second supported**	**Privacy**	**Used in**	**Security**
Increasing block size	On-chain	Decreases	Amount of transactions supported increases linearly	Does not change	Bitcoin, Bitcoin Cash, Bitcoin SV	Decreases because of increased centralization
Reducing transaction size	On-chain	Does not change	Increases by % of the amount tx size is reduced	Does not change	Bitcoin	Does not change
Sharding	On-chain	Decreases	Amount of transactions supported increases linearly	Does not change	Ethereum (planned)	Decreases because of increased complexity and additional attack vectors
State channels	Off-chain	Does not change	Virtually no limit to increase	Improved	Raiden Network, Perun, Nitro Protocol, PISA, L4, Celer Network	Does not change
Payment channels	Off-chain	Does not change	Virtually no limit to increase	Improved	Lightning Network	Does not change
Sidechains	On-chain	Does not change	Increases linearly	Can be improved, depending on the sidechain	Rootstock, Liquid	Does not change

Attributes and consequences of different scalability methods

IV. Conclusion:

At the current moment, scientific consensus is that if proof-of-work blockchains are to be scaled to current credit card processor levels, it can only be done using off-chain scalability methods. and that the most efficient method to do that without sacrificing decentralization and other benefits of proof-of-work blockchains, are payment channels. This is reflacted in the amount of research done and articles written about the subject. While some articles disagree which specific implementation of the payment channels should be preferred, most mention the Lightning Network as the current frontrunner. As the cryptocurrency space continues evolving extremely quickly, re-evaluation of newest technologies and scalability methods those technogies enable is constantly needed, in order to not fall behind of the newest innovations created.

References:

[1] M. Angrisani, K. Foster, and M. Hitczenko, “The 2016 and 2017 Surveys of Consumer Payment Choice: Technical Appendix,” *Fed. Reserv. Bank Atlanta Consum. Payments Res. Data Reports*, 2018, doi: 10.29338/rdr2018-04.

[2] “Population ages 0-14 (% of total population) | Data.” https://data.worldbank.org/indicator/SP.POP.0014.TO.ZS (accessed May 10, 2020).

[3] “Population ages 15-64 (% of total population) | Data.” https://data.worldbank.org/indicator/SP.POP.1564.TO.ZS (accessed May 10, 2020).

[4] “Population ages 65 and above (% of total population) | Data.” https://data.worldbank.org/indicator/SP.POP.65UP.TO.ZS (accessed May 10, 2020).

[5] P. W. Eklund and R. Beck, “Factors that impact blockchain scalability,” in *11th International Conference on Management of Digital EcoSystems, MEDES 2019*, Nov. 2019, pp. 126–133, doi: 10.1145/3297662.3365818.

[6] E. Ademi, “A Comprehensive Study on the Scalability Challenges of the Blockchain Technology,” 2018.

[7] “fix openssl linkage problems, · bitcoin/bitcoin@a30b56e · GitHub.” https://github.com/bitcoin/bitcoin/commit/a30b56eb (accessed May 10, 2020).

[8] “don’t count or spend payments until they have 1 confirmation, · bitcoin/bitcoin@a790fa4 · GitHub.” https://github.com/bitcoin/bitcoin/commit/a790fa46f40 (accessed May 10, 2020).

[9] “bips/bip-0141.mediawiki at master · bitcoin/bips · GitHub.” https://github.com/bitcoin/bips/blob/master/bip-0141.mediawiki (accessed May 10, 2020).

[10] “bips/bip-0144.mediawiki at master · bitcoin/bips · GitHub.” https://github.com/bitcoin/bips/blob/master/bip-0144.mediawiki (accessed May 10, 2020).

[11] J. Song, *Programming Bitcoin*. .

[12] “Why I Was Wrong About Segwit And Big Blocks - Jimmy Song - Medium.” https://medium.com/@jimmysong/why-i-was-wrong-about-segwit-and-big-blocks-405b0860dacb (accessed May 10, 2020).

[13] K. Croman *et al.*, “On scaling decentralized blockchains (A position paper),” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2016, vol. 9604 LNCS, pp. 106–125, doi: 10.1007/978-3-662-53357-4_8.

[14] S. Bano, M. Al-Bassam, and G. Danezis, “The Road to Scalable Blockchain Designs Functional Components of a Blockchain,” *;Login:*, vol. 42, no. 4, pp. 31–36, 2017, Accessed: May 10, 2020. [Online]. Available: www.usenix.org.

[15] A. Singh, R. M. Parizi, M. Han, A. Dehghantanha, H. Karimipour, and K. K. R. Choo, "Public blockchains scalability: An examination of sharding and segregated witness," in *Advances in Information Security*, vol. 79, Springer, 2020, pp. 203–232.

[16] P. McCorry, C. Buckland, S. Bakshi, K. Wüst, and A. Miller, "You Sank My Battleship! A Case Study to Evaluate State Channels as a Scaling Solution for Cryptocurrencies," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Feb. 2020, vol. 11599 LNCS, pp. 35–49, doi: 10.1007/978-3-030-43725-1_4.

[17] A. E. Gencer, S. Basu, I. Eyal, R. van Renesse, and E. G. Sirer, "Decentralization in Bitcoin and Ethereum Networks," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Feb. 2018, vol. 10957 LNCS, pp. 439–457, doi: 10.1007/978-3-662-58387-6_24.

[18] "Coin Dance | Bitcoin XT Nodes Summary." https://coin.dance/nodes/xt (accessed Oct. 17, 2020).

[19] "Coin Dance | Bitcoin Classic Nodes Summary." https://coin.dance/nodes/classic (accessed Oct. 17, 2020).

[20] "Coin Dance | Bitcoin Unlimited Nodes Summary." https://coin.dance/nodes/unlimited (accessed Oct. 17, 2020).

[21] "Coin Dance | Bitcoin Nodes Summary." https://coin.dance/nodes/btc1 (accessed Oct. 17, 2020).

[22] M. Schäffer, M. di Angelo, and G. Salzer, "Performance and Scalability of Private Ethereum Blockchains," in *Lecture Notes in Business Information Processing*, Sep. 2019, vol. 361, pp. 103–118, doi: 10.1007/978-3-030-30429-4_8.

[23] G. Wang, Z. Jerry Shi, M. Nixon, and S. Han, "SoK: Sharding on Blockchain," 2019, doi: 10.1145/3318041.3355457.

[24] S. Kim, Y. Kwon, and S. Cho, "A Survey of Scalability Solutions on Blockchain," in *9th International Conference on Information and Communication Technology Convergence: ICT Convergence Powered by Smart Intelligence, ICTC 2018*, Nov. 2018, pp. 1204–1207, doi: 10.1109/ICTC.2018.8539529.

[25] S. Bartolucci, F. Caccioli, and & Pierpaolo Vivo, "A percolation model for the emergence of the Bitcoin Lightning network," doi: 10.1038/s41598-020-61137-5.

[26] G. Malavolta, P. Moreno-Sanchez, C. Schneidewind, A. Kate, and M. Maffei, "Anonymous Multi-Hop Locks for Blockchain Scalability and Interoperability," Mar. 2019, doi: 10.14722/ndss.2019.23330.

[27] J. Herrera-Joancomartí and C. Pérez-Solà, "Privacy in bitcoin transactions: New challenges from blockchain scalability solutions," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2016, vol. 9880 LNAI, pp. 26–44, doi: 10.1007/978-3-319-45656-0_3.