Journal of Network and Computer Applications 195 (2021) 103232

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

A systematic review of blockchain scalability: Issues, solutions, analysis and future research

Abdurrashid Ibrahim Sanka *, Ray C.C. Cheung

Department of Electrical Engineering, City University of Hong Kong, Hong Kong

ARTICLE INFO

Keywords:
Blockchain
Scalability
Throughput
Latency
Solutions
Analysis

ABSTRACT

Blockchain is an inspiring emerging technology that takes much attention from various researchers and companies. The technology offers various benefits such as data security, autonomy, immutability, transparency, and auditability. Hence, blockchain is getting large adoptions for various applications besides cryptocurrencies. Despite these benefits, scalability is a big challenge to blockchain impeding its mainstream adoption. This paper gives a systematic review of blockchain scalability. We follow a systematic process to investigate the research trend on blockchain scalability and review its state of the art. We review the various proposed solutions and methods for blockchain scalability. We also review the performance analysis of blockchain systems. We assess the proposed scalability solutions, deduce future research directions on the blockchain scalability, and finally discuss the blockchain adoption. We hope this paper will serve as a guide for learning and research on blockchain scalability.

1. Introduction

Blockchain is a powerful emerging distributed ledger technology that provides many benefits of data security, autonomy, transparency, auditability, privacy, immutability, efficiency, speed, and cost savings. It evicts central authorities and facilitates the creation of autonomous, secure, and transparent systems with the provision of trust among non-trusting entities. Many companies, consortiums, and countries have currently incorporated blockchain into their systems for its benefits after successful trials (Makhdoom et al., 2019; Hewa et al., 2020). A survey by Deloitte (Pawczuk et al., 2019) revealed that blockchain will eventually reach mainstream adoption as a large number of its projects are now in the production stage. By 2025 and 2030, the business value of blockchain was forecasted by Gartner (Furlonger and Valdes, 2017) to be over 176 billion and 3.1 trillion USD, respectively. Another report by Cisco (Cisco, 2018) also forecasted that 10% of the global GDP will be on blockchain by 2027.

The success of blockchain was first seen in Bitcoin which is the most successful cryptocurrency. Blockchain underpins Bitcoin, Ethereum, and about 1200 more cryptocurrencies. Several other applications of blockchain besides cryptocurrencies exist in smart contracts, banking, insurance, supply chain, healthcare, registry, identity management, banking, stock marketing, IoT, energy, intellectual property, and more (B et al., 2020; Chen et al., 2020a; Liu et al., 2020a).

Despite all its success and strength, scalability is the major challenge that hinders the full adoption of blockchain in some areas (Singh

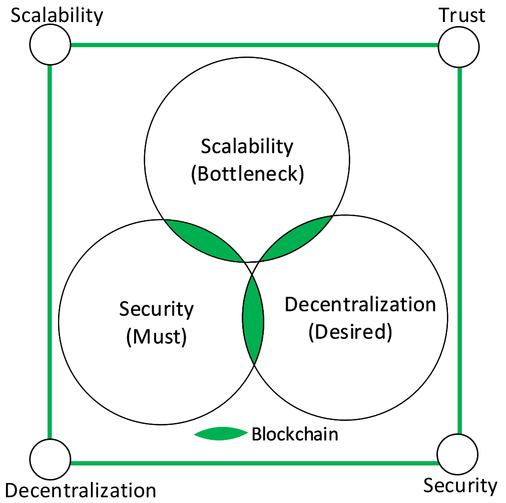

Fig. 1. Blockchain Scalability Quadrilemma (outer-square) and Trilemma (inner-diagram).

[*] Corresponding author.
E-mail addresses: iasanka2-c@my.cityu.edu.hk (A.I. Sanka), r.cheung@cityu.edu.hk (R.C.C. Cheung).

https://doi.org/10.1016/j.jnca.2021.103232
Received 2 December 2020; Received in revised form 13 August 2021; Accepted 23 September 2021
Available online 2 October 2021

et al., 2020). Blockchain systems have low throughput and latency performance compared to non-blockchain systems. For example, the throughput of Bitcoin and Ethereum blockchains are 3–4 and 15 transactions per second (TPS) respectively. In comparison, Visa and PayPal achieve 1667 and 193 TPS respectively. Besides its huge storage data, the read-performance of blockchain servers is also low compared to that of non-blockchain servers such as YouTube and Google.

There are several efforts and proposals on improving the scalability of blockchain. However, it is difficult to solve the blockchain scalability issues without compromising either the security, decentralization, or trust of the blockchain. There is always a tradeoff between security, scalability, decentralization, and trust in blockchain (blockchain quadrilemma as shown in Fig. 1). The blockchain quadrilemma issue is discussed in more detail in Section 2.4.

There are surveys and review papers on blockchain scalability. However, the existing related papers focus on one or a few specific aspects of blockchain scalability such as sharding. Only a few of them cover more but not all aspects of the scalability write-performance solutions. Other scalability aspects especially the read-performance, storage issues, and performance analysis are untreated. Singh et al. (Singh et al., 2020) is a good survey on sidechain technologies. Yu et al. (2020) also gave a good survey of blockchain sharding solutions. Hafid et al. (2020) discussed the solutions to blockchain scalability also focusing on sharding. Kim et al. (2018) was a brief survey on some scalability write-performance solutions. Zhou et al. (2020) is also a survey on the blockchain write-performance solutions. Eklund and Beck (2019) highlighted some of the factors that have an impact on the scalability of blockchain in relation to the consensus mechanisms and network patterns. Bai (2019) is a survey that discussed and compared the performances of Bitcoin, Ethereum, and DAG blockchains (IOTA, and Byteball). Other related surveys include (Xie et al., 2019; Mazlan et al., 2020; Mahony and Popovici, 2019).

In contrast to the existing surveys, this paper gives a wider review of the whole blockchain scalability studies covering both write-performance, read performance, storage solutions, and performance analysis. We also follow a systematic review process to identify the various researches and the research trend on blockchain scalability. Various databases were searched for academic and gray area papers. Based on our findings, we classify the blockchain scalability studies into three, namely scalability solutions, performance analysis, and reviews/surveys. We further classify the scalability solutions into write-performance, read-performance, and storage scalability solutions. Furthermore, we classify the write performance solutions into five groups according to the blockchain ecosystem model that we propose. We deduce future research directions on blockchain scalability and finally discuss the adoption of the blockchain technology. Our contributions are summarized as follows:

- We give an overview of blockchain and its scalability issues whereby we propose a five-layer conceptual model for the blockchain ecosystem.
- We conduct a systematic review process to investigate the research trend and state of the art on blockchain scalability.
- We classify the various proposed blockchain scalability solutions and performance analyses.
- We give a comprehensive review of the proposed blockchain scalability solutions and performance analyses.
- We deduce future research directions and opportunities on blockchain scalability.

The organization of the rest of the paper is as follows: Section 2 entails the background of blockchain, proposal of the five-layer model for the blockchain ecosystem as well as the overview of the blockchain scalability issues. Section 3 entails the methodology of the systematic review process while Section 4 discusses the systematic review findings. Section 5 is a review of the blockchain write-performance scalability solutions. On the other hand, the review of the blockchain read-performance and storage solutions is given in Section 6. Section 7 reviews the blockchain performance analysis while Section 8 discusses the future research directions. Section 9 discusses the adoption of blockchain and finally, a conclusion is given in Section 10.

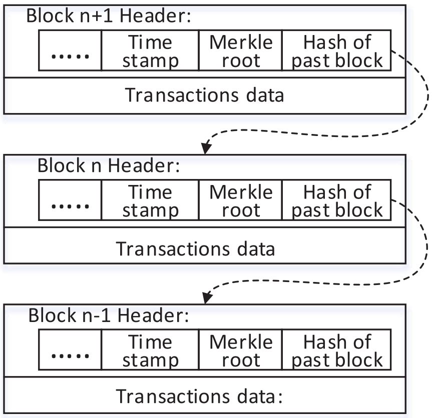

Fig. 2. Blockchain Structure.

2. Background

2.1. Blockchain description

Blockchain is a secure distributed ledger of interconnected blocks of data arranged in chronological order and maintained using consensus agreements. The nodes (computers) of the blockchain network have the same copy (duplicate) of the blockchain. The blocks are chained in such a way that each block references the hash of its previous block. In this way, blockchain data is protected against tampering. Any modification of blockchain data is detected since a change in any block will change the hash of the block which will differ from the previously stored hash in the next block. Hence, illegal tampering of the blockchain data is infeasible because it requires modification of the blocks on the majority of the network nodes.

Fig. 2 gives an overview of blockchain structure. Each block contains the block header and several transaction data (about 2000 for Bitcoin). The block header comprises the hash of the previous block, timespan, Merkle root of transactions, nonce, and other elements depending on the network. The transaction data contains the transactions in the block. All the block transactions are represented by the Merkle root in the block header. The Merkle root is the root of the tree of the hashes of all the transactions obtained by continuously hashing all the block transactions in pairs until a single hash value remains. Merkle root protects the transaction data against tampering since its value changes with any change in the transaction data.

Blockchain uses consensus protocol to create new blocks and maintain the network. A consensus protocol is an agreement between the network participants on how the network is maintained. It defines how the creator of a new block is selected. Proof of work is the most popular consensus protocol which is used in Bitcoin, Ethereum, and many cryptocurrencies. In PoW, special nodes called miners compete by continuously computing the hash of the block until a target result is obtained. The first node that gets the target result becomes the winner to create the new block. The competition repeats for subsequent blocks. Permissioned blockchains mostly use PBFT and Raft consensus which are voting-based. There are several other consensus mechanisms for blockchain. These include the Proof of Elapsed time (PoET), Proof of Stake (PoS), Delegated Proof of Stake (DPoS), Tendermint, Ripple, Proof of Burn (PoB), Proof of Capacity (PoC), Proof of Authority, and more.

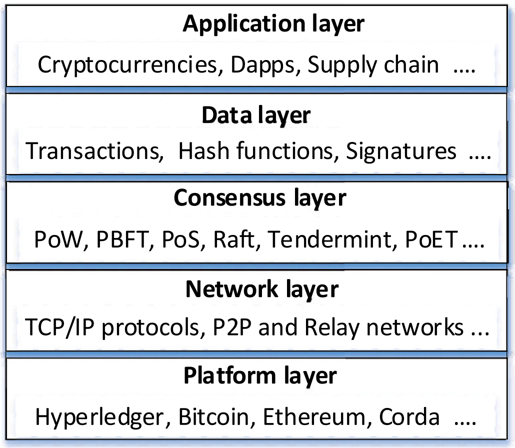
Application layer
Cryptocurrencies, Dapps, Supply chain ....
Data layer
Transactions, Hash functions, Signatures ....
Consensus layer
PoW, PBFT, PoS, Raft, Tendermint, PoET ....
Network layer
TCP/IP protocols, P2P and Relay networks ...
Platform layer
Hyperledger, Bitcoin, Ethereum, Corda ....

Fig. 3. Poposed Conceptual Model for Blockchain Ecosystem.

2.2. *Types of blockchain*

Blockchain is classified as either public (Open), private, or consortium blockchain (Cachin and Vukolić, 2017).

1. **Public Blockchain:**
A public blockchain is permissionedless. Anyone can join the blockchain network without prior permission and can participate in the network consensus with the full right of reading and writing the blockchain data. Bitcoin, Ethereum, and most cryptocurrencies are public blockchains.
2. **Private Blockchain:**
A private blockchain is permissioned in the sense that the participants require prior approval for joining the network. This type of network is centralized and mostly owned by a single organization where all the nodes are known and authorized. Multichain and BlockStack are examples of private blockchains.
3. **Consortium Blockchain:**
A consortium blockchain is also permissioned, its participants are known and require authorization to join the network. This type of network is formed by a group of organizations (consortium) that want to share data and have little or no trust among its members. A consortium blockchain is partially centralized. Hyperledger Fabric and Corda are examples of consortium blockchains.

2.3. *Conceptual model for blockchain ecosystem*

Similar to the OSI and TCP/IP networking models, we propose a five-layer conceptual model for blockchain ecosystem for a better description of its various components and architecture. The blockchain ecosystem consists of various components that perform different functions combined to achieve the blockchain technology. Hence blockchain technology is a collection of components and services. We classify these components with a five-layer conceptual model. The layers of the model are the application, data, consensus, network, and platform layers as shown in Fig. 3.

1. **Application Layer:**
This layer comprises the applications and use cases for which blockchain networks are created. They are the end products of a blockchain network that gives its actual value to the end-user. The application layer includes applications such as cryptocurrencies, Dapps, supply chain, and more.
2. **Data Layer:**
The data layer consists of components that make up the blockchain data and the cryptographic protocols used to create and verify the data. This layer includes data structures, transactions, block headers, databases, hashing algorithms, digital signatures, and zero-knowledge proofs. The data layer provides the basic ingredients of the creation and verification of blockchain data.
3. **Consensus Layer:**
The consensus layer provides the consensus protocols used to manage and maintain the blockchain. There are several consensus protocols used or proposed for blockchain. Proof of work (PoW) and Practical Byzantine Fault Tolerant (PBFT) are the most prominent consensus protocols mostly used in public and enterprise blockchains respectively. The other consensus protocols include the Proof of Stake(PoS), Proof of Elapsed Time (PoET), Raft, Tendermint, Ripple, Delegated Proof of Stake (DPoS), Proof of Capacity (PoC), and several others.
4. **Network Layer:**
The network layer consists of the protocols and services for propagating the blockchain data and messages among the network participants. This layer provides the means of communication between the blockchain network members. Hence, the blockchain network layer includes the networking protocols (such as the TCP/IP and gossip) and the networking software and hardware infrastructures such as the network stack and the Media access controller (MAC). The network layer also comprises the peer-to-peer network, relay networks, and data propagation algorithms such as data compression algorithms.
5. **Platform Layer:**
The platform layer provides the blockchain software and hardware framework and infrastructure using the above layers. It provides the technological backbone which serves as the back-end of the blockchain infrastructure. The platform layer includes several frameworks provided by different blockchain vendors. Blockchain end-users rely on these frameworks provided by the blockchain vendors. Examples of the frameworks provided in this layer include the Hyperledger Fabric, Bitcoin, Ethereum, and Multichain.

2.4. *Scalability issues in blockchain*

Scalability is a major challenge of blockchain technology despite its successes. The performance of blockchain systems in terms of throughput is low compared to non-blockchain systems. A typical example is that of Visa and Paypal compared to Bitcoin and Ethereum. Bitcoin and Ethereum handle 3–4 and 15 TPS respectively. On the other hand, Visa and Paypal can handle 1667 and 193 TPS respectively. Visa even claimed to support up to 56000 TPS at peak hours. Furthermore, the huge data of blockchain brings another storage scalability issue hindering the full adoption of blockchain in some applications especially the IoT. Currently, Bitcoin and Ethereum blockchains are over 280 GB and 562 GB respectively.

There is a common belief of the blockchain scalability trilemma (Altarawneh et al., 2020) now extended to quadrilemma as shown in Fig. 1. The blockchain trilemma (similar to the CAP theory of databases) means that the scalability, decentralization, and security of blockchain cannot perfectly coexist at the same time without compromising one of them. On the other hand, trust is very critical to blockchain scalability. However, there is also a tradeoff between trust and decentralization. Blockchains having trusted parties may adopt less complex consensus, communications, and computations to achieve higher scalability. Hence, the blockchain scalability trilemma is extended to quadrilemma with the addition of trust as shown in Fig. 1 (Fortino et al., 2020; Harz and Boman, 2019; Golan Gueta et al., 2019; Javaid et al., 2020; Li et al., 2021). Blockchain scalability quadrilemma is the tradeoff that exists between the scalability, decentralization, security, and trust in the current blockchain systems on top of the blockchain trilemma. It is very difficult to achieve these four properties at the same time in the current blockchain. For example, security and scalability are achieved in private and consortium blockchains which have fully trusted parties but are fully or partially centralized. Scalability and decentralization are achieved in DAG-based blockchains which are less secure with less trust. On the other hand, public blockchains have good security and decentralization but their scalability is weak (Altarawneh et al., 2020). Therefore, to achieve an optimum blockchain solution, optimum levels of security, decentralization, trust, and scalability must be determined.

The major scalability performances of blockchains are measured according to the transaction throughput/latency (write-performance), data read throughput/latency (read-performance), and data storage volume (storage performance). Some blockchain performance evaluations also measure success rate, as well as the CPU and network resources.

1. **Transaction Throughput/Latency (Write-Performance):**
The transaction throughput is the rate at which transactions are processed and added to the blockchain network. It is measured in transactions per second (TPS) and entails the entire network's throughput, not a single node's. Transaction throughput is the most important scalability performance measure of blockchain as it shows the processing power of the blockchain. Many people use TPS to measure the scalability or performance of blockchain solutions. Transaction throughput in blockchain depends on:

(a) **Block-Size:**
The throughput of blockchain increases with an increase in the number of transactions put in a block. However, due to security reasons, the block-size of most blockchain networks is limited. For example, the block-size in Bitcoin is 1 MB. The throughput could be increased by increasing the block-size but may lead to a security problem and increase the propagation delay. Very large blocks may allow denial of service attacks on the network. Hence an optimum tradeoff block-size needs to be determined to safeguard the security of the network.
(b) **Block Arrival Time (block-time):**
The throughput of a blockchain could be increased by sending blocks more frequently. The shorter the block-arrival time, the more transactions will be processed per second. However, there is a trade-off with the security of the network here. If blocks are processed so frequently, the chance of having fork increases as it will be difficult to synchronize the nodes since they have different processing power. With more forks, the chances of double spending and other attacks will also increase. Hence, an optimal tradeoff between the block arrival time and the network's security needs to be established.

Transaction latency on the other hand refers to the time taken from submission of a transaction to its addition to the blockchain. This latency actually depends on the propagation delay as well as the throughput of the network. In Bitcoin, six confirmations (about 1 h) are required before accepting payments due to security reasons (double spending). Other networks may require less confirmation. In most permissioned blockchains, transaction finality is achieved in the sense that once the transaction is added to the blockchain, it is the final decision and no more confirmation is needed.

2. **Read Throughput/Latency (Read-Performance):**
The read-performance of blockchain refers to the response of blockchain nodes upon request of data. Due to the huge size of the blockchain, many blockchain nodes such as the Simplified Payment Verification (SPV) nodes and IoT do not store the full blockchain data. Hence, they request this data from a full blockchain node (blockchain server). Therefore, the response throughput and latency of the blockchain server are important for maintaining the ecosystem of the blockchain network. In this regard, the read throughput is the number of blockchain data requests (queries) responded per second. On the other hand, read latency refers to the time to get the response of a blockchain data request from the time the request is sent. There are more instances where data can be requested from a full blockchain node. When a new node joins the network, it requests the blockchain data from the full nodes close to it. Likewise, when a particular node misses its data or goes down for some time, the missing data is requested from the full blockchain nodes. The read-performance of blockchain servers needs to be improved for better performance of blockchain systems.

3. **Storage Size (Storage-Performance):**
The huge size of the blockchain is obvious since the blockchain is a continuously growing ledger of append-only data blocks. As of August 2020, the Bitcoin and Ethereum blockchain sizes are over 280 GB and 562 GB respectively. This huge blockchain size hinders the full adoption of blockchain in many sectors especially the IoT and embedded systems whose devices have a small memory capacity that cannot store all the blockchain data. Blockchain users also find it inconvenient to store such a huge amount of data on their computers. This storage problem may also lead to a read-performance issue and make a large number of lightweight nodes depend on blockchain servers and putting so much workload on the servers.

Several pieces of research have been conducted and many proposals were made to scale blockchain for better scalability. We reviewed these scalability solutions in detail in Sections 5 and 6.

2.5. Enabling technologies for effective and bandwidth-efficient transmission of transactions in blockchain

Blockchain uses a peer-to-peer (P2P) network and several networking protocols and technologies for secure and bandwidth-efficient communications. In this section, we discuss the various technologies and methods for effective transmission of transactions with reduced requirements of bandwidth resources in the blockchain.

1. **Gossip and Remote Procedure Call (RPC) protocols:**
Since blockchain network is a P2P network and consists of a large number of nodes, it is inefficient and bandwidth-expensive for each node to directly transmit transactions to all other nodes of the network. Packet loss, churn, synchronicity requirement, and other constraints make randomized gossip algorithms more suitable for blockchains. Hence, blockchain uses randomized gossip protocols (Berendea et al., 2020; Boyd et al., 2006) to disseminate transactions for efficiency and bandwidth preservation. In addition to the gossip protocol, blockchain also uses remote procedure call (RPC) protocols especially for the communication between client apps and full blockchain nodes.
Gossip protocols are used to disseminate information across a large group of nodes just similar to the way epidemics spread across a community. A node passes the information to only its neighbors selected randomly or deterministically. Each neighbor after receiving the information transmits the information further to its neighbors. In this way, the information spreads across the whole network and can cover a wide range. Gossip is the major process of broadcasting transactions in a blockchain P2P network. It allows blockchains such as Bitcoin and Ethereum to reach wider coverage globally and efficiently (Berendea et al., 2020).
There are some pieces of works that proposed improvements to the existing gossip protocols used in blockchain. He et al. (2019) proposed an enhanced HNA-Gossip algorithm for blockchains using a semi-distributed structure. The proposal uses historical data of nodes to reduce the chances of duplicating gossip to a particular node. Berendea et al. (2020) also proposed an enhanced gossip algorithm for Hyperledger Fabric blockchain. The proposed gossip algorithm consumes lesser bandwidth (40% reduction) and has lesser propagation time and latency (10 times faster) compared to the exiting gossip algorithm used in the Hyperledger Fabric.
On the other hand, RPC protocols allow a client app executes a function on a server remotely. Blockchain uses RPC protocols such JSON-RPC and GRPC for the communications between the client apps and full blockchain nodes running the RPC server. For example, 'Bitcoind' is the RPC server run by a Bitcoin node for responding to client requests. JSON-RPC is a simple, lightweight, and stateless RPC using JSON structure (Group, 2013). On the other hand, gRPC (gRPC Authors, 2021) is a high-performance RPC based on protocol buffers and released by Google in 2015 (Riley, 2019). Ethereum, Bitcoin, MultiChain, Tendermint, and many more use JSON-RPC while the GRPC is used in

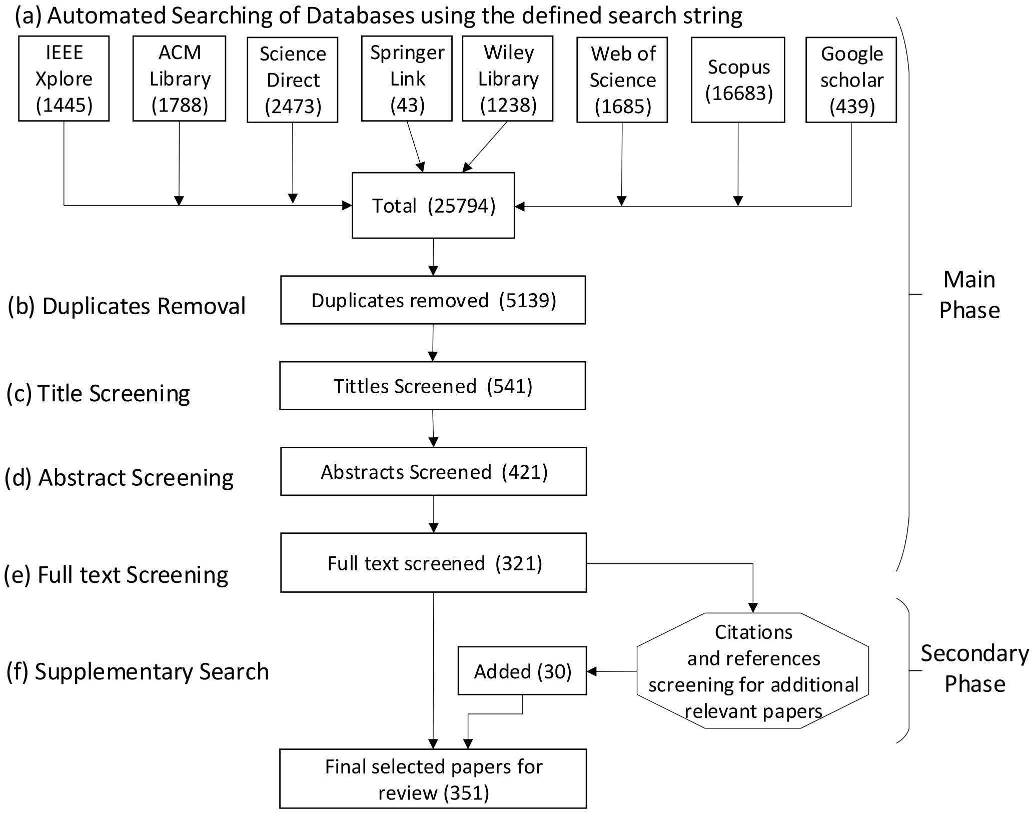

Fig. 4. Systematic Review Process.

blockchains such as the Hyperledger Fabric and Sawtooth. In Hyperledger Fabric, the communication between the ordering service and the anchor peers is achieved using gRPC while gossip is used between the anchor peer and other peer nodes in the same organization.

2. **External networks (Relay networks):**

To improve throughput and propagation time, some miners in blockchain networks such as Bitcoin use high-speed external networks called relay networks (Corallo, 2013) to transmit transactions (blocks) to various parts of the world. Relay networks are very fast networks utilized to broadcast blockchain (Bitcoin) transactions to various parts of the world for reduced block propagation time.

We discussed in detail the relay networks and other methods for conserving the bandwidth and reducing propagation delays in Section 5.3.1 under the network layer scalability solutions.

3. **Compact block relay and other data compression methods:**

The amount of data transmitted in a blockchain network could be reduced using compact block relay (Corallo, 2016) and other data compression methods. In Compact block relay, peers do not send transactions to other peers since most transactions are already received by peers in their memory pool. Hence, only a snap-shot (compact-block) of the block is transmitted. This reduces the requirement of the bandwidth resources as well as the propagation delay.

In Section 5.3.2, we discussed the compact block relay and other data compression methods (such as Txilm, Xthin, and Xtreme) used in blockchain for reducing the bandwidth consumption and propagation delay.

3. Research method of the systematic review

We followed a systematic review process to identify the various research works from published papers and the gray area to find the research trend and the state of the art of blockchain scalability. Fig. 4 gives an overview of the stages and the methods followed in this study as well as the number of publications found in each stage. After outlining the research questions and defining the search string (theme), we selected the final papers used in the review in two phases (main/primary and secondary phases).

In the main phase, we used an automated searching method. We defined and used a search string to extract papers from major science-related databases, namely IEEE Xplore, ACM Digital Library, ScienceDirect, Springer Link, Wiley Online Library, Web of Science, Google Scholar, and Scopus. The search string used is defined as: **((***Blockchain* OR *Bitcoin* OR *Ethereum* OR *Cryptocurrencies* OR *Hyperledger* OR *Corda***) AND (** *Scalability* OR *Scalable* OR *Scaling* OR *Scale* OR *Performance* OR *Throughput* **))**. The searches were done between the 20th–30th of April 2020 and covered publications in the range of 2012–2020. We selected papers written in English and from conferences, Journals, White papers, Archives, Reports, Symposiums, Workshops, and patents categories. We used the EndNote tool to analyze and screen the papers. After the full-text screening in the main phase, we selected 321 papers.

It is possible to have few relevant papers that are not captured in the main phase (automated searching method). Hence a combination of both the automated search and a manual (supplementary) search is recommended and used by many systematic reviews (Zhang et al., 2011). Therefore, similar to Akpinar et al. (2020), Thilakaratne et al. (2019), Azhar et al. (2012) and Bertolino et al. (2019), we conducted the secondary phase as a complement to the main phase to ensure that relevant papers are not left in the review as far as possible. In the secondary phase, we checked the references and citations of the 321 selected papers of the primary phase for the possible additional relevant papers. After several checkings and screenings, we got 30 additional relevant papers which we added to the initial 321 selected papers of the main phase. Hence we finally selected and used a total of 351 papers for the review as shown in Fig. 4.

3.1. Research questions

The research questions for this review are outlined as follows:

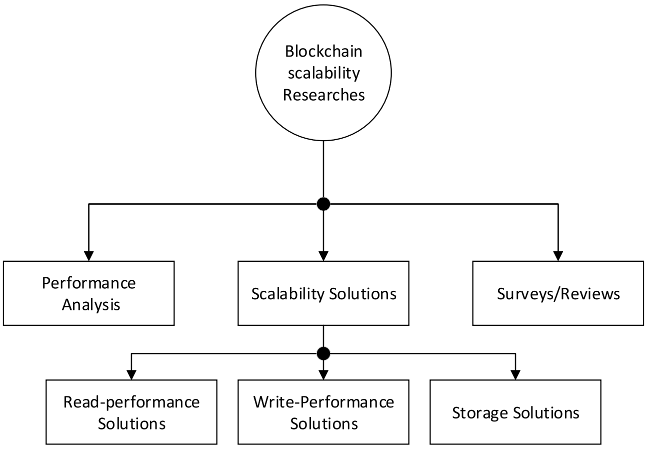
Blockchain scalability Researches
Performance Analysis
Scalability Solutions
Surveys/Reviews
Read-performance Solutions
Write-Performance Solutions
Storage Solutions

Fig. 5. Main Classification of Blockchain Scalability Researches.

1. Where are the researches on blockchain scalability?
2. Where are the blockchain write-performance solutions?
3. Where are the blockchain performance analysis?
4. What is the state of the art of blockchain scalability?
5. What are the research opportunities on blockchain scalability?

4. Research findings

In this section, we discuss the various findings from our systematic review. We answer research questions 1, 2 and 3 in this section while 4 and 5 are answered in Section 5, 6 and 7.

4.1. Where are the researches on blockchain scalability?

From the outcome of our systematic review, we classify the studies on blockchain scalability into three (3) main categories, namely scalability solution proposals, performance-analysis studies, and review/survey papers as shown in Fig. 5. We further classify the proposed scalability solutions into write-performance, read-performance, and storage solutions. It could be seen from Fig. 6 that most (71%) of the scalability studies are scalability solutions. The performance analysis and the reviews/surveys are 16% and 13% respectively.

Blockchain scalability studies span various sources (databases) and have emerged since around 2013. Fig. 7 shows the distribution of blockchain scalability studies by years. It could be seen that the year 2019 has the highest number of publications followed by 2018. Fig. 8 also shows the distribution of the blockchain scalability studies across the various databases searched. It could be seen that IEEE Xplore and Google Scholar contain the largest number of papers. Gray area articles such as patents and reports were obtained from the Google Scholar database. Furthermore, Fig. 9 gives the distribution of the blockchain scalability studies based on the publication type. It could be seen that about half (163) of the 351 papers studied are conference papers. Journal papers collected were about a quarter of the total 351 papers. Hence, there is a need for more journal publications on blockchain scalability. Lastly, Fig. 11 shows the distribution of the proposed blockchain scalability solutions based on our classification.

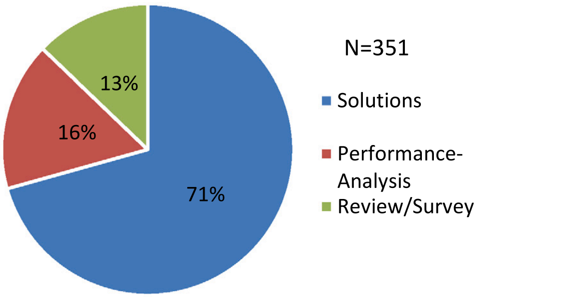

Fig. 6. Blockchain Scalability Publications Based on Category.

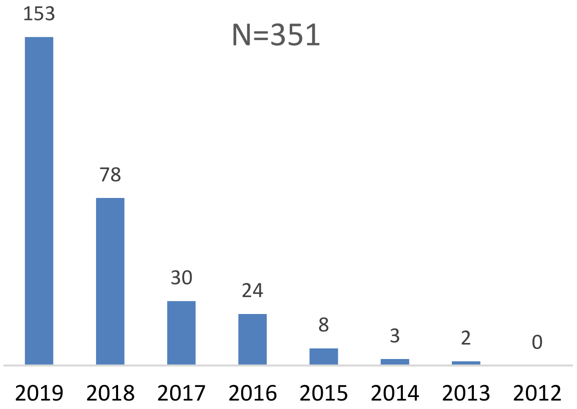

Fig. 7. Blockchain Scalability Yearly Publications.

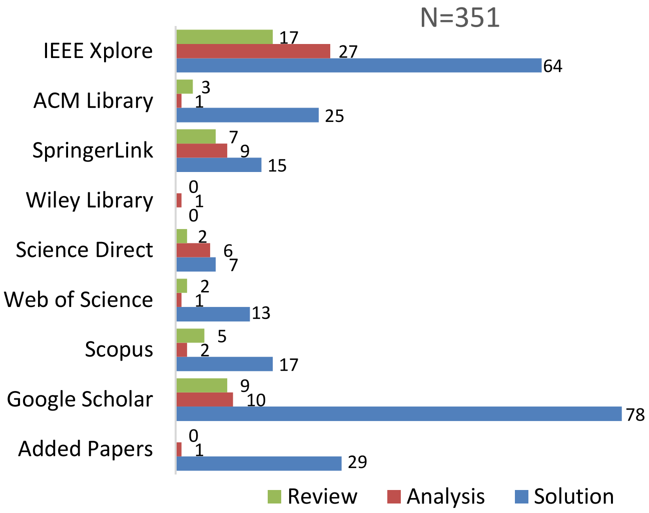

Fig. 8. Distribution of Blockchain Scalability Publications in Databases.

4.2. Where are the blockchain write-performance solutions?

Based on our systematic review findings, we classify studies on the proposed write-performance solutions of blockchain scalability into various classes and groups as shown in Table 1. The write-performance solutions are classified according to our conceptual model of the blockchain ecosystem proposed in Section 2.3. Hence we classify the blockchain write-performance solutions into data, consensus, network, and platform layer solutions. Each layer is then further classified as follows:

Data layer solutions are classified into on-chain and off-chain solutions. The on-chain solutions are the scalability solutions that work on the main chain structure by modifying some of its aspects such as size or structure. They do not introduce another chain or carry out any task outside the main chain. We further classify the on-chain solution into five sub-groups, namely reducing block data, increasing block size, Sharding, Graph (DAG), and parallel executions solutions.

The off-chain solutions on the other hand improve the blockchain scalability by carrying out transactions or tasks outside the main chain.

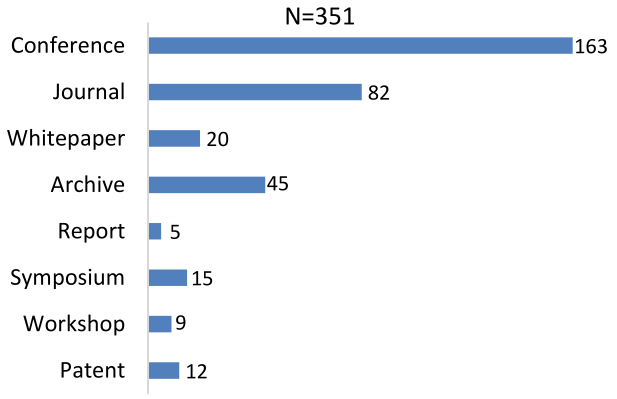

Fig. 9. Distribution of Blockchain Scalability Publications Based on Type.

The summary or aggregation of the transactions or the tasks carried out is later recorded on to the main chain as a single transaction. This is done to relieve the main chain, increase its throughput, reduce its storage burdens as well as lower the transaction fees. We further subdivide the off-chain solutions into payments channels, sidechain, cross-chains, and off-chain computations.

The consensus layer solutions are solutions that propose new consensus or improve an existing consensus protocol. We further classify them based on the type and nature of the proposed or improved consensus. Hence, these solutions are classified into three groups, namely probabilistic, non-probabilistic, and hybrid consensus solutions.

The network layer solutions improve the data propagation delay and the network aspect of the blockchain to achieve higher scalability. We classify the network layer solutions into network structure (relay networks, RINA, and RDMA-based proposals), data compression (such as compact block relay), and other network solutions.

Finally, the platform layer solutions consist of solutions adopted by various blockchain platforms to enhance the scalability of their blockchain. Different blockchain vendors such as Hyperledger adopt some peculiar method or technique to improve the scalability of their blockchain framework. The above-mentioned write-performance solutions of blockchain are discussed in detail in Section 5.

4.3. *Where are the blockchain performance analyses?*

Several pieces of research conducted performance analyses of the blockchain to evaluate, model, or compare the performance of one or more blockchains. We classify these blockchain performance analyses into modeling analysis studies, benchmarking studies, and performance evaluation studies.

The modeling analysis studies model a particular blockchain to predict and observe its performance. By using a given set of parameters and features of the blockchain, the model gives the performance value of the blockchain. In this way, the impact of such parameters on the blockchain can be studied. On the other hand, performance evaluation studies implement a particular blockchain and measure its performance based on certain settings and parameters. Finally, the performance benchmarking studies implement and compare the performances of two or more blockchains. Some of such studies compare the blockchain with other non-blockchain databases or systems.

Fig. 12 shows the distribution of the blockchain performance analysis studies from the total of 58 papers we screened. It could be seen that 41% of the studies are performance evaluation analysis, 35% are benchmarking studies while 24% are modeling analysis studies.

Fig. 10 gives the summary of our review and classification of the various blockchain scalability studies collected. All the 351 selected papers were used in the review and the classification. Also, all the 351 selected papers were classified and listed in their respective groups as shown in Fig. 10. Due to the manuscript size limitation, we chose and discussed only the key representative proposals in each group of our classification to the best of our knowledge and ability since the total selected papers in our review are many (351). Since the papers in each class/group did similar and related works, we believe it is enough to

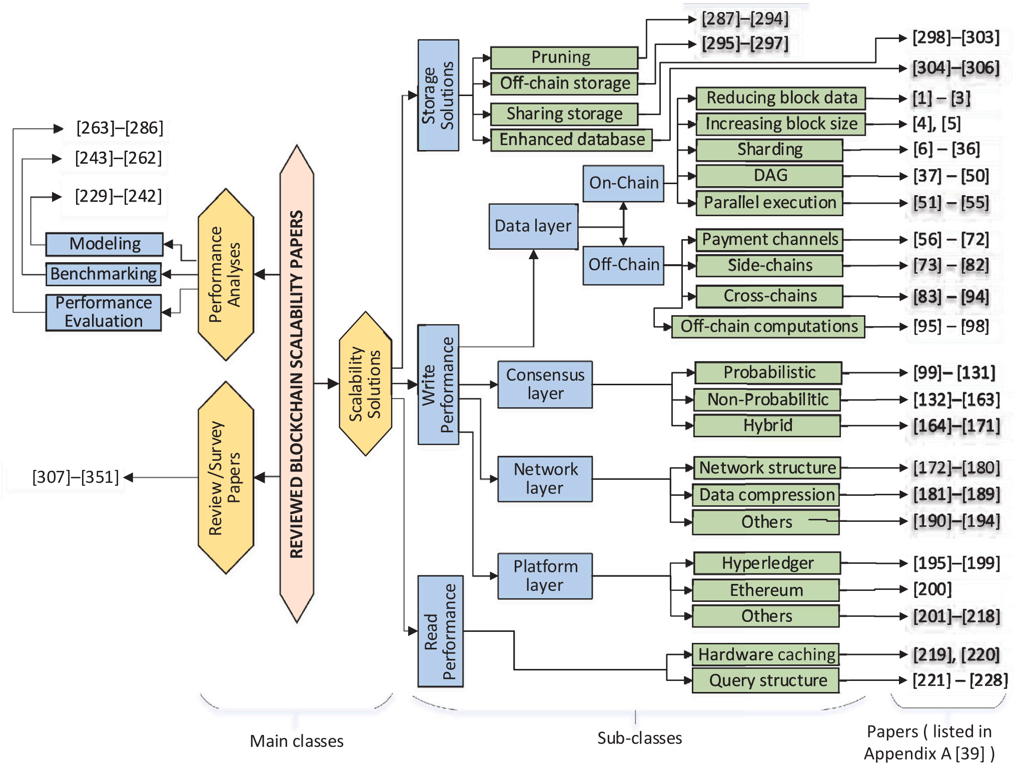

Fig. 10. Summary of the Blockchain Scalability Systematic Review and Classification.

Table 1
Proposed write-performance solutions to blockchain scalability.

Layer	Class	Sub-class	Examples
Data layer	On-Chain	Reducing block data	SegWit, MAST, LTCP
		Increasing block-size	Bitcoin-unlimited, Bitcoin-Cash
		Sharding	Elastico, Omnilegder, Monoxide, RapidChain
		Graph (DAG)	Spectre, Tangle(IOTA), CoDAG, ByteBall
		Parallel executions	Gao et al. (Gao et al., 2017), Yu et al. (Yu et al., 2018), Dickerson (Dickerson et al., 2019), Anjana (Anjana et al., 2019)
	Off-Chain	Payment channels	Lightning Network, Raiden Network, Trinity Network
		Sidechains	Plasma, RootStock, Liquid, ZK-Rollup
		Cross-chains	Cosmos, Geeq, Polkadot
		Off-chain computations	TrueBit, Arbitrum, Zokrate, ACE
Consensus layer	Probabilistic consensuses		Bitcoin-NG, GHOST, Bicomp, Ouroborous, Roll-DPoS
	Non-Probabilistic consensuses		PBFT, SBFT, FBFT, Raft
	Hybrid consensuses		ByzCoin, Casper, BA, Solida
Network layer	Network structure		Relay networks, RINA
	Data compression		Compact block-relay, Txilm, Graphene, Xtreme
	Others		Sallal (sallal et al., 2017), Wang (Wang and Kim, 2019)
Platform layer	Hyperledger		Execute-Order-Validate architecture
	Ethereum		Casper/Shasper
	Others		Parallel computing architectures (Limited, 2020)

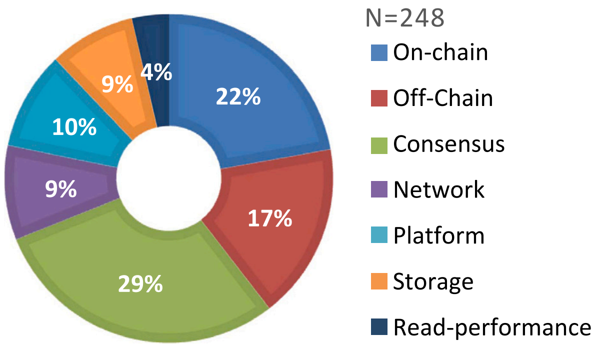

Fig. 11. Distributions of Proposed Blockchain Scalability Solutions.

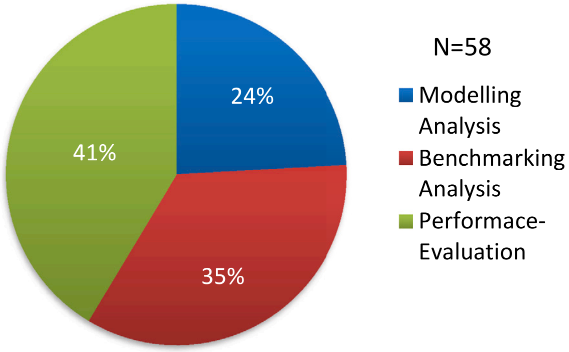

Fig. 12. Distributions of Blockchain Performance Analysis Publications.

discuss only the key proposals in each group to limit the size of our paper as well as the reference list. Furthermore, we listed all the papers (cited) in Fig. 10 in appendix A pointed by Sanka and Cheung (2020) instead of the reference list to limit the size of the reference list as explained.

5. Write-performance scalability solutions

In this section, we discuss the state of the art of the blockchain write-performance scalability solutions as shown in Table 1.

5.1. Data layer solutions

5.1.1. On-chain solutions

1. **Reducing Block Data:**
One factor causing the write-performance scalability issue in blockchain is the data size of the blocks. Large blocks result in a long propagation delay which may lead to frequent forks and reduce the security of the blockchain. Some proposals reduce the size of the block data without decreasing the number of transactions in the block for more transactions per second.

(a) **SegWit (Segregated Witness):**
Lombrozo et al. (2015), a Bitcoin improvement proposal BIP141 proposed in 2017 by Dr. Pieter Wuille is a soft fork on Bitcoin implemented to improve the scalability (Block size limit) of Bitcoin as well as prevent malleability of Bitcoin transactions. Segwit removes signatures from the transaction data and appends it to metadata together with scripts as a separate structure called Witness. Also, the signatures now count as a quarter of their original sizes. Signatures occupy about 65% of transaction data, thus removing them frees up some space in a block and allows more transactions to be included in the block. About 4 times more transactions are added in a block. Hence, the throughput (transactions per second) is increased. Segwit also increases the Bitcoin block size from 1 MB to 4 MB. It also solves the quadratic hashing problem and facilitates the running of payment channels such as the Lightning network which are other blockchain scaling protocols. Despite its benefits, the throughput improvement in SegWit is limited to 17–23 TPS.
(b) **Merkelized Abstract Syntax Tree (MAST):**
MAST (Lau, 2016) was proposed to reduce the block size of Bitcoin to get more space for more transactions. It is included in the Bitcoin Improvement Proposal BIP-114. Bitcoin allows for the addition of scripts in transactions. These scripts contribute to the big size of Bitcoin transactions and part of the scripts may not be used. Abstract Syntax Tree (AST) is a way of breaking program codes into a tree structure with each block of code connecting to its dependencies until all dependencies get connected. MAST proposed the presentation of Bitcoin scripts as a Merkle tree of its AST branches. In this way, the unused sub-script can be removed from a block. A coin spender is required to provide a Merkle proof of the missing script branch to spend the Bitcoin when the Merkle proof returns True. MAST achieves tremendous block size reduction on a logarithmic scale.
(c) **Lumino Transaction Compression Protocol (LTCP):**
Lumino (Lerner, 2017) is a scaling method on Lumino (RSK) network which is a Bitcoin payment channel similar to the lightning network.

LTCP like SegWit uses delta compression to remove selected fields (some signatures) from a block. This results in about 90% block data reduction. LTCP was aimed at scaling the RSK network to 2000 TPS and reaching about 1 billion active users. The protocol could scale Bitcoin to 100 TPS when implemented as a soft-fork. The protocol could also be applied to Ethereum and other token blockchains.

2. **Directly Increasing Block-Size:**

Some proposals directly increase the block-size to improve the transactions per second. Such an increase was done in Bitcoin-cash and Bitcoin-unlimited (Bitcoinunlimited, 2020). However, a direct increase in block size may result in security vulnerabilities due to the increase in the propagation delay leading to the possibility of forks and DoS attacks. After the activation of SegWit soft fork on Bitcoin, some miners that were not happy with SegWit pushed for Bitcoin-Cash (Bitcoin-cash, 2019) hard-fork as an alternative of scaling Bitcoin transactions. Bitcoin-Cash increased the Bitcoin block size limit to 8 MB and then later to 32 MB. Since the block size does not linearly relate to the throughput, the throughput increment here becomes limited.

3. **Sharding:**

Sharding is a scaling method adopted from distributed database systems. In native distributed database systems, the whole database is segmented and each segment is stored on a separate server to boost performance and reduce the workload of a single server. Likewise, in blockchain systems, sharding achieves scalability by dividing the blockchain network into groups called shards as shown in Fig. 13. Each shard processes transactions and stores data in parallel. The network functions could also be divided among the different shards. Sharding allows blockchain to scale horizontally by allowing parallel consensus and storage with an increasing number of nodes. Sharding also reduces the communication overhead in BFT consensus networks.

Despite its performance, shard assignments, small shard security, cross-shard communication overheads are the major challenges to handle for a successful sharding technique. Poor shard design may lead to a 1% attack and other security issues. An optimal shard size and the atomicity of cross-shard transactions must be ensured for optimum throughput and security. The major proposed sharding protocols include:

(a) **Elastico:**

Elastico (Luu et al., 2016) was the first sharding proposal for scaling public blockchains with byzantine fault tolerance. Elastico divides the miner nodes into shards (committees) each processing different transactions and creating blocks in parallel using PBFT consensus. The committee members are selected using PoW consensus. The least significant digits of the PoW result were used to re-shuffle the validators using a proposed scheme (distributed commit and XOR) for achieving randomness in the next epoch. One distinct feature of Elastico is the leader committee that receives blocks from each shard and creates the final aggregated block (main block) for the whole blockchain network. Elastico achieved a throughput of 40 TPS with 1600 nodes.

Despite its scaling efforts, the frequent committee selection and identity creations degrade the network performance. The fact that each node stores the blockchain data of the whole network makes Elastico less scalable in terms of storage. Also, Elastico could only tolerate 25% fault nodes due to the small size of the committee members and the PBFT consensus used. This could lead to chances of attacks with an appreciable number of faulty nodes. Furthermore, the atomicity of transactions could not be achieved in Elastico while its leader committee processings adds further transaction confirmation delays. Other sharding solutions try to improve on Elastico's challenges.

(b) **Omniledger:**

Omniledger (Kokoris-Kogias et al., 2018) is a sharding protocol that improves on Elastico. They improve Elastico's shard assignment by using a bias-resistant randomness method combining Algorand's verifiable random function (Gilad et al., 2017) and RandHound (Syta et al., 2017). Similar to Elastico, their protocol still requires a few committee members to scale and can tolerate only 25% of malicious nodes. However, Omniledger improved the ByzCoin consensus and proposed ByzCoinX as their consensus mechanism. ByzCoin combines BFT and PoW consensuses in a tree structure using collective signing (CoSi). On the other hand, ByzCoinX improves the tree structure of ByzCoin by fixing its depth to only three levels and spanning the number of branches horizontally. In this way, ByzCoinX achieves better latency and increases the shard size up to a thousand nodes to solve the 1% BFT attack vulnerability in ByzCoin and Elastico. Also, ByzCoinX removes the shifting window of ByzCoin to achieve more scalability by electing leaders in the current epoch.

Omniledger introduced *Atomix*, a 2-phase method to ensure the atomicity of transactions. Furthermore, Omniledger uses block-DAG to commit blocks in parallel for more scalability. With 600 nodes per shard, Omniledger achieved a throughput of 3500 TPS. However, the involvement of Omniledger participants in cross-shard transactions introduces a new bottleneck for lightweight nodes. Combining the Algorand and RandHound in Omniledger requires initial randomness to be set in the genesis block by a third-party which limits the usage as well the scalability of Omniledger for an asynchronous network.

(c) **RapidChain:**

RapidChain (Zamani et al., 2018) was designed to overcome the issues encountered by the previous sharding protocols i.e. Elastico and Omniledger. It is the first sharding protocol for public blockchain that requires no trusted initial setup and achieves parallel transaction processing, storage, and communication. RapidChain could tolerate up to 33% faulty nodes in the whole network unlike the 25% faulty nodes tolerance in Elastico and Omniledger. Furthermore, a shard can tolerate up to 50% faulty nodes. RapidChain is robust and uses a fast cross-shard communication protocol which avoids gossip in relaying transactions to the whole network nodes. It also uses Visual Secret Sharing (VSS) to achieve unbiased randomness and block pipelining in its intra-consensus to achieve greater throughput and scalability.

RapidChain achieves over 7300 TPS and 8.7 s confirmation time on its evaluation with a 4000 nodes network. The measured time to failure was estimated to be over 4,500 years. However, the BFT consensus used in RapidChain is only suitable for shards with small sizes. Increasing the shard size incurs communication overhead. Another vain for Rapidchain is its proposed DRG protocol which is unscalable just like other VSS schemes.

(d) **Monoxide:**

Monoxide (Wang and Wang, 2019) is a sharding protocol that uses PoW based on Chu-ko-nu mining as its intra-shard consensus. Shards process transactions in parallel, a system they called *Asynchronous Consensus zones*. The shards form the parallel zones with each shard handling its transactions without communication with the other zone members. Inter-zone communication is achieved using eventual atomicity to ensure the atomicity of transactions. Monoxide possesses one strong security feature due to its Chu-ko-nu mining which originated from merged mining (Judmayer et al., 2017). In this type of mining, the mining power of a shard is the same or close to the total mining power of the whole network. This is achieved by allowing miners to mine in all or multiple shards. As such, attacking a shard requires the same power as attacking the whole network. However, the kind of mining in Monoxides causes fear of centralization due to the sophistication involved. An experimental evaluation of Monoxide using 48,000 nodes revealed that Monoxide performs 1000x than Bitcoin and Ethereum.

(e) **Ethereum 2.0 Sharding:**

Ethereum has long been anticipated to upgrade to Ethereum 2.0 for higher scalability up to 100,000 TPS. Ethereum 2.0 entails sharding and Casper/shasper (BFT-PoS consensus). Ethereum 2.0 will have 64 parallel shards including the existing Ethereum 1. Each shard processes sets of transactions and stores data in parallel. A validator is elected locally in each shard every 8s to create new blocks until the validator group is re-shuffled globally. Each validator stores all headers for all the shards while the attesters may belong to a different shard other than the shard he is attesting for. The block headers of a bunch of shard blocks will be signed by some other selected validators (attestors) and stored on the main chain called the Beacon chain to maintain liveness. At the heart of Ethereum 2.0 is the *Beachon chain* which is the core of the sharding infrastructure. The Beacon chain manages and keeps the records of the network validators (with their stakes) and serves as the root validators registry for the shards chains. To become a validator in Ethereum 2.0, a node deposits coins (32ETH stake) in a special contract account on Ethereum 1.0 using a one-way transaction. After confirmation, the receipt could be used in the shards as an attestation for voting the validators. The Beacon chain will hence be used to manage validators and their stakes. It will also be used in electing block validators, rewards, and punishments as well as facilitating cross-shard transactions.

(f) **Other Sharding Proposals:**

Team (2018) uses PoW as its main consensus used in parallel in each shard. However, the rule that users in each shard store the whole blockchain data creates a storage scalability bottleneck. Harmony (Team, 2019) unlike Zilliqa, allows users to store only the blockchain data corresponding to their shards only. They introduced and used a new consensus they called *Effective PoS* (EPoS). Blocks in Harmony are created every 8s with finality in each shard supporting 250 nodes for fault tolerance. An unbiased randomness is ensured using a verifiable random function. Aspen, RSCoin, PolyShard, and SMChain are also sharding protocols. The other sharding proposals include Logos, Chainspace, Stegos, SSChain, and Ostraka using Axios, BFT, gPoS, PoW, and PoW consensus protocols respectively. Table 2 compares the prominent sharding protocols.

4. **Directed Acyclic Graph (DAG) Based Solutions:**

To improve blockchain scalability, the original structure of blockchain was changed with a directed acyclic graph (DAG) structure used in graph theory. In this structure, two or more blocks can reference the same previous block and a block can reference more than one previous block. Hence blocks can be created in parallel in DAG and may contain conflicting transactions. Sergio Demian was the first to propose a DAG-based cryptocurrency known as DagCoin in 2015 (Lerner, 2015). DAG structure reduces the latency and increases the throughput (TPS) of blockchain resulting in shorter confirmation time. However, security issues such as double-spending and fear of centralization are the major concerns of the current DAG-based blockchains. Fig. 14 compares original blockchain and DAG blockchain structures. The major blockchain scaling proposals using DAG include:

(a) **Inclusive Blockchain Protocols and Spectre:**

Inclusive blockchain (Lewenberg et al., 2015) and SPECTRE (Sompolinsky et al., 2016) were proposed by Sompolinsky, Zohar, and Lewenberg based on the directed acyclic graph (DAG) to facilitate higher transaction rates. In the inclusive, new blocks reference many previous blocks, and transactions from conflicting blocks can also be included. In this way, the propagation delay of larger blocks could be tolerated. They proposed an inclusive rule that determines the main chain not based on the longest chain as used in normal blockchains. Spectre also uses similar DAG approach to offer high throughput and secure protocol for cryptocurrencies. Spectre allows miners to mine blocks concurrently and can resist an adversary with up to 50% computing power. The major approach in Spectre is using a voting algorithm to order pairs of blocks depending on their position in the DAG; however conflict resolution is not guaranteed.

(b) **Tangle (IOTA):**

Unlike block-DAGs, Tangle (Popov, 2016) is a DAG structure (Tx DAG) formed by transactions instead of blocks. Hence there are no blocks and mining in Tangle. Tangle is the protocol used in the IOTA blockchain platform and cryptocurrency developed for IoT devices and the Internet of Everything for a machine to machine micropayments without transaction fees. Nodes in IOTA are allowed to create transactions if they validate two previous transactions and carry out simple PoW. The transaction is added to the network upon validation by other nodes. A coordinator node run by the IOTA foundation was used with the Tangle to achieve consensus. The coordinator releases transactions after every time interval. Only transactions referenced by these transactions are considered valid by the network.

IOTA is currently considered centralized due to its coordinator node. The network is susceptible to double spending, scams, and other security attacks, causing the loss of a large number of tokens. However, some algorithms were proposed to avert attacks in IOTA but not yet well-proven. The IOTA PoW hash function was upgraded due to some security vulnerabilities identified. Another issue of IOTA is the large metadata generated for not using blocks (Bai, 2019).

(c) **ByteBall:**

ByteBall is also another transaction DAG platform and cryptocurrency similar to IOTA but having *Byte* as its token. Since there are no blocks, Byteball transactions reference the hash of previous transactions. Byteball uses a witness group for its main chain. The witnesses are twelve powerful authenticated nodes from highly reputable organizations that validate transactions and add to the network (Bai, 2019). Any witness whose reputation drops to a certain level is removed and replaced with another. Byteball uses transaction fees as an incentive. The cost of the transaction fee is equivalent to the size of the transaction.

(d) **Phantom:**

Phantom (Sompolinsky and Zohar, 2020) is a PoW permissionless blockchain-based on the DAG structure. The authors of the Spectre (discussed above) proposed Phantom using a new protocol GHOSTDAG, a greedy algorithm for ordering transactions through solving an np-hard puzzle. Unlike Spectre, Phantom supports smart contracts as well as controls the number of blocks that can be created in parallel using a parameter k. Phantom could distinguish blocks created by honest miners and the blocks created by the protocol violators.

(e) **CoDAG:**

CoDAG (Yang et al., 2019) stands for compacted DAG-based blockchain that is efficient and secure. Unlike other DAG-based blockchains, CoDAG arranges blocks in levels having fixed widths. Thus, blocks in CoDAG form a compacted structure. CoDAG is a block DAG and is mined by miners by solving a puzzle simpler than in PoW consensus. The compacted DAG structure improves both the confirmation time and the security of the existing DAGs such as Tangle (IOTA) which has nondeterministic confirmation time. Upon the arrival of a new block, an algorithm was proposed to place the block to the appropriate level. Preliminary implementation result showed that CoDAG could resist adversary attacks and achieved a throughput of 394 TPS.

(f) **Other DAG-Based Blockchains:**

Conflux (Li et al., 2018) is another DAG-based blockchain proposed to scale PoW consensus to over 1000 TPS. Conflux divides consensus time into epochs similar to Bitcoin-NG. In each epoch, blocks are produced in parallel by multiple creators. Conflux uses the GHOST rule in selecting the main chain and achieves a throughput of 6400 TPS with a confirmation time between 4.5–7.4 min. DagCoin (Lerner, 2015) uses DAG and considers each transaction as a block. OperaChain is proposed in Fantom having two DAG blocks (main chain and event blocks). Other proposed DAG-based blockchains include Nano, Hashgraph, DLattice, Hycon, ITC, and 3D-Dag (Zou et al., 2018). Table 3 compares the various Dag-based blockchains.

5. **Parallel Executions**

Most blockchains use smart contracts for their operations. Hence there is a need to process these contracts as fast as possible for scalability. There are several efforts made to improve the execution of smart contracts for a more efficient and scalable blockchain. Gao et al. (2017) proposed a scheme for parallel execution of smart contracts instead of the currently serial execution. Their work relied on random assignment and contract partition algorithms. Similarly, Yu et al. (2018) proposed a model for parallel smart-contract processing. The model facilitates parallel execution of transactions to achieve higher throughput. The authors use multi-threading to implement the model. They also proposed

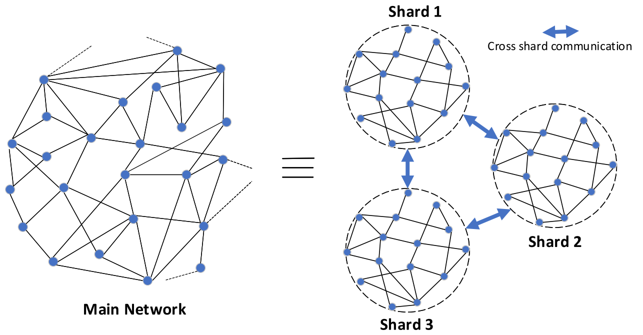

Fig. 13. Sharding in Blockchain.

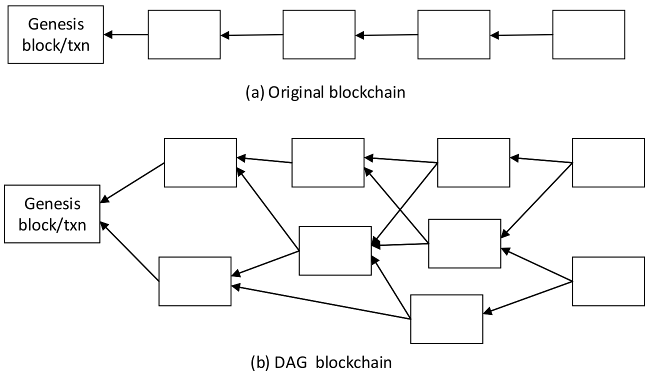

Fig. 14. DAG Vs Original Blockchain Architectures.

an algorithm to solve a synchronization issue in the proposed model. Other papers on parallel-smart contract execution include (Yu et al., 2017; Dickerson et al., 2019; Anjana et al., 2019).

5.1.2. *Off-chain solutions*

1. **Payment and State Channels:**

These off-chain solutions conduct instant micropayments outside the main chain of the blockchain through an established secure channel mostly used for cryptocurrencies. Smart contracts are used to facilitate the transactions and the creation of the channel without the involvement of third parties. The micropayments are recorded on the main chain as a single transaction at the end of payments. As such, transaction fees are reduced and less data is uploaded to the main chain. The major payment channels include:

(a) **Lightning Network:**

The Bitcoin Lightning network (Poon and Dryja, 2016) was proposed to improve the throughput of Bitcoin. The Lightning Network allows two or more parties to create a secure channel outside the main chain and perform instant transactions via the created channel without a third party. It provides high throughput transactions, instant payments as well as lesser transaction fees.

Assuming Alice and Bob want to carry out pieces of transactions totaling 20 dollars using the Lightning Network. In the beginning, Alice and Bob create a multiparty account on Bitcoin and each transfers 20 dollars. Next, they create a secure channel outside the main chain and carry out micropayments of smaller dominations (2 or 5 dollars, for example). After the completion of all the micropayments, they return to the main chain and deposit the total amount transacted on the payment channel from the multiparty account. Each of them also transfers the balance to his account. Lastly, the payment channel is finally closed. It is not necessary for two parties A, B to have a direct payment channel to carry out transactions in Lightning Network. They can do so by linking through another party that already established a payment channel with the other corresponding party.

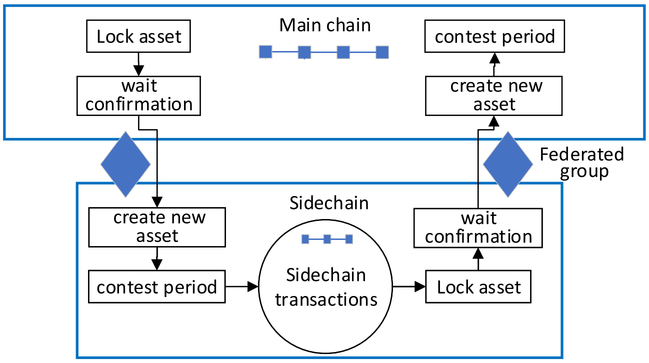

Fig. 15. Sidechain Scalability Method.

Lightning Network faces criticism that the transacting parties must be online throughout the transaction period and the fact that it is only limited to Bitcoin. The Lightning Network also does not scale to a large number of nodes and does not handle transactions of a large amount of currency.

(b) **Raiden Network:**

Raiden Network (Hees, 2016) is the off-chain solution for Ethereum that supports bidirectional multihop transfer. It is similar to the Lightning Network in Bitcoin except that the Raiden network supports Ethereum'ERC20 tokens while the Lightning Network supports Bitcoin. ERC20 is a smart contract standard used to implement tokens in Ethereum. Similar to the Lightning Network, the parties using this payment channel must deposit coins until the end of the channel. The payment channel also does not allow large deposits, limiting the number of tokens to be transferred.

(c) **Trinity Network:**

Trinity Network (Credit, 2018) is an open-source universal off-chain solution that provides a payment channel for instant transactions in the NEO blockchain platform at low fees and high throughput. It uses state channel technology to achieve high throughput. Trinity uses different technologies and proof of asset consensus for privacy, scalability, and security. Its cross-chain converter (TNC) facilitates data and token transfer between the different chains.

(d) **Other Payment Channels:**

Bitcoin Duplex Micropayment Channels (Decker and Wattenhofer, 2015) unlike the Lightning Network puts updates to the blockchain once, not for every update of the micropayment channel. μRaiden (labs, 2018) is a unidirectional payment channel for Ethereum. Unlike the Raiden Network, the μRaiden is cheaper but does not support multihop transfer. AMHL (Malavolta et al., 2019) tries to improve privacy in payment channels. It uses locks that are anonymous and multihop. As a result, communication overhead is reduced. Sprites (Miller et al., 2017) is an improved payment channel faster than Lightning Network, where the locking duration of an account is constant. This improves the throughput as well as allows partial deposits and withdrawals that do not disturb the payment channel.

2. **SideChains:**

A sidechain is a secondary ledger (blockchain) that is created as an attachment to a main (primary) blockchain to allow the transfer of the assets of the main chain on the sidechain at a predetermined rate for scalability. Sidechains are created using a two-way peg and may have its separate consensus protocol that can be different from the consensus of its main chain blockchain. Due to the low inter-dependence between the sidechain and its main chain, security issues of the sidechain do not affect the main chain and vice-versa. However, a sidechain only exists with the existence of its main chain blockchain. Sidechain allows

Table 2
Comparison of sharding based scalability protocols.

Protocol	Elastico	Omniledger	Monoxide	RapidChain	Ethereum2.0	Harmony	Zilliqa
Shard consensus	PBFT	ByzCoinX	PoW	50%BFT	BFT-PoS	BFT	PBFT
No. of shards	<100	<64	2^{10}–2^{18}	<256	64	–	–
Shard size	<100	4–1024	100–10 k	3–256	<100	–	–
Shard allocation	PoW	Nonce(R)	Fixed	PoW	PoS	PoS	PoW
Throughput (TPS)	40	3500	11,694	7380	<100 k	–	–
Latency (sec)	<900	800	23	8.7	6–8	–	–
Total tolerance	25%	25%	50%	33%	33%	25%	25%
Shard tolerance	33%	33%	50%	50%	33%	33%	33%
Smart contract support	No	No	No	No	Yes	Yes	Yes
Blockchain type	Public	Public	Public	Public	Public	Public	Public
Synchronization	Partial	Partial	Async	Partial	Partial	Sync	Async
Transaction mode	UTXO	UTXO	Account	UTXO	Account	Account	Account

Table 3
DAG-based scalability protocols.

Solution	Consensus	DAG nature	Throughput (TPS)	Smart contract?
Inclusive	PoW	block DAG	>65	No
Spectre	PoW	block DAG	–	No
Tangle	PoW	Txn DAG	>1000	Yes
CoDAG	PoW like	block DAG	1151	No
Nano	voting	lattice DAG	>10000	No
Phantom	PoW	block DAG	>1000	No
ByteBall	Witness-based	Txn DAG	20	No
Conflux	GHOST-PoW	block DAG	6400	No
DLattice	Panda	block DAG	1200	No

the use of additional features that are unavailable on the main chain and improves the throughput, privacy, or security of the main chain. For example, a sidechain can be used to use smart contract tokens on Bitcoin network.

Back et al. (2014) was the first to propose two-way pegged sidechains into scale blockchain. Fig. 15 describes the processes of a two-way pegged sidechain. Supposing Alice having a Bitcoin wallet and wants to carry out a 20 Bitcoin transaction on Ethereum network with his Bitcoins. Alice first sends the 20 Bitcoins to a special address (output) and locks it so that no one can spend it while locked. Alice then waits for the confirmation period to ensure his transaction is recorded on the blockchain. Once the transaction is sent to the special output, It will be broadcasted to federated group members belonging to both the main and the sidechain. Federated groups are members of both the main chain and the sidechain witnessing the locking and unlocking of transactions. To spend the Bitcoins on the Ethereum network, Alice produces the SPV proof of his locked transaction to create an equivalent currency on Ethereum. Finally, Alice waits for another time, the contest time before spending the created Ethereum coins. Several sidechain solutions exist, they include:

(a) **Plasma:**

Ethereum Plasma (Poon and Buterin, 2017) proposed by Vitalik Buterin (Ethereum cofounder) and Poon Joseph, is a framework for sidechains attached to the Ethereum network using a smart contract as its root. The smart contract transactions are carried on the Plasma instead of Ethereum to reduce the load of the Ethereum and increase its throughput. The block headers of the Plasma sidechain are periodically posted on the main chain for verification. Any invalid block found is removed, and the fraudulent nodes get penalized. Plasma uses MapReduce and PoS (or another consensus) as its building blocks to allow faster smart contract execution. The parent Plasma smart contract on the Ethereum (root smart contract) is invoked to create and connect many different plasma instances (Plasma sidechains or child chains forming a tree structure). Plasma sidechains can be 100x faster than the Ethereum blockchain.

Despite the scalability promise of Plasma, some security issues were raised regarding buggy constructions and smart contract vulnerabilities. Secondly, the Ethereum network needs to check and verify the sidechain blocks which is an additional workload. Thirdly, in case of a security breach, all the sidechain records have to be offloaded to the Ethereum (main chain). These concerns slow down the implementation and adoption of Plasma on Ethereum. However, some new Plasma variants were proposed with solutions to some of these concerns and possibly adding new security issues. Some of the proposed plasma solutions include Plasma cash, Plasma MVP, and Plasma Debit.

(b) **ZK-Rollup:**

Zero-Knowledge (ZK) proof Rollup (ZK-rollup) was introduced by the Ethereum cofounder (Vitalik Buterin) to scale Ethereum blockchain as an alternative to Plasma. It is an off-chain solution that uses ZK-proof to bundle many transactions sent to it into a single light transaction which it stores on the main chain. When transactions are sent (by transactors) to the ZK-rollup smart contract, nodes called relayers attached to the contract collect a large number of these transactions and generate the ZK-SNARK proof as a single transaction. The single transaction is much lighter than the bundled transactions, and each transaction can be verified on the main chain from the generated proof. ZK-Rollup solves the data availability of Plasma and has the same security level as the main chain (Thomson, 2020).

Currently, Ethereum will temporarily be using ZK-rollup (currently available on Ethereum test networks) for a few years before its major upgrade to Ethereum 2.0. The ZK-rollup can scale Ethereum to achieve 2000–3000 TPS (Thomson, 2020). Loopring exchange (Ltd, 2020) reported a throughput scaling of 2025 TPS using the ZK rollup protocol for exchange and trading on Ethereum. Despite its scaling ability, the creation of the ZK proof is computationally intensive. Furthermore, centralization and the fear of quantum computing are the main setbacks of ZK-rollup. For this reason, ZK-rollup is considered a short time solution by Ethereum. Optimistic Rollup removes the zero-knowledge proof from ZK-Rollup to reduce its computational intensity.

(c) **Liquid Network:**

Liquid Network (Nick et al., 2020) is a sidechain network of Bitcoin created by Blockstream (initial proposers of the two-way pegged sidechains). The sidechain is mainly used for exchanging cryptocurrencies and other digital assets (participated by crypto exchanges and traders) for confidential and faster Bitcoin transactions. The liquid network provides a means by which Bitcoin can be transferred to the sidechain with a two-way peg for exchange and faster transactions. The platform was built on the Elements code, an open-source Bitcoin code. Liquid Network uses a non-PoW consensus known as *Strong Federation* that achieves the finality of blocks in 2 min. The Liquid network hides the amount and the type of asset in its transactions for privacy purposes.

(d) **RootStock (RSK):**

The RootStock (RSK) (Lerner, 2019) is a Bitcoin sidechain (using federated two-way peg) used to run smart contracts that use Bitcoin as their cryptocurrency. It is open-source and uses PoW and merged mining to gain the same security as in Bitcoin by paying the miners some fees per execution. For fast execution, RSK executes smart contracts in parallel with the help of its Turing complete virtual machine (RVM).

RSK has its network developed from Ethereum and QixCoin which was a cryptocurrency created in 2013. Furthermore, RSK uses the DECOR+ protocol to reward miners for avoiding conflict among the miners.
With RSK, 300–1000 Bitcoin transactions per second could be achieved. For the performance of RSK and its compatibility with the Ethereum virtual machine, different kinds of smart contracts are possible on Bitcoin. RSK is used for retail payments, digital identity, and supply chains. However, RSK is not available to all developers even though it is open-source. Another limitation of the RSK is the fear of centralization due to its federated two-way peg.

3. **Cross-Chains:**
Cross-chain solutions connect several blockchains for interoperability and scalability. It has similar construction with sidechains except that the members of cross-chain are pre-existing independent blockchains. This is to say that each member of the cross-chain will exist in the absence of the other and also have its own and asset. Secondly, cross-chain does not store the summary of the whole transactions of one network on another. Rather, it uses smart contracts to pass transactions from one partner network to another. Examples of cross-chain blockchains are Cosmos and Polkadot.

(a) **Cosmos:**
Cosmos (Kwon and Buchman, 2020) is a network of independent blockchains referred to as zones. It connects different blockchain platforms independently running in parallel and interacting with each other for interoperability and scalability. Several zones (blockchains) are extended from the Cosmos Hub which is the first zone. The Cosmos Hub uses PoS cryptocurrency and is equipped with simple and effective governance capabilities. The extended zones communicate using the inter-blockchain communication protocol (IBC) through the Hub and exchange assets. Cosmos supported zones running Tendermint consensus. The cross-chain blockchain was contributed to the open-source community by the Tendermint team. Cosmos can achieve thousands of transactions per second and one second block time.

(b) **Geeq:**
Geeq (Conley, 2020) is another cross-chain blockchain that connects various Geeq instances (Geeqchains) to interoperate, scale, and exchange assets. Geeq uses a new consensus called the proof of Honesty (PoH). The consensus allows Geeq to achieve 99% Byzantine fault tolerance, unlike the lower value in the other BFT based blockchains. The blockchain also achieves very quick finality as well as scalability. The interconnected Geeq instances extend the same genesis blockchain and can easily exchange the Geeq token.

(c) **Polkadot:**
Polkadot (Wood, 2016) is another cross-chain blockchain that secures and allows interoperability and scalability among different independent blockchains called the *Parachains*. It was proposed in 2016 by Gavin Wood and uses relay-chain architecture. The relay Chain is a decentralized network that provides the interface and security services for the different inter-connected blockchains (Parachains). The relay-chain does not depend on the internal architecture or behaviors of its Parachain. The nodes in the Polkadot relay-chain run the Polkadot software lightweight nodes or full nodes. The nodes can take three main roles (validator, nominator, collator) as well as the Fishermen roles (Burdges et al., 2020).

4. **Off-Chain Computations:**
This approach performs computations of some tasks (state transitions) in a node outside the main chain for scalability and privacy. Instead of each node to compute the tasks, only the off-chain node does the computation, thus alleviating the redundant computations and scaling the blockchain. The computation result is sent back to the main chain where it is verified and used. This method is essential to relieve the main chain from complex and time-consuming tasks for scalability. Tasks in transactions that require privacy from the public view could also be performed in this way. Examples of the tasks computed off-chain include the gaming contract verifications, homomorphic encryption, smart contract emulations by miners, ring signatures, and private transactions.
Different methods are used to achieve off-chain computations. These include the use of verifiable computations, Trusted Execution Environment (TEE), secure multi-party computation (SMPC), and incentive mechanisms. However, security and privacy are the major challenges to be ensured in off-computations. Truebit and Arbitrum are examples of off-chain computations frameworks in Ethereum.

(a) **TrueBit:**
TrueBit (Teutsch and Reitwießner, 2019) allows complex computations of smart contracts to be done off the Ethereum main chain using a verifiable computation method. It extended the idea of computation markets of Ethereum and introduced solvers and verifiers of computations using an oracle. Any person can query the solver to solve a complex computation by giving some rewards as incentives. A solver is a third party that solves the complex computations and posts the results on the Ethereum together with the proof of their computation for verification. Ethereum miners called verifiers are incentivized to find to verify the computations and detect errors. TrueBit was initially created to expand the smart contract gas limit in Ethereum. However, more benefits such as increased throughput are realized. The TrueBit is built on three layers, namely incentive, computation, and dispute resolution layers.

(b) **Arbitrum:**
Arbitrum (Kalodner et al., 2018), (currently available on Ethereum testnet) is a cryptocurrency system that improves scalability by relieving miners from the execution (verification) of smart contracts which is now done off-chain. With arbitrum, transacting parties create a virtual machine and agree on its behavior in the execution of their smart contract. The parties assigned trusted managers (they can assign themselves) who execute their smart contract off-chain to verify the VM's behavior. In Arbitrum contracts, miners (Verifier) in the main chain (Ethereum) are not required to execute smart contracts for verification since is done off-chain by the managers. The verifier only verifies the digital signatures of the trusted managers indicating the parties' agreement on the behavior of their VM. The manager is incentivized for his execution as well as punished if found cheating. Arbitrum combines protocol, incentives, and virtual machine designs to achieve scalable and private smart contracts.
Other proposed off-chain computations include (Wüst et al., 2020; Eberhardt and Tai, 2018).

5.2. *Consensus layer scalability solutions*

Several consensus protocols have been used for blockchain. These include the PoW, PoS, PBFT, Raft, Ripple, Proof of Activity (PoA), Proof of Elapsed time (PoET), and others. However, the early protocols such as PoW, PoS, and PBFT fail to satisfactorily overcome the blockchain trilemma (achieve both scalability, security, and decentralization). Therefore, many proposals try to improve the existing protocols, especially for higher scalability. Some proposals bring new protocols as a means of scaling the blockchain. Here, we discuss the consensus-based blockchain scalability proposals.

5.2.1. *Proof of work (PoW) improvements*

PoW is the first blockchain consensus protocol implemented in Bitcoin and most cryptocurrencies. It has good security and decentralization but offers low scalability and consumes a huge amount of energy. Some proposals tried to improve PoW for better blockchain scalability. These include the Bitcoin-NG and GHOST protocols.

(a) **Bitcoin-NG:**
Bitcoin-NG (Eyal et al., 2016) was proposed in 2016 to scale blockchains using PoW. The proposal segregates the Bitcoin consensus into leader election and serialization of transactions. It divides time into time

frames called epochs. In each epoch, a leader who will create blocks within the time frame is elected through the normal PoW process. The elected leader in an epoch collects transactions and creates blocks (microblocks) continuously without PoW until the end of his epoch. When his time frame expires, a new leader is then elected. Hence, there are two different types of blocks in Bitcoin-NG, namely the key block and microblocks. The key block does not have transactions and is mainly generated by the miners for the block creator selection. The microblocks created by the elected leader in an epoch are the actual blockchain blocks containing transactions. In this way, greater transactions per second (10x) were achieved. The confirmation time in Bitcoin-NG is long, about 100 key blocks to be waited to avoid double-spending.

(b) **GHOST:**
In their effort to improve fairness, utilization of mining power as well as prevent double-spending, Sompolinsky and Zohar proposed the GHOST rule (Greedy Heaviest Observed Subtree) (Sompolinsky and Zohar, 2015) which also improves the scalability of PoW blockchains. The authors noticed that at high throughput of Bitcoin transactions, even non-sophisticated attackers may reverse accepted payments to perform double-spending. For this reason, they proposed the GHOST rule to allow for a safe, high throughput Bitcoin blockchain. The GHOST rule changes the selection of the main chain when a fork occurs from the longest chain to the chain with the heaviest sub-tree. The reason behind this is the fact that the heaviest sub-tree considers the PoW blocks that do not get into the main chain. GHOST in addition to security improvement, is capable of increasing the Bitcoin throughput to 200 TPS. However, finding the main chain in the GHOST is a challenging process that may lead to denial of service attacks. Ethereum implements a simple GHOST in some of its versions.

(c) **Bicomp:**
Bicomp (Jiao et al., 2018) improves on the Bitcoin-NG and reduces the power of the elected leader. Similar to Bitcoin-NG, two types of blocks are created, namely macroblocks and microblocks. The blocks are created in rounds, each having an elected leader. The macroblocks are used for the leader election using PoW among the contesting miners. On the other hand, transactions are packed into the microblocks by miners also using the PoW. For each round, the elected leader receives multiple microblocks simultaneously mined. The leader serializes the microblocks into one macroblock which he broadcasts to the whole network.

(d) **Other PoW Improvements:**
Other PoW improvements include ACCEL, Prism, OHIE, and many more. ACCEL was proposed by Hari et al. (2019) to scale Bitcoin for lower confirmation time on top of the PoW consensus. They leverage their idea of a singular block. Using the singular block, they came up with an optimal and secure block generation rate. Prism is an open-source PoW protocol proposed by Yang et al. (2020) to scale Bitcoin to achieve over 10000 tps. Prism is built on a DAG structure to allow parallel creation of blocks. However, Prism used structured DAG and separated blocks based on some features. The evaluation of Prism gave a throughput of 70,000 TPS and 10s latency. However, the network may be susceptible to spam. OHIE is a permissionedless PoW-based consensus proposed by (Yu et al., 2019) for simplicity and higher throughput. In OHIE, many instances of Bitcoin-PoW protocols are run in parallel. Evaluation of OHIE gave a throughput of 1000–2500 TPS through the linear scaling. Hazari and Mahmoud (2019), Gündlach et al. (2019), and Thilagavathi and Lopez (2020) proposed parallel PoW mining protocols to scale PoW blockchains. Likewise, Chainweb (Martino et al., 2018a) and Fitzi (Fitzi et al., 2018) proposed PoW parallel-chains approaches for scaling the PoW blockchains.

5.2.2. Proof of stake (PoS) improvements

The PoS consensus was created as an alternative to PoW due to its high energy consumption. Selecting the block creators (validators) in PoS depends on the coins owned by the validator. In addition to its weaker security, PoS still did not solve scalability issues. Hence, some proposals were made on top of PoS for better security and scalability.

(a) **Delegated PoS (DPoS):**
Delegated PoS consensus was proposed to improve the scalability of PoS. The consensus is currently used in the EOS.io platform and Bitshare cryptocurrency. In DPoS, the participants select delegates among themselves at intervals based on their stakes. The selected delegates serialized blocks on behalf of the other participants. DPoS achieves over 2000 TPS. EOS plans to achieve a million TPS (currently achieves thousands) even though its decentralization is questioned because 86% of their tokens are owned by less than 1% of the participants. Also, few (10) accounts own over 50% of the shares; hence, they will always produce blocks. EOS suffers from attacks from BOTs causing a huge amount of losses amounting to 2.6M USD.

(b) **Ouroboros:**
Ouroboros (Kiayias et al., 2017) is an improved PoS currently used in Ada cryptocurrency of Cardano blockchain network (Cardano, 2020). The protocol guarantees security and further scales PoS by electing delegates in an epoch using a coin-flipping algorithm. A verifiable random number is generated by nodes to prove their suitability for becoming delegates. Each epoch consists of many slots. The block creators for the slots in an epoch are selected randomly by the delegates using a multi-party computation scheme and based on their stakes. The delegates also elect the delegates of the next epoch. Ouroboros Praos (David et al., 2018) is a similar PoS protocol like the Ouroboros. It provides security in a semi-synchronous blockchain against corrupting the blockchain stakeholders.

(c) **Other PoS Improvements:**
Roll-DPoS (Fan and Chai, 2018) was proposed to modify the DPoS consensus for the IoT environment. Due to the low resources in IoT, using the DPoS will be challenging; therefore, the Roll-DPoS uses a randomized delegated DPoS with some cryptographic techniques. They divide time into epochs, with each subdivided further into sub-epochs. Roll-DPoS uses the Ethereum network to bootstrap its pool of potential block producers. A set of delegates is selected at the beginning of each epoch using a random beacon and the hashes generated in the previous epoch. Fractal (Zhou, 2019) is a high-performance blockchain that is provably secure and can achieve over 3000 TPS. Fractal introduces a new and improved PoS-based consensus protocol called *iChing*. The iChing uses pragmatic cryptographic mechanisms to overcome the security concerns in PoS especially, the nothing at stake and grind attacks. The protocol is secure, energy-efficient, and can scale to over 10,000 nodes. Other PoS improvement protocols include Snow-White (Daian et al., 2019), Proof of Lottery (Lee et al., 2020) and (Gao et al., 2019; Chaumont et al., 2019; Fitzi et al., 2020).

5.2.3. Non-probabilistic consensuses

Unlike PoW and PoS (probabilistic protocols), This category of protocols achieves finality on each block before it is added into the blockchain. Therefore, no fork exists in these protocols. Most protocols in this category are Byzantine fault-tolerant (BFT). However, few protocols such as Raft are non-BFT but have fault tolerance and high throughput. BFT protocols tolerate a certain number of anniversaries, mostly 33% of the network size (n). Many BFT protocols improve the scalability of blockchain over the PoW. Some proposals are also made to further improve the scalability of the existing BFT protocols. These include:

(a) **PBFT Improvements:**
PBFT consensus is the most widely used consensus in private and consortium blockchains due to its scalability as well as its byzantine fault tolerance. However, its performance is better with fewer nodes due to its massive messaging (communication overhead O(N^2)). PBFT tolerates up to $f = (n-1)/3$ adversaries in a network of $n = 3f+1$ nodes (1/3 tolerance). PBFT works in rounds called Views, where for each view, a leader is elected among other validators (replicas). The leader controls a three-phase communication decision-making process after which a block is agreed on to be added to the blockchain or rejected.

PBFT achieves a throughput of over 1000 and is used in Hyperledger Fabric (Wang et al., 2019).

Some proposals were made to improve PBFT for more scalability and handling a larger number of ber of nodes. T-PBFT (Gao et al., 2019) uses an Eigen-trust model to reduce the number of nodes required for the PBFT consensus by grouping. Reducing the number of nodes increases the throughput of the PBFT since its message complexity reduces. Other proposals improving the PBFT include (Wang et al., 2020; Chen et al., 2020b; Lao et al., 2020; Fan et al., 2018; Feng et al., 2018).

(b) **SBFT Consensus**

SBFT (Golan Gueta et al., 2019) is a BFT consensus that improves the scalability of blockchain for large-scale global deployment. It offers twice throughput as PBFT as well as 1.5x its latency. SBFT reduces the communication overhead in PBFT to O(N). The protocol is built on top of the PBFT protocol by incorporating four added items. The first is making the communication linear O(N) instead of quadratic O(N^2) using collector nodes. Instead of sending messages to everyone, replicas send messages only to collector nodes. The collector node then broadcasts messages to everyone. The collector message complexity is made a constant with the use of threshold signatures. The second item is the provision of an optimistic fast agreement path. Thirdly, client communication is reduced to unity (1) instead of $f + 1$. Lastly, SBFT adds servers that are redundant for better performance and resilience.

(c) **Tendermint:**

Tendermint is another scalable BFT-based consensus used in the Cosmos network. Like PBFT, Tendermint works in rounds consisting of three phases. It achieves absolute finality through voting among the block validators by messaging. Tendermint also has $1/3$ adversarial tolerance and achieves a throughput of up to 10,000 TPS (Monrat et al., 2019).

(d) **Other BFT Protocols:**

The other BFT consensus algorithms proposed for blockchain scalability include the Proteus (Jalalzai et al., 2019), Federated Byzantine Fault Tolerance (FBFT), DBFT (Zhang et al., 2019), and (Liu et al., 2019; Kim et al., 2020; Jiang and Lian, 2019a,b; Long and Wei, 2019).

5.2.4. Hybrid consensus protocols

Most single consensus protocols face either scalability or security challenges due to the tradeoff between the three blockchain properties i.e. scalability, security, and decentralization (blockchain trilemma). Hence, some researchers propose the use of two or more consensus to achieve a more robust consensus protocol with better scalability and security.

(a) **Ethereum Casper**

Casper (Buterin and Griffith, 2017) is a long-anticipated Ethereum upgrade expected to scale Ethereum to high transactions per second using both PoS and Byzantine fault tolerance. Casper will provide a high degree of finality, security, and liveness. Ethereum was expected to upgrade to Casper in the year 2020.

(b) **ByzCoin Protocol**

ByzCoin protocol (Kogias et al., 2016) combines the Bitcoin-NG's PoW style with PBFT for scaling Bitcoin transactions finality. It is currently used in ByzCoin cryptocurrency. The consensus protocol was built on top of the CoSi (Syta et al., 2016) (a collective signing method) to further improve the PBFT using a tree-structured communication. Hence the prepare and commit phases of the PBFT were enhanced to complete in less than 30 s. Byzcoin achieves finality (confirmation) in 15 to 20 s and over 1500 TPS.

(c) **Byzantine Agreement (BA) Protocol**

Byzantine Agreement (BA) is a new protocol used in Algorand cryptocurrency (Gilad et al., 2017) to achieve confirmation within a minute. Algorand uses another mechanism based on random verifiable functions to scale the BA consensus to a large number of users. In this way, users can privately check their selection in the BA's consensus for each set of transactions. Algorand's performance evaluation revealed that it could handle 500,000 users or more. The Algorand's throughput was 125 times that of Bitcoin.

(d) **Other Hybrid Consensus Protocols**

Hybrid Consensus (Pass and Shi, 2016) was proposed to acquire the security of permissionless protocols such as PoW and the throughput of typical byzantine consensus such as the PBFT. The authors bootstrap a scalable byzantine consensus with a slow Nakamoto's consensus (Snailchain) to get a scalable permissionless consensus. Solida (Abraham et al., 2018) is also a hybrid consensus combining a reconfigurable Byzantine consensus and PoW aimed at improving Bitcoin's confirmation time as well as defending against selfish mining. Solida uses PoW but claimed not to be using Nakamoto's consensus, unlike the (Pass and Shi, 2016). Furthermore, VBBFT-Raft (Tan et al., 2019) is another hybrid consensus protocol.

5.3. Network layer scalability solutions

Some blockchain scalability solutions improve the blockchain network layer for faster propagation delay which consequently allows for higher throughput. We classify the network layer solutions into:

5.3.1. Improving network structure

Various improvements to the blockchain networking structure were proposed to improve the scalability of the blockchain by reducing the propagation delay.

(a) **Relay Networks:**

Relay networks are networks of nodes that serve the purpose of effectively relaying and broadcasting blockchain (mostly Bitcoin) blocks and transactions for faster propagation. Broadcasting through the relay network is much faster compared to using the main blockchain network. Earlier around 2013, a fast but centralized Bitcoin relay network was created by Corallo (Corallo, 2013) to relay miner's blocks globally. However, the centralization poses security and selfish mining challenges despite its scalability benefits. Fiber (Fibre, 2019) is a more recent high-speed Bitcoin relay network that connects nodes and broadcasts Bitcoin compact blocks for shorter propagation delay. Miners connected to Fiber send and receive blocks through its six powerful nodes. Random hub and spoke network is used as a relay network in GeeqChain (Conley, 2020) to broadcast blocks and transactions.

(b) **Recursive Inter-Network Architecture (RINA):**

Recursive Inter-Network Architecture is a new network structure that is an alternative to network protocols such as the TCP/IP. It assumes networking as simply as inter-process communication. RINA is used in Cardano (Cardano, 2020), a blockchain platform for Ada cryptocurrency and smart contracts. It provides Cardano with fast blockchain data propagation and short propagation delay.

(c) **RDMA-Based Blockchain Networking:**

The Remote Direct Memory Access (RDMA) is a network protocol used in high-performance computing network architecture such as Infiniband and RoCE, mostly used in data centers for high throughput communications. Unlike TCP protocol, RDMA directly transfers data from the RDMA-enabled network adapter (NIC) to the application's memory without the involvement of the operating system (OS) and the CPU (kernel-bypassing). In this way, it avoids the multiple memory copying operations in the network stack of the OS and achieves greater throughput and lower latency.

Rüsch (Rüsch et al., 2018) proposed the use of the RDMA to improve the communication overhead in BFT consensuses. The authors proposed Rubin, a framework based on RDMA that offers similar functionality to the selector of Java NIO used in BFT protocols such as BFT-Smart. The Rubin uses a single thread to handle many connections and provides the RDMA capability for Java-based BFT frameworks. Similarly, BOR (Huang et al., 2020a) is an RDMA-based blockchain framework proposed for its high-performance. The authors used the primitives in the RDMA to improve the DPoS protocol. Their evaluation results showed better performance with BOR compared to the EOSIO blockchain platform using DPoS consensus.

5.3.2. Data compression for transmission

Some proposals suggested compression of the block content to save bandwidth and have faster propagation.

(a) **Compact Block Relay:**
Compact block relay (Corallo, 2016) is a Bitcoin upgrade (defined in BIP152) similar to SegWit aimed at saving the bandwidth of nodes and increasing the block propagation delay by reducing the size of the block data transmitted over the network. It works based on the fact that Bitcoin peers already receive transactions from senders in their memory pools. Hence when propagating blocks to neighbors, peers do not have to send the transactions (each about 300 to 400 bytes) in a block over the network. Instead, a snapshot of the block (compact block) is sent so that the receiving peers can use the provided information to construct the real block using the transactions in their memory pools. The compact block only consists of the block header and 32 bytes IDs of the block transactions (transactions replaced with their IDs). However, the peers request transactions that are missing in their memory pool from the sending peer. In this way, the compact block relay achieves transmission of 1 MB block (having 2500) with 15 KB only. Hence, compact block relay saves bandwidth by almost 10 times.
(b) **Txilm:**
Txilm (Ding et al., 2019) is another proposal built on top of the compact block relay to further save the transmission bandwidth of nodes. In this method, a short hash of transaction ID is used to represent a transaction in a block to be propagated to the peers. Receiving peers reconstruct the original block by adding the actual transactions from their memory pool. Here, the peers need to compute the hash of transaction IDs in their memory pool and compare them with the received transaction ID hash in the compact block. Txilm uses salt in the hashing to reduce the chances of collision. Hence, a bandwidth reduction of 80 times is achieved compared to the original blockchain data transmission.
(c) **Other Data Compression Methods:**
The other block compression proposals include the Xthin (Rizun, 2016), Xtreme (Tschipper, 2016), Graphene (Pinar Ozisik et al., 2017) and Xthinner (Lerner, 2017). Xthin is the first block compression suggested before the compact block relay. It represented the transactions in a block with 256-bit TXIDs. On the other hand, Xtreme and Graphene use bloom filters and or IBLTS to propagate a block (1 MB) with only 10 KB–20 kB and 2.6 KB respectively. Using state machine (stack-based), the prefix of transaction ID was used in Xthinner (Toomim, 2018) to compress blocks for reducing the transmission bandwidth.

5.4. Platform layer scalability solutions

(a) **Execute-Order-Validate Architecture of Hyperledger Fabric:**
To improve scalability, Hyperledger Fabric introduced the execute-order-validate approach for processing blockchain transactions. Instead of the block creators (orderers) to execute transactions with smart contracts (chaincodes) and then order blocks, the Fabric relieves them from the transaction execution by introducing the concept of endorsing service. Hence, endorser nodes now execute the transactions and endorse them if authenticated. The orders only receive the endorsed transactions and serialize them into blocks. This approach allows for better throughput and scalability in the Hyperledger Fabric.
Furthermore, some proposals further improve the scalability of the Hyperledger Fabric. Kwon and Yu (Kwon and Yu, 2019) improved the performance (throughput and latency) of Hyperledger Fabric by 20% through the optimization of its ordering and endorsing peers. FastFabric (Gorenflo et al., 2019) claimed to scale the Hyperledger Fabric to 20,000 TPS by making some changes to its architecture for faster ordering and validation. The other improvement proposals for Hyperledger Fabric include (Lu et al., 2018; Lee et al., 2019; Lee and Park, 2020).
(b) **Ethereum 2.0's Casper-Sharding Architecture:**
To improve its low scalability (15–20 TPS), Ethereum planned for an upgrade from Ethereum 1 to Ethereum 2.0 (Millman, 2020) which is expected to achieve up to 100,000 TPS. Ethereum 2.0 will use both sharding and a hybrid BFT-PoS consensus (Casper) with a new Ethereum virtual machine (eWASM). The sharding is the major feature of Ethereum 2.0. We discuss Ethereum 2.0 sharding in 5.1.1.3.e. According to Vitalik Buterin, the Ethereum 2.0 upgrade will take years to complete as it will be implemented in at least three phases. Scaling the Ethereum storage will come first before scaling its computations. During the transition time, ZK roll-up, an off-chain scaling method will scale the Ethereum to about 2000–3000 tps for some years before upgrading to 100,000 TPS upon the complete upgrade of the Ethereum 2.0 (Thomson, 2020).
Besides the 64 shards, the Casper-FFG consensus will provide the desired finality as well as the BFT based proof of stake consensus for the Ethereum 2.0. In the Casper-FFG, a group of validators and attesters are infrequently (6.4 mins) and randomly assigned to shards globally using their stakes on the whole network (not shard's stake). In Casper, Randao (randao.org, 2017) and Verifiable Delay Function (VDF) (Boneh et al., 2018) are used to generate the unbiased randomness required.
(c) **Other Platform Layer Solutions:**
Digital transaction platform (Limited, 2020) uses a high-performance parallel computing architecture in its hybrid private blockchain (ParallelChain) having a throughput of 100,000 TPS. The parallel computing engine allows the blockchain to concurrently run multiple parallel chains and achieve high performance. Other platform layer solutions include the Kadena (Martino et al., 2018b) and Multichain (Multichain, 2020) architectures capable of achieving over 10,000 TPS and 2500 respectively.
Table 4 assesses and summarizes the write-performance blockchain scalability solutions citing their advantages and disadvantages.

6. Storage and read-performance scalability solutions

6.1. Storage scalability solutions

Blockchain faces a storage scalability issue that makes participation difficult or impossible, especially by nodes having small memory. The nodes find it difficult to store such huge data that could not be deleted according to the Satoshi blockchain rule. Some proposals were made to improve the blockchain storage scalability issue. We classify the proposed storage scalability solutions into data pruning, off-chain storage, sharing storage among peers, and the enhanced database methods.

6.1.1. Storage data pruning and compression

Deleting the unused or old part of the blockchain has been proposed to reduced the storage burden. However, such methods may come with a new security issue. An example of this approach is the Mini-Blockchain scheme proposed by Bruce (2014). He proposed the deletion of old transactions on the blockchain and introduced a small-size additional database for keeping the account balances of cryptocurrencies. A small size structure, Proof-Chain, was also introduced to secure the Mini-blockchain against the possible insecurity introduced and the possible loss of coins. Cryptonite cryptocurrency was built on this Mini-blockchain scheme. Other storage reduction methods compress the storage data to reduce its size. To reduce data size in IoT nodes, Kim et al. (2019) proposed a storage compression method. On the other hand, Dai et al. (2018) proposed a network coding concept to reduce the storage data of blockchain.

Table 4
Assessment of blockchain write-performance scalability solutions.

Solutions	Class	Advantages	Disadvantages
On-Chain	Reducing block data	• More transactions per block • Bitcoin malleability solved • High privacy in MAST	• Limited throughput increase
	Increasing block-size	• More transactions per block • More transactions per second	• Security risks due to long propagation delay • Prone to centralization • Limited throughput increase
	Sharding	• Parallel block processing and massive scaling • Storage scalability • Less communication overhead	• 1% attack is possible • Design complexity
	Graph (DAG)	• Higher throughput, lower confirmation time • Parallel block creation • No mining, hence low energy waste • Low or no transaction fees	• Security issues are common • Fear of centralization • Weak consistency • Large storage data
Off-Chain	Payment channels	• Higher throughput and privacy • Instant payments • Low transaction fees	• Limited to cryptocurrencies and less secure • Less support for large value transactions • May require coins to be deposited and locked
	Off-chain computation	• Better scalability • No redundant computation by all nodes • Tasks can be computed in parallel	• May have privacy issues • May introduce security issues • Fear of centralization
	Sidechains	• Better scalability • Allow for interoperability • Security issues of the child-chain does not affect the parent-chain	• May need frequent checking by the main chain • May have storage burden on the main chain • Less userfriendly
	Cross-chain	• Better scalability • Allows interoperability	• May have privacy issues • Fear of centralization • Design complexity
	Parallel executions	• Higher throughput • Lower latency	• Limited throughput increase • May require expensive hardware
Consensus layer solutions		• Allow for massive scalability • Can be pluggable to give different options	• May introduce new security issues • Communication over head in BFT based consensuses • High energy consumption in PoW based consensuses
Network layer solutions		• Faster propagation delay • Higher throughput and lower latency • It does not tamper with the chain structure or consensus	• Fear of centralization and bias in relay networks • Throughput increase may be limited
Platform layer solutions		• Better scalability • Native to the platform • Provide variety of options to users	• May make interoperability difficult • May be complex

6.1.2. Off-chain storage

These types of methods store the blockchain data in off-chain secondary storage such as IPFS or using a distributed hash table (DHT) on other storages.

(a) **Distributed Hash Table (DHT) Storage:**
A distributed hash table (DHT) is used to store blockchain raw data on an off-chain data storage while the hash of the raw data is stored on the blockchain. At the same time, the hash serves as the reference to the raw data on the off-chain storage.

(b) **InterPlanetary File System (IPFS) Storage:**
IPFS is a decentralized and distributed system of storing files. There are several blockchain proposals based on IPFS to relieve the storage scalability issue on blockchain nodes. FileCoin works based on IPFS and provides decentralized storage with the use of proof of Storage (PoS) consensus. Disema (Klems et al., 2017) also provides an Ethereum blockchain-based marketplace where its raw data is stored on IPFS while storing the IPFS data address on Ethereum. Other off-chain storages include (Zheng et al., 2018; Bai et al., 2019).

6.1.3. Sharing storage burden among peers

These methods distribute the storage data among the participating peers. Hence, the nodes are not required to store all the blockchain data. Examples of this method are CUB and Jidar.

(a) **CUB:**
CUB (Xu et al., 2018) is a blockchain storage proposal that distributes the storage burden among nodes divided into groups. They divide nodes into groups known as consensus units. Each unit consists of some nodes that together store a complete copy of the blockchain ledger. The authors proposed a block assignment optimization (BAO) to optimally assign blocks to the nodes in a unit. They further proposed a method of dynamic block assignment as well as heuristic algorithms for static assignment issues. Using Blockbench and synthetic data analysis, they prove the efficacy of their proposed CUB.

(b) **Jidar:**
Dai et al. (2019) proposed Jidar as a trustless data reduction method for Bitcoin systems. In this approach, nodes only store block data that is relevant to them without storing the whole blockchain data. They select

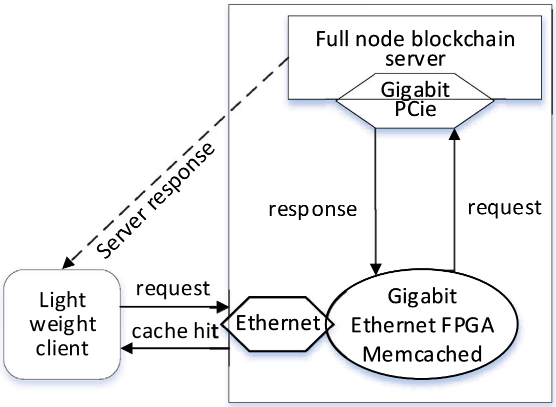

Fig. 16. Hardware Caching for Read-performance Scalability.

transactions that are of interest to them together with their related Mekle branches. This reduces the storage burden for the nodes. The nodes use bloom filters to verify transactions. They can also request other nodes for missing blocks whenever necessary. Jidar was proved to improve the storage issue by 1.03% when compared to the normal storage approach.

6.1.4. Enhanced databases

BigchainDB (GmbH, 2018) is a scalable and decentralized database designed for blockchain. It can handle up to million writes in one second, store petabytes of data, and has less than a second latency. BigchainDB was designed based on a native decentralized database with the addition of blockchain characteristics to it. Therefore, BigchainDB has characteristics of both native database and blockchain. It has an efficient NoSQL query language and partitioning ability making it suitable for both public and private blockchains. In BigchainDB, separate databases S and C are used to store transactions and blocks respectively. BigchainDB 2.0 uses Tendermint consensus and solves the issues realized in BigchainDB 1.0. Other enhanced databases for blockchain include (Wang et al., 2018; Sahoo and Baruah, 2018).

Table 5 compares the various blockchain storage scalability solutions.

Table 5
Blockchain storage solutions.

Proposal	Type	Approach
Mini-Blockchain	Pruning	Deleting old transactions
DHT storage	Off-chain storage	Using distributed hash table
IPFS storage	Off-chain storage	Raw data stored on IPFS
CUB	Storage sharing	Dividing peers into zones
Jidar	Storage sharing	Only relevant data stored
BigchainDB	Enhanced database	Native database with blockchain properties
Forkbase	Enhanced database	Efficient folkable database

6.2. Read-performance solutions

Besides the write-performance of the blockchain (transactions per second, and confirmation time), read-performance is another scalability measure and issue for the blockchain systems. Current blockchain systems are lagging in terms of response performance as well as an efficient query. We found few works in this direction and are discussed as follows:

1. **Hardware Caching:**
To reduce the workload of blockchain servers and improve their response performance upon requests from clients, FPGA hardware caching systems were proposed in Sanka (Sanka and Cheung, 2018) and Sakikabara (Sakakibara et al., 2018). Both papers work on the similar principle of caching the blockchain data on FPGA hardware attached to the blockchain server. As shown in Fig. 16, the FPGA intercepts blockchain data requests from a client and responds with the requested data to the client on behalf of the server if the data is already cached in its memory. When the data is not cached, the FPGA forwards the requests to the blockchain server for the data. It is much faster to get the data from the FPGA cache compared to getting the data from the blockchain server. In this way, the workload on the server is reduced, and the system can handle many requests per second thus, increasing the performance of blockchain applications.

2. **Improving Blockchain Query:**
Blockchain lacks a standard and efficient query. It requires blocks to be searched individually or sequentially using an identifier such as block hash or height. This (in addition to the huge size of the blockchain) causes high latency in accessing blockchain data. Santiago (Bragagnolo et al., 2018) proposed an efficient query language for Ethereum blockchain. They provide SQL-like queries for easier searching of Ethereum blockchain. The language is rich in syntax and has the capabilities of being an efficient query language similar to SQL. Other proposals for improving the blockchain query include (Liu et al., 2020; Qu et al., 2019; Trihinas, 2019).

7. Blockchain performance analysis

There are various analyses conducted on the performance of blockchain. We categorize these analyses into modeling analysis, benchmarking analysis, and performance evaluations. Furthermore, the analysis can be based on consensus protocols, platforms, comparing blockchain with other databases or other scaling methods.

7.1. Blockchain benchmarking tools

Benchmarking blockchain is getting more interest from researchers; however, few benchmarking and evaluation tools are available while more blockchain platforms exist. The existing benchmarking and evaluation tools include the BlockBench, Hyperledger Caliper, and Prism.

7.1.1. BlockBench

Blockbench developed by Dinh et al. (2017) is the first framework for evaluating and benchmarking private blockchain. Blockbench is open-source and can be used to analyze any private blockchain using APIs. With Blockbench, the performance of different private blockchains (latency, throughput, and tolerance to a fault) could be evaluated and compared.

7.1.2. Hyperledger Caliper

Hyperledger Caliper (Hyperledger, 2020) is another blockchain benchmarking tool developed by Hyperledger. The framework is also open-source and could be used for evaluating the performance of different blockchain platforms with preset use cases. Currently, Hyperledger caliper supports Hyperledger (Fabric 1.x, 2.x, Sawtooth 1.0+, Besu, Iroha 1.0 (beta3) and Burrow 1.0), Ethereum, and FISCO BCOS blockchain solutions. The tool provides performance results including the transaction throughput (TPS), transaction latency, read-performance, success rate, and resources (CPU, Network resources, and memory) utilization.

7.1.3. Prism

Recently, Liu et al. (2020b) implemented another benchmarking tool called Prism for blockchains. Like the other blockchain benchmarking tools, Prism is also open-source. The project was aimed at investigating resource utilization in blockchains. All the modules in Prism are run in Docker containers giving the Prism high accuracy and compatibility with several blockchain solutions.

7.2. Performance analysis based on consensus protocols

Sukhwani et al. (2017) analyzed the performance of PBFT consensus by modeling the PBFT consensus by Stochastic Reward Nets (SRN). The mean time for completing the consensus process was computed by the model in a network of up to 100 nodes. They also analyzed the sensitivity of the network to various parameters. Hao et al. (2018) conducted a performance evaluation of the consensus mechanisms used in private blockchains using Hyperledger Fabric and Ethereum. By sending varying transactions to the experimental systems, the throughputs and latencies of the systems were obtained. The results quantitatively showed that the consensus mechanism seriously affects the performance of blockchain systems. As expected, the PBFT consensus in Hyperledger Fabric outperforms the PoW used in the Enterprise Ethereum consensus.

Cao et al. (2020) analyzed and compared the performances of PoW, PoS, and DAG consensus mechanisms. The performance model analyzed the transaction per second, block time, confirmation time, and failure probability. They also found out that network resources affect PoW and PoS while the DAG is affected by the load conditions of the network.

Huang et al. (2020b) was a performance analysis on the Raft consensus mechanism which is a popular consensus for private blockchains. They proposed a model to analytically analyzed the split probability in the blockchain network. The model was analyzed using a Raft simulator they implemented. Hence, thus using this model, the Raft consensus parameters could be optimized.

Jalalzai et al. (2019) evaluated the performance and reliability of BFT consensus protocols, namely PBFT, SBFT, Musch BFT, and Bchain-3. They implemented the protocols in Go language and tested their performance with a blockchain on Amazon cloud EC2 instances. Both the number of nodes and block sizes were varied for the experiment. They found out that the block and network sizes affect performance. Furthermore, Musch BFT performs better than PBFT. On the other hand, the performances of Bchain-3 and SBFT degraded when failures were introduced.

Aljassas and Sasi (2019) quantitatively evaluated and compared the performances of PoW and the Proof of Collatz Conjecture (PCC) consensus mechanisms on the latency, execution, and deployment times. Their test was carried out on a varying number of transactions that are multiples of 10 from 1 to 10000. They found out that the PCC has almost a fixed execution time which is $1/1000$ times compared to the PoW. Durand (Durand et al., 2019) is another performance analysis on blockchain consensus mechanisms.

7.3. Performance analysis based on platforms

7.3.1. Performance analysis on hyperledger fabric

Thakkar (Thakkar et al., 2018) analyzed the effects of some configuration parameters such as endorsement policies and block size in Hyperledger Fabric on its scalability. The parameters include endorsement policies, block size, choice of database, and channels. The analysis found out three bottlenecks of Hyperledger fabric i.e. the verification of endorsement policy, sequential transaction policy validation, and validation and commit of states in the CouchDB database. They finally recommended aggressive caching for verification of endorsement policy as it gave 3x performance improvement. Verification of endorsement policy in parallel was also recommended as it resulted in 7x performance improvement. Finally, they optimized the CouchDB bulk read and write, leading to another 2.5x improvement. In total, this work achieved 2250 TPS which is 16 times the initial 140 TPS of the Fabric.

Baliga et al. (2018) analyzed the latency and throughput of Fabric 1.4 based on different workloads to evaluate its performance. They varied some benchmark parameters such as chaincode parameters, payload size of events, and chaincode invocations. They also studied the impact of endorsement policy on throughput and latency. They found out that the throughput is approximately linear below 1000 TPS and degrades above this value while the latency increases. The throughput also increases with a lesser number of required endorsements. The other parameters that affected their throughput and latency include the orderer setting, transaction read/write size, event payload, and chaincode. The throughput was found to be unaffected by the data in the chaincode. Furthermore, an overhead ranging from 5.2% to 7.45% was discovered for chaincode to chaincode calls.

Unlike other performance studies on Hyperledger fabric, Wang (2019) evaluated the performance of Hyperledger Fabric in an adversary setup. They simulated different adversarial behaviors and measured the blockchain's performance. They recommended how to improve the system performance under such malicious behaviors.

Jiang et al. (2020) followed a hierarchical modeling approach to model the performance of the transaction process in Hyperledger Fabric 1.4. Based on the endorsement and block timeout constraints, they came up with equations that could be used to estimate the performance of the system. The model was validated upon extensive simulation and numerical analysis.

Yuan et al. (2020) used the generalized-Stochastic-Petri-Nets (GSPN) to model the performance (latency and throughput) of Hyperledger Fabric blockchain. They also varied the ordering service configurations and observed their impact on the performance. The other performance analyses on Hyperledger Fabric include (Kuzlu et al., 2019; Nasir et al., 2018; Kocsis et al., 2017; Wickboldt, 2019).

7.3.2. Performance analysis on cryptocurrencies

Shahsavari (Shahsavari et al., 2020) was a performance analysis on Bitcoin network where a random graph performance model was developed to model the network's data propagation. They used an OMNet++ network simulator to implement the model. The model uses a set of equations to estimate the network's traffic overhead and propagation delay of blocks given the blockchain parameters. The impact of relay-network on performance and decentralization of the network was also studied. Through this analysis, the authors discovered a trade-off between the bandwidth, connection per node, and block size on the propagation delay. The use of many relay-networks was also found to have a large impact on the decentralization of the network.

Schäffer et al. (2019) examined and evaluated the effect of some blockchain configuration parameters on the performance of private Ethereum blockchains. Private blockchains can agree to preset some of these parameters for better performance. Such parameters include the block size, block interval, network size, and the type of hardware used. They discovered that these parameters are inter-related in the sense that the effect of one depends on how the other is configured. Therefore, they developed a hierarchy of bottlenecks where the parameters are ranked according to the impact of the blockchain's performance.

Rouhani and Deters (2017) conducted a similar performance analysis on Ethereum transactions. They evaluated and compared the two types of Ethereum clients, that is Parity and Geth. Using the same configurations, they found out that the Parity client is much faster (89.8%) in terms of transaction processing compared with the Geth client. Ochôa et al. (2019) is another performance analysis on Ethereum.

Park et al. (2019) analyzed and modeled the performance of a DAG-based cryptocurrency. Since the performance of the DAG blockchain depends on some parameters, the authors derived a model to analyze the performance by changing those parameters. They also propose a scheme that detects the state of the system and varies the validation parameter to maintain high performance. The scheme allows nodes to also fluctuate their transaction fees during network traffic. About 46% improvement in transaction finality was reported.

Zhang et al. (2019) conducted a performance analysis on Facebook's cryptocurrency, Libra. They formulated a procedure for the performance evaluation and experimented to evaluate the performance of the Libra cryptocurrency compared to other blockchains. Based on their findings, the Libra could only support up to 1000 TPS. The performance degrades as the number of validators increases. Thus, the Libra performs slower than Hyperledger Fabric and requires performance improvements for the effective global micropayments it is intended for.

Table 6
Blockchain performance analysis studies.

Analysis	Category	Example
Benchmarking tools	NA	Blockbench, Hyperledger Caliper
Based on consensus mechanisms	Modeling Analysis Benchmarking Analysis	Sukhwani (Sukhwani et al., 2017), Huang (Huang et al., 2020b) Hao (Hao et al., 2018) and Jalalzai (Jalalzai et al., 2019)
Based on other scaling methods	Performance Evaluation	McCorry (McCorry et al., 2020)
Based on platforms (Hyperledger, Bitcoin and Ethereum)	Modeling Analysis Benchmarking Analysis Performance Evaluation	Yuan (Yuan et al., 2020)and Shahsavari (Shahsavari et al., 2020) Rouhani (Rouhani and Deters, 2017) and Han (Han et al., 2020) Thakkar (Thakkar et al., 2018) and Zhang (Zhang et al., 2019)
Blockchain vs other databases	Benchmarking Analysis	Chen (Chen et al., 2018) and Bergman (Bergman et al., 2020)

7.3.3. Hyperledger Sawtooth performance analysis

Ampel et al. (2019) analyzed the performance of Hyperledger Sawtooth blockchain through performance modeling. They sent 30,000 transactions between two nodes and used Hyperledger Caliper to measure the performance of the blockchain which was 2300 TPS. The authors also observed the effects of the rate of input transaction and batch size on throughput as well as the effect of the throughput on latency.

Shi et al. (2019) is a similar performance analysis on Hyperledger Sawtooth. However, the blockchain in this case is built on the cloud. They found out that by tuning the maximum batches in a block and the scheduler, optimum performance of the blockchain could be obtained.

7.3.4. Analysis on other platforms

Han et al. (2020) analyzed and evaluated the performance of up to five different blockchain platforms with a varying number of nodes for IoT. The platforms include Corda, Hyperledger fabric 0.6 and 1.0, Tendermint, and Ripple. They found out the performance to decrease with an increase in the number of nodes. The other performance analysis based on platforms include (Dhulavvagol et al., 2020; Roehrs et al., 2019; Li et al., 2020; Alrubei et al., 2020; McCorry et al., 2020; Han et al., 2020).

7.4. Blockchain vs other databases analysis

Chen et al. (2018) experimented with both blockchain and a relational database to compare their data read and write performance. They use Ethereum to represent blockchain and MySQL for the relational database. Their result showed that the maximum transaction data capacity of Mysql is 10 times that of blockchain. Also, the transaction processing time in blockchain was 80 to 2000 times that of Mysql.

Bergman et al. (2020) is another similar comparative study of blockchain vs. other databases. The authors studied and compared the performances of Hyperledger Fabric blockchain and Cassandra distributed database. They measured the performance values while varying the sizes of the networks as well as their workloads. Their result indicated that blockchain is comparable to Cassandra in terms of latency. However, the result may differ with different setup and consistency models.

Table 6 summarizes the blockchain performance Analysis studies.

8. Future research directions on blockchain scalability

Here, we deduce the future research directions and opportunities on blockchain scalability.

1. **Hybrid Scalability Solutions:**
It is difficult for a single solution to efficiently solve the scalability issue of blockchain as well as give optimum security. Most proposed solutions either give limited scalability increase or introduce new security concerns. However, two or more scalability solutions may be combined for a better and secure scalability solution. Hence we recommend more hybrid solutions such as the Sharding with Casper in Ethereum 2.0.
2. **Enhancing the Read-Performance of Blockchain:**
The Read-performance issue is among the scalability issues in blockchain, but few pieces of research are done in this direction. SPV and many other nodes (such as IoT) incapable of storing the whole blockchain depend on blockchain servers for the blockchain data. However, the response of blockchain servers is low compared to native servers such as Google and YouTube. Hence there is a need to improve the read-performance of blockchain. Secondly, blockchain lacks an efficient query language and structure compared to native NoSQL and relational databases such as Cassandra and MySQL.
3. **Blockchain Performance Analysis**:
Many blockchains have been developed, but there are few performance evaluations and analyses. Compared to the total number of blockchain scalability publications we collected, the performance analyses take only (17%) percentage. Hence, there is a need to evaluate and analyze the existing blockchains to allow for more improvement and assessment of the blockchain platforms for adoptions in various applications. Likewise, more performance models are required for better parameter selections in the development of new blockchains based on the existing platforms.
4. **Improved Blockchain Storage System:**
The huge size of the blockchain is often discussed as a big issue to the wide adoption of the technology globally. However, we found fewer proposals for the improvement of the blockchain storage issues. Out of the 229 scalability solutions we collected, only 9% were on storage issues. Hence there is an urgent need for more research on reducing the storage requirements of blockchain for wider and global adoption, especially in the Internet of Things (IoT).
5. **Low Storage and Secure DAG Design:**
DAG architecture allows blockchain to scale by allowing multiple blocks to be added produced at the same time. However, the existing DAGs have larger storage data and have security and centralization issues. Therefore, there is a need for more designs of secure DAGs that have less storage and are free from centralization, especially for public blockchains.
6. **Blockchain Evaluation/Benchmarking Tools and Standards:**
Few blockchain evaluation and benchmarking tools are available while more blockchains are created. Hence there is a need to design more evaluation and benchmarking tools to allow for better testing and evaluation of blockchains using effective and standard methods. International Standards also are needed to be developed for standardizing the blockchain evaluation, benchmarking, and testings.
7. **Efficient Shard Assignment and Cross Shard Communication Methods:**
Sharding is an excellent way to scale blockchain. However, poor shard-assignment and cross-shard communications design lead to new security as well as scalability issues. Hence, more effective shard-assignment is required for better security and scalability of blockchains. Likewise, more scalable and secure cross-shard communication protocols are also required for better and scalable blockchains. The 1% attack threat in small shards is also a research area that needs to be addressed to allow secure smaller shards for better scalability. Small shards have little communication overhead and massive scaling.

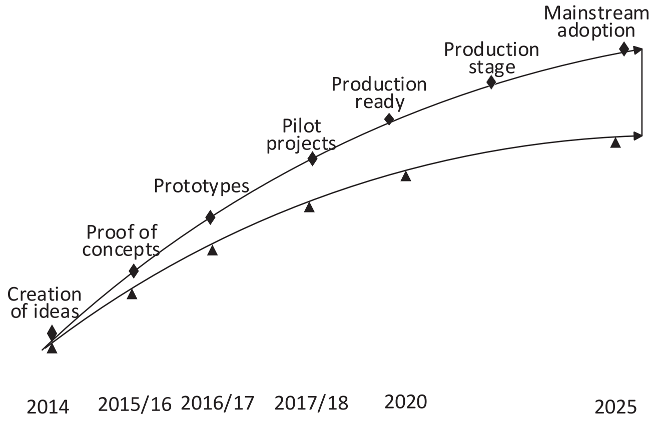

Fig. 17. Blockchain adoption timeline.

9. Adoption of blockchain technology

Blockchain technology gets wider adoption and acceptance from several companies and countries for many applications. Many blockchain trials yielded positive results. PWC's survey of 600 respondents (China, 2018) revealed that 84% of the respondents were engaging with blockchain technology. Fig. 17 shows the blockchain adoption timeline and progress. Blockchain use cases started to be conceptualized in 2014 after realizing the benefits of the blockchain from Bitcoin. In 2015/2016, the generated ideas were put into proof of concepts (PoCs). After successful PoCs, prototypes were created, followed by pilot projects. Around 2019, many pilot projects gave positive results, making them ready for production. Many blockchain projects are already in production. By 2025, blockchain is expected to mature and achieve mainstream adoption. In Deloitte's 2019 blockchain survey of 1386 respondents (Pawczuk et al., 2019), 86% of the respondents believed that blockchain will reach mainstream adoption, 81% planned to change their systems with the blockchain, while 53% put the blockchain among their top 5 critical priorities (Sanka et al., 2021).

There are several applications of blockchain, among which cryptocurrencies are the most trending. Currently, there are about 1200 cryptocurrencies globally. Bitcoin and Ethereum are the most successful cryptocurrencies having the market capital of 1.02 Trillion USD and 194 Billion USD respectively. As of 23rd March 2021, The prices of Bitcoin and Ethereum were 54,790 USD and 1692 USD respectively. Other cryptocurrencies include the Binance coin, Monero, Dash, Zcash, Cardano, and IOTA (Cryptoreport, 2021).

Besides cryptocurrencies, blockchain is used for several other applications such as smart contracts, insurance, healthcare management, and many more. Smart contracts are hosted on blockchains such as Ethereum and Hyperledger for Decentralized Applications (Dapps) and the Decentralized Autonomous Organizations (DAOs). Similarly, insurance companies such as Insurwave and Etherisc use blockchain to avoid fraud of claiming duplicate insurance benefits. Blockchain is also used to hold health records for the provision of common records and better healthcare management. With the use of the blockchain, duplicate and inconsistent healthcare records could be avoided. Furthermore, blockchain is used to provide a decentralized domain name service (DNS) and decentralized storage systems. Blockchain has got many applications in IoT. It has been used for security, privacy, smart contract, trust management, and smart trading in the IoT (Butun and Österberg, 2021). Blockchain has also been used in the IoT for reputation and multi-agent systems (Fortino et al., 2019; Giancarlo et al., 2021; Malik et al., 2019; Liu et al., 2019; Debe et al., 2019). Some of these research works used the blockchain for grouping agents using a reputation-based strategy. In Giancarlo et al. (2021) and Fortino et al. (2019), the social reputation capital in IoT was optimized using a blockchain-based group formation strategy. The optimization allows for effective cooperation and coordination between the smart IoT devices. Other applications of blockchain include its use in banking and finance, stock exchange, asset registry, supply chain, energy, identity management, cybersecurity, and more.

Several big and small companies adopt blockchain after realizing its benefits. Microsoft, IBM, and Oracle each have blockchain cloud platforms. Corda is a blockchain platform of R3, which is a consortium of over 200 financial institutions globally. Ford, BMW, and Renault formed the Mobility Open Blockchain Initiative (MOBI) consortium for sharing data among the giant automobile companies. Tradelens is also a blockchain platform used by Maersk and other top global shipping companies for supply chain. Libra is a stable cryptocurrency of Facebook expected to be released. The other big companies using blockchain technology include LG, J.P. Morgan, Walmart, HealthBank, Civic, and the Australian Stock Exchange (AES).

Many countries also use blockchain technology for several applications. Estonia keeps health records on the blockchain, while Georgia uses the technology to keep its land registry. United Arab Emirates (UAE) and Saudi Arabia use blockchain for internal banking payments, while Singapore uses it to protect fraud in trade invoices. Japan, Switzerland, and Indonesia use blockchain for identity management. China advocates for research and adoption of blockchain technology besides its effort in creating its national digital currency. Other countries using blockchain technology include Russia, Chile, UK, Australia, Sweden, India, Canada, and South Korea (Sanka et al., 2021).

10. Conclusion

Blockchain provides a secure distributed ledger that allows parties with little or no trust to share information and data without central authority or intermediary. Blockchain provides data security, autonomy, speed, privacy, transparency, and efficiency. Despite its benefits, scalability is a big challenge to the blockchain. The scalability issues are due to lower throughput, high latency, large storage, and low read-performance. In this paper, we conducted a systematic review to find the research trend and the state of the art on blockchain scalability. We collected and screened various blockchain scalability research publications from various databases through the systematic process. We classified the various blockchain scalability studies based on the blockchain ecosystem layered model we also proposed. We also gave a comprehensive review of the state of the art of blockchain scalability studies. We deduced the future research directions and the opportunities on blockchain scalability. Finally, we discussed the adoption of the blockchain technology in several companies and countries for various applications.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Abraham, I., Malkhi, D., Nayak, K., Ren, L., Spiegelman, A., 2018. Solida: A blockchain protocol based on reconfigurable Byzantine consensus. In: Leibniz International Proceedings in Informatics, Vol. 95. LIPIcs, http://dx.doi.org/10.4230/LIPIcs.OPODIS.2017.25, cited by 4.

Akpinar, E., Yeşilada, Y., Temizer, S., 2020. The effect of context on small screen and wearable device users' performance - a systematic review. ACM Comput. Surv. 53 (3), http://dx.doi.org/10.1145/3386370, URL https://doi-org.ezproxy.cityu.edu.hk/10.1145/3386370.

Aljassas, H.M.A., Sasi, S., 2019. Performance evaluation of proof-of-work and collatz conjecture consensus algorithms. In: 2019 2nd International Conference on Computer Applications Information Security. ICCAIS. pp. 1–6.

Alrubei, S.M., Ball, E.A., Rigelsford, J.M., Willis, C.A., 2020. Latency and performance analyses of real-world wireless IoT-blockchain application. IEEE Sens. J. 20 (13), 7372–7383.

Altarawneh, A., Herschberg, T., Medury, S., Kandah, F., Skjellum, A., 2020. Buterin's scalability trilemma viewed through a state-change-based classification for common consensus algorithms. In: 2020 10th Annual Computing and Communication Workshop and Conference. CCWC, pp. 0727–0736. http://dx.doi.org/10.1109/CCWC47524.2020.9031204.

Ampel, B., Patton, M., Chen, H., 2019. Performance modeling of hyperledger sawtooth blockchain. In: 2019 IEEE International Conference on Intelligence and Security Informatics. ISI. pp. 59–61.

Anjana, P.S., Kumari, S., Peri, S., Rathor, S., Somani, A., 2019. An efficient framework for optimistic concurrent execution of smart contracts. In: 2019 27th Euromicro International Conference on Parallel, Distributed and Network-Based Processing. PDP, pp. 83–92. http://dx.doi.org/10.1109/EMPDP.2019.8671637.

Azhar, D., Mendes, E., Riddle, P., 2012. A systematic review of web resource estimation. In: Proceedings of the 8th International Conference on Predictive Models in Software Engineering. PROMISE '12, Association for Computing Machinery, New York, NY, USA, pp. 49–58. http://dx.doi.org/10.1145/2365324.2365332.

B, N.Z., Aminian, M., Javadi, B., 2020. Blockchain-based decentralized storage networks: A survey. J. Netw. Comput. Appl. 162, 102656. http://dx.doi.org/10.1016/j.jnca.2020.102656.

Back, A., Corallo, M., Dashjr, L., Friedenbach, M., Maxwell, G., Miller, A., Poelstra, A., Timón, J., Wuille, P., 2014. Enabling blockchain innovations with pegged sidechains. 72, URL: http://www.opensciencereview.com/papers/123/enablingblockchain-innovations-with-pegged-sidechains.

Bai, C., 2019. State-of-the-art and future trends of blockchain based on DAG structure. In: Structured Object-Oriented Formal Language and Method. Springer, Cham, pp. 183–196.

Bai, L., Hu, M., Liu, M., Wang, J., 2019. BPIIoT: A light-weighted blockchain-based platform for Industrial IoT. IEEE Access 7, 58381–58393.

Baliga, A., Solanki, N., Verekar, S., Pednekar, A., Kamat, P., Chatterjee, S., 2018. Performance characterization of hyperledger fabric. In: 2018 Crypto Valley Conference on Blockchain Technology. CVCBT. pp. 65–74.

Berendea, N., Mercier, H., Onica, E., Rivière, E., 2020. Fair and efficient gossip in hyperledger fabric. In: 2020 IEEE 40th International Conference on Distributed Computing Systems. ICDCS, pp. 190–200. http://dx.doi.org/10.1109/ICDCS47774.2020.00027.

Bergman, S., Asplund, M., Nadjm-Tehrani, S., 2020. Permissioned blockchains and distributed databases: A performance study. Concurr. Comput.: Pract. Exper. 32 (12), e5227. http://dx.doi.org/10.1002/cpe.5227, e5227 cpe.5227. arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/cpe.5227.

Bertolino, A., Angelis, G.D., Gallego, M., García, B., Gortázar, F., Lonetti, F., Marchetti, E., 2019. A systematic review on cloud testing. ACM Comput. Surv. 52 (5), http://dx.doi.org/10.1145/3331447, URL https://doi-org.ezproxy.cityu.edu.hk/10.1145/3331447.

Bitcoincash, 2019. Bitcoin cash. URL http://118.31.72.178/wiki/CryptoCurrency/Bitcoin_Cash.pdf.

Bitcoinunlimited, 2020. Bitcoin unlimited: The peer-to-peer electronic cash system for planet earth. URL https://www.bitcoinunlimited.info/.

Boneh, D., Bonneau, J., Bünz, B., Fisch, B., 2018. Verifiable delay functions. In: Advances in Cryptology. CRYPTO 2018, Springer, Cham, pp. 757–788.

Boyd, S., Ghosh, A., Prabhakar, B., Shah, D., 2006. Randomized gossip algorithms. IEEE Trans. Inform. Theory 52 (6), 2508–2530. http://dx.doi.org/10.1109/TIT.2006.874516.

Bragagnolo, S., Rocha, H., Denker, M., Ducasse, S., 2018. Ethereum query language. In: Proceedings of the 1st International Workshop on Emerging Trends in Software Engineering for Blockchain. WETSEB '18, Association for Computing Machinery, New York, NY, USA, pp. 1–8. http://dx.doi.org/10.1145/3194113.3194114.

Bruce, J., 2014. The mini-blockchain scheme. White paper.

Burdges, J., Cevallos, A., Czaban, P., Habermeier, R., Hosseini, S., Lama, F., Kilinc Alper, H., Luo, X., Shirazi, F., Stewart, A., Wood, G., 2020. Overview of polkadot and its design considerations. arXiv:2005.13456.

Buterin, V., Griffith, V., 2017. Casper the friendly finality gadget. CoRR abs/1710.09437. arXiv:1710.09437. URL http://arxiv.org/abs/1710.09437.

Butun, I., Österberg, P., 2021. A review of distributed access control for blockchain systems towards securing the internet of things. IEEE Access 9, 5428–5441. http://dx.doi.org/10.1109/ACCESS.2020.3047902.

Cachin, C., Vukolić, M., 2017. Blockchain consensus protocols in the wild. http://dx.doi.org/10.4230/LIPIcs.DISC.2017.1, arXiv preprint arXiv:1707.01873, arXiv:1707.01873.

Cao, B., Zhang, Z., Feng, D., Zhang, S., Zhang, L., Peng, M., Li, Y., 2020. Performance analysis and comparison of PoW, PoS and DAG based blockchains. Digit. Commun. Netw. http://dx.doi.org/10.1016/j.dcan.2019.12.001.

Cardano, 2020. What is cardano. URL https://www.cardano.org/en/what-is-cardano/.

Chaumont, G., Bugnot, P., Hildreth, Z., Giraux, B., 2019. DPoPS: Delegated Proof-of-Private-Stake, a DPoS implementation under X-Cash, a Monero based hybrid-privacy coin. Yellowpaper.

Chen, F., Xiao, Z., Cui, L., Lin, Q., Li, J., Yu, S., 2020a. Blockchain for Internet of things applications: A review and open issues. J. Netw. Comput. Appl. 172, 102839. http://dx.doi.org/10.1016/j.jnca.2020.102839.

Chen, J., Zhang, X., Shangguan, P., 2020b. Improved PBFT algorithm based on reputation and voting mechanism. J. Phys. Conf. Ser. 1486, 032023. http://dx.doi.org/10.1088/1742-6596/1486/3/032023.

Chen, S., Zhang, J., Shi, R., Yan, J., Ke, Q., 2018. A comparative testing on performance of blockchain and relational database: Foundation for applying smart technology into current business systems. In: Distributed, Ambient and Pervasive Interactions: Understanding Humans. Springer, pp. 21–34.

China, P., 2018. PwC Global Blockchain Survey 2018 - Blockchain is here. What's your next move? Res. Insights URL www.pwccn.com/global-blockchain-survey-2018.

Cisco, 2018. Blockchain by Cisco - Build trust-based business networks for digital transformation. Cisco Blockchain White Paper.

Conley, J.P., 2020. The geeq white paper. URL https://geeq.io/geeq-white-paper-2/.

Corallo, M., 2013. High-speed bitcoin relay network. URL http://sourceforge.net/p/bitcoin/mailman/message/31604935/.

Corallo, M., 2016. Compact block relay. BIP152. URL https://github.com/bitcoin/bips/blob/master/bip-0152.mediawiki.

Credit, T.N., 2018. Universal off-chain scaling solution. Trinity, URL https://trinity.tech/#/.

Cryptoreport, 2021. Live crypto prices and trading. URL https://cryptoreport.com/all.

Dai, X., Xiao, J., Yang, W., Wang, C., Jin, H., 2019. Jidar: A jigsaw-like data reduction approach without trust assumptions for bitcoin system. In: 2019 IEEE 39th International Conference on Distributed Computing Systems. ICDCS. pp. 1317–1326.

Dai, M., Zhang, S., Wang, H., Jin, S., 2018. A low storage room requirement framework for distributed ledger in blockchain. IEEE Access 6, 22970–22975.

Daian, P., Pass, R., Shi, E., 2019. Snow white: Robustly reconfigurable consensus and applications to provably secure proof of stake. In: Financial Cryptography and Data Security. Springer, pp. 23–41.

David, B., Gaži, P., Kiayias, A., Russell, A., 2018. Ouroboros praos: An adaptively-secure, semi-synchronous proof-of-stake blockchain. In: Advances in Cryptology. EUROCRYPT 2018, Springer, pp. 66–98.

Debe, M., Salah, K., Rehman, M.H.U., Svetinovic, D., 2019. IoT public fog nodes reputation system: A decentralized solution using ethereum blockchain. IEEE Access 7, 178082–178093. http://dx.doi.org/10.1109/ACCESS.2019.2958355.

Decker, C., Wattenhofer, R., 2015. A fast and scalable payment network with bitcoin duplex micropayment channels. In: Stabilization, Safety, and Security of Distributed Systems. Springer, Cham, pp. 3–18.

Dhulavvagol, P.M., Bhajantri, V.H., Totad, S.G., 2020. Blockchain ethereum clients performance analysis considering E-voting application. Procedia Comput. Sci. 167, 2506–2515. http://dx.doi.org/10.1016/j.procs.2020.03.303, International Conference on Computational Intelligence and Data Science.

Dickerson, T., Gazzillo, P., Herlihy, M., Koskinen, E., 2019. Adding concurrency to smart contracts. Distrib. Comput. 1–17.

Ding, D., Jiang, X., Wang, J., Wang, H., Zhang, X., Sun, Y., 2019. Txilm: Lossy block compression with salted short hashing. CoRR abs/1906.06500. arXiv:1906.06500. URL http://arxiv.org/abs/1906.06500.

Dinh, T., Wang, J., Chen, G., Liu, R., Ooi, B., Tan, K.-L., 2017. BLOCKBENCH: A framework for analyzing private blockchains. pp. 1085–1100. http://dx.doi.org/10.1145/3035918.3064033.

Durand, A., Hamida, E.B., Leporini, D., Memmi, G., 2019. Asymptotic performance analysis of blockchain protocols. CoRR abs/1902.04363. arXiv:1902.04363. URL http://arxiv.org/abs/1902.04363.

Eberhardt, J., Tai, S., 2018. ZoKrates - scalable privacy-preserving off-chain computations. In: 2018 IEEE International Conference on Internet of Things (iThings) and IEEE GreenCom and IEEE CPSCom and IEEE Smart Data (SmartData). pp. 1084–1091. http://dx.doi.org/10.1109/Cybermatics_2018.2018.00199.

Eklund, P.W., Beck, R., 2019. Factors that impact blockchain scalability. In: Proceedings of the 11th International Conference on Management of Digital EcoSystems. MEDES '19, Association for Computing Machinery, New York, NY, USA, pp. 126–133. http://dx.doi.org/10.1145/3297662.3365818.

Eyal, I., Gencer, A.E., Sirer, E.G., Renesse, R.V., 2016. Bitcoin-NG: A scalable blockchain protocol. In: 13th USENIX Symposium on Networked Systems Design and Implementation. NSDI 16, USENIX Association, Santa Clara, CA, pp. 45–59, URL https://www.usenix.org/conference/nsdi16/technical-sessions/presentation/eyal.

Fan, X., Chai, Q., 2018. Roll-DPoS: a randomized delegated proof of stake scheme for scalable blockchain-based internet of things systems. In: Proceedings of the 15th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services. pp. 482–484.

Fan, X.X., Chai, Q., Assoc Comp, M., 2018. Roll-DPoS(sic): A randomized delegated proof of stake scheme for scalable blockchain-based internet of things systems. In: Proceedings of the 15th Eai International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services. pp. 482–484. http://dx.doi.org/10.1145/3286978.3287023.

Feng, L., Zhang, H., Chen, Y., Lou, L., 2018. Scalable dynamic multi-agent practical byzantine fault-tolerant consensus in permissioned blockchain. Appl. Sci. 8 (10), 1919.

Fibre, 2019. Fibre: What is fibre? URL http://bitcoinfibre.org/index.html.

Fitzi, M., Gazi, P., Kiayias, A., Russell, A., 2018. Parallel chains: Improving throughput and latency of blockchain protocols via parallel composition. IACR Cryptol. ePrint Arch. 2018, 1119.

Fitzi, M., Gazi, P., Kiayias, A., Russell, A., 2020. Proof-of-stake blockchain protocols with near-optimal throughput. IACR Cryptol. ePrint Arch. 2020, 37.

Fortino, G., Fotia, L., Messina, F., Rosaci, D., Sarné, G.L., 2020. Trust and reputation in the internet of things: State-of-the-art and research challenges. IEEE Access 8, 60117–60125. http://dx.doi.org/10.1109/ACCESS.2020.2982318.

Fortino, G., Messina, F., Rosaci, D., Sarné, G.M.L., 2019. A reputation capital and blockchain-based model to support group formation processes in the internet of things. In: 2019 6th International Conference on Control, Decision and Information Technologies. CoDIT, pp. 888–893. http://dx.doi.org/10.1109/CoDIT.2019.8820294.

Furlonger, D., Valdes, R., 2017. Practical blockchain: a gartner trend insight report. URL https://blockcointoday.com/wp-content/uploads/2018/04/Practical-Blockchain_-A-Gartner-Trend-Insight-Report.pdf.

Gao, Y., Kawai, S., Nobuhara, H., 2019. Scalable blockchain protocol based on proof of stake and sharding. J. Adv. Comput. Intell. Intell. Inf. 23 (5), 856–863.

Gao, Z., Xu, L., Chen, L., Shah, N., Lu, Y., Shi, W., 2017. Scalable blockchain based smart contract execution. In: 2017 IEEE 23rd International Conference on Parallel and Distributed Systems. ICPADS. pp. 352–359.

Gao, S., Yu, T., Zhu, J., Cai, W., 2019. T-PBFT: An EigenTrust-based practical Byzantine fault tolerance consensus algorithm. China Commun. 16 (12), 111–123. http://dx.doi.org/10.23919/JCC.2019.12.008.

Giancarlo, F., Lidia, F., Fabrizio, M., Domenico, R., Giuseppe, M.S., 2021. A blockchain-based group formation strategy for optimizing the social reputation capital of an IoT scenario. Simul. Model. Pract. Theory 108, 102261. http://dx.doi.org/10.1016/j.simpat.2020.102261, URL https://www.sciencedirect.com/science/article/pii/S1569190X20301891.

Gilad, Y., Hemo, R., Micali, S., Vlachos, G., Zeldovich, N., 2017. Algorand: Scaling byzantine agreements for cryptocurrencies. In: Proceedings of the 26th Symposium on Operating Systems Principles. SOSP '17. New York. pp. 51–68. http://dx.doi.org/10.1145/3132747.3132757.

GmbH, B., 2018. BigchainDB2.0 - the blockchain database. White paper. URL https://www.bigchaindb.com/whitepaper/bigchaindb-whitepaper.pdf.

Golan Gueta, G., Abraham, I., Grossman, S., Malkhi, D., Pinkas, B., Reiter, M., Seredinschi, D., Tamir, O., Tomescu, A., 2019. SBFT: A scalable and decentralized trust infrastructure. In: 2019 49th Annual IEEE/IFIP International Conference on Dependable Systems and Networks. DSN, pp. 568–580. http://dx.doi.org/10.1109/DSN.2019.00063.

Gorenflo, C., Lee, S., Golab, L., Keshav, S., 2019. FastFabric: Scaling hyperledger fabric to 20,000 transactions per second. In: 2019 IEEE International Conference on Blockchain and Cryptocurrency. ICBC, pp. 455–463. http://dx.doi.org/10.1109/BLOC.2019.8751452.

Group, J.-R.W., 2013. JSON-RPC 2.0 specification. JSON-RPC. URL https://www.jsonrpc.org/specification.

gRPC Authors, 2021. Introduction to gRPC. GRPC.Io. URL https://grpc.io/docs/what-is-grpc/introduction/.

Gündlach, R., Hoepman, J.-H., van der Hofstad, R., Koens, T., Meijer, S., 2019. Hydra: A multiple blockchain protocol for improving transaction throughput. arXiv preprint arXiv:1910.06682.

Hafid, A., Hafid, A., Samih, M., 2020. Scaling blockchains: A comprehensive survey. IEEE Access 1.

Han, R., Shapiro, G., Gramoli, V., Xu, X., 2020. On the performance of distributed ledgers for internet of things. Internet Things 10, 100087. http://dx.doi.org/10.1016/j.iot.2019.100087, Special Issue of the Elsevier IoT Journal on Blockchain Applications in IoT Environments.

Hao, Y., Li, Y., Dong, X., Fang, L., Chen, P., 2018. Performance analysis of consensus algorithm in private blockchain. In: 2018 IEEE Intelligent Vehicles Symposium. IV. pp. 280–285.

Hari, A., Kodialam, M., Lakshman, T.V., 2019. ACCEL: Accelerating the bitcoin blockchain for high-throughput, low-latency applications. In: IEEE INFOCOM 2019 - IEEE Conference on Computer Communications. pp. 2368–2376. http://dx.doi.org/10.1109/INFOCOM.2019.8737556.

Harz, D., Boman, M., 2019. The scalability of trustless trust. In: Financial Cryptography and Data Security. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 279–293.

Hazari, S.S., Mahmoud, Q.H., 2019. A parallel proof of work to improve transaction speed and scalability in blockchain systems. In: 2019 IEEE 9th Annual Computing and Communication Workshop and Conference. CCWC, pp. 0916–0921. http://dx.doi.org/10.1109/CCWC.2019.8666535.

He, X., Cui, Y., Jiang, Y., 2019. An improved gossip algorithm based on semi-distributed blockchain network. In: 2019 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery. CyberC, pp. 24–27. http://dx.doi.org/10.1109/CyberC.2019.00014.

Hees, H., 2016. Raiden network: Off-chain state network for fast DApps. In: Devcon Two. Ethereum Foundation.

Hewa, T., Ylianttila, M., Liyanage, M., 2020. Survey on blockchain based smart contracts: Applications, opportunities and challenges. J. Netw. Comput. Appl. 102857. http://dx.doi.org/10.1016/j.jnca.2020.102857.

Huang, B., Jin, L., Lu, Z., Zhou, X., Wu, J., Tang, Q., Hung, P.C.K., 2020a. BoR: Toward high-performance permissioned blockchain in RDMA-enabled network. IEEE Trans. Serv. Comput. 13 (2), 301–313. http://dx.doi.org/10.1109/TSC.2019.2948009.

Huang, D., Ma, X., Zhang, S., 2020b. Performance analysis of the raft consensus algorithm for private blockchains. IEEE Trans. Syst. Man Cybern. A 50 (1), 172–181.

Hyperledger, 2020. Measuring blockchain performance with hyperledger caliper. URL https://github.com/hyperledger/caliper.

Jalalzai, M.M., Busch, C., Richard, G.G., 2019. Proteus: A scalable BFT consensus protocol for blockchains. In: 2019 IEEE International Conference on Blockchain. pp. 308–313. http://dx.doi.org/10.1109/Blockchain.2019.00048.

Jalalzai, M.M., Richard, G., Busch, C., 2019. An experimental evaluation of BFT protocols for blockchains. In: Blockchain. ICBC 2019, Springer, pp. 34–48.

Javaid, U., Aman, M.N., Sikdar, B., 2020. A scalable protocol for driving trust management in internet of vehicles with blockchain. IEEE Internet Things J. 7 (12), 11815–11829. http://dx.doi.org/10.1109/JIOT.2020.3002711.

Jiang, L., Chang, X., Liu, Y., Mišić, J., Mišić, V.B., 2020. Performance analysis of hyperledger fabric platform: A hierarchical model approach. Peer Peer Netw. Appl. 1–12.

Jiang, Y., Lian, Z., 2019a. High performance and scalable byzantine fault tolerance. In: 2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference. ITNEC, IEEE, pp. 1195–1202.

Jiang, Y., Lian, Z., 2019b. Scalable efficient byzantine fault tolerance. In: 2019 IEEE 3rd Advanced Information Management, Communicates, Electronic and Automation Control Conference. IMCEC, IEEE, pp. 1736–1742.

Jiao, Z., Tian, R., Shang, D., Ding, H., 2018. Bicomp: A bilayer scalable nakamoto consensus protocol. CoRR abs/1809.01593. arXiv:1809.01593. URL http://arxiv.org/abs/1809.01593.

Judmayer, A., Zamyatin, A., Stifter, N., Voyiatzis, A.G., Weippl, E., 2017. Merged mining: Curse or cure? In: Data Privacy Management, Cryptocurrencies and Blockchain Technology. Springer, Cham, pp. 316–333.

Kalodner, H., Goldfeder, S., Chen, X., Weinberg, S.M., Felten, E.W., 2018. Arbitrum: Scalable, private smart contracts. In: 27th USENIX Security Symposium. USENIX Security 18, USENIX Association, Baltimore, MD, pp. 1353–1370, URL https://www.usenix.org/conference/usenixsecurity18/presentation/kalodner.

Kiayias, A., Russell, A., David, B., Oliynykov, R., 2017. Ouroboros: A provably secure proof-of-stake blockchain protocol. In: Advances in Cryptology. CRYPTO 2017, Springer, Cham, pp. 357–388.

Kim, S., Kwon, Y., Cho, S., 2018. A survey of scalability solutions on blockchain. In: 2018 International Conference on Information and Communication Tech. Convergence. ICTC. pp. 1204–1207.

Kim, S., Lee, S., Jeong, C., Cho, S., 2020. Byzantine fault tolerance based multi-block consensus algorithm for throughput scalability. In: 2020 International Conference on Electronics, Information, and Communication. ICEIC, IEEE, pp. 1–3.

Kim, T., Noh, J., Cho, S., 2019. Scc: storage compression consensus for blockchain in lightweight IoT network. In: 2019 IEEE International Conference on Consumer Electronics. ICCE, IEEE, pp. 1–4.

Klems, M., Eberhardt, J., Tai, S., Härtlein, S., Buchholz, S., Tidjani, A., 2017. Trustless intermediation in blockchain-based decentralized service marketplaces. In: International Conference on Service-Oriented Computing. Springer, pp. 731–739.

Kocsis, I., Pataricza, A., Telek, M., Klenik, A., Deé, F., Cseh, D., 2017. Towards performance modeling of hyperledger fabric. In: International IBM Cloud Academy Conference. ICACON.

Kogias, E.K., Jovanovic, P., Gailly, N., Khoffi, I., Gasser, L., Ford, B., 2016. Enhancing bitcoin security and performance with strong consistency via collective signing. In: 25th USENIX Security Symposium. USENIX Security 16, USENIX Association, Austin, TX, pp. 279–296, URL https://www.usenix.org/conference/usenixsecurity16/technical-sessions/presentation/kogias.

Kokoris-Kogias, E., Jovanovic, P., Gasser, L., Gailly, N., Syta, E., Ford, B., 2018. OmniLedger: A secure, scale-out, decentralized ledger via sharding. In: 2018 IEEE Symposium on Security and Privacy. SP. pp. 583–598.

Kuzlu, M., Pipattanasomporn, M., Gurses, L., Rahman, S., 2019. Performance analysis of a hyperledger fabric blockchain framework: Throughput, latency and scalability. In: 2019 IEEE International Conference on Blockchain. pp. 536–540.

Kwon, J., Buchman, E., 2020. Cosmos white paper: A network of distributed ledgers. Cosmos. URL https://cosmos.network/resources/whitepaper.

Kwon, M., Yu, H., 2019. Performance improvement of ordering and endorsement phase in hyperledger fabric. In: 2019 Sixth International Conference on Internet of Things: Systems, Management and Security. IOTSMS, pp. 428–432. http://dx.doi.org/10.1109/IOTSMS48152.2019.8939202.

labs, B., 2018. What is μraiden? URL https://microraiden.readthedocs.io/en/latest/.

Lao, L., Dai, X., Xiao, B., Guo, S., 2020. G-PBFT: A location-based and scalable consensus protocol for IoT-blockchain applications. In: 2020 IEEE International Parallel and Distributed Processing Symposium. IPDPS, IEEE, pp. 664–673.

Lau, J., 2016. Merkelized abstract syntax tree. BIP: 114. URL https://github.com/bitcoin/bips/wiki/Comments:BIP-0114.

Lee, S.-b., Hwang, D., Kim, J., Kim, K.-H., 2020. Proof-of-lottery: Design for block producing algorithm based on PoS for scalability. In: 2020 International Conference on Information Networking. ICOIN, IEEE, pp. 666–669.

Lee, J.W., Park, S., 2020. A study on performance improvement of hyperledger fabric through batched chaincode message. In: 2020 21st Asia-Pacific Network Operations and Management Symposium. APNOMS, pp. 259–262. http://dx.doi.org/10.23919/APNOMS50412.2020.9236779.

Lee, H., Yoon, C., Bae, S., Lee, S., Lee, K., Kang, S., Sung, K., Min, S., 2019. Multi-batch scheduling for improving performance of hyperledger fabric based IoT applications. In: 2019 IEEE Global Communications Conference. GLOBECOM, pp. 1–6. http://dx.doi.org/10.1109/GLOBECOM38437.2019.9013551.

Lerner, S.D., 2015. DagCoin: a cryptocurrency without blocks. White paper.

Lerner, S.D., 2017. Lumino transaction compression protocol (LTCP). RSK Labs-Rev10. URL https://docs.rsk.co/LuminoTransactionCompressionProtocolLTCP.pdf.

Lerner, S.D., 2019. RSK-Rootstock platform: Bitcoin powered smart contracts. White paper, revision 11. URL https://www.rsk.co/Whitepapers/RSK-White-Paper-Updated.pdf.

Lewenberg, Y., Sompolinsky, Y., Zohar, A., 2015. Inclusive block chain protocols. In: Financial Cryptography and Data Security. Springer, Berlin, Heidelberg, pp. 528–547.

Li, H., Li, Z., Tian, N., 2020. Resource bottleneck analysis of the blockchain based on tron's TPS. pp. 944–950. http://dx.doi.org/10.1007/978-3-030-32591-6{_}103.

Li, C., Li, P., Xu, W., Long, F., Chi-Chih Yao, A., 2018. Scaling nakamoto consensus to thousands of transactions per second. CoRR abs/1805.03870. arXiv:1805.03870. URL http://arxiv.org/abs/1805.03870.

Li, S., Yu, M., Yang, C., Avestimehr, A.S., Kannan, S., Viswanath, P., 2021. PolyShard: Coded sharding achieves linearly scaling efficiency and security simultaneously. IEEE Trans. Inf. Forensics Secur. 16, 249–261. http://dx.doi.org/10.1109/TIFS.2020.3009610.

Limited, D.T., 2020. Digital transaction. URL https://www.digital-transaction.com/.

Liu, D., Alahmadi, A., Ni, J., Lin, X., Shen, X., 2019. Anonymous reputation system for IIoT-enabled retail marketing atop PoS blockchain. IEEE Trans. Ind. Inf. 15 (6), 3527–3537. http://dx.doi.org/10.1109/TII.2019.2898900.

Liu, Y., He, D., Obaidat, M.S., Kumar, N., Khan, M.K., Raymond Choo, K.-K., 2020a. Blockchain-based identity management systems: A review. J. Netw. Comput. Appl. 166, 102731. http://dx.doi.org/10.1016/j.jnca.2020.102731.

Liu, J., Li, W., Karame, G.O., Asokan, N., 2019. Scalable byzantine consensus via hardware-assisted secret sharing. IEEE Trans. Comput. 68 (1), 139–151. http://dx.doi.org/10.1109/TC.2018.2860009.

Liu, Y., Qian, K., Yan, J., Wang, K., He, L., 2020b. Effective scaling of blockchain beyond consensus innovations and Moore's law. arXiv:2001.01865v1. URL arXiv:2001.01865v1.

Liu, X., Yu, X., Ma, X., Kuang, H., 2020. A method to improve the fresh data query efficiency of blockchain. In: 2020 12th International Conference on Measuring Technology and Mechatronics Automation. ICMTMA, pp. 823–827. http://dx.doi.org/10.1109/ICMTMA50254.2020.00179.

Lombrozo, E., Lau, J., Wuille, P., 2015. Segregated witness (consensus layer). BIP141. URL https://github.com/bitcoin/bips/blob/master/bip-0141.mediawiki.

Long, J., Wei, R., 2019. Scalable BFT consensus mechanism through aggregated signature gossip. In: 2019 IEEE International Conference on Blockchain and Cryptocurrency. ICBC, pp. 360–367.

Ltd, L.P., 2020. Remarkable throughput. LoopRing. URL https://loopring.org/#/protocol.

Lu, F., Gan, L., Dong, Z., Li, W., Jin, H., Zomaya, A.Y., 2018. A cache enhanced endorser design for mitigating performance degradation in hyperledger fabric. In: 2018 IEEE International Conference on Internet of Things (IThings) and IEEE GreenCom and IEEE CPSCom and IEEE Smart Data (SmartData). pp. 1001–1006. http://dx.doi.org/10.1109/Cybermatics{_}2018.2018.00188.

Luu, L., Narayanan, V., Zheng, C., Baweja, K., Gilbert, S., Saxena, P., 2016. A secure sharding protocol for open blockchains. In: 2016 ACM SIGSAC Conference on Computer and Communications Security. CCS '16. New York. pp. 17–30. http://dx.doi.org/10.1145/2976749.2978389.

Mahony, A.O., Popovici, E., 2019. A systematic review of blockchain hardware acceleration architectures. In: 2019 30th Irish Signals and Systems Conference. ISSC, IEEE, pp. 1–6.

Makhdoom, I., Abolhasan, M., Abbas, H., Ni, W., 2019. Blockchain's adoption in IoT: The challenges, and a way forward. J. Netw. Comput. Appl. 125, 251–279. http://dx.doi.org/10.1016/j.jnca.2018.10.019.

Malavolta, G., Moreno-Sanchez, P., Schneidewind, C., Kate, A., Maffei, M., 2019. Anonymous multi-hop locks for blockchain scalability and interoperability. In: NDSS.

Malik, S., Dedeoglu, V., Kanhere, S.S., Jurdak, R., 2019. TrustChain: Trust management in blockchain and IoT supported supply chains. In: 2019 IEEE International Conference on Blockchain (Blockchain). pp. 184–193. http://dx.doi.org/10.1109/Blockchain.2019.00032.

Martino, W., Quaintance, M., Popejoy, S., 2018a. Chainweb: A proof-of-work parallel-chain architecture for massive throughput. Chainweb whitepaper. 19.

Martino, W., et al., 2018b. The kadena public blockchain project summary whitepaper. 1, pp. 1–7, Version.

Mazlan, A.A., Daud, S.M., Sam, S.M., Abas, H., Rasid, S.Z.A., Yusof, M.F., 2020. Scalability challenges in healthcare blockchain system—A systematic review. IEEE Access 8, 23663–23673.

McCorry, P., Buckland, C., Bakshi, S., Wüst, K., Miller, A., 2020. You sank my battleship! a case study to evaluate state channels as a scaling solution for cryptocurrencies. In: Financial Cryptography and Data Security. Springer, Cham, pp. 35–49.

Miller, A., Bentov, I., Kumaresan, R., McCorry, P., 2017. Sprites: Payment channels that go faster than lightning. CoRR abs/1702.05812. arXiv:1702.05812. URL http://arxiv.org/abs/1702.05812.

Millman, R., 2020. What is ethereum 2.0 and why does it matter? Decrypt. URL https://decrypt.co/resources/what-is-ethereum-2-0.

Monrat, A.A., Schelén, O., Andersson, K., 2019. A survey of blockchain from the perspectives of applications, challenges, and opportunities. IEEE Access 7, 117134–117151.

Multichain, 2020. Enterprise blockchain that actually works.

Nasir, Q., Qasse, I., Talib, M., Nassif, A., 2018. Performance analysis of hyperledger fabric platforms. Secur. Commun. Netw. 2018, 1–14. http://dx.doi.org/10.1155/2018/3976093.

Nick, J., Poelstra, A., Sanders, G., 2020. Liquid: A bitcoin sidechain. Liquid white paper. URL https://blockstream.com/assets/downloads/pdf/liquid-whitepaper.pdf.

Ochôa, I., Piemontez, R., Martins, L., Leithardt, V., Zeferino, C., 2019. Experimental analysis of the scalability of ethereum blockchain in a private network. In: Proceedings of the 2nd Workshop Em Blockchain: Theory, Technology, and Applications. SBC, Porto Alegre, RS, Brasil, http://dx.doi.org/10.5753/wblockchain.2019.7481.

Park, S., Oh, S., Kim, H., 2019. Performance analysis of DAG-based cryptocurrency. In: 2019 IEEE International Conference on Communications Workshops. ICC Workshops. pp. 1–6.

Pass, R., Shi, E., 2016. Hybrid consensus: Scalable permissionless consensus.

Pawczuk, L., Massey, R., Holdowsky, J., 2019. Deloitte 2019 global blockchain survey - blockchain gets down to business. Deloitte Insights URL https://www2.deloitte.com/content/dam/Deloitte/se/Documents/risk/DI_2019-global-blockchain-survey.pdf.

Pinar Ozisik, A., Andresen, G., Bissias, G., Houmansadr, A., Levine, B., 2017. Graphene: A new protocol for block propagation using set reconciliation. In: Data Privacy Management, Cryptocurrencies and Blockchain Technology. Springer, pp. 420–428.

Poon, J., Buterin, V., 2017. Plasma: Scalable autonomous smart contracts. White paper. pp. 1–47.

Poon, J., Dryja, T., 2016. The bitcoin lightning network: Scalable off-chain instant payments. URL https://www.bitcoinlightning.com/bitcoin-lightning-network-whitepaper/.

Popov, S., 2016. The tangle. p. 131, Cit. on. URL https://www.iota.org/foundation/research-papers.

Qu, Q., Nurgaliev, I., Muzammal, M., Jensen, C.S., Fan, J., 2019. On spatio-temporal blockchain query processing. Future Gener. Comput. Syst. 98, 208–218.

randao.org, 2017. Randao: Verifiable random number generation. Randao whitepaper. URL https://www.randao.org/whitepaper/Randao_v0.85_en.pdf.

Riley, C., 2019. Know your API protocols: SOAP vs. REST vs. JSON-RPC vs. gRPC vs. graphql vs. Thrift. Mertech Data Syst. URL https://www.mertech.com/blog/know-your-api-protocols.

Rizun, P.R., 2016. Towards massive on-chain scaling: Block propagation results with xthin. URL https://medium.com/@peter_r/towards-massive-on-chain-scaling-block-propagation-results-with-xthin-a0f1e3c23919.

Roehrs, A., André da Costa, C., da Rosa Righi, R., Ferreira da Silva, V., Goldim, J.R., Schmidt, D.C., 2019. Analyzing the performance of a blockchain-based personal health record implementation. J. Biomed. Inform. 92, 103140. http://dx.doi.org/10.1016/j.jbi.2019.103140.

Rouhani, S., Deters, R., 2017. Performance analysis of ethereum transactions in private blockchain. In: 2017 8th IEEE International Conference on Software Engineering and Service Science. ICSESS. pp. 70–74.

Rüsch, S., Messadi, I., Kapitza, R., 2018. Towards low-latency byzantine agreement protocols using RDMA. In: 2018 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops. DSN-W, pp. 146–151. http://dx.doi.org/10.1109/DSN-W.2018.00054.

Sahoo, M.S., Baruah, P.K., 2018. HBasechainDB – A scalable blockchain framework on hadoop ecosystem. In: Supercomputing Frontiers. Springer, pp. 18–29.

Sakakibara, Y., Morishima, S., Nakamura, K., Matsutani, H., 2018. A hardware-based caching system on FPGA NIC for blockchain. IEICE Trans. Inf. Syst. E101.D (5), 1350–1360. http://dx.doi.org/10.1587/transinf.2017EDP7290.

sallal, M.F., Owenson, G., Adda, M., 2017. Proximity awareness approach to enhance propagation delay on the bitcoin peer-to-peer network. In: 2017 IEEE 37th International Conference on Distributed Computing Systems. ICDCS, pp. 2411–2416. http://dx.doi.org/10.1109/ICDCS.2017.53.

Sanka, A.I., Cheung, R.C.C., 2018. Efficient high performance FPGA based NoSQL caching system for blockchain scalability and throughput improvement. In: 2018 26th International Conference on Systems Engineering. ICSEng. pp. 1–8.

Sanka, A.I., Cheung, R.C., 2020. Appendix A: List of final screened/reviewed papers of our systematic review of blockchain scalability paper. URL https://tinyurl.com/Sanka2020review.

Sanka, A.I., Irfan, M., Huang, I., Cheung, R.C., 2021. A survey of breakthrough in blockchain technology: Adoptions, applications, challenges and future research. Comput. Commun. 169, 179–201. http://dx.doi.org/10.1016/j.comcom.2020.12.028.

Schäffer, M., di Angelo, M., Salzer, G., 2019. Performance and scalability of private ethereum blockchains. In: Business Process Management: Blockchain and Central and Eastern Europe Forum. Springer, pp. 103–118.

Shahsavari, Y., Zhang, K., Talhi, C., 2020. A theoretical model for block propagation analysis in bitcoin network. IEEE Trans. Eng. Manage. 1–18.

Shi, Z., Zhou, H., Hu, Y., Jayachander, S., de Laat, C., Zhao, Z., 2019. Operating permissioned blockchain in clouds: A performance study of hyperledger sawtooth. In: 2019 18th International Symposium on Parallel and Distributed Computing. ISPDC. pp. 50–57.

Singh, A., Click, K., Parizi, R.M., Zhang, Q., Dehghantanha, A., Choo, K.-K.R., 2020. Sidechain technologies in blockchain networks: An examination and state-of-the-art review. J. Netw. Comput. Appl. 149, 102471.

Sompolinsky, Y., Lewenberg, Y., Zohar, A., 2016. SPECTRE: A fast and scalable cryptocurrency protocol. IACR Cryptol. ePrint Arch. 2016, 1159.

Sompolinsky, Y., Zohar, A., 2015. Secure high-rate transaction processing in bitcoin. In: Financial Cryptography and Data Security. Springer, Berlin, Heidelberg, pp. 507–527.

Sompolinsky, Y., Zohar, A., 2020. Phantom, ghostdag.

Sukhwani, H., Martínez, J.M., Chang, X., Trivedi, K.S., Rindos, A., 2017. Performance modeling of PBFT consensus process for permissioned blockchain network (hyperledger fabric). In: 2017 IEEE 36th Symposium on Reliable Distributed Systems. SRDS. pp. 253–255.

Syta, E., Jovanovic, P., Kogias, E.K., Gailly, N., Gasser, L., Khoffi, I., Fischer, M.J., Ford, B., 2017. Scalable bias-resistant distributed randomness. In: 2017 IEEE Symposium on Security and Privacy. SP. pp. 444–460.

Syta, E., Tamas, I., Visher, D., Wolinsky, D.I., Jovanovic, P., Gasser, L., Gailly, N., Khoffi, I., Ford, B., 2016. Keeping authorities “Honest or Bust” with decentralized witness cosigning. In: 2016 IEEE Symposium on Security and Privacy. SP. pp. 526–545.

Tan, D., Hu, J., Wang, J., 2019. VBBFT-raft: An understandable blockchain consensus protocol with high performance. In: 2019 IEEE 7th International Conference on Computer Science and Network Technology. ICCSNT, IEEE, pp. 111–115.

Team, T.Z., 2018. The zilliqa project: A secure, scalable blockchain platform.

Team, T.H., 2019. Open consensus for 10 billion people. URL https://harmony.one/.

Teutsch, J., Reitwießner, C., 2019. A scalable verification solution for blockchains. TrueBit white paper. URL arXiv preprint arXiv:1908.04756.

Thakkar, P., Nathan, S., Viswanathan, B., 2018. Performance benchmarking and optimizing hyperledger fabric blockchain platform. In: 2018 IEEE 26th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems. MASCOTS. pp. 264–276.

Thilagavathi, M., Lopez, D., 2020. Enhancing blockchain performance using parallel merkle root and parallel proof of work. URL https://www.jardcs.org/abstract.php?id=3564.

Thilakaratne, M., Falkner, K., Atapattu, T., 2019. A systematic review on literature-based discovery: General overview, methodology, and statistical analysis. ACM Comput. Surv. 52 (6), http://dx.doi.org/10.1145/3365756, URL https://doi-org.ezproxy.cityu.edu.hk/10.1145/3365756.

Thomson, G., 2020. Ethereum 2.0 will walk and ‘roll’ for two years before it can run. Decrypt. URL https://decrypt.co/34204/ethereum-2-0-will-walk-and-roll-for-two-years-before-it-can-run.

Toomim, J., 2018. Benefits of LTOR in block entropy encoding (Xthinner). URL https://medium.com/@j_73307/benefits-of-ltor-in-block-entropy-encoding-or-8d5b77cc2ab0.

Trihinas, D., 2019. Datachain: A query framework for blockchains. In: Proceedings of the 11th International Conference on Management of Digital EcoSystems. pp. 134–141.

Tschipper, P., 2016. Xtreme thinblocks. Buip010. URL https://bitco.in/forum/threads/buip010-passed-xtreme-thinblocks.774/.

Wang, S., 2019. Performance evaluation of hyperledger fabric with malicious behavior. In: Blockchain. ICBC 2019, Springer, Cham, pp. 211–219.

Wang, S., Dinh, T.T.A., Lin, Q., Xie, Z., Zhang, M., Cai, Q., Chen, G., Fu, W., Ooi, B.C., Ruan, P., 2018. Forkbase: An efficient storage engine for blockchain and forkable applications. arXiv preprint arXiv:1802.04949.

Wang, K., Kim, H.S., 2019. FastChain: Scaling blockchain system with informed neighbor selection. In: 2019 IEEE International Conference on Blockchain (Blockchain). pp. 376–383. http://dx.doi.org/10.1109/Blockchain.2019.00058.

Wang, Y., Song, Z., Cheng, T., 2020. Improvement research of PBFT consensus algorithm based on credit. In: Blockchain and Trustworthy Systems. Springer, Singapore, pp. 47–59.

Wang, J., Wang, H., 2019. Monoxide: Scale out blockchains with asynchronous consensus zones. In: 16th USENIX Symposium on Networked Systems Design and Implementation. NSDI 19, USENIX Association, Boston, MA, pp. 95–112, URL https://www.usenix.org/conference/nsdi19/presentation/wang-jiaping.

Wang, X., Zha, X., Ni, W., Liu, P., Jay, Y.G., Niu, X., Zheng, K., 2019. Survey on blockchain for Internet of Things. Comput. Commun. 136, 10–29. http://dx.doi.org/10.1016/j.comcom.2019.01.006.

Wickboldt, C., 2019. Benchmarking a Blockchain-based Certification Storage System, No. 2019/5. Freie Universität Berlin, Fachbereich Wirtschaftswissenschaft, Berlin, URL http://hdl.handle.net/10419/195585.

Wood, G., 2016. Polkadot: Vision for a heterogeneous multi-chain framework. White paper.

Wüst, K., Matetic, S., Egli, S., Kostiainen, K., Capkun, S., 2020. Ace: Asynchronous and concurrent execution of complex smart contracts. In: Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security. pp. 587–600.

Xie, J., Yu, F., Huang, T., Xie, R., Liu, J., Liu, Y., 2019. A survey on the scalability of blockchain systems. IEEE Netw. 33 (5), 166–173.

Xu, Z., Han, S., Chen, L., 2018. CUB, a consensus unit-based storage scheme for blockchain system. In: 2018 IEEE 34th International Conference on Data Engineering. ICDE. pp. 173–184.

Yang, L., Bagaria, V., Wang, G., Alizadeh, M., Tse, D., Fanti, G., Viswanath, P., 2020. Prism: Scaling bitcoin by 10,000x. arXiv:1909.11261.

Yang, S., Chen, Z., Cui, L., Xu, M., Ming, Z., Xu, K., 2019. CoDAG: An efficient and compacted DAG-based blockchain protocol. In 2019 IEEE International Conference on Blockchain (Blockchain). pp. 314–318.

Yu, W., Luo, K., Ding, Y., You, G., Hu, K., 2018. A parallel smart contract model. In: Proceedings of the 2018 International Conference on Machine Learning and Machine Intelligence. MLMI2018, pp. 72–77. http://dx.doi.org/10.1145/3278312.3278321.

Yu, H., Nikolic, I., Hou, R., Saxena, P., 2019. OHIE: Blockchain scaling made simple. arXiv:1811.12628.

Yu, L., Tsai, W.-T., Li, G., Yao, Y., Hu, C., Deng, E., 2017. Smart-contract execution with concurrent block building. pp. 160–167. http://dx.doi.org/10.1109/SOSE.2017.33.

Yu, G., Wang, X., Yu, K., Ni, W., Zhang, J.A., Liu, R.P., 2020. Survey: Sharding in blockchains. IEEE Access 14155–14181.

Yuan, P., Zheng, K., Xiong, X., Zhang, K., Lei, L., 2020. Performance modeling and analysis of a Hyperledger-based system using GSPN. Comput. Commun. 153, 117–124. http://dx.doi.org/10.1016/j.comcom.2020.01.073.

Zamani, M., Movahedi, M., Raykova, M., 2018. RapidChain: Scaling blockchain via full sharding. In: Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security. CCS ’18, Association for Computing Machinery, New York, NY, USA, pp. 931–948. http://dx.doi.org/10.1145/3243734.3243853.

Zhang, H., Babar, M.A., Tell, P., 2011. Identifying relevant studies in software engineering. Inf. Softw. Technol. 53 (6), 625–637. http://dx.doi.org/10.1016/j.infsof.2010.12.010, Special Section: Best papers from the APSEC.

Zhang, J., Gao, J., Wu, Z., Yan, W., Wo, Q., Li, Q., Chen, Z., 2019. Performance analysis of the libra blockchain: An experimental study. In: 2019 2nd International Conference on Hot Information-Centric Networking. HotICN. pp. 77–83.

Zhang, J., Rong, Y., Cao, J., Rong, C., Bian, J., Wu, W., 2019. DBFT: A Byzantine fault tolerant protocol with graceful performance degradation. In: 2019 38th Symposium on Reliable Distributed Systems. SRDS, IEEE, pp. 123–12309.

Zheng, Q., Li, Y., Chen, P., Dong, X., 2018. An innovative IPFS-based storage model for blockchain. In: 2018 IEEE/WIC/ACM Int. Conference on Web Intelligence. WI, pp. 704–708. http://dx.doi.org/10.1109/WI.2018.000-8.

Zhou, H.-S., 2019. Fractal: A new paradigm for high-performance proof-of-stake blockchains. In: Proceedings of the Seventh International Workshop on Security in Cloud Computing. SCC ’19, Association for Computing Machinery, New York, NY, USA, p. 3. http://dx.doi.org/10.1145/3327962.3331459.

Zhou, Q., Huang, H., Zheng, Z., Bian, J., 2020. Solutions to scalability of blockchain: A survey. IEEE Access 8 (""), 16440–16455.

Zou, J., Dong, Z., Shao, A., Zhuang, P., Li, W., Zomaya, A.Y., 2018. 3D-DAG: A high performance DAG network with eventual consistency and finality. In: 2018 1st IEEE International Conference on Hot Information-Centric Networking. HotICN. pp. 262–263.

Abdurrashid Ibrahim Sanka received B.Eng. Electrical from Bayero University Kano, Nigeria in 2011. He got MSc. in Embedded Microelectronics and Wireless Systems from Coventry University, United Kingdom, in 2014. He has been working with the Bayero University, Kano as a lecturer since 2012. Currently, he is working towards Ph.D. degree with the department of Electrical Engineering, City University of Hong Kong. His research interests include blockchain technology, digital systems design, network computing and information security.

Ray C. C. Cheung received B.Eng. and M.Phil. degrees in computing engineering from CUHK in 1999 and 2001 respectively. He received his Ph.D. degree in computing from Imperial College London (IC) in 2007. He received the Hong Kong Croucher Foundation Fellowship for postdoctoral and doctoral research work at UCLA and IC. In 2009, he visited Princeton University as a visiting research fellow. He is currently an associate professor in the department of Electrical Engineering, City University of Hong Kong. His current research interests include blockchain technology, cryptographic hardware design; rapid prototyping trusted computing platforms and high-performance biomedical VLSI designs.