Decentralization, Scalability, and Security Trade-off in Blockchain System: Comparison on Different Approaches

RICCO PRATAMA HALIM, University of Twente, The Netherlands

With all the features and benefits it brings, blockchain technology has gained popularity in various industries and businesses sector in this modern day. This wide-used adoption has left researchers and technical developers with some challenges to optimize the blockchain structure in general. One well-known problem involves the state of decentralization, scalability, and security of the blockchain itself, which is further termed as a blockchain *trilemma*. The problem exists as trade-offs are usually taken place, preventing a blockchain to have a fully maximized state of decentralization, scalability, and security simultaneously. Various approaches and ideas have been formulated to overcome the mentioned trade-off. This paper aims to investigate and view the work done by other researchers regarding the mentioned *trilemma*. A taxonomy and comparison of blockchain solutions to achieve the trade-off between the *trilemma* will also be done and a conclusion to be drawn.

Additional Key Words and Phrases: Blockchain, Decentralization, Scalability, Security, Trade-off, *Trilemma*

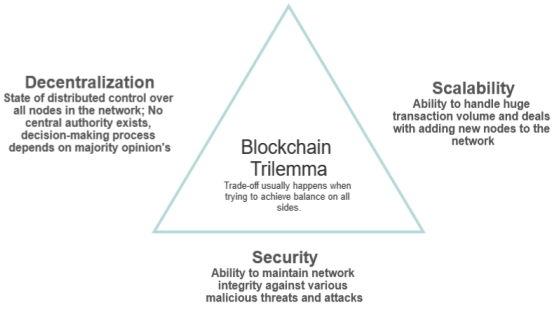

Fig. 1. A conclusion on the *trilemma*.

1 INTRODUCTION

Making its first appearance back in 2008, blockchain has been known for its potential to create new foundation structures in many different areas. It was considered as a digital public transaction ledger of the first decentralized cryptocurrency, Bitcoin [44]. As time progress, the usage has shifted from a mere backbone of digital currency to a more demanding broad application. Especially with the emergence of Web 3.0 [4] and Internet of Things (IoT) [5], blockchain has been a breakthrough technology that is worth looking. It becomes chart-topping technology and has been proposed to be applied in various industries and business sectors, such as financial, healthcare, and food domains [3], as it offers far greater benefits compared to the traditional approach in the mentioned sectors. The basic nature of blockchain, which involves decentralization, scalability, and security can be considered as the key features and requirements that attract [13].

Decentralization is the state of blockchain referring to distributed control over the network rather than a single central point [32]. This leads to equally distributed power for everyone using the network. This can lead to a more secure environment since every node in the network has its own copy, attackers should breach all participating nodes in the network to change a single piece of data. Compared to the centralized system, there is a sole authority that has a single point of failure when attacked [64].

While scalability refers to how scalable is a blockchain in handling a massive number of transactions, without altering user experiences such as processing speed and the cost. It also covers the ability to add new nodes to the network [16]. This factor can be considered the most problematic when it comes to realism. Bitcoin, for example, has been reported to can only handle approximately 7 transactions per second. As a comparison, a centralized finance organization, such as VISA, is able to process around 24000 transactions per second [58]. Issues also occur in the attempt of deploying blockchain in IoT systems [5]. IoT is considered to have a vast number of devices that should be connected, while blockchain has scaling difficulties when the number of nodes increases [61].

Last but the most important, security refers to the ability in maintaining a blockchain's network integrity from threats and malicious attacks. Blockchain would simply fail to operate without robust security as a node can alter and change transaction details abundantly. Various attack patterns have been known and available solutions have been discussed thoroughly by researchers and developers [28]. The state of security is also directly proportional to decentralization as explained in the example above.

These mentioned key features are critical to achieving a stable and wide adoption. Nevertheless, when it comes to practicality, for the most part, it is impossible to achieve a balance between decentralization, scalability, and security in a blockchain network. One condition is most likely to be traded off and negatively affected for the sake of achieving the other two. The initial idea was mentioned by Vitalik Buterin, the founder of Ethereum blockchain [14], which was then further termed as the blockchain *trilemma*.

Researchers and developers are actively looking for solutions, and lots of significant attempts have been made. With all available attempts, each holds its own weaknesses and strengths compared to the other approaches. However, extensive discussions and comparisons among these available solutions are lacking. This lead to a question of what approaches have been tried before and what is the best approach closest to success.

This paper will investigate and review the work done by other researchers and developers regarding the achievement of the *trilemma*. The goals of this research project then can be structured in two parts:

TScIT 37, July 8, 2022, Enschede, The Netherlands

Goal 1: To investigate the most effective and viable way to overcome the decentralization, scalability, and security problem in a blockchain system.

Goal 2: To produce an extensive comparison of various works and attempts that have been made by other researchers.

These goals will be achieved with the support of these research questions (RQ) below:

- **RQ1:** How does decentralization, scalability, and security, are being traded off and achieved in the context of blockchain? What does the trade-off do in each of the works mentioned in the literature?
- **RQ2:** What is the best approach to maximize the decentralization, scalability, and security of a blockchain, without compromising each other state?

To achieve the goals and answer the research questions mentioned, several steps will be planned. First, a literature review on the *trilemma* problem of blockchain will be done to have a solid understanding. This will also help to bound the research areas and focus on the three main problems. Only after that, RQ1 and RQ2 will be answered.

To answer RQ1, systematic literature reviews on the work that has been done regarding the problem by other researchers will be done. Each of them will be broken down into details, making sure that differences among those works can be observed. This will give comprehensive views on how the current solutions attempt to solve the trade-off between decentralization, scalability, and security.

Once the literature reviews are done and RQ1 has been answered, a comparison table is to be made. This is settled to view the weakness and strengths of each approach so that there is a clear overview and the best approach can be concluded, which is the main point of RQ2.

The rest of the paper is arranged as follows. Section 2 discusses related works and literatures regarding the *trilemma*. Section 3 presents the analysis comparison of each approaches. Section 4 provides a discussion of the analysis and future direction of blockchain *trilemma* problem. Finally, section 5 concludes and summarizes this paper.

2 RELATED WORK

The literatures are gathered from various research domain such as Scopus, IEEE, Arxiv and Google Scholar, using the search terms of *"blockchain", "trilemma", "tradeoff", "decentralization", "scalability"* and *"security"*. For the various works, the name of the specific approach is searched within the aforementioned domain.

Various approaches on different layers have been made to tackle the *trilemma*. These diverse attempts can be categorized into three different bigger general sections, namely, First Layer Solutions, Second Layer Solutions, and Scalable Distributed Ledgers. Some of the works done from each category are discussed in the below sections.

2.1 First Layer Solutions

First Layer Solutions, also known as *on-chain scaling solutions*, refer to the approaches made which necessitate changes to the main blockchain network's codebase to overcome the problem. Therefore, the network will have to be hard forked [23] or soft forked [24] depending on the solution. Various commercial attempts have been made, for example, [1, 34, 38, 68, 72]. Different types of theoretical research [19, 33] are also produced throughout the last decade. The first layer solution has two main common techniques in general, to either modify the block size or partition the network into a smaller subset called *sharding*. Both are mentioned and explained further below.

2.1.1 ***Block Size***. It is mentioned that the block period in Bitcoin is around 10 minutes, and the block size of approximately one megabyte(MB) [44]. This restricts the number of transactions that may be stored in each block. The key idea of this category is to modify the size of each block, either make it larger or smaller. Theoretically, more transactions can be included in a block when the block size is increased, thus increasing the transaction throughput and improved scalability. A similar result can be achieved via block compression, which will minimize storage usage over the blocks. This section will discuss several notable attempts, both commercial and non-commercial.

Segregated Witness. Developed back in 2015, Segregated Witness [38] was a Bitcoin protocol that was created to optimize transactions. It accomplishes the aims by increasing block capacity and segregating the transaction into two different parts, the wallet addresses of the receiver and the *"witness"* data that holds the transaction signatures for verification. Figure 3 in Appendix A gives an illustration of this, where the signature script of each transaction block is stored outside of the block storage. It is kept in an extended block or *witness*, where it will be weighted accordingly when determining the overall block limit. SegWit was pushed in via a soft fork to the former Bitcoin network in July 2017.

Bitcoin Cash. Bitcoin Cash [1] was born as a hard fork of the Bitcoin network after the Bitcoin community rejected the idea in August 2017. It was designed to have a bigger block size than Bitcoin, ranging from 8 MB to 32 MB, enabling even more transactions inside a single block and thus higher throughput.

Merkle Trees. Back in 1980, R. Merkle [42] presented a cryptographic protocol that helps hash huge transaction data. Today, it is used in Bitcoin as a way to efficiently store transactions [60]. In a blockchain system, it is often used to verify the data integrity that is stored inside the chain. A user may confirm if it contains a transaction in a block by using the Merkle tree, which adds up every transaction in the block and creates a digital signature of the complete set of activities.

Merkelized Abstract Syntax Trees. It is a data structure designed by Rubin *et al.* [60] for Bitcoin network in 2014. It integrates the concept of Merkle Trees [42] and Abstract Syntax Trees (ASTs). It aims to compress a block by eliminating useless script segments on it while maintaining data integrity and compression. The script here refers to a program that can be written by users and default in every wallet, which is used as dynamic public keys and signatures. ASTs create a separation of a program into its constituent pieces, while Merkle Trees are used to verify these small individual units belong to a full program even if the entire program is not present. They

argued that a good code compression scheme has a huge role in scaling blockchain cryptosystems. Some benefits of MASTs involve smaller transactions, boosting privacy, and enlarging smart contract size. According to Rubin *et al.*, MASTs data structure allows for a program of length n to be compressed to O(log n).

Txilm. In 2019, Ding *et al.* [20] proposed Txilm, a protocol that aims to compress transaction's size in each block to conserve network bandwidth. The compression lies in the fact that instead of full transactions, a block contains truncated hashes of Transaction IDs (TXID). They were aware that when a shorter hash is utilized, hash collisions are more common, and they also analyze the chance of it and give methods to resolve such collisions. Txilm enhances the algorithm by applying *salt* while constructing the hash of TXIDs to lower the chance of hash collision and help protect the system from a collision attack. They claimed that Txilm reduces data size by up to 80 times when compared to the traditional blockchain approach.

2.1.2 ***Sharding.*** Instead of increasing or compressing block size, this approach takes advantage by splitting the network into smaller multiple sets. These sets are represented as shards, and each of them will process different data types on a transaction simultaneously, enhancing the intensity of processing and transaction verification. Each node is no longer in charge of handling the whole network's transactional load, instead, it simply keeps data on its partition. Consequently, overall network performance will scale as the number of shards increases, as it can operate larger transactions each time. This fact also forces a communication from each shard to update the current status of the blockchain. Information can be broadcasted from shard to shard via intra-shard nodes, which are in charge of maintaining the chain, a committee for instance.

In 2016, Luu *et al.* [39] proposed Elastico, which is the first paper to introduce the idea of sharding in the blockchain. Afterward, lots of research on sharding arise to create the further perfect solution, such as OmniLedger [34], Monoxide [72], and Zilliqa [68]. Theoretical approaches also made such as SecuSca [19] by Del Monte *et al.* in 2020 and Dynamic Sharding [33] proposed by Khacef *et al.* in 2021. These are discussed further in the paragraphs below.

Elastico. Developed in 2016, with the aim of scaling transaction rates, Elastico [39] become the first protocol to implement a sharding mechanism. It utilizes Proof-of-Work consensus to build committees and Practical Byzantine Fault Tolerance (PBFT) [15] to achieve intra-committee consensus. In brief, the network participants must complete a Proof-of-Work problem to decide on the consensus committee. Each committee will then work in a form of a shard, running PBFT to agree upon the consensus. The outcome will be submitted to a leader committee, which will be in charge of making final decisions on the consensus outcomes of other shards. These shards amount increases about linearly with network growth. Nevertheless, there are some security issues such as it can only tolerate approximately 25% of malicious nodes in total and 33% in each committee, resulting in a significant failure probability.

Zilliqa. A novel blockchain platform was proposed by the Zilliqa Team [68] in 2017 to improve scalability via transaction rates. With the increased number of miners on the platform, the transaction rates are also expected to increase. They claimed that compared to Ethereum, which has a network size of 30.000 miners, Zilliqa would have processed approximately a thousand times the transaction speed of Ethereum. Still using the concept of sharding, Zilliqa increases transaction rates by processing it in numerous different shards, however, each Zilliqa node must still retain the full data of the network, limiting the system's ability to grow.

Monoxide. In 2019, J. Wang & H. Wang [72] developed Monoxide, which is claimed as an *Asynchronous Consensus Zones*, that it can scales blockchain linearly without losing the state of decentralization and security. The main idea is to partition and manage workloads across numerous independent and parallel instances of single-chain systems known as Consensus Zones. The condition of the entire network is divided into *Zones*, with each *zone* in charge of its particular piece. This holds the same concept as a shard. They also introduce *eventual atomicity*, ensuring transaction correctness and atomicity across *zones*. For the consensus, they propose *Chu-konu* mining, a revolutionary PoW system that guarantees effective mining power in each zone at the same level as the entire network, making an attack on any one zone as difficult as an attack on the entire network. Monoxide claims that the system achieves 1000 times throughput and 2000 times capacity compared to Bitcoin and Ethereum networks.

SecuSca. Khacef *et al.* [33] constructed *SecuSca* system with the goal of reducing storage load by trimming replication in each block in a distributed ledger. The idea is to break down and divide the full global blockchain across the available nodes. So, instead of having a full copy of the whole transactions, each node takes a minimized partial copy, and only stores the block header for the rest. As a result, the memory required for each node is reduced significantly, thus more transactions can be stored and contribute to the state of scalability. However, it has been mentioned in the paper that the approach may be vulnerable to attack, and can be seen as a trade-off in scalability and security.

Scaling Blockchains Without Giving up Decentralization and Security. In 2020, Del Monte *et al.* [19] come up with theoretical approach that is believed under common assumptions, trade-off between the *trilemma* does not need to take place while scaling the blockchain. Derived from the idea of sharding, this approach attempts to spread the burden of creating the next block across several parallel executing committees, including nearly all nodes and evades broadcast in all circumstances where scalability is crucial. They introduced the term *committees*, referring to selected nodes working together, carrying the computation required to check and confirm transactions as well as compute the new block. Analysis of how scalability does not impair the decentralization and security state is also presented. However, as concluded in the paper, it is still necessary to look at synchronization and committee behavior under consensus failure, and formal security proof is still needed.

2.2 Second Layer Solutions

Contrary to the First Layer, Second Layer Solutions attempts to solve the problem *off-chain*, creating a framework that handles

transactions on top of the main blockchain structure [63]. The essential principle is to host transactions while simply broadcasting a "summary" of them to the main chain. This eventually led to a higher transaction throughput since it reduces the load that the main blockchain needs to handle. This category can be further broken down into three different approaches, namely the State/Payment Channels, Sidechains, and Rollups.

2.2.1 ***State Channels.*** Developed for Ethereum in 2018, this solution essentially provides an *off-chain* communication network among nodes. Transactions involving two parties are managed outside the main chain, and they can execute an almost infinite number of transactions without overloading the main chain. The entire procedures only burden the main blockchain with two transaction records, the opening and closing state. As a result, the number of transactions that can be handled hugely increased and contribute to the scalability state of the blockchain. The term is often used interchangeably with *Payment Channel* since it is the main usage of State Channel. A more detailed implementation is explained in the paragraph below, discussing some of the well-known State Channels developed for different chains.

Lightning Network. In 2016, Poon & Dryja [55] proposed Lightning Network, a payment channel for Bitcoin. The fundamental concept can be broken down into three stages, opening the channel, executing the transactions, and closing the channel. Whenever a node wants to make a transaction with some other node, they set up an *off-chain* channel between them. A certain amount of coins that are going to be traded should be deposited in the channel. Then, both parties can make multiple trade logs, before closing the channel and report the end balance of both parties to the main blockchain. As a security measure, during the trading process, if one node behaves dishonestly, the total accumulated deposit on that specific channel will be sent to the counterparty. To further optimize the off-chain network, a Payment Channel Network is established. For instance, node A has created a channel with node B, and node C has already transacted with node B as well. So there exists a direct channel between nodes A and B, and C and B. When node A wants to make a transaction with node C, they do not need to establish a fresh channel. Instead, node B can become an intermediate between them. Figure 2 depicts the Lightning Network and the channel interactions. This Payment Channel Network can grow tremendously and a perfect routing mechanism is needed. Lighting Network has proven its strengths including swift payment rate, high throughput, and low cost due to the off-chain structure. However, some limitations of Lighting Network do exist, hindering its widespread adoption of it. The *off-chain* channel necessitates that both parties be online at the same moment to execute the transactions. Several studies [30, 31, 69] have also shown that Lightning Network may increase security risks.

Raiden Network. Developed as a payment channel to support Ethereum's scaling problem, Raiden Network [50] holds a similar concept to Lightning Network, as in both construct an *off-chain* ledgers that record all transactions between two parties without the need for mining power to validate them as they happened *off-chain*. The main difference is that Raiden Network runs on a focus group

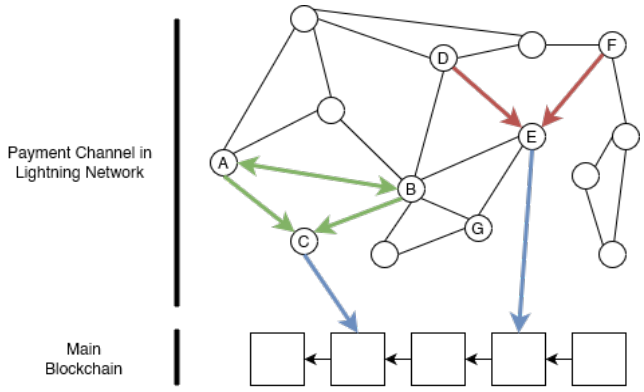

Fig. 2. Payment Channel Network illustration. Nodes A and B create a new channel between them. While nodes D and F, instead of creating a new channel, they transact via node E. After transactions are finished, the final state is reported back to the main chain (blue arrow).

of tokens that operates on top of the Ethereum network, while Lightning Network covers Bitcoin.

Trinity. Adopting the State Channel concept, Trinity [70] introduced itself as a universal *off-chain* scaling solution. It claims to achieve real-time payments, low transaction fees, scalability, and privacy protection. The technical implementation includes Proof-of-Assets (PoA) consensus and smart contract. PoA is a consensus mechanism that requires the participant to lock their tokens as collaterals to do the transaction. Whereas smart contract is used to determine rules that are agreed upon between participants and as channel management. The framework can be divided into 4 layers, Channel Service Layer, Channel Network Layer, State Channel Layer, and Block Layer, each of them providing different advantages.

Celer Network. In 2018, ScaleSphere Foundation created Celer Network [62], a platform that is claimed to bring scalability to every blockchain. The system is not a mere blockchain but rather a networked system that sits on top of current and upcoming blockchains. Its main component called *cStack*, consists of layered architecture with three main technologies, *cChannel, cRoute*, and *cOS*. These layers are based on the generalized state channel and work collectively to handle decentralized applications that are built on various chains. It has been claimed in the paper that Celer Network is scalable, trust-free, decentralized, and protects privacy.

Perun. Introduced by Dziembowski *et al.* in 2017, Perun [21] is an off-chain channel system that is claimed to provide a more efficient mechanism for linking channels than the traditional methodology of "routing transactions" over numerous channels. The system is based on smart contracts and a hub network. They proposed two types of payment channels, *Ledger Channels* and *Virtual Channels*. The *Ledger Channels* refers to a direct connection between two parties that want to transact, and it is built over the blockchain. While *Virtual Channels* is created for avoiding the involvement of an intermediary for every single payment. They believed that using this protocol helps create micro-transactions to be cheap, fast,

offline, and secure. A Perun's smart contracts are also implemented in Ethereum as proof of their concept.

2.2.2 ***Sidechains.*** The general concept of Sidechains [6] resembles State Channels, with a difference in *off-chain* structure. Sidechains are based on their *off-chain* transactions on a blockchain, whereas State Channels use a simple two-way *state channel*. Sidechains essentially transfer transaction procession to a different separated blockchain. And to eliminate the same scalability problem as the main chain, the sidechain usually contains far fewer nodes and makes a trade-off on decentralization to achieve higher throughput [63]. Similar to the state channel, the only reports to be sent to the main chain are the opening and closing states.

Plasma. A proposed framework by J. Poon & V. Buterin [54], Plasma, intends to expand the notion of sidechains to lower the number of transactions handled by the first layer chain. Plasma's architecture consists of two fundamental components, reframing blockchain computation into a set of MapReduce functions, and an optional method to discourage block withholding attacks or selfish mining. This design is made possible by creating smart contracts on the primary blockchain that use fraud proofs. A fraud-proof mechanism is used to verify transactions when a user decides to challenge or irregularities exist. Different consensus mechanisms are also introduced to improve scalability, such as Proof-of-Authority, granting higher throughput with negligible fees. Theoretically, many Plasma chains can be constructed, each with a distinct function and application, thus further lightening the process.

RootStock. Started in 2015, S. Lerner [37] introduced RootStock. It is the first open-source Bitcoin sidechain that comes up with a smart contract. It is also suitable with Ethereum, giving users and businesses running on Ethereum a new platform to launch their solutions, utilizing Bitcoin mining architecture as the security layer. The goal of this sidechain is to enable higher scalability and reduced transaction costs. Some drawbacks of the system include the necessary deposit tokens for transactions and the PoW-based consensus results in high energy consumption.

Loom. Since early 2018, Loom Network [49], created by M. Campbell *et al.*, has been up in production. It is a second-layer sidechain focused on a set of products that covers developer tools, education material, games, and scalable side chains. Loom's major service is the creation of these side chains, in which developers may establish their own highly targeted blockchain networks based on the goals of each application. Instead of depending entirely on Ethereum's basic characteristics, these customizable side chains enable developers to alter decentralization, security, and scalability requirements. It also supports integrations with Bitcoin, Ethereum, Binance Chain, and Tron, allowing developers to integrate assets from all major chains.

Liquid. In 2020, Nick *et al.* [52] established Liquid Network, which allows Bitcoin transactions into and out of the system using a cryptographic peg. The underlying concept of the platform is the Strong Federation, which helps to achieve faster transactions and private settlements by tweaking different security models. It claims to remain safe as long as more than two-thirds of its nodes are honest.

2.2.3 ***Rollups.*** The term was first introduced by an alias named Barry WhiteHat [8] in 2018. It was labeled as *"scale Ethereum with snarks"* in the repository. Rollups refers to the concept in which a transaction is executed *off-chain* or outside the main chain, but each of the transaction data is reported back to the main chain at all times. Consequently, the main chain will not be overcrowded by transactions while still keeping transactions history. Rollups can be divided into two different categories based on the transaction security mechanisms. ZK Rollups used *validity proof* and *zero-knowledge proofs*, while Optimistic Rollups utilized *fraud proofs*, same validity method that is used in Plasma that has been discussed in section *2.2.2* about Sidechains. The paragraphs below further discuss these two categories and their applications.

ZK Rollups. Derived from the Rollups idea, this type of Rollups [65] used *validity proof* and *zero-knowledge proof* [27] as verification methods. This validity proof can come in the form of STARKS [9] and SNARKS [10]. This system allows low fees to do a verification but is expensive to compute. This makes ZK Rollup a suitable system to manage transactions, but not complicated contract execution as it can be very costly. Several notable ZK Rollups projects include zkSync [26], Loopring [71], ZkSwap [35], Hermez [48], and AZTEC [73].

Optimistic Rollups. This type of rollups [22] can be seen as a combination of ZK Rollups [65] and Plasma [54]. It holds the same concept as ZK Rollups, but with the security mechanism of Plasma, which is *fraud proof*. A contract is maintained to keep track of the states, and anyone can publish evidence of falsity using the fraud-proof concept if they find a false post-state root. Some well-known Optimistic Rollups projects includes Arbitrum One [36] and Optimism [53].

2.3 Scalable Distributed Ledgers

Blockchain is a subset of a distributed ledgers technology scope. This last approach is another option still in the cluster of distributed ledgers, which strive to change the blockchain data structure itself. When it comes to storing information, instead of using chains of hashed blocks with linear structure as used in the blockchain approach, it attempts to make it non-linear. Directed Acyclic Graphs, or DAGs, is the most well-known example of a scalable distributed ledger as a solution to blockchain scalability. The key difference can be observed in Figure 5 of Appendix A. Each square represents the block or transaction's data, while the arrow indicates direct reference from the newer created block to the latest. Blockchain is built in a linear way, where the newer created block has a single direct reference to the previous block. On the other hand, In DAG, several blocks can be formed at the same time. For instance, blocks B and C are constructed simultaneously, both have direct reference to block A. Therefore, theoretically, a transaction can be processed faster, which raises the amount of throughput and increases the scalability of the system [11]. Numerous notable frameworks have been built using this concept, such as IOTA [56], SPECTRE [66], DEXON [17], Hedera [7], and Meshcash [12]. Further discussions

on these frameworks are covered below.

2.3.1 ***Directed Acyclic Graph***. In the graph theory field, a directed acyclic graph or DAG refers to a directed graph that has no directed cycles therefore a closed loop will not exist [18]. In a blockchain world, it is used as a data structure to keep transaction records. DAG architecture aims to address blockchain's major drawbacks, including transaction fees, throughput, and scalability. It eliminates the block concept, which means no mining is required to produce a new block, thus no transaction fees are needed. It offers a huge efficiency in storing data and processing transactions, thanks to the DAG structure itself. The paragraphs below discussed some DAG-based protocols.

IOTA. Starting in 2016, S. Popov introduced a *cryptocurrency* named IOTA [56], to be used within the Internet-of-Things domain. The highlight of IOTA is the underlying structure for storing transaction records, the *tangle*. It is a directed acyclic graph (DAG) that binds transactions with each other directly. It helps to scale the system as approval of transactions relies on other transactions, in the form of DAG instead of fitting it in a block with a determined size. In consequence, the block size limitation problem explained in section 2.1.1 can be taken out. As the number of transactions increases, the faster a new transaction can be processed.

SPECTRE. In 2016, Sompolinsky *et al.* presented SPECTRE [66], a payment protocol that utilizes DAG, claimed to remain secure even under high throughput and fast confirmation times. It improves transaction rates as block creation can be done in parallel. It is also claimed that the system is resistant to an attacker with up to 50% of the processing power.

DEXON. Chen *et al.* [17] introduced DEXON in 2018, which use the DAG structure. It uses the total ordering algorithm to execute many chains concurrently and uses the DEXON Byzantine Agreement to obtain consensus. DEXON claimed to break three limitations of the blockchain, which are transaction throughput, confirmation time, and probabilistic finality.

Hedera Hashgraph. In 2018, Baird *et al.* [7] created Hedera Hashgraph, which is a new platform that provides a *hashgraph* data structure based on DAG. Every transaction container is added to the ledger and none are removed, unlike blockchain where two blocks can be created at the nearly same time, and the network nodes should choose one and discard the other. Hedera claimed to achieve up to 100.000 transactions per second using the DAG structure. For security, it utilized a system called Asynchronous Byzantine Fault Tolerance (aBFT), which maintains the network health even when malicious nodes are present on the network.

Meshcash. In 2017, Bentov *et al.* proposed Meshcash [12] framework, a layered DAG, aiming to solve and diminish risks that Bitcoin suffers. It reduced pool mining incentives, which damage the state of decentralization. It also improves scalability by removing the "race" condition in mining. Several other advantages such as incentive-compatible verification, propagation, resistance to bribe attacks, and forking are also mentioned.

3 ANALYSIS

Approaches that have been discussed in Related Work section can be summarized per categories as presented on Table 1. Paragraph below will further analyze each approach on the strengths and weaknesses.

3.1 First Layer Solutions

3.1.1 ***Block Size***.

Improves scalability. Making a good compression so more transaction data can be stored or making the block size larger both increase transaction throughput and improve the scalability of the system in general. However, block propagation time, the typical amount of time required for the new block to be broadcasted to the majority of network nodes also grows. This endangered the nodes to several attacks, such as the 51% attack.

Less decentralized. Taking the approach to increase the block size, like SegWit, means that more data should be retained by single nodes as the network expands. A bigger data block leads to a bigger blockchain, forcing the node to have the resourceful computing power and data storage to process. This limit the accessibility to nodes that have a standard processing device but the robust one, suggesting a more centralized environment of powerful machines.

Less secure. Less decentralized due to small participating nodes would decrease the security of the network. A more centralized system is exposed to a single point of failure. For instance, executing a 51% attack on 10000 nodes means the attacker needs to control at least 5100 nodes. While doing it on much smaller nodes, say 500 nodes, it only takes 255 nodes to be manipulated, thus higher vulnerability. Also, taking example specifically on Segwit, by removing the *wit* parts, which acts as a digital signature for transactions, leads to higher exploits and verification issues.

3.1.2 ***Sharding***.

Improves scalability. Since each node is grouped into a smaller environment and only needs to handle transactions within its *shard*, more transactions can be processed thus increasing transaction throughput. This is directly proportional to the state of the scalability in the chain. Better throughput is achieved at the expense of decreased security when shard sizes are smaller. *Omniledger* reported to reach up to 3500 transactions per second with 600 nodes per committee. Confirmation time for transaction also decreased significantly.

Decreased security. Several attack patterns and exploitation have been discovered throughout the sharding implementation. An attack such as a single-shard takeover and single-shard flooding [51] raises the issue of the sharding environment. This is a trade-off between scalability and security as mentioned in the paragraph above. The concept of *committee resiliency* and *total resiliency* can be used to describe the security state. *Committee resiliency* refers to the number of harmful nodes that a committee can control without jeopardizing security. While *total resiliency* suggests the number of malicious nodes that the entire network can withstand while remaining safe. For instance, *OmniLedger* has 33% of *committee resiliency*, denoting that it can only handle 330 malicious nodes out

Table 1. Taxonomy of Blockchain *trilemma* approaches

Name of Approach	Categories	Applications
First Layer Solutions	Block Size	SegWit [38], Bitcoin Cash [1], Merkle Trees [42], MAST [60], Txilm [20]
	Sharding	OmniLedger [34], Monoxide [72], Zilliqa [68], Dynamic Sharding [33], SecuSca [19]
Second Layer Solutions	State / Payment Channels	Lightning Network [55], Raiden Network [50], Trinity [70], Celer [62], Perun [21]
	Sidechains	Plasma [54], RootStock [37], Loom [49], Liquid [52]
	ZK Rollups	zkSync [26], Loopring [71], ZkSwap [35], Polygon Hermez [48], AZTEC [73]
	Optimistic Rollups	Arbitrum One [36], Optimism [53], Boba Network [45], Fuel Network [47], Cartesi [67]
Scalable Distributed Ledgers	Directed Acyclic Graph	IOTA [56], SPECTRE [66], DEXON [17], Hedera Hashgraph [7], Meshcash [12]

of 1000 nodes in a committee before the network went down. Some other projects have a higher number, such as *Monoxide*, which has 50% for both committee and total resiliency. *Sybil attack* is another concern regarding the security state in the sharding approach. It is a manipulation created by one party to take control of the network decision. Rajabi *et al.* [57] did an analysis of the attack on *Elastico* and conclude that it is vulnerable to such an attack from an adversary with a network hash power as small as 25%. The attacker has up to a 20% chance of breaking the consensus process in at least one shard.

Less decentralized. To deal with the vulnerability issues explained above, most common ideas is to add a watch node for each available set. This node will track activities that happened within that specific *shard* and handle malicious nodes if observed. Practically, this watch node holds higher role and substantial power to affect the network in general, thus lower the decentralization state.

3.2 Second Layer Solutions

Extra structure. Second-layer solutions need an additional structure on top of the main chain. This can be seen as an advantage and disadvantage at the same time. For a huge network with massive transactions on going, Bitcoin and Ethereum for instance, of course, changes are less expected as they may break the current system, creating economical loss and damaging the community. This can be solved by making and testing changes in the second layer. Since changes are made in the separate layer, the main network can still fully operate even if the second layer fails. The downside is then an extra layer needs to be generated and can be resourceful.

Higher transaction speed. In general, the second layer achieves higher transaction speed as all transactions are being executed *off-chain*, reducing congestion in the main chain. This contributes to the scalability of the network, as it can process more transactions per second. Lightning Network [55] claimed that it can handle an estimated average of 11.000 TPS and billions of transactions each day.

Low transaction fees. As the transaction speed boost, most of the frameworks have a very low transaction fee. This makes the state channel a suitable environment to carry out micro payments. Of course, some of the approaches can still have high fees, depending on the consensus and underlying design. For example, validity proofs on ZK Rollups can be consuming in cost and complexity.

Lower decentralization. In state channel, the decentralization state might fail in the payment channel network. Nodes tend to connect to a *hub* that already has a connection with lots of other nodes. This is a natural response as connecting to a *hub* with higher channel counts allows a node to transact with wider parties. While in sidechain, the second layer is composed of a separate blockchain structure and is usually more centralized to handle transactions faster. Rollups also suffer from a less decentralized environment since it is organized mainly by a finite set of smart contracts, and block production heavily depends on it.

Higher privacy. Thanks to the *off-chain* transaction execution, the main chain does not contain the whole transactions that is done by different parties since it only records the initial balance and the closing balance. Only involved nodes can view and entitled to record this, resulting in higher privacy.

Security issues. Some vulnerability in second-layer solutions has been known and studied by researchers and developers. It includes Wormhole attack [41], Flood and loot attack [30], and Congestion attack [43]. A paper by Gangwal *et al.* [25] has summarized several major attacks on second-layer solutions. Tikhomirov *et al.* [69] also did quantitative research on security, anonymity, and scalability for Lightning Network and state channels in general. According to them, payment channels are vulnerable to security breaches, such as the aforementioned Wormhole attack, anonymity issues [40], and scalability limitations. E. Rohrer and F. Tschorsch [59] researched about attacks on privacy in state channel.

3.3 Scalable Distributed Ledgers

Improves scalability. Since there is no miner in the DAG structure, users are expected to manage their own transactions to use the network. And before joining the network, they have to verify two different previous transactions. This contributes to the scalability of the chain since a new transaction means an increase in computing power. It also eliminates the single-chain issue in blockchain, since multiple blocks can be issues at the same time to record transactions.

Less decentralized. Most DAG projects tend to have a more centralized system in their network. IOTA, for instance. According to IOTA whitepaper [56], the foundation is in control of the so-called *coordinator node*, which is needed to control and ensure the network safety as it is not yet able to maintain itself. This harm the state of decentralization in the sense that the developers play a huge role and have a single point of power to change the system if they want to.

Prone to attack. Transaction volume plays a crucial role in the DAG system. Low transaction volume directly affects the verification process as resources become limited, thus, vulnerable to attacks. It has been known in practice that malicious nodes can breach the network by gaining only one-third of the total hash power. The double-spending problem becomes another issue as the node in DAG does not possess the network global history. It is a problem where a single token can be used more than once for different transactions. Unlike blockchain, where each node has a full copy of the entire ledger, only several nodes have it in DAG (full node). And when database size grows, pruning is done to lighten the process, making the node unable to guarantee no double-spending has occurred. These vulnerabilities mentioned are the reason why the DAG system needs a central node to keep its network from malicious nodes.

Lower transaction fees. Since no miner and mining process is involved in the system, transaction fees are not required. This makes DAG well-suited for micro transactions.

4 DISCUSSION

From the Analysis section, a comparison table can be created and available in Appendix B to summarize the advantages and disadvantages of each approach. There is a clear trend of trade-off that can be observed. The state of decentralization is usually directly proportional to the security level, while escalating the scalability is usually the main focus of these approaches. The section below discussed further the trade-off that occurs depending on the approaches.

4.1 First Layer Solutions

A trade-off between security and decentralization can be observed. In the sharding approach, for instance, grouping nodes into a smaller set and making them only responsible for the thing within its set leads to better throughput but also makes it more vulnerable to attack [29]. Aiyar *et al.* [2] share a probability distribution model to visualize the trade-off between scalability and security in the sharding approach.

4.2 Second Layer Solutions

The idea of adding a supplementary layer to process transactions improves scalability as it eliminates congestion on the main chain and increases transaction throughput. This comes with the price of lower decentralization to achieve higher performance on the second layer. Regarding security, although practically it increases privacy level, some vulnerabilities pattern have also known and discuss in the Analysis section.

4.3 Scalable Distributed Ledgers

Since the data structure is changed to DAG, the number of transactions that can be processed increase and improve scalability. However, new security challenges exist, that require a trade-off with a decentralization state to ensure the network's safety. As mentioned in the analysis, most DAG projects tend to have these coordinators or *witness* nodes, which function as network guards from malicious parties. This lower the decentralization state to achieve higher security.

4.4 Future direction of blockchain *trilemma* problem

With all the trade-offs mentioned, so far, no available universal solutions so to speak. The trade-off is still observed in most of the works done, although some approaches can be considered more popular and outshine the other with diverse motives such as lower vulnerability and maintained decentralization. Researchers and developers are still actively creating and polishing solutions. At the moment of writing, Ethereum is planning to release "The Merge" in Q3/Q4 2022 [46], which combines the idea of sharding and Proof-of-Stake (PoS) consensus to escalate scalability, environmental sustainability, and security. Many other solutions are expected to surface in upcoming years and the combination of different layers becoming more common.

5 CONCLUSION

Blockchain *trilemma* problem is the central point that hinders the mass adoption of blockchain in various industries. This paper compares the state of decentralization, scalability, and security among popular and conventional approaches that have been made publicly available by researchers and developers. A systematic literature review is done to have an understanding of these approaches. Based on the changes made, all the approaches can be categorized into first-layer solutions, second-layer solutions, and distributed ledger types. An analysis of these different approaches is presented, covering the strengths and weaknesses from the *trilemma* problem view. To summarize, up until this paper is written, there is no one-size-fits-all solution that excels on every side of the *trilemma*. A trade-off seems like a small price to be paid to surpass the other state. Which solution to use should be adjusted according to the end goal and business needs.

ACKNOWLEDGMENTS

I would like to express my gratitude to my supervisor, Dr. Mohammed Elhajj, for his continuous support and constructive critiques of this research work. I would also like to thank the track chairs, Wallace Corbo Ugulino and Leon de Vries, along with fellow

students, for giving track sessions and peer feedback. Finally, the credit goes to both of my parents for their endless encouragement throughout my bachelor's studies.

REFERENCES

[1] 2017. Bitcoin Cash: Peer-to-Peer Electronic Cash. https://bitcoincash.org/

[2] Kamalani Aiyar, Malka N. Halgamuge, and Azeem Mohammad. 2021. Probability distribution model to analyze the trade-off between scalability and security of sharding-based blockchain networks. In *2021 IEEE 18th Annual Consumer Communications and Networking Conference, CCNC 2021*. Institute of Electrical and Electronics Engineers Inc., 1–6. https://doi.org/10.1109/CCNC49032.2021.9369563

[3] Jameela Al-Jaroodi and Nader Mohamed. 2019. Blockchain in Industries: A Survey. *IEEE Access* 7 (2019), 36500–36515. https://doi.org/10.1109/ACCESS.2019.2903554

[4] Faten Adel Alabdulwahhab. 2018. Web 3.0: The Decentralized Web Blockchain networks and Protocol Innovation. In *2018 1st International Conference on Computer Applications Information Security (ICCAIS)*. 1–4. https://doi.org/10.1109/CAIS.2018.8441990

[5] Hany F. Atlam, Ahmed Alenezi, Madini O. Alassafi, and Gary B. Wills. 2018. Blockchain with Internet of Things: Benefits, challenges, and future directions. *International Journal of Intelligent Systems and Applications* 10, 6 (6 2018), 40–48. https://doi.org/10.5815/ijisa.2018.06.05

[6] Adam Back, Matt Corallo, Luke Dashjr, Mark Friedenbach, Gregory Maxwell, Andrew Miller, Andrew Poelstra, Jorge Timón, and Pieter Wuille. 2014. *Enabling Blockchain Innovations with Pegged Sidechains*. Technical Report.

[7] Leemon Baird, Mance Harmon, and Paul Madsen. 2018. *Hedera: A Public Hashgraph Network & Governing Council*. Technical Report.

[8] barryWhiteHat. 2018. Barrywhitehat/roll_up: Scale Ethereum with snarks. https://github.com/barryWhiteHat/roll_up

[9] Eli Ben-Sasson, Iddo Bentov, Yinon Horesh, and Michael Riabzev. 2018. *Scalable, transparent, and post-quantum secure computational integrity*. Technical Report.

[10] Eli Ben-Sasson Technion Alessandro Chiesa, Eran Tromer, and Madars Virza MIT. 2019. *Succinct Non-Interactive Zero Knowledge for a von Neumann Architecture*. Technical Report.

[11] Federico Matteo Benčić and Ivana Podnar Žarko. 2018. Distributed Ledger Technology: Blockchain Compared to Directed Acyclic Graph. (4 2018). http://arxiv.org/abs/1804.10013

[12] Iddo Bentov, Pavel Hubácek, Tal Moran, and Asaf Nadler. 2017. Tortoise and Hares Consensus: the Meshcash Framework for Incentive-Compatible, Scalable Cryptocurrencies. In *IACR Cryptol. ePrint Arch.*

[13] Umesh Bodkhe, Sudeep Tanwar, Karan Parekh, Pimal Khanpara, Sudhanshu Tyagi, Neeraj Kumar, and Mamoun Alazab. 2020. Blockchain for Industry 4.0: A comprehensive review. *IEEE Access* 8 (2020), 79764–79800. https://doi.org/10.1109/ACCESS.2020.2988579

[14] Vitalik Buterin. 2014. *Ethereum: A Next-Generation Smart Contract and Decentralized Application Platform*. Technical Report.

[15] Miguel Castro and Barbara Liskov. 1999. *Practical Byzantine Fault Tolerance*. Technical Report.

[16] Anamika Chauhan, Om Prakash Malviya, Madhav Verma, and Tejinder Singh Mor. 2018. Blockchain and Scalability. In *Proceedings - 2018 IEEE 18th International Conference on Software Quality, Reliability, and Security Companion, QRS-C 2018*. Institute of Electrical and Electronics Engineers Inc., 122–128. https://doi.org/10.1109/QRS-C.2018.00034

[17] Tai-Yuan Chen, Wei-Ning Huang, Po-Chun Kuo, Hao Chung, and Tzu-Wei Chao. 2018. *DEXON : A Highly Scalable, Decentralized DAG-Based Consensus Algorithm*. Technical Report.

[18] Acyclic Graph DAG. 2013. Directed acyclic graph. (2013).

[19] Gianmaria Del Monte, Diego Pennino, and Maurizio Pizzonia. 2020. Scaling Blockchains Without Giving up Decentralization and Security. (5 2020), 71–76. http://arxiv.org/abs/2005.06665

[20] Donghui Ding, Xin Jiang, Jiaping Wang, Hao Wang, Xiaobing Zhang, and Yi Sun. 2019. *Txilm: Lossy Block Compression with Salted Short Hashing*. Technical Report.

[21] Stefan Dziembowski, Lisa Eckey, Sebastian Faust, and Daniel Malinowski. 2017. *Perun: Virtual Payment Hubs over Cryptocurrencies*. Technical Report. https://eprint.iacr.org/2017/635

[22] Karl Floersch. 2019. Ethereum smart contracts in L2: Optimistic rollup. https://medium.com/plasma-group/ethereum-smart-contracts-in-l2-optimistic-rollup-2c1cef2ec537

[23] Jake Frankenfield. 2022. Hard fork (blockchain) definition. https://www.investopedia.com/terms/h/hard-fork.asp

[24] Jake Frankenfield. 2022. Soft fork definition. https://www.investopedia.com/terms/s/soft-fork.asp

[25] Ankit Gangwal, Haripriya Ravali Gangavalli, and Apoorva Thirupathi. 2022. *A Survey of Layer-Two Blockchain Protocols*. Technical Report.

[26] Alex Gluchowski. 2021. Introducing zkSync: the missing link to mass adoption of Ethereum.

[27] S Goldwasser, S Micali, and C Rackoff. 1985. The Knowledge Complexity of Interactive Proof-Systems. In *Proceedings of the Seventeenth Annual ACM Symposium on Theory of Computing (STOC '85)*. Association for Computing Machinery, New York, NY, USA, 291–304. https://doi.org/10.1145/22145.22178

[28] Debasis Gountia. 2019. Towards Scalability Trade-off and Security Issues in State-of-the-art Blockchain. *ICST Transactions on Security and Safety* 5, 18 (4 2019), 157416. https://doi.org/10.4108/eai.8-4-2019.157416

[29] Abdelatif Hafid, Abdelhakim Senhaji Hafid, and Mustapha Samih. 2020. Scaling Blockchains: A Comprehensive Survey. *IEEE Access* 8 (2020), 125244–125262. https://doi.org/10.1109/ACCESS.2020.3007251

[30] Jona Harris and Aviv Zohar. 2020. Flood & Loot: A Systemic Attack On The Lightning Network. (6 2020). http://arxiv.org/abs/2006.08513

[31] Jordi Herrera-Joancomarti, Guillermo Navarro-Arribas, Alejandro Ranchal Pedrosa, Perez-Sola Cristina, Joaquin Garcia-Alfaro, Jordi Herrera-Joancomartí, Telecom SudParis, and Universitat Rovira Virgili Cybercat Joaquin Garcia-Alfaro. 2019. *On the difficulty of hiding the balance of lightning network channels*. Technical Report. https://hal.archives-ouvertes.fr/hal-02086536

[32] Michał R. Hoffman, Luis-Daniel Ibáñez, and Elena Simperl. 2020. Toward a Formal Scholarly Understanding of Blockchain-Mediated Decentralization: A Systematic Review and a Framework. *Frontiers in Blockchain* 3 (8 2020). https://doi.org/10.3389/fbloc.2020.00035

[33] Kahina Khacef, Salima Benbernou, Mourad Ouziri, and Muhammad Younas. 2021. Trade-off Between Security and Scalability in Blockchain Design: A Dynamic Sharding Approach. (2021), 77–90. https://doi.org/10.1007/978-3-030-84337-3{_}7

[34] Eleftherios Kokoris-Kogias, Philipp Jovanovic, Linus Gasser, Nicolas Gailly, Ewa Syta, and Bryan Ford. 2018. OmniLedger: A Secure, Scale-Out, Decentralized Ledger via Sharding. In *Proceedings - IEEE Symposium on Security and Privacy*, Vol. 2018-May. Institute of Electrical and Electronics Engineers Inc., 583–598. https://doi.org/10.1109/SP.2018.000-5

[35] L2 Lab. 2020. *ZKSwap: a Layer-2 Token Swap Protocol based on ZK-Rollup*. Technical Report. https://github.com/l2labs/zkswap-whitepaper/blob/master/zkswap_en.pdf

[36] Offchain Labs. 2021. Introducing Arbitrum One: Our mainnet beta. https://offchain.medium.com/introducing-arbitrum-one-our-mainet-beta-ed0e9b63b435

[37] Sergio Demian Lerner. 2019. *RSK: Bitcoin powered Smart Contracts*. Technical Report.

[38] Eric Lombrozo, Johnson Lau, and Pieter Wuille. 2015. Segregated witness (consensus layer). *Bitcoin Core Develop. Team, Tech. Rep. BIP* 141 (2015).

[39] Loi Luu, Viswesh Narayanan, Chaodong Zheng, Kunal Baweja, Seth Gilbert, and Prateek Saxena. 2016. A secure sharding protocol for open blockchains. In *Proceedings of the ACM Conference on Computer and Communications Security*, Vol. 24-28-October-2016. Association for Computing Machinery, 17–30. https://doi.org/10.1145/2976749.2978389

[40] Giulio Malavolta, Pedro Moreno-Sanchez, Aniket Kate, Matteo Maffei, T U Wien, and Srivatsan Ravi. 2017. *Concurrency and Privacy with Payment-Channel Networks*. Technical Report. https://eprint.iacr.org/2017/820

[41] Giulio Malavolta, Pedro Moreno-Sanchez, Clara Schneidewind, Aniket Kate, and Matteo Maffei. 2019. Anonymous Multi-Hop Locks for Blockchain Scalability and Interoperability. Internet Society. https://doi.org/10.14722/ndss.2019.23330

[42] Ralph C Merkle. 1980. Protocols for Public Key Cryptosystems. In *1980 IEEE Symposium on Security and Privacy*. 122. https://doi.org/10.1109/SP.1980.10006

[43] Ayelet Mizrahi and Aviv Zohar. 2021. Congestion Attacks in Payment Channel Networks. In *Financial Cryptography and Data Security: 25th International Conference, FC 2021, Virtual Event, March 1–5, 2021, Revised Selected Papers, Part II*. Springer-Verlag, Berlin, Heidelberg, 170–188. https://doi.org/10.1007/978-3-662-64331-0{_}9

[44] Satoshi Nakamoto. 2008. *Bitcoin: A Peer-to-Peer Electronic Cash System*. Technical Report. www.bitcoin.org

[45] Boba Network. 2022. Layer 2 ethereum scaling and augmenting solution. https://boba.network/

[46] Ethereum Network. 2022. The Merge. https://ethereum.org/en/upgrades/merge/

[47] Fuel Network. 2020. Fuel network. https://fuel.network/

[48] Hermez Network. 2020. Hermez whitepaper.

[49] Loom Network. 2017. Loom network. https://loomx.io/

[50] Raiden Network. 2019. What is the Raiden Network? https://raiden.network/101.html

[51] Truc Nguyen and My T. Thai. 2020. Denial-of-Service Vulnerability of Hash-based Transaction Sharding: Attack and Countermeasure. (7 2020). https://doi.org/10.1109/TC.2022.3174560

[52] Jonas Nick, Andrew Poelstra, and Gregory Sanders. 2020. *Liquid: A Bitcoin Sidechain*. Technical Report.

[53] Optimism. 2022. Optimism. https://medium.com/ethereum-optimism/optimism-cd9bea61a3ee

[54] Joseph Poon and Vitalik Buterin. 2017. *Plasma: Scalable Autonomous Smart Contracts.* Technical Report. https://plasma.io/
[55] Joseph Poon and Thaddeus Dryja. 2016. *The Bitcoin Lightning Network: Scalable Off-Chain Instant Payments.* Technical Report.
[56] Serguei Popov. 2018. *The Tangle.* Technical Report.
[57] Tayebeh Rajab, Mohammad Hossein Manshaei, Mohammad Dakhilalian, Murtuza Jadliwala, and Mohammad Ashiqur Rahman. 2020. On the Feasibility of Sybil Attacks in Shard-Based Permissionless Blockchains. (2 2020). http://arxiv.org/abs/2002.06531
[58] Sabrina Rochemont. 2019. *Understanding Central Bank Digital Currencies (CBDC).* Technical Report. https://www.researchgate.net/publication/338792619
[59] Elias Rohrer and Florian Tschorsch. 2020. Counting Down Thunder: Timing Attacks on Privacy in Payment Channel Networks. (6 2020). http://arxiv.org/abs/2006.12143
[60] Jeremy Rubin and M Venkatesh Naik. 2014. Merkelized Abstract Syntax Trees.
[61] Mayra Samaniego and Ralph Deters. 2017. Blockchain as a Service for IoT. In *Proceedings - 2016 IEEE International Conference on Internet of Things; IEEE Green Computing and Communications; IEEE Cyber, Physical, and Social Computing; IEEE Smart Data, iThings-GreenCom-CPSCom-Smart Data 2016.* Institute of Electrical and Electronics Engineers Inc., 433–436. https://doi.org/10.1109/iThings-GreenCom-CPSCom-SmartData.2016.102
[62] ScaleSphere Foundation Ltd. 2018. *Celer Network: Bring Internet Scale to Every Blockchain.* Technical Report.
[63] Cosimo Sguanci, Roberto Spatafora, and Andrea Mario Vergani. 2021. *Layer 2 Blockchain Scaling: a Survey.* Technical Report.
[64] Saurabh Singh, A. S.M. Sanwar Hosen, and Byungun Yoon. 2021. Blockchain Security Attacks, Challenges, and Solutions for the Future Distributed IoT Network. *IEEE Access* 9 (2021), 13938–13959. https://doi.org/10.1109/ACCESS.2021.3051602
[65] Corwin Smith, Sam Richards, Hursit Tarcan, Sora Nature, and Joshua. 2022. Zero-knowledge rollups. https://ethereum.org/en/developers/docs/scaling/zk-rollups
[66] Yonatan Sompolinsky, Yoad Lewenberg, and Aviv Zohar. 2016. SPECTRE: A Fast and Scalable Cryptocurrency Protocol. *IACR Cryptol. ePrint Arch.* 2016 (2016), 1159.
[67] Augusto Teixeira and Diego Nehab. 2018. *The Core of Cartesi.* Technical Report.
[68] The ZILLIQA Team. 2017. *The ZILLIQA Technical Whitepaper.* Technical Report. www.zilliqa.com
[69] Sergei Tikhomirov, Pedro Moreno-Sanchez, and Matteo Maffei. 2020. A quantitative analysis of security, anonymity and scalability for the lightning network. In *2020 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW).* 387–396.
[70] Trinity Tech Team. 2020. *Trinity White Paper: Universal Off-chain Scaling Solution.* Technical Report.
[71] Daniel Wang, Jay Zhou, Alex Wang, and Matthew Finestone. 2018. *Loopring: A Decentralized Token Exchange Protocol.* Technical Report. https://loopring.org
[72] Jiaping Wang and Hao Wang. 2019. *Monoxide: Scale Out Blockchain with Asynchronous Consensus Zones.* Technical Report. 95–112 pages. https://monoxide.io
[73] Zachary J Williamson. 2018. *The AZTEC Protocol.* Technical Report. https://github.com/AztecProtocol/AZTEC/blob/master/AZTEC.pdf

A STRUCTURAL DESIGN

A.1 Segwit Structure

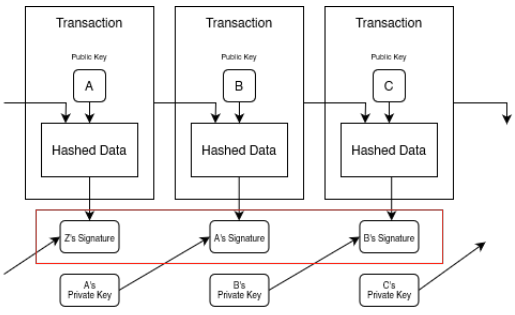

Fig. 3. SegWit structure; The signatures are stored in extended block or *witness*, instead of the main transaction block.

A.2 State Channel and Sidechains Structure

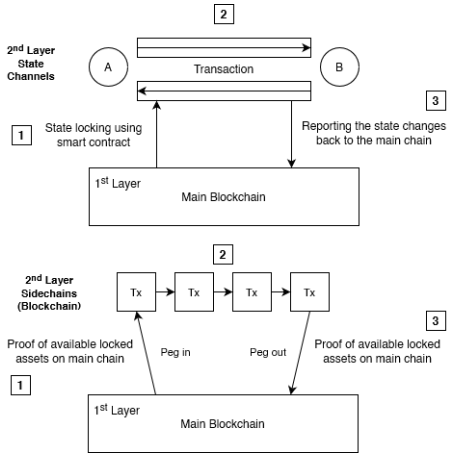

Fig. 4. State Channels and Sidechains difference.

A.3 Blockchain and Directed Acyclic Graph Structure

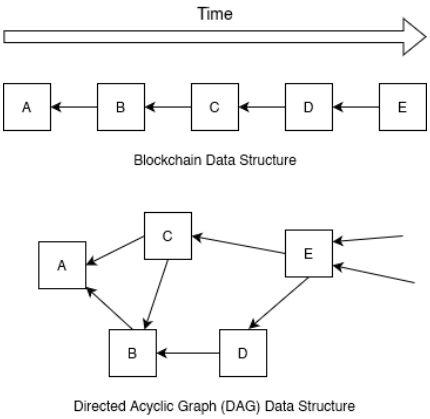

Fig. 5. Data structure of blockchain compared to DAG.

B COMPARISON TABLE

The comparison table created from the Analysis section available on the next page.

Table 2. Comparison Table of each approach in view of the *trilemma*

Name of Approach	Categories	Decentralization	Scalability	Security
First Layer Solutions	Block Size	• Limit nodes with smaller power to participate due to high operating cost	• Good throughput • fast transaction time • Lower transaction fee • Slower block propagation time	• Lower resources needed to launch an attack • Prone to 51% attack • Project like SegWit may lead to verification issues since it removes the *wit* data, which acts as a digital signature.
	Sharding	• The need of watch node to deal with malicious activities lower the decentralization state.	• Smaller committees size leads to better throughput • *OmniLedger* could achieve up to 3500 transactions per second with 600 nodes per committee • Low latency	• Prone to 51% attack/double spending attack • Prone to Sybil attack • Prone to single-shard takeover attack (1% attack)
Second Layer Solutions	Payment Channels	• Payment Channel Network has high tendency to be centralized • Reliance in monitoring service while node going offline	• Reduce transaction volume on the main chain • Increase transaction throughput, excellent scalability • Fast to real-time confirmation time	• Higher privacy level compared to the main chain • The use of Hash Lock Time Contract to avoid uncooperative behaviour among nodes • The longer the routing, the higher the privacy and security
	Sidechains	• Mining power tends to be centralized • Sidechain usually less decentralized to achieve higher scalability	• Faster transaction time • Low transaction costs	• Ensure *atomicity* of transfers • Safe of double-spending problem
	Rollups	• Smart contracts used in main chain usually handled by centralized parties • Development process still centralized for higher security state	• Broadcast a compressed small amount of data • High throughput • Faster transaction time • Verification can be quite costly (ZK Rollups)	• High privacy due to the fraud and ZK proofs mechanism
Scalable Distributed Ledgers	Directed Acyclic Graph	• Data is stored in decentralized manner • The need of centralized node, i.e. coordinator node in IOTA	• Improve scalability due to change in data structure	• Vulnerable to double spending problem