DAG-based Blockchain Systems

Lukáš Radovanský
Technical University of Munich
Munich, Bavaria, Germany
go73hij@mytum.de

Linus Kratz
Technical University of Munich
Munich, Bavaria, Germany
linus.kratz@tum.de

ABSTRACT

The increasing popularity of blockchain technologies is evident with each passing year. As the number of nodes participating in blockchain networks grows, addressing the scalability of these systems becomes a critical challenge. This paper examines Directed Acyclic Graph (DAG) based blockchain systems as a potential solution to the scalability and performance limitations inherent in traditional linear blockchains. Unlike linear blockchains, which store transactions in a single chain, DAG structures permit multiple branches, thereby enhancing throughput and reducing confirmation times. This paper provides an extensive overview of the DAG-based blockchain systems. We introduce and thoroughly describe the DAG-based blockchain IOTA and proceed to compare and contrast it with other DAG-based blockchains, including Hedera Hashgraph, Nano, Avalanche, GHOST, Spectre and Byteball.

1 INTRODUCTION

In recent years, blockchain technologies have experienced a significant surge in popularity, both in the academic sphere [18] and the industry [4]. This growing interest is driven by the potential of blockchain to revolutionize various industries through its decentralized and transparent nature. Despite the widespread enthusiasm, traditional blockchain systems are encountering several significant challenges, most notably scalability. With the increasing adoption of blockchain technology in recent years, this issue has become more pronounced. According to the well-known blockchain trilemma [31], it is impossible to simultaneously perfectly achieve decentralization, scalability, and security in a blockchain system; typically, two of these properties are prioritized at the expense of the third. The primary scalability problem in linear blockchains arises from a performance bottleneck caused by all participants in the network competing for a single valid position in the chain. In the Bitcoin network, if two valid blocks are mined simultaneously, one of these blocks will eventually be orphaned, and the transactions within that block will not be included in the blockchain [22]. Consequently, not only is the computational work rendered futile, but the miner also does not receive any incentives. One potential solution to enhance network throughput is to decrease the block period time, which can increase the network's throughput; however, it simultaneously reduces security because the network has less time to reach consensus, making it more vulnerable to certain types of attacks. Various solutions, including layer-2 protocols [16], sidechain techniques [1], and sharding [33], have been proposed to address this bottleneck, but they are all still based on the linear backbone protocol. Maintaining the history of transactions in a single linear chain inevitably leads to congestion due to the concurrent nature of the system.

Another challenge associated with linear blockchains, for which certain DAG-based blockchains provide a potential solution, is the centralization of miners/validators within pools. A recent topic of discussion in the field of blockchain technology is the centralization of the system caused by these pools [12]. As of July 1st 2024, the top three mining pools controlled over 65% of the total Bitcoin hashrate [17]. Currently, obtaining mining incentives necessitates membership in a mining pool. Consequently, these three main mining companies not only create the majority of blocks in the chain but also hold the majority voting power for protocol changes. Furthermore, the heterogeneous nature of the system, wherein the goals of miners often conflict with those of token owners, is suboptimal.

The DAG-based blockchains discussed in this paper propose solutions to the aforementioned issues in blockchain technology, specifically addressing slow confirmation times, low throughput, poor scalability, high transaction fees (which inhibit the feasibility of microtransactions), and centralization in mining pools. In this paper, we will compare and contrast these blockchains and provide a concise overview of DAG-based blockchains for the reader.

2 DAG-BASED BLOCKCHAIN MODEL

A directed acyclic graph (DAG) is defined as a pair $G = (V, E)$, where V is a set of vertices and E is a set of directed edges, with each edge represented as an ordered pair (u, v) such that $u, v \in V$. The acyclic nature of a DAG ensures that there are no directed cycles, meaning there is no path

$$v_1 \rightarrow v_2 \rightarrow \ldots \rightarrow v_n$$

where $v_1 = v_n$. This property is critical for the structure and functionality of DAG-based blockchain models. In such models, each vertex in the set V corresponds to a unit, which may take the form of a single transaction, a block of transactions, or an event, depending on the specific blockchain implementation. The edges E represent partial order relationships between these units. Specifically, if there exists a directed edge from unit u to unit v in the graph, it can be stated that transaction u *directly* approves transaction v. Conversely, if there is a directed path from unit u to unit v, it can be stated that transaction u *indirectly* approves transaction v.

The **in-degree** of a unit v is the number of units that directly approve this unit, given by:

$$d^-(v) = |\{u \in V \mid (u, v) \in E\}|$$

The **out-degree** of a unit v is the number of units that this unit directly approves, given by:

$$d^+(v) = |\{w \in V \mid (v, w) \in E\}|$$

Units at the "end" of the graph, with an in-degree of zero, are called **tips**. The **genesis unit**, found at the "beginning" of the graph, has an out-degree of zero. For each unit, we can also define the following terms: **Height** is the length of the longest oriented path to the genesis unit. **Depth** is the length of the longest reverse-oriented path from the genesis unit to some tip.

Different DAG-based blockchains employ varying terminologies for the concepts defined above, which can cause confusion for readers. In this paper, we adhere to a consistent terminology, specifically the one presented in [25, 35], and will always relate newly introduced terms to our defined DAG-based model. It is also important to note that the discussed blockchains may differ in unit representation (e.g., block of transactions, single transaction, event). Furthermore, the network topology may vary. We will classify chains into three categories according to the graphs they form: **Divergence ($\hat{D}$)**, **Parallel ($\hat{P}$)**, and **Convergence ($\hat{C}$)** as introduced in [35]. Divergence refers to the phenomenon where units are dispersed in various unpredictable directions without any predefined order. Parallelism denotes the maintenance of units in the form of multiple parallel sequences. Convergence, on the other hand, indicates that units are arranged or tend to align in a predetermined sequence.

3 CASE STUDY: IOTA

IOTA is a cryptocurrency specifically designed for the Internet of Things (IoT) industry. Instead of utilising a global blockchain, IOTA employs a divergent ($\hat{D}$) topology structure known as the Tangle. Within IOTA, a unit represents a single transaction. At the inception of the Tangle, a genesis transaction was conducted, distributing tokens from genesis addresses to several founder addresses. All IOTA tokens were created during this initial transaction; no additional IOTA tokens will be generated in the future, and there will be no mining rewards producing new tokens [25].

The core principle of the Tangle is as follows: if a user wishes to issue a new transaction, they must validate and directly approve two previous transactions. These two nodes must not conflict, either directly or indirectly. In detail, to issue a transaction, a node performs the following steps:

(1) Selects two tip transactions to approve using a Tip Selection Algorithm.
(2) Verifies that the selected transactions do not conflict.
(3) Solves a cryptographic puzzle by finding a nonce (similar to the one used in Bitcoin, but requiring significantly less computation).

The metric used in the Tangle to express the importance of a transaction is called **weight**. It can be assumed that weight is proportional to the amount of work done in solving the cryptographic puzzle; however, for simplicity, one may assume that it is always 1. The proof of work mechanism in the IOTA functions as an anti-spam filter. It is designed to ensure that no entity can generate a substantial number of transactions with acceptable weight within a short period of time.

The metric used in the Tangle to express how deeply a transaction is nested within the network is called **cumulative weight**. Cumulative weight is defined as the weight of the transaction plus the sum of the individual weights of all transactions that directly or indirectly approve this transaction. With more cumulative weight, we can be more certain that this transaction will remain in the main (genuine) branch of the Tangle.

To mitigate various types of attacks and replace complex consensus mechanisms within the network, IOTA introduces the Tip Selection Algorithm. This algorithm offers a highly scalable solution with reasonable resistance to forks, although it compromises strict consistency. The core concept of the Tip Selection Algorithm involves utilizing the Markov Chain Monte Carlo (MCMC) algorithm to select two tips for confirmation when initiating a new transaction. This technique assures the issuer that the selected tips are part of the main, legitimate section of the Tangle, thereby ensuring that the newly added transaction will persist in the blockchain and be confirmed by subsequent transactions. The approach involves placing a number N of random walkers in the Tangle and allowing them to traverse the reversed edges of the graph towards the tips, meaning that the transition from unit u to unit v is possible only if v approves u. The transition probabilities of the walker P_{uv} are defined by:

$$P_{uv} = \exp\left(-\alpha(\mathcal{H}_u - \mathcal{H}_v)\right)\left(\sum_{w:\, w \rightsquigarrow u} \exp\left(-\alpha(\mathcal{H}_u - \mathcal{H}_w)\right)\right)^{-1},$$

where $\alpha > 0$ is a parameter to be chosen, $\mathcal{H}_u$ is the cumulative weight of unit u, and $\mathcal{H}_v$ is the cumulative weight of unit v.

From the equation above, we can infer that the random walkers are more likely to transition to the unit with a similar cumulative weight. This algorithm is performed multiple times, and the tips are selected with proportional probabilities. For example, if the simulation is run 100 times and the walker ends up at tip T 86 times, then tip T is selected with 86% confidence.

Consider the following attack scenario: an adversary constructs a subtangle that intermittently references the main tangle to achieve a higher score. Honest tips possess a score approximately equivalent to the aggregate of all individual weights in the main tangle, whereas the attacker's tips incorporate the sum of all individual weights within the subtangle. Given that network latency is negligible for a solitary attacker, they can assign greater height to the parasite tips using a powerful computer. Moreover, the attacker can augment their tip count during the assault by broadcasting numerous new transactions that endorse previous transactions in the subtangle. To mitigate this attack, we rely on the superior hashing power of the main tangle relative to the attacker. Consequently, the main tangle can generate larger increases in cumulative weight for more transactions than the attacker is capable of producing. By employing the MCMC Tip Selection Algorithm, the probability of selecting the attacker's parasite chain is significantly reduced. This is because the parasite chain will have substantially lower cumulative weights, thereby minimizing the likelihood of the random walker transitioning from the main tangle to the parasite chain.

In a splitting attack, an adversary aims to bifurcate the Tangle into two distinct branches and subsequently sustain equilibrium between them. To initiate this attack, the adversary introduces two conflicting transactions at the onset of the split. To counteract this type of attack, IOTA proposes implementing a more stringent threshold for the transition probabilities of the random walkers. This adjustment is designed to render it increasingly impractical for the adversary to maintain a perfect balance between the two conflicting branches [25].

The algorithm employed in IOTA is quantum-resistant because solving the cryptographic puzzle requires a similar amount of time

as other tasks necessary for issuing a new transaction. Consequently, quantum computing does not offer any significant advantage in context of these other tasks. This ensures that the network remains secure even as advancements in quantum computing emerge.

In the IOTA network, there are no distinct roles such as miners/validators and token owners (transaction initiators). Instead, the network employs a single type of entity that encompasses both functions (miner/validator and transaction initiator). The primary motivation for approving previously added transactions arises from the desire to add one's own transaction. To achieve this, an individual must solve a simple cryptographic puzzle within a short timeframe. This puzzle serves as a spam protection and Sybil control mechanism. This structure presents significant advantages, notably the disincentive for users to centralize into mining or staking pools, as such centralization yields no additional benefits. Additionally, the absence of transaction fees is a notable advantage, as there is no need to compensate miners or validators. Their incentive is solely derived from the ability to add their own transaction to the tangle [25].

However, IOTA, as described above, is known to be vulnerable to the large weight attack, wherein a heavily weighted conflicting transaction invalidates a newly added transaction. The adversary enhances the cumulative weight of the conflicting transaction by appending numerous meaningless transactions to their own conflicting transaction. Given that honest tips are uniformly distributed across the network, the attacker requires significantly less than 50% of the computing power [32]. The temporary solution to prevent this attack in IOTA involves using a central entity called the Coordinator, which, however, transforms the network into a centralized one. Also during the initial stages of the IOTA network, the Coordinator helps maintain overall network stability and ensures that the Tangle grows in a secure and organized manner. The IOTA Foundation asserts its intention to eliminate this central entity in the future and replace it with a decentralized solution known as Coordicide [26].

4 OVERVIEW OF POPULAR AND LATEST DAG-BASED PLATFORMS

In this section, five different blockchains and one protocol will be compared and contrasted to each other and primarily to IOTA. The blockchains discussed are Hedera Hashgraph, Nano, Avalanche, Byteball, and Spectre. Additionally, the DAG-based GHOST protocol, which has been used in the Ethereum network before the Ethereum 2.0 update, is discussed.

Hedera Hashgraph [3] is a DAG-based blockchain that launched in 2018. It relies on a set of distributed nodes operated by a governing council to validate transactions. This governing council consists of 39 members, primarily comprised of reputable companies such as Google, IBM, and Deutsche Telekom. Transactions enter the network through one or more of these trusted nodes and are subsequently propagated across the network via a gossip-about-gossip protocol. Unlike IOTA, this structure renders Hedera Hashgraph a permissioned network, although there are plans to allow anyone to run a node in the future. In addition to transaction validation, the Hedera Hashgraph network facilitates file storage mechanisms. The network topology is parallel ($\hat{P}$), as each node maintains a parallel copy of the network. A notable innovation introduced by this DAG-based blockchain is Asynchronous Byzantine Fault Tolerance (aBFT). Unlike traditional blockchains, which are not inherently Byzantine fault-tolerant and assume an increasing certainty of agreement over time, Hedera Hashgraph achieves this robustness through aBFT. Moreover, the blockchain assigns a timestamp to a transaction once the majority of the network has acknowledged it via the gossip-about-gossip protocol. This feature enables a total ordering of transactions, thereby supporting the deployment of smart contracts and higher-level solutions.

Quite similar to Hedera Hashgraph in terms of its topology is the **Nano** [19] network. In this network, each account maintains a local copy of the state of the network, thereby enabling each user to participate in the validation of transactions. However, the Nano blockchain also allows accounts to delegate their voting stake to a representative, who then validates transactions on their behalf. The blockchain employs blocks that consist of a single transaction, and these blocks can be of different types. This structure facilitates the fast and asynchronous processing of transactions within the network. For example, Person A can create a transaction a by generating a send block. Subsequently, Person B can create a second transaction b by generating another send block. Person C can then asynchronously process the incoming transactions a and b by creating receive blocks. The ordering of transactions depends on their time of arrival at Person C. Furthermore, unlike other DAG-based blockchains, the Nano network is solely limited to the cryptocurrency.

Avalanche [27] is again a more elaborate network in terms of its functionality. Besides its cryptocurrency functionality it can handle smart contracts. This is done by having separate chains for these functionalities. The C-chain (for smart contracts), the X-chain (for currency transactions) and the P-chain (for payment of miner incentives) work in parallel and interact with each other. Another specialty of this network is that it allows for subnets within its network. These subnets are highly customizable to the extent that different consensus algorithms to the standard avalanche consensus can be chosen. In terms of topology and consensus its similar to IOTA in the sense that newer transactions verify older transactions. The difference lies in the fact that IOTA makes use of the tip selection algorithm meanwhile Avalanche consensus randomly selects the child set of transactions each round. Another difference is that instead of making use of PoW it relies on PoS, which makes a transaction fee necessary to prevent DDoS attacks/sybil attacks on the network.

The GHOST [29] protocol (Greedy Heaviest-Observed Sub-Tree) represents a proof-of-work blockchain analogous to Bitcoin. However, its distinction lies in the consensus mechanism it employs for resolving the correct chain. Rather than adhering to the longest chain consensus rule, GHOST prioritizes the subtree with the highest cumulative proof-of-work difficulty. The underlying premise is that by increasing the network throughput through the reduction of the difficulty threshold, block propagation times are extended, leading to geographically dispersed miners discovering blocks more rapidly yet becoming aware of new blocks later. This scenario diminishes security as honest nodes distribute their computational power across multiple forks. Nevertheless, considering all potential

subtrees, an adversary attempting to construct the longest secret chain would still require the majority of computational resources, even in the presence of these forks, when the GHOST consensus is applied. The GHOST protocol was prominently utilized in Ethereum 1.0, where it has evolved into the LMD GHOST (Latest Message-Driven GHOST) protocol. This variant was integral to Ethereum's consensus mechanism, facilitating nodes to reach agreement on the ledger's valid state more efficiently and securely compared to traditional longest chain rules. The LMD GHOST protocol mitigates disputes regarding transaction legitimacy by employing a multi-step voting process to determine the heaviest chain, defined by its computational difficulty [7]. Unlike IOTA, the GHOST protocol features a converging topology ($\hat{C}$), according to introduced classification. Transactions are aggregated into blocks, where each block unit possesses an out-degree of 1 and an in-degree that of any positive integer.

After the merge of the Ethereum chain and the Beacon chain in September 2022, Ethereum transitioned from a Proof-of-Work sybil control mechanism to a Proof-of-Stake sybil control mechanism [15]. This shift rendered the GHOST protocol unusable for the post-merge Ethereum 2.0.

Spectre [28] (Serialization of Proof-of-work Events: Confirming Transactions via Recursive Elections) is a DAG-based blockchain with a divergent topology ($\hat{D}$). It employs a structure analogous to IOTA's Tangle, known as BlockDAG. However, within Spectre, the unit representations are blocks of transactions. Another distinction from IOTA is the out-degree of a unit, which can exceed two, thereby allowing a single unit to approve multiple other units.

The consensus protocol functions by evaluating each pair of units and soliciting opinions from all units regarding the sequence of these blocks. This process resembles an election where the majority opinion prevails. The majority's opinion establishes the consensus within the BlockDAG. The voting mechanism operates as follows: if unit u directly or indirectly approves unit v, then u votes that v precedes it (thereby maintaining the topological order). In scenarios where unit w directly or indirectly approves both units u and v, w determines the order based on the opinion of the majority of units that directly or indirectly approves (denoted as Past(w)). Conversely, if unit w does not directly or indirectly approve either block u or v, it decides the order according to the majority opinion of the units that are directly or indirectly approving w (denoted as Future(w)). The primary issue with this algorithm is the excessive number of recursive calls required to determine the ordering, rendering it infeasible. One potential improvement involves iterating over the pairs of units in the BlockDAG in the correct order, thereby eliminating the need for redundant calculations. A second enhancement focuses on identifying the appropriate point at which to cease vote counting once a strong majority is established, ensuring that the outcome cannot be reversed. With these two improvements, the complexity of the voting mechanism is significantly reduced, making it feasible to implement [28].

In **ByteBall** (also known as Obyte) [9] each user maintains a local copy of the state of the network. The network topology is convergent ($\hat{C}$), the unit is a single transaction. To prevent attacks, the system employs central authorities known as witness nodes. The base currency, bytes, is used to pay for adding data to the Byteball database, where users pay 1,000 bytes to add 1KB of data. This model introduces a measure of utility and establishes a negative feedback loop for the price of bytes, driven by demand and actual usage rather than speculation. Byteball whitepaper asserts that its currency will exhibit less volatility compared to most cryptocurrencies, as its value is anchored to real-world usage through the utility of bytes [9]. This premise is not substantiated, as evidenced by the price graph showing that the Byteball price is as volatile as that of any other cryptocurrency [11]. The Byteball network incentivizes users to include as many recent transactions as possible when adding data, paying part of the unit's fees to those who include it first. To handle double-spending, Byteball employs a method where if there is no partial order between two units trying to spend the same output, both are accepted initially. A total order is later established when units are buried deep enough under newer units. If a user posts two units without partial order between them, these are treated as double-spends even if they do not attempt to spend the same output. This approach ensures clarity and prevents abuse within the network. Byteball allows users to issue other assets and use them as a means of payment. These assets might represent debt in fiat currencies or natural units like kWh or barrels of oil, and their prices are naturally bound to the underlying currencies or commodities.

Table 1 compares central differences between the blockchains. The topology category has already been widely discussed in other sections and is therefore omitted here. Notable with regard to the In/Out degree of transactions is that Nano relies on one receive block confirming one send block. This is different from many other chains, which mostly rely on each new transaction confirming multiple old ones. This mechanism ensures more certainty in the other networks.

In terms of the blocklessness criterion, it can be seen that all blockchains except for Ghost and Spectre do not aggregate transactions into blocks but rather handle them individually. Nano uses the term block, but each block is made up of only one transaction. IOTA has a unique role in terms of the voting class, as the transactions (cumulative weight) of the subtangle do the voting. In all other systems, representatives, witnesses, or validators take on this role, using stake assigned to them to validate transactions.

Fees and power usage per transaction are low for all DAG-based blockchains, as this is their main area of improvement over traditional blockchain systems. Additionally, the confirmation time of a transaction in the network is in the range of seconds to minutes compared to a confirmation time of approximately 1 hour for Bitcoin. A. Baczkowski [8] compared confirmation time and throughput of the different blockchains mentioned in this paper. The numbers were gathered in 2022 and are bound to change with updates to the blockchain systems and should be viewed as a rough reference. Bitcoin (without layer 2 solutions), for comparison, has a throughput of 7 transactions per second and is thus far slower than any DAG-based blockchain.

All systems discussed are already permissionless except for Hedera, which is on its path to becoming a permissionless blockchain as its ecosystem grows further. Viewed through the perspective of monetary inflows, the Hedera and Avalanche networks are the most relevant networks in the sphere of DAG-based blockchains

Table 1: Comparison Table of Selected DAG-based Blockchains

Category	Hedera [3]	Iota [25]	Byteball [9]	Avalanche [27]	Nano [19]	Ghost [30]	Spectre [28]
Topology [34]	Parallel ($\hat{P}$)	Diverging ($\hat{D}$)	Converging ($\hat{C}$)	Diverging ($\hat{D}$)	Parallel ($\hat{P}$)	Converging ($\hat{C}$)	Diverging ($\hat{D}$)
In/Out degree	(x, 2)	(x, 2)	(x, y)	(x, y)	(1, 1)	(x, 1)	(x, y)
Blockless	y	y	y	y	y	n	n
Voting class	Representatives	Transactions	Witnesses	Validators	Accounts/Representatives	Validators	Validators
Voting power	POS	POW	Multi-layer	POS	POS	Validators	Validators
Fees (USD)	0.0001	0	per used bytes	0.3	0	Low	Low
Confirmation Time (s) [8]	3-5	60-300	Minutes	1-2	0.14	Seconds	Seconds
Power usage (KWH/T)	0.00017	0.00016	Low	Medium	0.000112	Medium	Medium
Market Cap (USD) [10]	3.8B	700M	7M	14B	157M	-	1M
Permissionless	n (planned)	y	y	y	y	y	y
Throughput (TPS) [8]	10k+	1.5k	Medium	5k+	1k	Low	Low

in 2024. A reason for this is that they both offer a wide range of functionality and have big, trusted users of the network.

5 SECURITY ANALYSIS

In this section, we will provide a concise introduction to five potential attack scenarios that can occur in DAG-based blockchains. Following this, we will evaluate the effectiveness of the countermeasures implemented by these blockchains. We will mention the blockchain that addresses this particular vulnerability in its whitepaper and specifies any countermeasures. If the vulnerability is not discussed or if no other paper has analyzed it, it will be omitted due to the high complexity of such analysis. We included this section because recent developments have revealed that initially unnoticed vulnerabilities in DAG-based blockchains may represent a the main drawback to their use [32, 35].

Parasite chain attack [25], previously detailed in the IOTA section, involves an attacker creating a subgraph that occasionally references the main chain. The objective of this attack is to gain profit through double spending. This vulnerability exists in DAG-based blockchains utilizing probabilistic consensus mechanisms. The countermeasures against such attacks are discussed in both Spectre (voting mechanism) [28] and IOTA (Tip Selection Algorithm) [25], and both are expected to be robust.

Splitting/Balance attack [23] is a potential vulnerability in Proof-of-Work (PoW) based protocols, wherein the attacker seeks to divide the blockchain or graph into multiple subchains or subgraphs, maintaining equilibrium among them to facilitate the introduction of conflicting transactions. This type of attack has been analyzed and demonstrated to be infeasible in the GHOST [29], OHIE [36], IOTA [25], and Prism [2] protocols. Conversely, research has indicated that the Conflux blockchain [20] was susceptible to this form of attack [36]. Subsequently, Conflux proposed a new consensus mechanism analogous to that utilized in GHOST to mitigate this vulnerability.

Sybil attack [13] involves an adversary generating multiple pseudonymous identities to engage in malicious activities. Blockchains that employ variations of Proof-of-Work (PoW) or Proof-of-Stake (PoS) mechanisms are inherently resistant to such attacks due to their underlying consensus protocols. This type of attack has been examined in the context of Nano [19] and Blockclique [14] blockchains, both of which have demonstrated resistance to Sybil attacks.

Large weight attack [5], as discussed in the IOTA section, occurs when adversaries invalidate a recently confirmed transaction by introducing a conflicting transaction with a high weight (confidence). This type of attack presents challenges in the divergent topologies ($\hat{D}$), necessitating the implementation of central synchronization authorities to mitigate its effects [32].

Censorship attack [6] takes place when adversaries collude with enough committee members responsible for the consensus process in permissioned systems. These members may collaboratively block specific transactions from being included in blocks. If members can gain extra profits by censoring certain blocks or transactions, more members might be enticed to join the attack. Thus, the attack is dependent on the incentive structures in place and primarily targets permissioned systems with fixed committees. This type of attack has been found ineffective in the Prism protocol [2].

Table 2 provides an overview summarizing the aforementioned attacks.

Table 2: Overview of Discussed Attacks [35]

Attack	Assumption	Scope	Instance
Parasite Chain	– Enough power to generate a parasite chain outpacing than peers – No instant finality	Probabilistic consensus	IOTA Spectre
Balance Att.	– Split/partition the network – Secretly wander across subgraphs – Enough power to extend subgraphs	PoW-based protocols	GHOST Conflux OHIE Prism
Large Weight	– Units heavier than others – No instant finality	Probabilistic consensus	IOTA Conflux
Censorship Att.	– A closed committee – Collude the majorities – Static committee selection	Permissioned systems	Prism
Sybil Att.	– Generate identities without any cost – Enough power to create identities	Committee selection decouples with block production	Blockclique Nano

6 SYSTEM CHALLENGES

The most pressing current challenges of DAG-based blockchain systems are the lack of incentive mechanisms (1), development difficulty (2), the compatibility with smart contracts (3), the unseen vulnerabilites and need of trusted authority (4) and the lack of standardization and adoption (5).

(1) Low fees throughout the network allow for fast microtransactions but come at the cost of a lack of incentives for miners. This makes DAG systems either rely on newer transactions to verify the older transactions or make the transaction processing as lightweight as possible.
(2) In terms of development difficulty, the same problems face DAG-based systems as do face distributed database systems. Ensuring ACID guarantees meanwhile having a distributed system is hard. Some DAG chains hence drop certain features like the exact ordering of transactions in order to simplify the development. Additional problems in the development are that many chains do not publicly release their source code but rather only explain their designed protocols. If the source code is released often proper documentation is lacking/missing [35].
(3) The third problem is that functionality such as smart contracts are hard to provide in a DAG based setting. Some systems hence do not allow for this functionality (Nano) meanwhile other employ complicated workarounds, which add much overhead to the chain. Avalanche outsourced this problem to a separate chain, meanwhile IOTA uses an upper layer solution to the problem. The general problem why handling smart contracts is hard is that no total ordering of transactions exist in DAG based systems.
(4) Unseen vulnerabilites – The complicated design can lead to previously unseen vulnerabilities, necessitating the introduction of trusted authorities. These authorities are powerful entities that make final decisions and can take various forms, such as the coordinator in IOTA [25], the representatives in Nano [19], the proposer block in Prism [2], and the witness nodes in Byteball [9]. Trusted authorities can either directly influence the consensus or indirectly resolve conflicts. While this approach accelerates the confirmation of units, it compromises decentralization, which in the context of blockchain technologies is a major drawback.
(5) Standardization and adoption – DAG-based blockchains have not achieved widespread acceptance. This is partly due to the fragmentation within the DAG-based blockchain ecosystem, where various projects implement differing protocols and architectures, leading to a lack of uniform standards. This fragmentation complicates the development of interoperable solutions and discourages developers and businesses from investing in and integrating these technologies. Consequently, the limited adoption hinders the growth of a robust ecosystem, making it difficult for DAG-based systems to compete with more established blockchain technologies.

7 CONCLUSION

In this paper, we introduced the concept of Directed Acyclic Graph (DAG) structures within the context of blockchain technologies. We defined several key terms used across multiple DAG-based blockchain systems, as the terminology can vary significantly between different blockchains, which may be confusing for newcomers.

The paper then compares and contrasts seven different DAG-based blockchains, identifies potential attack vectors, discusses critical challenges, and explains key properties of the mentioned blockchains such as topology and consensus mechanisms. A particular focus is placed on the widely used blockchain IOTA, which serves as a reference point for comparing the other blockchains. DAG-based blockchains are acyclic in structure, meaning that, unlike traditional blockchain systems, there is no need for total ordering of transactions. This allows for the parallel processing of transactions, thereby increasing throughput compared to traditional blockchain systems

In terms of topology, there are notable differences: IOTA, Avalanche, and Spectre are classified as having a diverging topology; Hedera and Nano have a parallel topology; and Byteball and Ghost have a converging topology. Common attack vectors addressed in different ways by these chains include the parasite chain attack, the splitting attack, the Sybil attack, and the large weight attack. Beyond addressing potential attack vectors, developers face challenges due to the highly parallel nature of DAG systems. A criticism that can be mentioned here is that certain systems do not release their source code for review, which raises concerns about relying on security by obfuscation rather than the quality of the system.

In summary, DAG-based chains take on various shapes and implementations but all aim to make blockchain systems more scalable, faster, and cheaper. Common use cases include microtransactions, Internet of Things (IoT) integrations, and scenarios that rely on real-time data processing. Currently, much of the research is focused on ensuring real-time data processing, which may lead to extensions of existing systems or the development of new DAG-based blockchains [21, 24].

However, current developments in blockchain systems indicate that the more popular methods for scalability continue to be vertical scaling (increasing block size, decreasing block time), sharding techniques (which may be perceived as similar to blockchains in the parallel ($\hat{P}$) category of DAG-based blockchains), and most importantly, layer-2 techniques such as payment channels (e.g., Bitcoin Lightning Network) and scaling via rollups on Ethereum (e.g., zkSync, Optimism).

Another significant drawback is the difficulty associated with implementing smart contracts. As smart contracts and Web3 technologies become increasingly popular, it becomes necessary to integrate them into blockchain systems. However, DAG-based blockchains employing partial ordering are unable to support them. The only viable solution is to introduce an additional fully ordered layer, which essentially reverts to a linear blockchain structure.

While some ideas introduced in DAG-based blockchain research might be adopted by other blockchains (such as the GHOST consensus protocol in Ethereum 1.0), in our opinion, the most popular blockchains are likely to remain linear, at least in the near future.

REFERENCES

[1] 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. *URL: http://www. opensciencereview. com/papers/123/enablingblockchain-innovations-with-pegged-sidechains* 72 (2014), 201–224.

[2] Vivek Bagaria, Sreeram Kannan, David Tse, Giulia Fanti, and Pramod Viswanath. 2019. Prism: Deconstructing the blockchain to approach physical limits. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*. 585–602.

[3] Leemon Baird, Mance Harmon, and Paul Madsen. 2019. Hedera: A public hashgraph network & governing council. *White Paper* 1, 1 (2019), 9–10.

[4] 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.

[5] Gewu Bu, Wassim Hana, and Maria Potop-Butucaru. 2019. Metamorphic iota. *arXiv preprint arXiv:1907.03628* (2019).

[6] Vitalik Buterin. 2015. The Problem of Censorship. https://blog.ethereum.org/2015/06/06/the-problem-of-censorship. Accessed: 2024-07-11.

[7] Vitalik Buterin, Diego Hernandez, Thor Kamphefner, Khiem Pham, Zhi Qiao, Danny Ryan, Juhyeok Sin, Ying Wang, and Yan X Zhang. 2020. Combining GHOST and casper. *arXiv preprint arXiv:2003.03052* (2020).

[8] Aleksander Bączkowski. 2024. What is the fastest blockchain and why? Analysis of 43 blockchains. https://medium.com/aleph-zero-foundation/what-is-the-fastest-blockchain-and-why-analysis-of-43-blockchains-800b3b9aa9ee Accessed: 2024-06-15.

[9] Anton Churyumov. 2016. Byteball: A decentralized system for storage and transfer of value. *URL https://byteball. org/Byteball. pdf* (2016), 11.

[10] CoinMarketCap. [n. d.]. https://coinmarketcap.com/. Accessed: 2024-05-22.

[11] CoinMarketCap. 2024. Obyte (GBYTE) Price. https://coinmarketcap.com/currencies/obyte/. Accessed: 2024-07-23.

[12] Lin William Cong, Zhiguo He, and Jiasun Li. 2020. Decentralized Mining in Centralized Pools. *The Review of Financial Studies* 34, 3 (04 2020), 1191–1235. https://doi.org/10.1093/rfs/hhaa040

[13] John R Douceur. 2002. The sybil attack. In *International workshop on peer-to-peer systems*. Springer, 251–260.

[14] Sébastien Forestier, Damir Vodenicarevic, and Adrien Laversanne-Finot. 2018. Blockclique: scaling blockchains through transaction sharding in a multithreaded block graph. *arXiv preprint arXiv:1803.09029* (2018).

[15] Ethereum Foundation. 2022. The Merge. https://ethereum.org/en/roadmap/merge/. Accessed: 2024-07-23.

[16] Lewis Gudgeon, Pedro Moreno-Sanchez, Stefanie Roos, Patrick McCorry, and Arthur Gervais. 2020. Sok: Layer-two blockchain protocols. In *Financial Cryptography and Data Security: 24th International Conference, FC 2020, Kota Kinabalu, Malaysia, February 10–14, 2020 Revised Selected Papers 24*. Springer, 201–226.

[17] Hashrate Index. [n. d.]. Bitcoin Mining Pool Data: Hashrate Index. https://hashrateindex.com/hashrate/pools. Accessed July 01, 2024.

[18] Artyom Kosmarski. 2020. Blockchain adoption in academia: Promises and challenges. *Journal of Open Innovation: Technology, Market, and Complexity* 6, 4 (2020), 117.

[19] Colin LeMahieu. 2018. Nano: A feeless distributed cryptocurrency network. *Nano [Online resource]. URL: https://nano. org/en/whitepaper (date of access: 24.03. 2018)* 16 (2018), 17.

[20] Chenxin Li, Peilun Li, Dong Zhou, Zhe Yang, Ming Wu, Guang Yang, Wei Xu, Fan Long, and Andrew Chi-Chih Yao. 2020. A decentralized blockchain with high throughput and fast confirmation. In *2020 {USENIX} Annual Technical Conference ({USENIX}{ATC} 20)*. 515–528.

[21] Guoqiong Liao, Hao Ding, Chuanling Zhong, and Yinxiang Lei. 2024. RT-DAG: A DAG-Based Blockchain Supporting Real-Time Transactions. *IEEE Internet of Things Journal* (2024), 1–14. https://doi.org/10.1109/JIOT.2024.3409025

[22] Satoshi Nakamoto. 2008. Bitcoin: A peer-to-peer electronic cash system. (2008).

[23] Christopher Natoli and Vincent Gramoli. 2016. The balance attack against proof-of-work blockchains: The R3 testbed as an example. *arXiv preprint arXiv:1612.09426* (2016).

[24] Junpei Ni, Jiang Xiao, Shijie Zhang, Bo Li, Baochun Li, and Hai Jin. 2023. FLUID: Towards Efficient Continuous Transaction Processing in DAG-Based Blockchains. *IEEE Transactions on Knowledge and Data Engineering* 35, 12 (2023), 12679–12692. https://doi.org/10.1109/TKDE.2023.3272312

[25] Serguei Popov. 2018. The tangle. *White paper* 1, 3 (2018), 30.

[26] Serguei Popov, Hans Moog, Darcy Camargo, Angelo Capossele, Vassil Dimitrov, Alon Gal, Andrew Greve, Bartosz Kusmierz, Sebastian Mueller, Andreas Penzkofer, et al. 2020. The coordicide. *Accessed Jan* 12, 18 (2020), 1–30.

[27] Kevin Sekniqi. 2020. Whitepapers. https://www.avalabs.org/whitepapers.

[28] Yonatan Sompolinsky, Yoad Lewenberg, and Aviv Zohar. 2016. Spectre: A fast and scalable cryptocurrency protocol. *Cryptology ePrint Archive* (2016).

[29] Yonatan Sompolinsky and Aviv Zohar. 2015. Secure high-rate transaction processing in bitcoin. In *Financial Cryptography and Data Security: 19th International Conference, FC 2015, San Juan, Puerto Rico, January 26-30, 2015, Revised Selected Papers 19*. Springer, 507–527.

[30] Yonatan Sompolinsky and Aviv Zohar. 2018. Phantom. *IACR Cryptology ePrint Archive, Report 2018/104* (2018).

[31] Surya Viswanathan and Aakash Shah. 2018. The scalability trilemma in blockchain. *Medium online* 20 (2018).

[32] Bozhi Wang, Qin Wang, Shiping Chen, and Yang Xiang. 2020. Security analysis on tangle-based blockchain through simulation. In *Information Security and Privacy: 25th Australasian Conference, ACISP 2020, Perth, WA, Australia, November 30–December 2, 2020, Proceedings 25*. Springer, 653–663.

[33] Gang Wang, Zhijie Jerry Shi, Mark Nixon, and Song Han. 2019. Sok: Sharding on blockchain. In *Proceedings of the 1st ACM Conference on Advances in Financial Technologies*. 41–61.

[34] Qin Wang, Jiangshan Yu, Shiping Chen, and Yang Xiang. 2020. SoK: Diving into DAG-based blockchain systems. *arXiv preprint arXiv:2012.06128* (2020).

[35] Qin Wang, Jiangshan Yu, Shiping Chen, and Yang Xiang. 2023. Sok: Dag-based blockchain systems. *Comput. Surveys* 55, 12 (2023), 1–38.

[36] Haifeng Yu, Ivica Nikolić, Ruomu Hou, and Prateek Saxena. 2020. Ohie: Blockchain scaling made simple. In *2020 IEEE Symposium on Security and Privacy (SP)*. IEEE, 90–105.